ABSTRACT

APPLICATION OF CHEMICAL KINETIC MODELING TO IMPROVE DESIGN AND PERFORMANCE CRITERIA FOR A PRACTICAL INCINERATION SYSTEM

by
Charles A. Bass, Jr.

In this study, detailed thermo-chemical kinetics with networked ideal reactor model were applied to simulate a practical combustion system - the Secondary Combustion Chamber (SCC) of the Rotary Kiln incineration Simulator (RKIS) at the EPA facility at Research Triangle Park, NC. The networked ideal model was developed using analysis of reactor geometry, temperature profile measurements, and SO_2 tracer data provided by EPA. A computer simulation of the networked model was developed using the *CHEMKIN II* library. A parallel effort considered the effects of non-ideal mixing on detailed thermo-kinetic simulations. Specifically, an alternate approach was developed to solve the Partially Stirred Reactor (PaSR) model that allowed the incorporation of large detailed mechanisms. Both ideal and non-ideal modeling approaches were compared with experimental data gathered on a Toroidal Jet Stirred Combustor (TJSC) and the SCC at EPA. SCC experiments measured Product of Incomplete Combustion (PIC) formation of surrogate chlorinated wastes (CCl_4 and CH_2Cl_2), while the TSJC experiments measured PIC formation in ethylene/air combustion for fuel-lean conditions near blowout and fuel-rich conditions.

Analysis of the geometry and temperature profiles of the SCC suggested the existence of up to four distinct mixing zones. The RTDs, which were resolved from the tracer studies, further supported a multiple PSR model. A model was chosen based on the best fit to SO_2 tracer data and consistency with physical geometry, resulting flow patterns, and temperature

measurements. A thermo-kinetic mechanism developed by Chiang (1995) was applied to the model. The model results did not agree well with the experimental data. However, it followed many of the underlying trends revealed by the data. Sensitivity analysis of the parameters was used to further explore trends and recommend potential design improvements to reduce PIC formation.

An alternate solution technique was developed for the PaSR which approximated mean conditions and solved the deterministic model to refine the approximation and eventually converge on a solution. The approximation, direct integration, and convergence technique compared favorably with the published Monte Carlo modeling calculations, but used, on average, less than $1/200^{th}$ of the CPU time. This new technique allowed use of considerably larger detailed mechanisms. Additionally, a generalized PaSR model was proposed to account for the effects of non-ideal macromixing.

APPLICATION OF CHEMICAL KINETIC MODELING TO IMPROVE DESIGN AND PERFORMANCE CRITERIA FOR A PRACTICAL INCINERATION SYSTEM

by
Charles A. Bass, Jr.

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfilment of the Requirements for the Degree of
Doctor of Philosophy

Department of Chemical Engineering

January 2002

APPROVAL PAGE

APPLICATION OF CHEMICAL KINETIC MODELING TO IMPROVE DESIGN AND PERFORMANCE CRITERIA FOR A PRACTICAL INCINERATION SYSTEM

Charles A. Bass, Jr.

Dr. Robert B. Barat, Dissertation Advisor
Associate Professor of Chemical Engineering, NJIT

Dr. Piero Armenante, Committee Member
Distinguished Professor of Chemical Engineering, NJIT

Dr. Joseph W. Bozzelli, Committee Member
Acting Chairperson and Distinguished Professor of Chemistry and Environmental Science, NJIT

Dr. Dana E. Knox, Committee Member
Associate Professor of Chemical Engineering, NJIT

Dr. Paul M. Lemieux, Committee Member
Air Pollution Prevention and Control Division, National Risk Management Research Laboratory, Environmental Protection Agency

---

Dr. Adel F. Sarofim, Committee Member
Presidential Professor of Chemical and Fuels Engineering, University of Utah

BIOGRAPHICAL SKETCH

Author: Charles A. Bass, Jr.

Degree: Doctor of Philosophy

Date: January 2002

Undergraduate and Graduate Education:

- Doctor of Philosophy in Chemical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2002

- Masters of Military Arts and Science
  Command and General Staff College, Fort Leavenworth, KS, 1997

- Master of Chemical Engineering,
  Johns Hopkins University, Baltimore, MD, 1988

- Bachelor of Science in Engineering
  University of California, Los Angeles, CA, 1980

Major: Chemical Engineering

Professional Qualifications:

- Professional Engineer License (Chemical Engineering), Commonwealth of Virginia, awarded 1997

Papers and Presentations:

Bass, C. A., R. B. Barat, G. Sacchi, and P. M. Lemieux, "Fundamental Studies on the Characterization and Failure Modes of Incinerator Afterburners," *Proceedings of the 1995 Incineration Conference* (1995).

Papers and Presentations (continued):

Bass, C., R. B. Barat, G. Sacchi, T. Lee, J. P. Longwell, A. F. Sarofim, P. M. Lemieux, "Fundamental Experimental and Modeling Studies of Chlorocarbon Incineration in Afterburners," Work-in-Progress Poster Session, *25th International Symposium on Combustion*, The Combustion Institute (1994).

Bass, C. A., and R. B. Barat, "Detailed Kinetic Modeling of Ethylene/Air Combustion in a Toroidal Jet Stirred Combustor Using a Partially Stirred Reactor Model," Presentation, *2001 Technical Meeting of the Eastern States Section, Combustion Institute*, (2001).

Lemieux, P., J. Ryan, C. Bass, and R. Barat, "Emissions of Trace Products of Incomplete Combustion from a Pilot-Scale Incinerator Secondary Combustion Chamber," *Journal of the Air and Waste Management Association*, **46**, 309-316 (1996).

"When first under fire an' you're wishful to duck,
Don't look nor take 'eed at the man that is struck,
Be thankful you're livin', and trust to your luck
And march to your front like a soldier. . ."

– R. Kipling

ACKNOWLEDGMENT

I would like to express my most sincere appreciation to my thesis advisor, Dr. Robert Barat, for his help, continual encouragement, and ability to keep faith in me over the past eight years. Likewise, I would like to express thanks to my committee for their encouragement and patience. Special thanks is extended those who ran the experiments and allowed me the contribute. These include Dr. Paul Lemieux and Jeffrey Ryan from the Air Pollution Technology Branch, of the National Risk Management Research Laboratory, who ran the tracer and chlorocarbon combustion experiments at the EPA Research Triangle Park Facility, and Tara Salem, who obtained the data from the NJIT toroidal jet stirred combustor experiments.

Additionally, I would like to acknowledge colleagues and supervisors who help me through this process. All help, whether a sympathetic ear, a spirited exchange of ideas, or funding provided to attend a conference, was much appreciated. I would like to especially thank Colonel David Allbee, Chairman of the Chemistry Department, United States Military Academy, for his encouragement and providing me the time to get started, and John Hill and Bob Fithian from UTD Incorporated for helping me finish.

My deepest gratitude goes to my family for enduring the absences and bearing the load over the past eight years. My wife, Sue Marie, is a true inspiration for the way she has managed the family during five household moves, and brought two wonderful children in the world, while I continued my research. She has provided me with the best encouragement when I needed it the most.

TABLE OF CONTENTS

Chapter **Page**

Chapter **Page**

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

Figure **Page**

Figure **Page**

Figure **Page**

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

Incineration is a viable and necessary means of disposal for hazardous waste. It has the advantages of reducing the hazard of the waste, reducing its volume, and allowing for energy recovery. However, hazardous waste incineration has received much criticism for the pollutants formed by incomplete combustion called Products of Incomplete Combustion (PICs). Incinerators can fail to adequately destroy the Principle Organic Hazardous Constituent (POHC) or produce pollutants above regulatory limits for a number of reasons. These include kinetic and thermodynamic failures such as improper stoichiometry (too little or too much excess oxygen), inadequate temperature, insufficient time to react, and mixing failures.

Modeling of combustion processes constitutes one of many research approaches to understanding PIC production. While not a substitute for experiments, it has many advantages. A model can be analyzed under a variety of conditions much faster and at a lower cost than these conditions can be replicated in bench and pilot scale experiments. Models are useful in determining trends and identifying relationships between easily measured compounds - target analytes - and the production of hazardous, hard-to-measure PICs. They can identify potential incinerator conditions that minimize PIC production. Elaine Oran and Jay Boris (2001) argue that each simulation should be considered as a unique computer experiment, "The simulation can tell us about something new unexpected when the model is

complete enough, much as a laboratory experiment can teach us something new about the physical environment." The insights gained from modeling can be used for improving design of experiments, combustor designs and process control schemes.

The modeling of combustion processes combines three areas: thermodynamics, kinetics, and mixing. Thermodynamic modeling predicts the thermodynamic properties of all reaction species (reactants, products, and intermediates) over the range of reaction conditions. Since ideal gas conditions generally apply in a combustor, statistical mechanics can successfully extrapolate the measured properties of stable species and predict properties of unstable radical intermediates through group additivities. Kinetic modeling can range in detail between fundamental mechanisms at one extreme and global mechanisms at the other. Global mechanisms use empirically fitted parameters that apply for a narrow range of conditions. Fundamental mechanisms use detailed elementary reaction mechanisms. The parameters of each reaction step are either based on published experimental results, estimated from an appropriate analogy, or derived from fundamental thermo-chemical kinetic principles. Fundamental mechanisms can apply over a broader range of conditions than global mechanisms. Because they usually consider all possible kinetic pathways, fundamental mechanisms are useful in predicting PICs. Mixing models attempt to approximate how reactants are brought together and the products dispersed. The detail of these models can range from ideal reactor models to complicated turbulent mixing models. Computational limitations confine the detail of the overall combustor model. One has to trade-off between the detail of the kinetic mechanism and the detail of the mixing model.

The role of mixing in the modeling process depends on the ratio of the mixing time over the reaction time, the Damköhler number. A system with a large Damköhler number is mixing limited and requires a detailed mixing model to describe the combustion. A system with a small Damköhler number is chemically limited and requires a detailed chemical mechanism to describe the combustion. Such a system behaves like a well stirred reactor and may be approximated by a perfectly stirred reactor (PSR). The advantage of the PSR is that it is numerically easy to solve. For cases where the temperature is specified, the solution can be rapidly found on a computer, even with a mechanism that involves hundreds of species. An approach in reactor engineering has been to approximate complex mixing with a series of ideal reactors that model the macro mixing zones of the system. However, the effect of non-ideal micro-mixing must be considered at some point in order to accurately model the reaction.

This study focuses modeling efforts on the combustion of chlorinated hydrocarbons. These are of interest because a significant portion of hazardous wastes consists of chlorinated hydrocarbons which are more difficult to destroy completely by incineration than non-chlorinated hydrocarbons. These compounds affect combustion in two significant ways: they reduce combustion efficiency (increase the ratio of carbon monoxide to carbon dioxide) and they increase the rate of molecular weight growth in the combustion process leading to more and heavier PICs. Because of these characteristics, chlorinated hydrocarbons, particularly carbon tetrachloride, are commonly chosen as the surrogate POHC in trial burns.

1.2 Research Objectives

This study applies detailed thermochemical kinetics with simplified mixing models to model a practical combustion system - the Secondary Combustion Chamber (SCC) of the Rotary Kiln Incinerator Simulator (RKIS) at the EPA combustion facility at Research Triangle Park, NC. Its overall objective is to apply this model to make generalizations about the design and control of SCCs for hazardous waste incineration, and suggest strategies to mitigate the formation of products of PICs. Specific objectives are to:

1. develop an ideal reactor model that approximates the macro mixing zones in the SCC of the RKIS.
2. determine the ideal reactor model parameters - the mean residence times of the ideal reactors and bypasses - using SO_2 tracer data provided by EPA.
3. code a *CHEMKIN* driver program that simulates the ideal reactor model and contains a post processing capability to analyze the results of the simulations.
4. investigate the effects and trends of departing from perfect micro-mixing within the ideal reactors of the above model on PIC formation.
5. validate mixing models on data from experimental studies on a bench-scale toroidal jet-stirred combustor (TSJC).
6. compare experimental results from carbon tetrachloride and methylene chloride injections in the RKIS to the model.
7. analyze reaction mechanisms within the SCC model by exploring reaction pathways and parameter sensitivity.

8. use the model to investigate strategies to mitigate PIC formation, explain the mechanisms of various failure modes on PIC production, and suggest design improvements to the SCC.

1.3 Research Approach

This study focuses on the application of fundamental combustion mechanisms. It accomplishes this through a reactor engineering approach by developing a multi-parameter ideal reactor model, then analyzing the departures from ideal mixing. The model employs detailed chemical kinetic mechanisms developed by other researchers and uses them to analyze important reaction pathways to selected PICs. Chapter 2 presents a literature review on use of ideal reactor models and non-ideal reactor models. Additionally, it reviews techniques for reactor characterization. It also reviews developments in kinetic modeling for chlorocarbon combustion and formation of aromatics in the combustion process.

EPA contractors ran tracer studies on the SCC of the RKIS using step inputs of SO_2 as the tracer gas, and measured SCC temperature profiles during stoichiometric RKIS operation. These studies delivered data on both the SCC and the SO_2 sample train. This study uses these data to determine mean residence times and variances between different points of the SCC and estimate temperature cross section contours. Chapter 3 describes the methods and results in detail. This characterization provided a basis for SCC model development.

Finding an optimal model, a model that provides the best overall fit to tracer results, was not adopted as a goal. Instead, models were developed that explain observations, such as distinct temperature zones, and consider reactor geometry, while using the fewest possible

parameters. These candidate models were then narrowed to a single model that provided the best fit to the tracer data using methods described in Chapter 4. The best fit model and its parameters were used in a *CHEMKIN* driver program developed with sufficient flexibility to specify reactor conditions, feeds, and model parameters at run time.

A parallel effort considered non-ideal reactor models (Chapter 5). The reactor dispersion model and the segregated stirred reactor models provided two points of departure. The dispersion model is a departure from the ideal plug flow reactor (PFR) that incorporates axial turbulent diffusion. The segregated flow model departs from the PSR by supposing that fluid travels through a stirred reactor in infinitesimally small packets that do not mix. This study used an approach that regulated the rate at which these fluid packets mixed with the mean reactor conditions. This is the partially stirred reactor (PaSR) model.

Both ideal and non-ideal modeling approaches were compared with experimental data gathered on a TJSC at NJIT and the RKIS at EPA. The TJSC studies explored the formation of benzene during premixed ethene/air combustion under fuel rich conditions. The RKIS studies examined the formation of PICs in fuel lean, stoichiometric, and fuel rich conditions when dichloromethane and tetrachloromethane were injected into the SCC (Chapter 6). The results were compared to the model results including sensitivity analysis of the ideal reactor model parameters. The mechanism was analyzed by using reaction pathway analysis, which was used to gain insights into incinerator failure modes.

CHAPTER 2

LITERATURE SURVEY

Studies of both full scale incinerators and pilot scale research reactors show the need for combining detailed thermo-kinetic mechanisms with the complexities of mixing models in order to gain insights between incinerator design and the formation of pollutants. Fundamental approaches to combustion kinetics and thermodynamics provide the insight into the mechanism of pollutant formation. Normally, the complexity of these mechanisms require the use of simple ideal mixing models to obtain a numeric solution. Research in this area often relies on bench-scale research reactors such as tubular flow reactors, premixed laminar flames, and jet-stirred combustors to emulate ideal mixing. In contrast, the modeling of practical systems usually focuses on the complexity of mixing in computationally demanding simulations which become intractable with detailed chemistry. Applying detailed thermo-kinetic models to practical combustion systems requires some simplifying assumptions and often use ideal mixing models as a point of departure. This literature survey looks broadly at work performed in these areas.

2.1 The Problem of Incineration

Many studies support the use of incineration as a viable and safe means to dispose of hazardous materials. However, the tendency to form different hazardous materials as a result of combustion, especially in the destruction of chlorinated compounds, throws the use of incineration into question. EPA-sponsored testing conducted by Trenholm, Gorman, and

Jungclaus (1984) studied eight full scale hazardous waste incineration facilities. They observed that Destruction and Removal Efficiency (DRE) was strongly correlated to concentration of Principal Organic Hazardous Component (POHC) in the waste feed. Specifically, as the POHC concentration decreased in the waste stream the DRE, expressed as the percentage of the loss of the POHC, decreased below the required 99.99 percent DRE standard. Further, they found that CO and THC levels, typical indicators for combustion efficiency, were not good indicators of POHC emissions or DRE, but temperature correlated well with DRE. Three of six sites tested had stack emissions of pentachlorodibenzofurans (PCDF), and pentachlorodibenzodioxins (PCDD). While the authors did not suggest a mechanism for chlorinated dioxin/furan formation, they observed that formation occurred during cooling in the stack. In order to mitigate these problems a better understanding of the fundamental kinetic and mixing processes that produce these results is required.

Mixing studies have been applied to full-scale mass burn incinerators using Computational Fluid Dynamic (CFD) models. In two separate studies, Ravichandran and Gouldin (1993) and Nasserzadeh *et al.* (1994) used transient 2-dimensional numerical simulations of tracer injection to determine mixing dynamics. Both studies emphasized significance of reactor geometry, and the proper placement of burner jets to achieve a desired level of macromixing. They also highlighed the need for *in situ* tracer experiments that determine the Residence Time Distribution (RTD) rather than relying on physical geometry to approximate mean residence time.

A number of studies that employ pilot-scale incinerators have been conducted to gain a better understanding of combustion of chlorinated hydrocarbons. Cundy *et al.* (1989)

performed *in situ* sampling and kiln-simulator experiments using CCl_4 as a surrogate waste. These studies revealed a relationship between turbulent mixing and samples of CO, and fuel (methane) in the kiln, which had little effect on the methane and CO measured in the Secondary Combustion Chamber (SCC) and stack. Effects of the activation of turbulent air in the kiln were smoothed and reduced by additional mixing in the SCC. Computer simulations of the turbulent combustion of CCl_4 (used as a surrogate waste) using the eddy break-up model, tended to over predicted CCl_4 concentration thus highlighting the need of a realistic finite rate kinetic for CCl_4 destruction. Wendt and Linak (1988) and Lemieux *et al.* (1992) explored the transient effects from batch incineration. Transient puffs from batch feeding of surrogate wastes create temporary system failures even in the presence of 100 percent excess air. The magnitude of the failure increased with temperature and kiln rotation speed. These studies highlight the complex relationship between kinetics and mixing for waste destruction and that a useful design model should incorporate both factors.

2.2 Detailed Thermo-Kinetic Modeling

Detailed thermo-kinetic mechanisms are compilations of relevant elementary reactions applied with the corresponding thermodynamic properties of each species. When applied to hazardous material incineration, these models provide a fundamental approach to the understanding of stable intermediates production that become the body of pollutants which make incineration controversial. Using broad sets elementary reactions with kinetic and thermodynamic parameters valid over a large range of conditions allows the researcher to

infer how reaction pathways and PIC production changes as conditions change. Detailed thermo-kinetic models are necessary to gain a broad understanding of PIC production, but this approach is not without its limitations.

To faithfully represent nature, one would have to include all possible elementary reactions and intermediate species in a mechanism. Instead, available thermo-kinetic mechanisms bound the problem by focusing on a limited range of fuels and PICs of interest. Mechanisms range in detail from a half dozen species in a few dozen reactions to hundreds of intermediate species and thousands of reactions. The required thermodynamic and kinetic parameters are either taken from literature or estimated.

Many of these estimation techniques are summarized by Dean and Bozzelli (2000). They include group additivity estimation of thermodynamic properties (Cohen and Benson, 1993), transitions state theory estimation of Arrhenius *A*-factors, and estimation of activation energies for hydrogen atom abstractions using the "Evans–Polanyi relationship." The pressure dependence of kinetic parameters for unimolecular reactions and bimolecular reactions with an intermediate activated adduct is modeled by one of several approaches. The pressure fall-off modeled by Lindemann (Kee, Rupley, and Miller, 1989) inadequately describes the phenomenon to provide good predictions (Dean and Bozzelli, 2000). Other models such as Troe, SRI, and Landau-Teller alone with the use of enhanced third body collisions have supplemental parameters that work directly with the *CHEMKIN-II* intrepeter (Kee, Rupley, and Miller, 1989). The Quantum Rice-Ramperger-Kassel (QRRK) approach

(Westmorland, Howard, Longwell and Dean, 1986) recognizes the quantized states of the energized adduct and is particularly useful for predicting results for multiple channel reactions.

Once thermo-kinetic mechanisms are compiled, the process of adjustment and optimization is required to gain agreement with a real combustion system to validate the mechanism. Real combustion systems are chosen to limit the effects of mixing. The three systems most often used are the isothermal flow reactor, the premixed laminar flat flame, and the jet-stirred reactor. The isothermal flow reactor limits the effects of mixing by maintaining a plug-like flow at a constant temperature down the length of the reaction tube. Dilute concentrations and temperatures lower than those necessary for sustained combustion slows the reactions rates and keeps the system in a reaction controlled rather than a mixing-controlled regime. The premixed laminar flat flame produces an adiabatic flat flame front that can be suitably modeled in one dimension. Detailed presentations of the energy and mass conservation equations are made by Turns (1996) for calculation flame speeds and Kee *et al.* (1993) for the computer code *PREMIX* that simulates a one-dimensional laminar flame. Molecular diffusion in the laminar flame requires a additional set of parameters for each species. Finally the jet-stirred reactor (Nenniger *et al.*, 1984) produces turbulent mixing rates within the reactor rapid enough to approximate composition and temperature as homogeneous. The computer code PSR (Glarborg *et al.*, 1992), a supplemental program to the *CHEMKIN* package, readily simulates this reactor even for large thermo-kinetic mechanisms. Adjustments made to sensitive parameters within the bounds of experimental

certainty are often adequate to get good agreement between model and experimental results, but over-optimization can limit the utility of a mechanism beyond the conditions originally addressed.

An excellent example of a mechanism that has undergone optimization is the GRI 3.0 mechanism for natural gas and air combustion (Smith *et al.*, 1999). This mechanism, along with preceding versions, is frequently cited and has become an emerging standard in for modeling of C_1 and C_2 hydrocarbon combustion. It uses 53 species and 325 reactions to model a complete C_1 and C_2 as well as NO_x chemistry. It includes a limited set of C_3 combustion products to account for the minor quantities of propane in natural gas. The GRI mechanism has been optimized for natural gas combustion in a premixed laminar one-dimensional flame. The mechanism covers a broad spectrum of natural gas components including methane and minor quantities for ethane, ethylene, and acetylene.

Another useful hydrocarbon mechanism was developed by Marinov *et al.* (1998) to model the formation of polyaromatic hydrocarbons (PAH) from the combustion of *n*-butane in air under fuel-rich conditions. It has 155 species in 689 reactions, which is small considering the range of possible intermediate species between *n*-butane and polyaromatic incomplete combustion products. Like the GRI mechanism, it was optimized for the fuel rich *n*-butane combustion in a laminar flat flame. The computational model was a one-dimensional premixed laminar flat fame code (*PREMIX*). Optimization of the mechanism focused on dominant pathways to PAH growth. Analysis found the recombination of the propargyl (C_3H_3) radical as the dominant reaction for the formation of benzene.

Other recent studies in hydrocarbon combustion includes work done by Bikas and Peters (2001) who developed a mechanism for the combustion of *n*-decane. This mechanism was validated with experiments using a pre-mixed laminar flat flame as well as jet-stirred reactors. It extends upon the C_1 and C_2 model in the GRI mechanism to handle the oxidation and pyrolysis of large alkanes along a few prominent pathways using 67 species and 600 reactions. This approach contrasts with a study conducted by Heyberger *et al.* (2001) which employed a computer code to automatically generate a thermo-kinetic mechanism of propene and air combustion. The oxidation and pyrolysis reactions of C_1 through C_3 are exhaustive requiring 262 species and 1295 reactions. Validation was done against jet-stirred reactor experiments, but optimization to reduce unnecessary intermediates and reactions was not performed.

Modeling the oxidation and pyrolysis of chlorine containing compounds is of particular interest because of the relationship chlorine has to the formation of pollutants. The presence of chlorine has major effects, which inhibit flame propagation, and increase the formation of higher molecular weight aromatics. The low bond dissociation energy of the C–Cl bond favors chlorine dissociation over hydrogen dissociation especially at lower temperatures. At 1000 K the rate of C–Cl dissociation is 2000 times greater than C–H dissociation (Senkan, 2000). Additionally, the higher electronegativity of Cl results in molecular elimination of HCl as a major channel and the free chlorine radicals rapidly abstract hydrogens from other hydrocarbons. The resulting rapid rise of hydrocarbon radicals promote molecular weight growth and soot (Huang and Senkan, 1996). Free chlorine

radicals and hydrogen chloride scavenge hydrogen radicals inhibiting the chain branching of molecular oxygen. Additionally, hydrogen chloride scavenges the hydroxyl radical regenerating free chlorine radicals and inhibiting CO burnout (Ho, Barat, and Bozzelli, 1992a).

Important mechanisms have been developed by Ho *et al.* (1992a) of pyrolysis and oxidation of methylene chloride in H_2 / O_2 mixtures using tubular flow reactors, and Ho and Bozelli (1992b) of pyrolysis and oxidation of methylene chloride in the presence of C_1 and C_2 hydrocarbons. Both experiments rely on the relatively low temperature tubular reactor for validation. Sgro *et al.* (2000) also validated the Ho mechanism using a post flame injection of methylene chloride at relatively low temperatures between 900 to 1200 K. Using a similar tubular flow set-up Lou and Chang (1997) investigated the oxidation of chloroform under dilute fuel-lean conditions. Both studies found that important intermediates include CCl_3 and $COCl_2$.

Investigation into the production of PCDDs and PCDFs continues to be one of the most important areas of research in chlorinated hydrocarbon combustion. Experimental and theoretical studies of chlorobenzene pyrolysis in a dilute H_2 atmosphere (Ritter, Bozzelli, and Dean, 1990a) resulted in the development of mechanisms based upon fundamental thermodynamic and kinetic principles that explained experimental results. Extension of these models to PCDD and PCDF radical precursors demonstrated thermodynamically favorable pathways from chlorobenzene (Ritter and Bozzelli, 1990b). However, experimental evidence

that shows no correlation between chlorine content in the feed and PCDD/PCDF production (Townsend, Wilson, and Park, 1995) suggests a far more complicated mechanism that could be addressed by better combustion practices. More research is required in this area.

2.3 Ideal Reactors

Ideal reactors provide a relatively simple mixing model to solve the large equation set generated by a detailed thermo-kinetic mechanism. Because of this, a considerable amount of work has been done applying ideal modeling to physical combustion systems. Practical systems such as a swirl burner in a furnace suggest a well-stirred zone followed by a plug-flow section. Beer and Lee (1965) investigated the mixing in this system by building a $1/10^{th}$ scale water model and using a salt solution as a tracer to measure the RTD, which agreed with the two stage, Perfectly Stirred Reactor (PSR) to Plug Flow Reactor (PFR) model. The RTD (on a dimensionless basis) agreed with the RTD measured from an argon tracer used in a full-scale pulverized coal furnace.

Using a simple kinetic model for pulverized coal combustion, Beer and Lee demonstrated that theoretical conversions for a PSR followed by a PFR yield the highest combustion efficiencies for a given reactor volume. The reverse flow region of the swirl flame that corresponds to PSR region can be controlled by the adjusting the degree of swirl or geometric factors. A divergence angle beyond the burner increases the recirculating volume in the combustor for a given swirl number (Syred and Beer, 1974). Using the PSR to PFR model, combustion efficiency can be maximized by adjusting swirl and combustor design to optimize the time spent in the PSR stage.

The approach of reducing complex mixing issues to a network of ideal reactors follows the classical reactor engineering approach (Levenspiel, 1972). Swithenbank *et al.* (1973) applied this reactor engineering approach to a turbine can combustor using an extension of the PSR to PFR model to a more complex system. The combustor had a swirl burner at one end with three subsequent sets of fuel and air inlets downstream. Each inlet set reintroduced additional turbulent energy. Thus, the PSR to PFR sequence is cut short and begins again resulting in a 7 reactor model. In a follow-up study, Ewan, and co-workers (1984) deduced equivalent PSR volumes for the can combustor model from RTDs. These measurements used a technique developed by Topps (1978) that measures the adsorption of mercury vapor pulse. The resulting model was used to predict rich and lean blow-off limits.

The PSR itself provides a useful research to for exploring thermo-kinetic mechanisms. Computer codes such as *PSR* developed by Sandia National Laboratories (Glarborg, 1992) can efficiently solve large mechanisms. The research reactor must reasonably approximate the mixing conditions assumed in ideal mixing. Nenniger *et al.* (1984) developed the Toroidal Jet-Stirred Combustor (TJSC) to approximate these conditions and achieved significant improvements over existing jet-stirred research designs. Temperature profiles in the axial cross section of the torus were flat for fuel-lean and stoiciometric combustion and slightly fluctuated for fuel-rich combustion. Using the assumption of isotropic, homogeneous, turbulence they estimated turbulent parameters to deduce break-up or turbulent mixing time that reasonably approximated PSR behavior under certain conditions.

Analysis of inhomogeneous behavior in the TJSC conducted by Longwell and Bar-Ziv (1989) employed both cold-flow tracer analysis and one-dimensional, spatially dependent, modeling analysis. During fuel-rich conditions, the cross-torus temperature profile dipped in the center which contrasted with the flat temperature profiles exhibited in fuel-lean conditions. Longwell and Bar-Ziv contend that since mixing time was finite, the delayed combustion at higher fuel/air equivalence ratios resulted in a lower temperature in the jet stream in the center of the torus. This demonstrated that some reactor conditions are inadequately modeled by a PSR. This conclusion is further reinforced by experiments conducted by Barat (1992) where the dilution ratio of a fuel-lean mixture was steadily increased until combustion blowout. He demonstrated that modeling of the TJSC with a PSR inadequately predicted blowout.

The inadequacies of an ideal PSR based solely on the apparent system dimensions have been long apparent. These inadequacies arise from inhomogeneous behavior both at the macro and micro-scale. Danckwerts (1958a) distinguished between mixing at the macro and micro-scales and demonstrated how the RTD can be used to quantify macromixing. He quantified micro-scale inhomogeneities by defining the degree of segregation which varies from 0 (completely homogenous) to 1 (completely segregated). For the completed segregated case, he proposed a model with "points," defined a relatively small volumes of fluid, moving through the reactor without mixing. This model can be generalized from a system with the exponential RTD of a PSR to arbitrary RTDs. Zwietering (1959) generalized the completely mixed case with the Maximum Mixedness Model. He demonstrated that, while the PSR is the maximum mixing case for an exponential RTD, an RTD that reflects a

series of PSRs achieves the maximum mixing condition with his model and not with a series of PSRs. The Dankwerts and Zwietering models represent the limiting cases and "real" systems lie somewhere in between.

Spencer *et al.* (1980) showed how a reacting tracer can determine the degree of segregation. They used two intermediate models with mixing rules that produce conditions between the limiting cases of the Danckwerts and Zwietering models. The first model suggests a mixing mechanism where each "point" spends the same fraction of its time in a segregated state. The second model assumes that all fluid spends the same fixed time in a segregated state. David (1994) criticizes these phenomenological models in that they ignore the physical structure of the flow. The modeling of the structure of turbulent flow, however, creates computational demands that prevent the solution of all but the simplest thermo-kinetic mechanisms.

2.4 Modeling of Turbulent Reacting Flows

Even with the rapid increase in computational power over the last decade, research computers still do not have the needed capability to integrate detailed thermo-kinetic models with Navier-Stokes Direct Numerical Simulation (DNS) of reactive turbulent flows. If the problem involves a relatively simple geometry with only a few species whose chemical reactions do not cause significant energy changes in the flow, the simulation might be within the realm of what can be done (Oran and Boris, 2001). For example, Bédat *et al.* (1999) developed a methodology that calibrated a reduced order 4-step mechanism against the detailed GRI 2.11 mechanism with a one-dimensional laminar transport model (*PREMIX*). The reduced order

mechanism was then applied to DNS of a turbulent system operating under similar conditions to gain detailed profiles of NO production. While it is possible to model PIC production with detailed turbulent mixing resolution in limited circumstances, these techniques are not applicable, yet, as engineering tools.

To make turbulent reacting flow problems tractable, certain simplifications are made and those simplifications on based on the regime of the turbulent conditions. A useful parameter is the turbulent Damköhler number, Da_T, which is the ratio of the turbulent time scale over the flame or chemical time scale. When $Da_T > 1$, the reaction times are shorter than the mixing time. So, mixing is the rate determining step. In this regime, a useful model is the eddy-break-up model, which was first suggested by Spalding (Peters, 2001). When $Da_T < 1$, the mixing times are shorter and chemical reactions are the rate determining step. This region is often listed as the well stirred reactor region after the Borghi Diagram (Warnatz *et al.*, 1999; and Peters, 2001). In this region, stochastic approaches describing small scale mixing work well (Oran and Boris, 2001).

Probability Distribution Function (PDF) methods achieve closure by describing the chaotic turbulent process by its statistics. S. B. Pope's review article of the subject (1990) and further development in *Turbulent Flows* (2000) discusses how PDF's derived from Navier-Stokes equations have described combinations of velocity, dissipation and composition. He provides a detailed discussion of a case of constant density, homogeneous, turbulence where the composition PDF is independent of position, and presents a simple closure approximation for diffusion that was first proposed by Dopazo *et al.* (1974). This closure model relaxes local conditions to average conditions at a rate proportional to the

mixing frequency of the system. In many respects, this approach is similar to other mixing models where local composition is interchanged with surrounding conditions. The most well known of these is the coalescence and dispersion model (Curl, 1963).

Curl's model, originally developed to describe the mixing of reactants in a liquid phase, uses a pair-wise mixing rule. The model coalesces two randomly selected drops and immediately re-disperses them into two equal volumes with equal composition at a rate governed by the mixing frequency. Solution techniques employ a Monte Carlo simulation that uses an array of drops to randomly select among for coalescence. Kridiotis *et al.* (1989) modified Curl's approach by using an array of fixed cells that mixed only with adjacent cells. They successfully demonstrated an improved fit to experimental data of CO/H_2 and air combustion in a toroidal jet-stirred combustor over a perfectly stirred reactor model.

The Interaction by Exchange with the Mean (IEM) concept uses a simpler scheme of relaxing local conditions toward the mean conditions at a rate proportional to the mixing frequency, ω, that is ratio of the turbulent kinetic energy, k, to the dissipation rate, ε. Correa's (1993) application of the IEM model, termed the Partially Stirred Reactor (PaSR), demonstrated a solution method using a Monte Carlo simulation. The reactor consisted of an ensemble of N_p discrete fluid packets or particles. At each time step, one or several packets enter the reactor, and an equal number are randomly selected for removal as the effluent. He demonstrated solutions to a 18 species, 43 reaction, CO/H_2 and air system using 300 particles; a 27 species, 77 reaction, methane and air system using up to 640 particles (Correa and Braaten, 1993); and a 11 species, 23 reaction, CO/H_2 and air system using about 500 particles (Correa, 1995). These studies underscored the large amount of CPU time

required for solution. For a system with N_p particles and N_s chemical species, a total of $N_p \times (N_s + 1)$ ordinary differential equations must be solved for each time step until a solution converges. The methane mechanism required 166 hours on a Sun SPARC 1 workstation to solve. Even with greater microprocessor capability found in contemporary computers, larger thermo-kinetic mechanisms takes prohibitively large amounts of CPU time. Additional studies conducted by Cannon *et al.* (1998) explored the use of the 276 reaction GRI 2.11 mechanism for methane and air. The full mechanism required 36 hours to solve using a HP 735 workstation, but only 28 minutes when reduced to a 4-step mechanism. Consequentially, efforts have been made to reduce processing time. The two approaches to this end have been developing a systematic approach to reduced kinetic mechanisms (Norris and Pope, 1995) and developing efficient reaction look-up tables (Saxena and Pope, 1999) that reduce the demand for numeric integration.

The IEM and PaSR approaches have limitations. When turbulence lacks homogeneity, as is the usual case, dissipation has a spatial dependence, but the PaSR uses a single spatially-independent mixing frequency. Additionally, the IEM approach fails to take local composition into account, which makes this technique less useful for combustion regimes with sharp local divisions (e.g. flamelets). PDF methods to overcome these limitations include the application of combined velocity-composition-dissipation joint PDFs (Norris and Pope, 1995, and Xu and Pope, 2000) and a local mixing approach termed Euclidean minimum spanning tree (Subramaniam and Pope, 1998). These techniques are more demanding on CPU time making application detailed mechanisms such as GRI 2.11 intractable (Xu and Pope, 2000).

2.5 Summary

Research has shown the complex nature of PIC formation in practical incineration systems. Incinerators have modes which cause failure despite the presence of excess air and adequate apparent residence times. In research, detailed thermo-kinetic mechanisms have been generally applied to well defined mixing systems, whereas the modeling of combustion in complex geometries has focused on detailed turbulent modeling. The middle ground between these approaches is the classical application of ideal reactor models to practical systems. This approach is not without its inherent fault. Even research reactors specifically designed to produce ideal mixing conditions fail to completely achieve it and these deviations cannot be ignored under certain conditions. The IEM approach of the PaSR model can also bridge this gap. It makes the simplifying assumptions to show the departure from ideal mixing, but maintains computational simplicity that allows the incorporation of detailed thermo-kinetic models.

CHAPTER 3

CHARACTERIZATION OF THE SECONDARY COMBUSTION CHAMBER OF THE ROTARY KILN INCINERATOR SIMULATOR

A networked ideal reactor model preserves the mathematical simplicity of an ideal reactor while attempting to introduce some of the complexities of the mixing process. This study applies this approach to a pilot-scale system and allows the application of detailed thermo-chemical kinetic mechanisms in a computationally inexpensive mixing model. A physical characterization of the reactor was the first step toward model synthesis. Characterization incorporated the analysis of reactor geometry, temperature profiles, and the residence time distribution of tracers between various points. This chapter covers the characterization of the secondary combustion chamber (SCC) of the rotary kiln incinerator simulator (RKIS) which leads to the synthesis of a candidate model. Chapter 4 covers the final synthesis of the model and its parameter identification.

3.1 Description of the RKIS and SCC

Figure 3.1 shows a simplified diagram of the RKIS and SCC located at the Air Pollution Technology Branch, EPA in Research Triangle Park, NC (see Appendix C for a more detailed diagram). The system includes the salient features of a commercial rotary kiln system. The 73 kW (250,000 Btu/hr) SCC is attached to a 73 kW pilot-scale rotary kiln incinerator. The kiln burner is an Eclipse 82 MTVA burner and the afterburner is an ACI Pyrotron

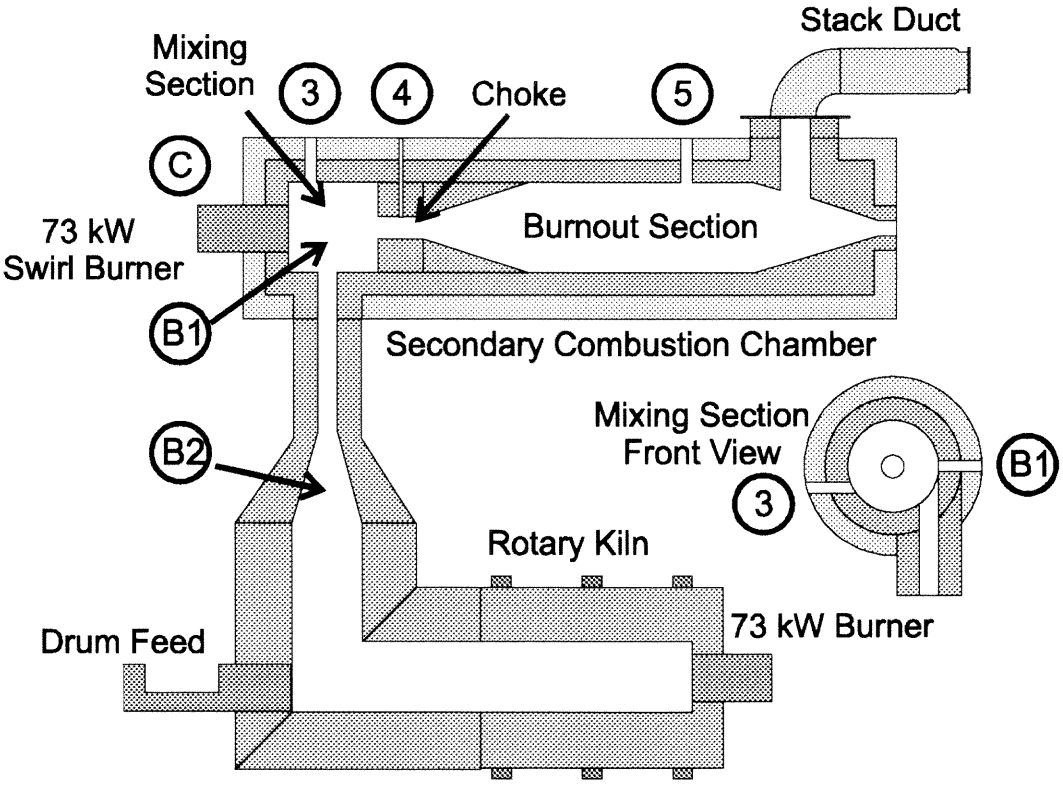


Figure 3.1 Schematic of the Rotary Kiln Incinerator Simulator (RKIS) and the Secondary Combustion Chamber (SCC)

swirl burner. Both burners use natural gas as the primary fuel. The SCC contains a removable choke which divides mixing chamber from the burnout section. During SCC experiments, liquid surrogate wastes were injected at point B2 in the kiln transition duct using a nitrogen and air atomizer. These studies included both dichloromethane (CH_2Cl_2), and tetrachloromethane (CCl_4) as the liquid surrogate wastes. The system has an on-line continuous emission monitor at point 5 for CO, CO_2, O_2, NO, and total hydrocarbon (THC). An on-line HP 5890 Series II (FID, ECD) GC, with a Tekmar Purge and Trap, can be attached to any of the sample ports. (Lemieux *et al.*, 1995). This set-up gave the ability to sample the volatile organics as well as continuously monitor the standard emissions.

The geometry of the SCC gives a starting point for model synthesis by identifying distinct mixing regions that might be modeled as ideal reactors. The two chamber design of the SCC automatically suggests at least two mixing zones divided by the choke: the mixing chamber and the burnout section. The off-radial entrance of the kiln transition duct into the mixing chamber forces a cyclone chamber-like swirl, while the SCC burner creates another swirl through adjustable vanes of the fuel and air inputs. Beer and Lee (1965) demonstrated that the recirculating zone of the swirling flame can be modeled as a well-stirred reactor followed by a plug flow reactor as the gases move downstream. However, since the mixing chamber has two sources of swirl from two separate feeds the result may be two mixing zones and possibly a third representing an entrainment zone between them. Additionally, there is potential for another mixing zone downstream of the choke as the emerging gases slow to the bulk velocity in the burnout section. Hence, the geometry of SCC suggests the existence of up to four mixing zones that might be modeled as perfectly stirred reactors (PSR).

3.2 Experimental Approach

3.2.1 Residence Time Distribution (RTD)

Residence Time Distributions (RTD) are useful in characterizing non-ideal flows. Such analysis can reveal the presence of stagnant zones (dead space) and flow bypasses while characterizing the macro-mixing zones within the reactor. However, the RTD does not fully characterize a non-ideal flow. It provides a linear response of a tracer injected into the

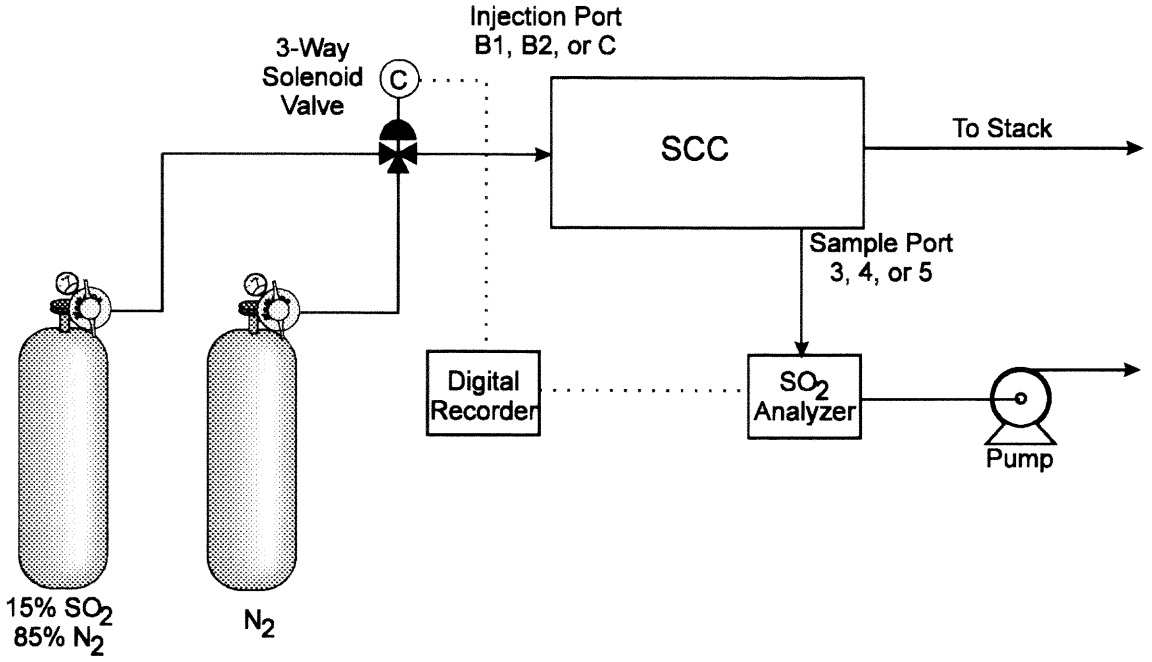


Figure 3.2 SO_2 Tracer Experimental Set-up

reactor and can be used only to directly model conversions for 1st order reactions (Levenspiel, 1972). Even though the RTD can help distinguish the size and number of in-series ideal reactors applied to model a non-ideal flow, it cannot distinguish the order of that series. Thus, a single RTD resolved from a single input/output combination would not contribute much to the characterization of the SCC. Rather, a more thorough approach necessitated the measurement of RTDs from various inputs and output combinations.

Tracer studies, using sulfur dioxide (SO_2) as the tracer gas, identified the RTD of the mixing chamber and the burnout sections of the SCC. The main kiln burner and SCC afterburner were operated at the "stoichiometric" conditions set listed in Table 6.1. These correspond to an overall fuel/air equivalence ratio, Φ, of 0.92 with 2.42×10^4 std cm^3/s (30.0 g/s) entering through the kiln transition duct and 1.28×10^4 std cm^3/s (15.8 g/s) entering

through the SCC swirl burner. Figure 3.2 shows the experimental setup. Tracer gas entered the system as step input. This allowed two data sets, a step-up and step-down, to be produced per data run. The analyzer had its own characteristic response. This was separately measured using the same sample train. Even though SO_2 reacts with oxygen to form an equilibrium with SO_3 at typical SCC operating temperatures, this equilibrium occurs rapidly with respect to the time scales being measured and should not have affected the results.

Preparation included a series of test runs and the gathering of a preliminary data set. The test runs provided several findings that impacted on the experimental set-up. For instance, tracer gas could not be detected at port 3 when inputted in C. This indicated a complete bypass. Next, these tests demonstrated the need to modify the SCC and create port 4 directly through the wall of the reactor at the choke instead of sampling with a long probe through the mixing chamber. Finally, it revealed several limitations of the SO_2 detector. This included a minimum (analyzer limiting) digital sampling interval, Δt, of approximately 0.05 seconds. Also, the range of the instrument did not give it enough sensitivity at low levels to allow the use of a pulse input of tracer gas.

With the above modifications, an initial set of data was collected for all possible input/output port combinations. Analysis of these data provided troublesome results. Particularly, the volume of the mixing chamber calculated from the RTD significantly exceeded the physical volume of the entire SCC. The source of this problem was attributed to how sample gas entered the sample train to characterize the analyzer; it entered through

the high pressure lines off the cylinders. Modifying this procedure by introducing tracer gas into the sample train at ambient pressure (near the operating pressure of the SCC) produced very consistent results.

A second data set was obtained by injecting SO_2 tracer gas into the SCC at ports B1, B2, and C, and the response measured at 3 (except C), 4, and 5, while the SCC operated at the same stoichiometric conditions used for dopant injection (Chapter 6). Each input/output combination was measured as a separate run. The analyzer step response was measured before and after each run. Table 3.1 summarizes runs the schedule of runs.

Table 3.1 SO_2 Tracer Runs

Run No.*	Input Port	Output Port	No. Sequences	Analyzer Runs
T1	B1	3	3	before/ after
T2	B1	4	3	before/ after
T3	B1	5	3	before/ after
T4	B2	3	3	before/ after
T5	B2	4	3	before/ after
T7	B2	5	3	before/ after
T8	C	4	3	before only
T10	C	5	4	before only

* Runs T6, T9, and T11, used a random input that was not analyzed

A typical run had 3 sample sequences with each sequence consisting of a step-up and step-down response. Each represents a distinct response. This set of data produced an unexpected difference between the step-up and step-down responses of the system, which should have been symmetrical. This phenomenon was not observed in previous trials of the system response and had not been observed for the analyzer trials. A noticeable data spike

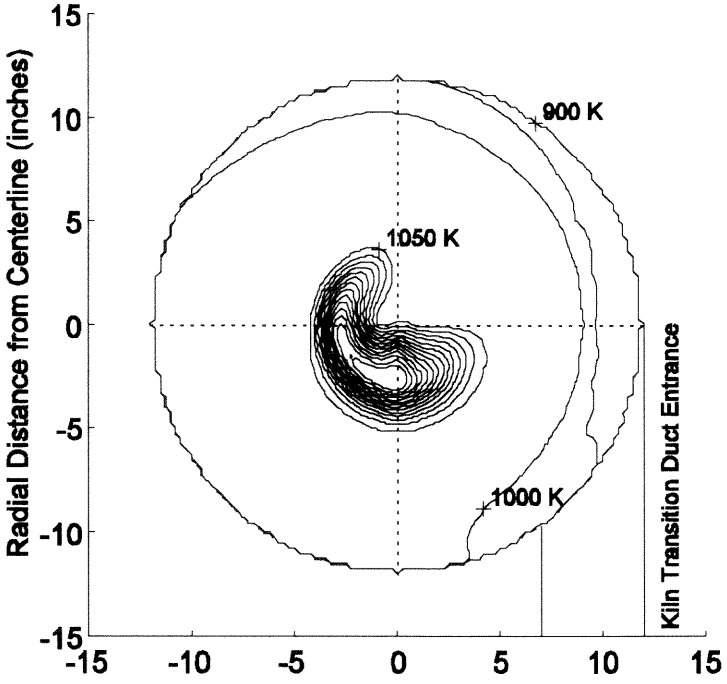


Figure 3.3 Radial Temperature Profile - SCC Mixing Chamber; Temperature Contours from 900 to 1700 K at 50 K Increments

had been observed during several system response runs during the step-up portion. The spike was attributed to the electrical signal sent to the solenoid on the three-way valve. Hence, the integrity of the system response step-up data was suspect and thus rejected.

3.2.2 Temperature Profiles

Temperature profiles were taken along the horizontal and vertical diameter midway in the mixing chamber, on the vertical diameter midway in the burnout section, and midway on the vertical diameter in the choke (see Figure 3.1). The experimenters used a suction pyrometer for the mixing chamber and burnout section temperatures and a type R thermocouple in the choke. The temperature was recorded every inch.

Figure 3.3 shows a rendered contour temperature profile in the mixing chamber. This plot was produced from the data by first linearly interpolating points every 15 degrees along the arcs connecting equal radius data points on adjacent axes, then rendering the contour with the MATLAB® contour function. Interpolating along the arcs followed the logic of the cylindrical geometry and the swirling flow in the mixing chamber.

The rendering shows the presence of a relatively tight flame region and suggests the presence of at least two distinct mixing zones in the chamber. The alignment of the kiln transition duct is depicted on the graph. This graph displays the effect of the cooler gas entering the chamber on the right and the relatively small flame zone of the swirl burner near the axial centerline. Using the 1050 K contour as an approximation of the flame boundary, the flame cross section occupies 10.2 percent of the reactor cross-section area and has an average (weighted on the heat capacity of N_2) temperature of 1343 K. Using the same weighting technique, the reactor cross-section has the average temperature of 1009 K.

Figure 3.4 shows the vertical temperature profiles in the choke and the burnout section. Since the choke measurement did not use a suction pyrometer, the temperatures shown on the graph have been corrected for radiation heat loss,[1] which yields gas temperatures in the center of the choke about 90 K higher than the thermocouple temperature. This graph indicates significant temperature difference (300 K) between the gas near the lower choke wall and the gas just above the center in the choke, which reveals the presence of two distinct, but converging, temperature zones. Since the choke wall heavily

[1]Correction is determined from an energy balance between the radiative heat loss from the thermocouple bead to the cooler reactor wall, and the heat transfer across a boundary layer estimated by a Nusselt number–Reynolds number correlation (Fristrom, 1995).

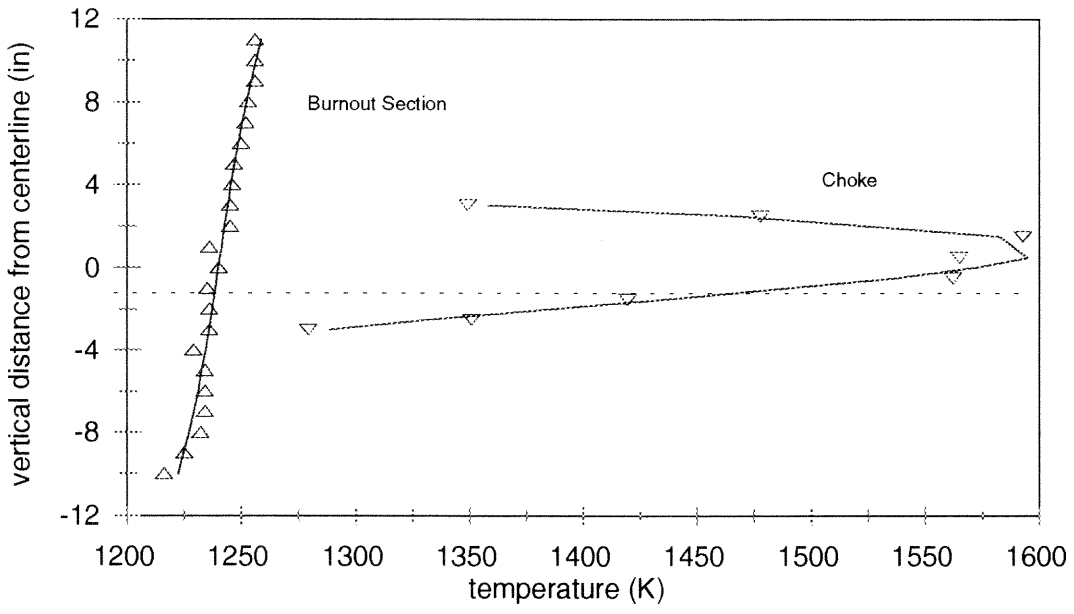


Figure 3.4 Vertical Temperature Profile in the Choke and Burnout Section

insulates the passage and it is at thermal equilibrium, heat loss through the walls would not account for this behavior. Downstream in the burnout section, the temperature profile is approximately even with a slight degree of buoyant stratification. This suggests that streams converge past the choke and should be modeled as a single homogenous (mixed) stream. Interpolating along the arcs connecting temperature measurements at equal radii above and below the centerline produced a cross-section contour for the choke and burnout section similar to Figure 3.3. From these graphs (see Appendix C), the average cross-section temperatures for the choke and the burnout section (at point 5) are 1213 K and 967 K, respectively.

3.3 Least Square Identification of the Residence Time Distribution

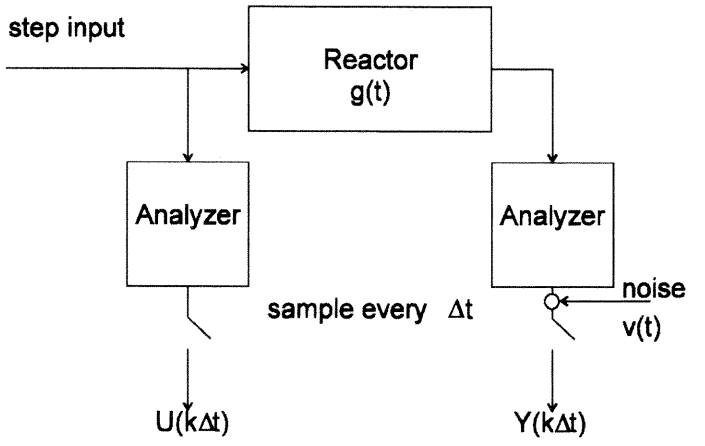
3.3.1 Modeled System

Identifying a system involves first modeling the fundamental characteristics of that system, based on observations and assumptions, then identifying the system as it pertains to that model. The SCC will be modeled based on the assumption that the reactor behaves as a linear system for the passage of an inert tracer. In other words, the passage of the tracer can be described with a set of linear first order differential equations, a condition that applies to a reactor describable by a network of ideal reactors, and, when normalized, it does not depend on the initial concentration of the tracer. Unless the tracer is consumed or adsorbed, this assumption normally holds true.

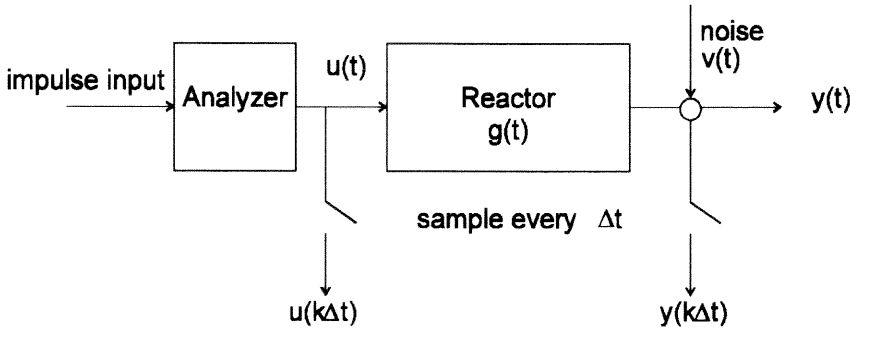
A linear system that has a Single Input and a Single Output (SISO), such as the tracer injected at a single point and measured at a single point downstream, can be described by a single transfer function, $g(t)$. The output of the system, $y(t)$, is a convolution of the input, $u(t)$, with the transfer function.

$$y(t) = \int_0^t g(t-\tau)\,u(t)\,d\tau \tag{3.1}$$

For the tracer experiments, the system represents both the effects of the reactor and the analyzer on the distribution of the tracer (Figure 3.5a). As a linear SISO system, the order of the reactor and analyzer do not matter; the effect of the analyzer on the tracer can be considered either before or after the reactor. This provides an alternate view where the



a. Physical system



b. Alternative view assuming linearality

Figure 3.5 Modeled System

output of the analyzer can be considered as the input of the reactor. Using this view, the **analyzer step response** is the input, $U(t)$, and the **analyzer (impulse) response** will be represented as $u(t)$. Only the analyzer step response is directly measured. However, differentiating the analyzer step response with respect to time gives the analyzer impulse response, which will be referred to simply as the analyzer response (Levenspeil, 1972).

$$u(t) = \frac{dU(t)}{dt} \tag{3.2}$$

In the same fashion, differentiating the **system step response**, *Y(t)*, gives the **system response**, *y(t)*. The system response is the combined response of tracer through the analyzer and the reactor with the addition of any uncorrelated noise external to the analyzer or reactor, *v(t)* (Figure 3.5b). Equation 3.3 shows this relationship for a continuous system.

$$y(t) = \int_0^t g(t-\tau)\,u(t)\,d\tau + v(t) \tag{3.3}$$

3.3.2 Least Squares Estimation

The transfer function, *g(t)*, in the modeled system (Figure 3.5b) is also the residence time distribution (RTD). This can be identified from the input, *U(t)*, and output, *Y(t)*. Hsia (1977) provides a suitable technique which identifies this function as the estimator of a least squares formula. This method computes the auto-correlation (input) and cross-correlation (input vs. output) functions in the process, so the input signal must be stationary. That is, the expected value is invariant with time (Hsia, 1977). The step input function does not meet this criterion. However, converting the step input and response function to an impulse input and response by using the first derivative (Levenspiel, 1972) makes it a quasi-stationary process. A Savitsky-Golay (Press *et al.*, 1992) filter that fits a 2nd order polynomial to a window 10 points on each side of a given point estimates the derivative by using the polynomial coefficients estimated at each point.

The system has a continuous input and output, but it is sampled on a discrete time interval, Δt. When Δt is small, relative to the system response, a discrete system approximation becomes a valid approach. Equation 3.3 approximates a discrete system by defining the weighting function, h, as

$$h(k\Delta t) = \Delta t\, g(k\Delta t) \tag{3.4}$$

where k is the sample index (where $t = k\Delta t$ for $k = 1, 2, 3 \ldots$). Equation 3.5 gives the discrete version of Equation 3.3.

$$y(k\Delta t) = \sum_{i=k-p}^{k} h(k\Delta t - i\Delta t)\, u(i\Delta t) + v(k\Delta t) \tag{3.5}$$

Here, p is an integer such that $g(t) \approx 0$ for $t \geq p\Delta t$ and $0 \leq k \leq p$. The discrete system response, $y(k\Delta t)$, depends on the weighted discrete analyzer response, $u(i\Delta t)$, for the preceding p measurements or $i = k\text{-}p, k\text{-}p+1, \ldots, k$.

Despite the fact that the experiment measured a single system response to a single input, it is necessary, for this method, to consider the experiment as a periodic series of responses to a periodic series of inputs where the system has time to come to rest after each input. The system response is measured for an interval between 0 and p where p is chosen sufficiently large such that $y(t) \approx 0$ for $t \geq p\Delta t$. Equation 3.5 can now be expressed in matrix form

$$\mathbf{y} = \mathbf{Uh} + \mathbf{v} \tag{3.6}$$

where

$$\begin{aligned}\mathbf{y} &= [y(0) \quad y(1) \quad \ldots \quad y(p)]' \\ \mathbf{v} &= [v(0) \quad v(1) \quad \ldots \quad v(p)]' \\ \mathbf{h} &= [h(0) \quad h(1) \quad \ldots \quad h(p)]'\end{aligned}$$

$$\mathbf{U} = \begin{bmatrix} u(0) & u(-1) & \ldots & u(-p) \\ u(1) & u(0) & \ldots & u(1-p) \\ \vdots & \vdots & \ddots & \vdots \\ u(p) & u(p-1) & \ldots & u(0) \end{bmatrix} \tag{3.7}$$

Applying the matrix form of the linear least squares formula (see Draper and Smith, 1998) to Equation 3.6 gives the estimator of **h** in Equation 3.8. Equation 3.8 is the estimate of the **h** vector that minimizes the sum of the squared errors, **v**′**v** (Note: the prime denotes the transpose).

$$\hat{\mathrm{h}} = (\mathbf{U}'\mathbf{U})^{-1}\mathbf{U}'\mathbf{y} = \mathbf{\Phi}^{-1}\mathbf{\Gamma} \tag{3.8}$$

Here, **Φ** and **Γ** represent the groupings that relate to the autocorrelation and cross-correlation vectors, respectively, shown in Equations 3.9.

$$\mathbf{\Phi} = \frac{1}{1+p}\mathbf{U}'\mathbf{U} = \begin{bmatrix} \varphi_{uu}(0) & \varphi_{uu}(1) & \ldots & \varphi_{uu}(p) \\ \varphi_{uu}(1) & \varphi_{uu}(0) & \ldots & \varphi_{uu}(p-1) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{uu}(p) & \varphi_{uu}(p-1) & \ldots & \varphi_{uu}(0) \end{bmatrix} \tag{3.9a}$$

$$\mathbf{\Gamma} = \frac{1}{1+p}\mathbf{U}'\mathbf{y} = \begin{bmatrix} \varphi_{uy}(0) \\ \varphi_{uy}(1) \\ \vdots \\ \varphi_{uy}(p) \end{bmatrix}$$

The autocorrelation function of the inputs **u** and cross-correlation function of the inputs **u** with the outputs **y** in the discrete form are defined below.

$$\varphi_{uu}(k) = \frac{1}{1+p}\sum_{j=0}^{p} u(j)u(j+k)$$
$$\varphi_{uy}(k) = \frac{1}{1+p}\sum_{j=0}^{p} u(j+k)y(j) \qquad (3.9b)$$

Using this least squares approach eliminates the uncorrelated noise, $v(k\Delta t)$, from the solution. This has advantages over other deconvolution techniques, such as the application of Fourier transforms, which can algebraically inverse Equation 3.3 in frequency-space, but accentuate the effects of the noise. Instead, the auto and cross correlations reduce the uncorrelated noise.

A problem arises, however, when finding the inverse of $\mathbf{\Phi}$. Because of reasons that will be elaborated in Chapter 4, $\mathbf{\Phi}$ is singular or near singular – it is rank deficient, and thus, has a determinate near or equal to zero. Hence, $\mathbf{\Phi}^{-1}$ as such does not exist and cannot be determined by standard techniques such as Gaussian elimination. This also means that $\hat{\mathbf{h}}$ will not have a unique solution (Draper and Smith, 1998). Lack of a unique solution has implications for model selection which will be discussed in the next chapter. Rather than dwell on the implications here, it is sufficient to select an approach to solve for $\hat{\mathbf{h}}$ without directly computing $\mathbf{\Phi}^{-1}$ and apply that approach consistently.

A constrained iterative reconstruction technique was chosen to find $\hat{\mathbf{h}}$. Schafer *et al.* (1981) presented a class of constrained discrete iterative deconvolution methods that are applicable to this problem. This iterative approach works by multiplying the auto-correlation

matrix, $\mathbf{\Phi}$, with an initial guess of $\hat{\mathbf{h}}$, to give an estimated cross-correlation vector, $\mathbf{\Gamma}$. Since $\mathbf{\Gamma}$ is already known, the error between the known and estimated $\mathbf{\Gamma}$ may be used to find the next guess of $\hat{\mathbf{h}}$. To show this symbolically, Equation 3.8 is rearranged to show $\mathbf{\Gamma}$ as the product,

$$\mathbf{\Gamma}^{(0)} = \mathbf{\Phi}\hat{\mathbf{h}} \tag{3.10}$$

where the parenthetical zero superscript represents the known $\mathbf{\Gamma}$. Henceforth, parenthetical **n** superscripts of integers 1 or greater represent successive iterations. These iterations follow the sequence:

1. Make an initial guess of $\hat{\mathbf{h}}^{(1)} = \mathbf{\Gamma}^{(0)}$
2. First iteration n = 1
3. Calculate new $\mathbf{\Gamma}^{(n)} = \mathbf{\Phi}\,\hat{\mathbf{h}}^{(n)}$
4. Error $\mathbf{e} = \mathbf{\Gamma}^{(0)} - \mathbf{\Gamma}^{(n)}$
5. $\mathbf{Z} = \hat{\mathbf{h}}^{(n)} + \lambda\mathbf{e}$ where $1 \le \lambda \le 2$
6. $\hat{\mathbf{h}}^{(n+1)}$ = constraints($\mathbf{Z}$) : $Z_k = 0$ when $Z_k < 0$ for all k
7. n = n + 1
8. goto step 3; repeat until error converges

For this problem, a λ with a chosen value of 1 resulted in a stable convergence. When applying constraints in step 6, $\hat{\mathbf{h}}$ was normalized such that $\sum_{i=1}^{p} \hat{h}_i = 1$.

3.3.3 Least Squares Results

Each tracer run had at least three data step-down system response and analyzer response sequences (see Table 3.1). To use these data in the least squares reconstruction of the RTD, the information from each of these samples was normalized, combined, differentiated, and re-

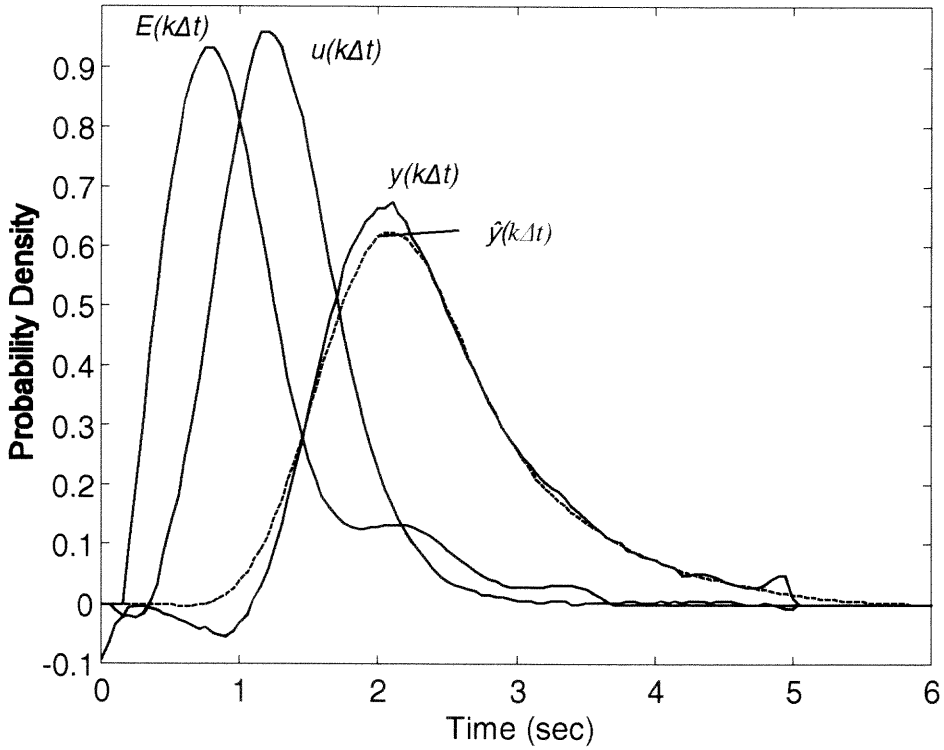


Figure 3.6 LS Reconstructed RTD, $E(k\Delta t)$, from System Response, $y(k\Delta t)$, and Analyzer Response, $u(k\Delta t)$; Estimated System Response, $\hat{y}(k\Delta t)$, Shown for Comparison

sampled at a constant interval, Δt, for the input function, $u(k\Delta t)$, and the response function, $y(k\Delta t)$. Each run was normalized and numerically converted to a step-up function. The sampling interval for the raw data was approximately 0.05 seconds and the interval varied slightly between sequences. So, the combination of data from each sample run sequence (either system or analyzer step response) resulted in an inconsistent, non-uniform, interval, but the numerical techniques covered in section 3.3.2 require a uniform Δt. The application of the Savitsky-Golay filter in the differentiating process solved this problem. The algorithm uses an imposed time array at a 0.05 second interval and fits a second order polynomial to the data points within a 0.5 second window of each point on the imposed time sequence.

The parameters of the polynomial give the derivative for each point on the interval and, in essence, combined the data from each sample within the tracer run. Additionally, the data points within a 0.05 second interval about each point on the imposed time sequence were used to determine the local data variance (after correction for the local slope), which was applied in the error analysis of model parameter estimation (Chapter 4).

The graph in Figure 3.6 shows the normalized responses and transfer function for run T2. $E(k\Delta t)$ is the least squares reconstruction estimate of the transfer function, $\hat{g}(k\Delta t)$, equal to $\hat{h}(k\Delta t)/\Delta t$ (see Equations 3.4 and 3.5). As noted earlier in section 3.3.2 the singularity of $\mathbf{\Phi}$ allows for other solutions of $E(k\Delta t)$ that meet the least square criterion and produce the same estimate of the system response function, $\hat{y}(k\Delta t)$, when convoluted by $u(k\Delta t)$ as shown in Equation 3.11.

$$\hat{\mathbf{y}} = \mathbf{U}\hat{\mathbf{h}} \tag{3.11}$$

The differences within the set of possible solutions are removed in the convolution process and cannot be reconstructed. As Chapter 4 will elaborate, $u(k\Delta t)$ removes the higher frequencies of $g(k\Delta t)$ when viewed in a frequency domain. Thus, solutions of $E(k\Delta t)$ can differ in the higher frequencies, but produce the same $\hat{y}(k\Delta t)$ when Equation 3.11 is applied. The constrained iterative reconstruction technique starts with a guess that does not have these higher frequencies, so it consistently produces the "safest" solution that does not assume

their existence. This results in a consistent estimate, but lacks the higher frequencies. So, any sharp peaks of the "true" $g(k\Delta t)$ will appear smeared, that is, slightly flattened and spread to both the left and right.

3.3.3.1 Analysis of Moments. The first three moments of the RTD are often used to characterize the system. These are the mean, variance, and skewness coefficient shown below in their discrete form (Levenspiel, 1972) and (Fogler, 1992).

$$\mu = \frac{\sum_{i=0}^{p} E_i t_i}{\sum_{i=0}^{p} E_i}, \qquad \sigma^2 = \frac{\sum_{i=0}^{p} E_i t_i^2}{\sum_{i=0}^{p} E_i} - \mu^2, \qquad s^3 = \frac{1}{\sigma^3}\left(\frac{\sum_{i=0}^{p} E_i t_i^3}{\sum_{i=0}^{p} E_i} - 3\mu\sigma^2 - \mu^3 \right) \quad (3.12)$$

Before discussing the moments calculated from the experimental results, it is useful to consider them in the context of ideal reactor models. Consider three archetype ideal reactor models: (1.) PSR and PFR in series with mean resident times τ_1 and τ_d respectively; (2.) two different size PSRs in series with mean residence times τ_1 and τ_2; and (3.) n equal PSRs in series, each with mean residence time τ_i. Table 3.2 shows the RTD formulas and resulting moments, derived from the continuous versions of Equation 3.12.

From these archetypes, several deductions may be made. First, for a simple series with no recirculation, the PFR makes no contribution to the variance and skewness coefficient, and PSR mean residence time for archetype 1 is better measured through the variance. Next, the skewness coefficient has the value of 2 for a single PSR in that series and becomes smaller as more PSRs are added. Finally, for the series of equal sized PSRs

Table 3.2 Moments of Model Archetypes

	RTD	mean (μ)	variance (σ^2)	skewness (s^3)
1.	$E(t) = \dfrac{e^{(\tau_d - t)/\tau_1}}{\tau_1} \quad t \geq \tau_d$	$\tau_d + \tau_1$	τ_1^2	2
2.	$E(t) = \dfrac{\left(e^{-t/\tau_1} - e^{-t/\tau_2}\right)}{(\tau_1 - \tau_2)}$	$\tau_1 + \tau_2$	$\tau_1^2 + \tau_2^2$	$\dfrac{2(\tau_1^3 + \tau_2^3)}{(\tau_1^2 + \tau_2^2)^{3/2}}$
3.	$E(t) = \dfrac{t^{(n-1)}}{(n-1)!\tau_i^n} e^{-t/\tau_i}$	$n\tau_i$	$n\tau_i^2$	$\dfrac{2}{\sqrt{n}}$

(archetype 3), the number of reactors in the series can be deduced from the skewness coefficient, but since higher moments heavily weight perturbations farthest from the mean, and thus, are less reliable, a better deduction can be made from the first two moments (Fogler, 1992):

$$n = \frac{\mu^2}{\sigma^2} = \frac{(n\tau_i)^2}{n\tau_i^2} \tag{3.13}$$

Table 3.3 lists the results of these moment calculations for the runs in Table 3.1. These show consistency with the physical configuration shown in Figure 3.1. As the tracer was collected from sampling ports downstream, the mean residence time as well as the variance increased. The variance reveals the nature of mixing, and, for an ideal reactor series, should be additive from one section of the reactor to the other, as demonstrated with archetype 2. From this fact, variance should increase when sampled further downstream, which was observed. However, the variances measured for tracer runs from input port B2

were in two cases slightly smaller than those measured from the runs starting at port B1. One would expect some degree of mixing to occur in the kiln transition duct which makes this result counter-intuitive; at a given output port, the variances resulting from the B2 input should be greater than or equal to those measured from the B1 input. This can be explained by the lower degree of accuracy in the higher order moments. Higher moments weight measurements furthest from the mean, which are most susceptible to noise and the least accurate.

Table 3.3 Moments from Tracer Runs

run	in/out port	mean (sec)	variance (sec^2)	skewness coeff.
average	analyzer	1.30	0.21	1.67
T1	B1/3	0.56	0.32	2.57
T2	B1/4	1.08	0.40	1.45
T3	B1/5	3.59	2.13	0.68
T4	B2/3	0.86	0.26	1.36
T5	B2/4	1.38	0.60	1.12
T7	B2/5	3.99	2.00	0.68
T8	C/4	0.31	0.17	2.03
T10	C/5	3.60	1.71	0.76

Despite its questionable accuracy, the skewness coefficient reveals a trend. In a multiple PSR model the skewness coefficient decreases with the number of reactors (see Table 3.2). The analysis of the tracer data revealed the same trend. As the sample point moved down stream away from the injection point the skewness coefficient decreased indicating additional (PSR) mixing zones. The RTD of a PSR gives a skewness coefficient of 2 in contrast to a PFR (with or without gaussian diffusion) which has a skewness coefficient

of zero. For ideal reactor models, the RTD of PSRs in series approach that of a PFR as more reactors are added (Levenspiel, 1972). The RTDs from B1 to 3 and C to 4 have a skewness near 2 indicating a distinct single mixing region for each of these streams. Using the skewness coefficient formula derived for multiple reactors (Table 3.2) and solving for n gives:

$$n = \left(\frac{2}{s^3}\right); \text{ where } s^3 \text{ is the skewness coefficient.} \qquad (3.14)$$

Using this formula, $n = 0.61$ for B1 to 3 and $n = 0.97$ for C to 4. This is further reinforced by applying Equation 3.13 for multiple PSRs in series, which gives $n = 0.98$ for B1 to 3 and $n = 0.57$ for C to 4. Since relying on higher moments is subject to error, no single tool is absolute. However, the combination of both of these approaches reinforces a conclusion of a single mixing area between B1 and 4 and another between C and 4. This also suggests that very little, if any, of the afterburner gas (C to 4) contributes to a postulated entrainment area discussed in section 3.1.

The B1 to 4 RTD has a smaller skewness coefficient that could indicate the presence of an additional mixing region for the kiln gas stream prior to the choke. Applying Equation 3.13 gives $n = 2.9$, and applying Equation 3.14 gives $n = 1.9$. This reinforces the assertion of an additional mixing zone after point 3 and prior to the choke. The skewness coefficients for the C to 5 and B1 to 5 RTDs are much smaller which supports a model of two or three additional PSRs between the choke and point 5.

3.3.3.2 Reactor Volumes. As a final check for the consistency of the results, a quick comparison was made between the volumes measured by the RTDs and the physical volume of the SCC. Using the temperature profile in the mixing chamber (Figure 3.2) and assuming two distinct constant temperature streams, one for the kiln gas (900 K) and one for the afterburner (1314 K), the volumes were estimated to be 78 L and 35 L respectively within the mixing section. This leaves a dead volume of 65 L in the 178 L mixing section. The long tail of the RTD, $E(k\Delta t)$ for B1 to 4 (shown in Figure 3.6), is characteristic of this dead space (Himmelblau and Bischoff, 1968). Most of this dead space is probably adjacent to the wall near the choke since it is not in the flow path going out of the mixing chamber and through the choke.

3.4 Summary

The geometry of the SCC suggests the existence of up to four distinct mixing zones (flame zone, kiln gas zone, entrainment zone, and post choke mixing zone), that is, four distinct zones which could be modeled as PSRs. The temperature profiles support this conjecture by showing the presence of two distinct zones in the mixing chamber, and two emerging streams from the choke that rapidly mix as they move downstream. The RTDs, which were resolved from the tracer studies, further support a multiple PSR model. From tracer injections at B1, B2 and C, the skewness coefficient decreased as the sampling point moved downstream. This was a result of an RTD more symmetrical about its mean which follows the multiple PSR model (Levenspiel, 1972).

The tracer results validate the supposition that the flame and the kiln gas maintain two distinct zones in the mixing chamber. This has important consequences for secondary combustion chamber designs that rely on the tangential mixing of the kiln gas. Little mixing between the two streams occurs until further downstream. Such designs, are unlikely to actually meet requisite retention time - temperature requirements, even though averaged (assumed mixed) conditions are within recommended standards. This analysis seem to suggest that some turbulence producing obstruction, like a choke or bluff body, is essential to ensue complete mixing.

CHAPTER 4

AN IDEAL REACTOR MODEL

Chapter 3 described the characteristics of the Secondary Combustion Chamber (SCC) in terms of geometry, temperature profiles and residence distribution time (RTD). This chapter discusses the development of a networked ideal reactor model that is consistent with these characteristics and fits the parameters to that model. The modeling approach makes reasonable approximations of the actual flow characteristics in order to allow for the incorporation of detailed reaction mechanisms. Additionally, the extent and detail of physical measurements limits model detail. This chapter deals with the issue of making a reasonable approximation that is consistent with observations and provides a good fit to the tracer data without introducing unsupportable or physically inconsistent complexities. Since this issue introduces a degree of subjectivity to model selection, the goals and constraints of this approach must be developed to provide for a measure of merit for model selection.

Developing and fitting an ideal reactor network employs a *mechanistic* modeling approach, as opposed to an *empirical* modeling approach (Draper and Smith, 1998). It uses assumptions based on a mathematical description of ideal reactors along with observations on geometry, and temperature profiles within the reactor, rather than fitting a suitably large polynomial to the tracer data. Unlike mechanistic modeling approaches that attempt to replicate fundamental mixing behavior, an ideal reactor approach begins with a simplifying approximation. It leads only to an approximation of the "true" mixing characteristics.

An empirical approach, on the other hand, makes no assumptions about the mixing process and does not attempt to describe it. Instead, it simply finds an equation and parameter set, usually a sufficiently large polynomial, that faithfully reproduces the data. The goal of a mechanistic approach, in this work, is not to resolve the true mixing characteristics of the reactor, but to find a networked ideal reactor model that is consistent with observation and approximates the mixing characteristics for application of the detailed reaction model.

The number of tracer run combinations limited the complexity of the network model. The model was fit against the data from five different tracer combinations. Each combination consist of a single-input/single-output (SISO) between one of the input points, B1 or C, and an output point 3, 4 or 5 (see Figure 3.1). Tracer data from an SISO system can resolve the flow parameters of the ideal reactors between points, but cannot resolve the order of ideal reactors in a series. For instance, a model with a perfectly stirred reactor (PSR) followed by a plug flow reactor (PFR) has the same RTD as the reverse sequence, but the sequence clearly matters for modeling a complex reaction (Fogler, 1992). If the model had several PSRs in series the tracer data could have resolved the mean residence time (the flow parameter) for each ideal reactor, but not the order of reactors. So it would be fruitless to have multiple differently sized PSRs in a series without having other evidence to distinguish the order of the different sizes.

Other approaches to ideal reactor model development have employed a combination of reactor geometry analysis and studies of temperature profiles, in combination with the application of empirical formulas that describe the turbulent discharge of a fluid from a nozzle. Swithenbank, *et al.* (1972) relied primarily on reactor geometry when developing a

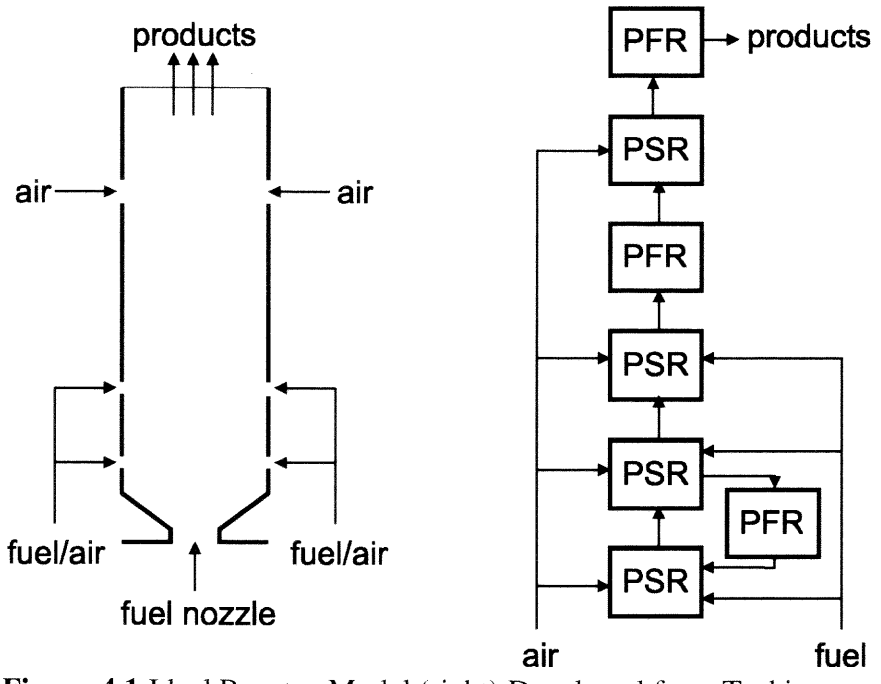


Figure 4.1 Ideal Reactor Model (right) Developed from Turbine Can Combustor (left) (Swithenbank, *et al.*, 1972)

12-parameter ideal reactor model for a turbine combustor. The fuel-rich flame originated from a swirl nozzle and was successively augmented and diluted downstream from three sets of holes. The authors developed the model from the physical volumes between the fuel and air inlets to develop the ideal reactor network that was subsequently refined using the cross-section temperature profiles and cold flow analysis (Figure 4.1). They successfully used the model to predict the rich blow-off limits of the combustor. Ewan, *et al.* (1984) extended this study by measuring the RTDs between various points in the can combustor in a cold-flow mode which favorably compared with RTDs generated from computational fluid dynamic (CFD) models. The authors employed a technique that used a spark on mercury amalgam to generate a vapor pulse, which was triggered and measured at various points in the combustor.

The data were used to identify the model parameters in each of the zones of the combustor. These techniques taken together show a good modeling approach which uses the information available from reactor geometry, temperature profiles and tracer studies.

This chapter proposes a "base model" from the inferences developed in Chapter 3, which takes reactor geometry, temperature profiles, and RTD into account. From the base model, other candidate models were developed to improve the goodness of fit to the tracer data while maintaining consistency with temperature observations and reactor geometry. Selection of the best model was based on a subjective analysis of the model residuals, the parameter standard error, and agreement with the RTD moments derived in Chapter 3. To stay within the limitations of the data, merit was also placed on the simplicity of the model as a final selection criterion between otherwise equal candidates.

4.1 PFR and PSR Models

Ideal reactor networks use plug flow reactors (PFR) and perfectly stirred reactors (PSR) as their building blocks. Within these networks, PSRs and PFRs are distinct entities with a single premixed feed and a single effluent stream. Micro-mixing in a PSR is assumed to be perfect with no mixing time. Concentrations of all species and temperature are homogenous throughout the reactor. In contrast, a PFR has no mixing. A homogenous plug of infinitesimal thickness moves down the reactor without mixing or distorting in the axial direction. These ideal reactors portray the two limiting conditions of micro-mixing.

The material and energy equations for the PSR and PFR are well known. They are presented here to introduce the form and notation that will be further developed in Chapter 5. Equation 4.1 (Glarborg *et al.*, 1986) is the mass balance for a PSR which is presented in the notation common to CHEMKIN (Kee *et al.*, 1993) and its associated programs. The equation assumes steady state conditions of the reactor with mass fractions of each species and temperature remaining constant over time. The change in the mass fraction, Y, of the k^{th} species with respect to time, equal to zero, is given by

$$\frac{dY_k}{dt} = -\frac{1}{\tau}(Y_k - Y_k^*) + \frac{\dot{w}_k W_k}{\rho} = 0 \qquad (4.1)$$

where W_k, and $\dot{w}_k$ represent molar mass and molar production rate of the k^{th} species, respectively, and the asterisk superscript denotes the input stream conditions. τ is the mean residence time and ρ is the density. This shows the balance between convective mass transfer, the observed differences between feed and effluent, and the production rate of each species. The energy balance (Equation 4.2) follows in a similar form and shows the balance between convective heat transfer of the effluent stream, heat of reaction, and conduction through the walls of the reactor.

$$c_p \frac{dT}{dt} = \frac{1}{\tau}\sum_{k=1}^{K} Y_k^*(h_k^* - h_k) - \sum_{k=1}^{K} \frac{h_k \dot{w}_k W_k}{\rho} - \frac{Q}{\rho V} = 0 \qquad (4.2)$$

Here, h_k is the specific enthalpy of the k^{th} species. Other variables are: the reactor external heat loss, Q; reactor volume, V; mass weighted average specific heat, c_p, and reaction temperature, T. Since $\dot{w}_k$ is a function of the concentration of one or more species and

temperature, Equations 4.1 and 4.2 represent a set of $k + 1$ nonlinear equations, which may be solved by a number of techniques. A Sandia distributed *CHEMKIN* driver program titled *PSR* (Glarborg *et al.*, 1986) solves these equations.

Using the same notation, the PFR mass balance describes the change of a single fluid element moving through the reactor. Equation 4.3 shows the change of the mass fraction of species k with respect to time as a function of local production rate:

$$\frac{dY_k}{dt} = \frac{\dot{w}_k W_k}{\rho} \tag{4.3}$$

Since the fluid packet is not gaining or losing mass, there is no convective term. Likewise, the energy balance describes the change in enthalpy as a function of heat of reaction production and external conduction:

$$c_p \frac{dT}{dt} = - \sum_{k=1}^{K} \frac{h_k \dot{w}_k W_k}{\rho} - \frac{Q}{\rho V} \tag{4.4}$$

Taking Equations 4.3 and 4.4 together gives $k + 1$ nonlinear ordinary differential equations (ODEs). When the conditions of the input stream are known, the PFR mass and energy balances are solved as initial value ODEs with time as the independent variable. These ODEs are numerically stiff, which occurs when the dependent variables (mass fractions) change on two or more different scales of the independent variable (Press *et al.*, 1992). Such conditions are typical of combustion reaction models that include both very rapid and relatively slow reactions (Warnatz, Maas and Dibble, 1999). Finding the solution requires a program with a robust ODE algorithm to maintain both the accuracy and stability. One such program is *DVODE* (Brown, Hindmarsh, and Byrne, 1989) that is distributed with the *CHEMKIN*

package. The *CHEMKIN* library subroutines and interpreter program allow the use of standard chemical kinetic and thermodynamic input files and provide the ODE parameters at run-time.

4.2 Model Synthesis

An ideal reactor model should contain as few fitted parameters as possible while remaining consistent with observation. The goal is to make a general approximation of flow and mixing dynamics, but not an exact representation. In this way, the model will represent the approximate temperature-time history of the fluid traveling through the SCC while maintaining the simplicity of ideal reactors. An ideal reactor model accommodates detailed chemical kinetic mechanisms without undue computational cost.

The SCC (see Figure 3.1) is physically divided by a choke into two volumes: the mixing section and the burnout section. It has two separate streams entering the mixing chamber: fuel and air entering axially in the burner stream, and effluent gas from the rotary kiln entering tangentially in the kiln gas stream, which carries about twice the mass of the axial stream. Figure 4.2 shows four distinct mixing zones that arise based on the observations and analysis developed in Chapter 3. Zone 1 constitutes the recirculating flame present in a swirl burner. This behavior is well documented by Beér and Chigier (1972) and modeled as a well stirred reactor by Beér and Lee (1965). The cross-section temperature profile (Figure 3.3) further validates treating the flame zone as a distinct region. Zone 2 represents the swirling flow created by the tangential entry of the kiln gas into the mixing section (Syred and Beér, 1974). The geometry of the mixing chamber affects the confluence of these two

regions. Forcing both flows through a narrow choke should create an entrainment region in the mixing section (zone 3). The temperature profile at the choke (Figure 3.4) shows a distinctive hot center that is most likely due to a direct bypass from zone 1. The tangential mixing zone is likely to include dead volume along the refractory wall abutted against the choke wall. This was deduced from initial placements of trace experiment sampling probes which found little to no trace when sampled from that location. Another recirculating zone (zone 4) is likely as the swirling gases emerge from the choke.

The mixing section resembles a tangentially entry swirl burner (Syred and Beér, 1974; Beér and Chigier, 1972; and Hallett, 1986) feeding the burnout section. From analysis of the mixing chamber geometry, the tangential entry of the kiln gas creates a large axial flux of

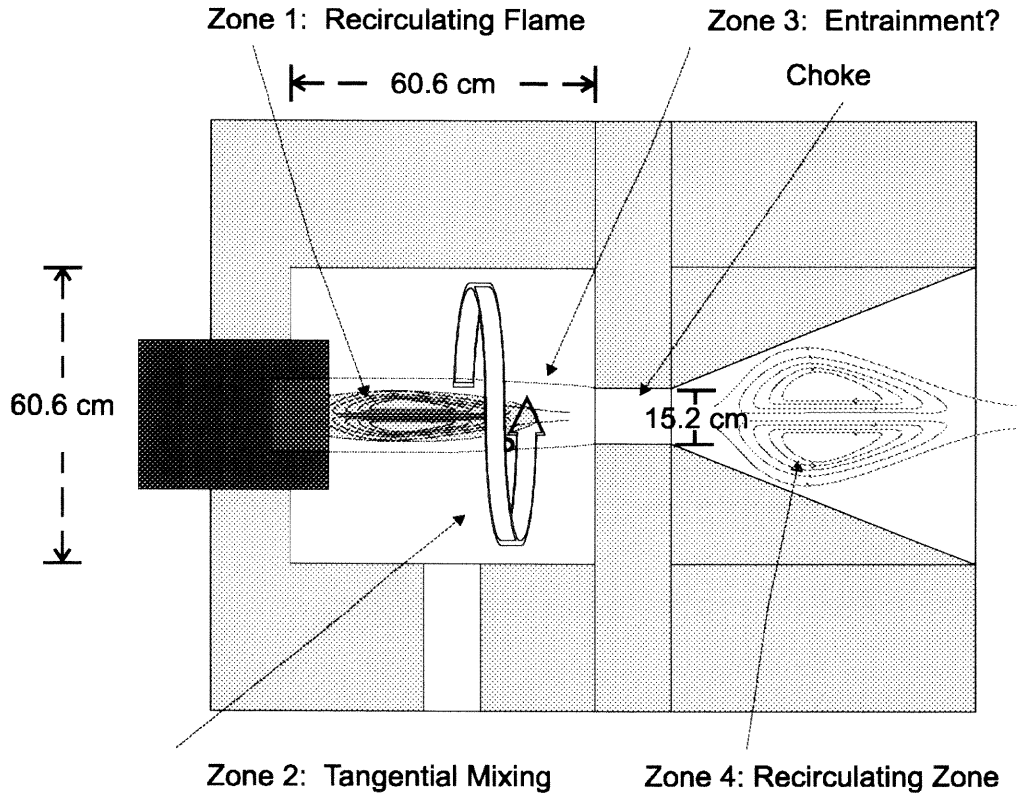


Figure 4.2 Mixing Zones

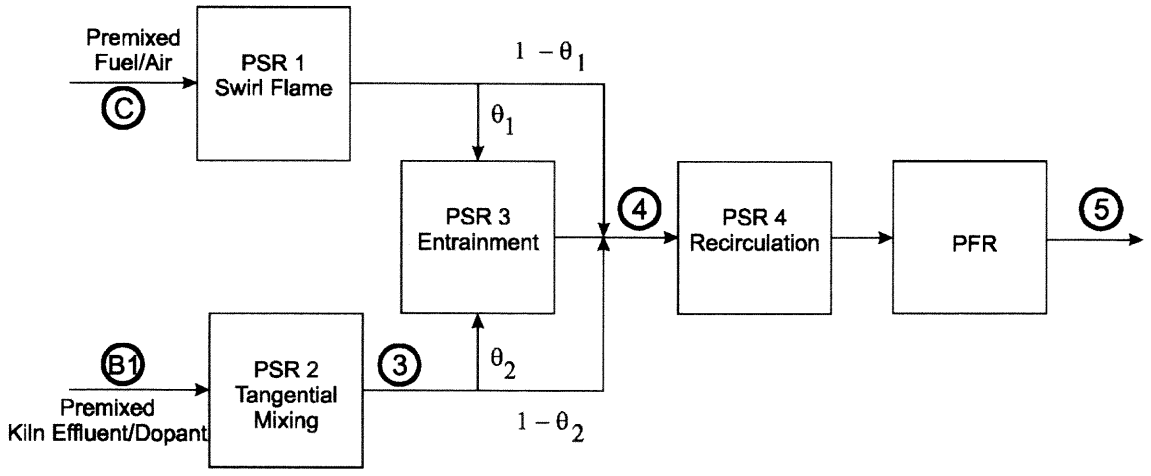


Figure 4.3 Postulated Ideal Reactor Model (Model 0)

angular momentum that transfers through the choke to create another recirculating zone in the same manner as the nozzle on a swirl burner. The dimensionless angular momentum flux or swirl number, *S*, quantifies the intensity of this swirl (Beér and Chigier 1972). Considering only the angular momentum generated from the kiln transition duct under isothermal conditions *S* has a value of about 30 at the choke exit, indicating strong swirl (see calculations in Appendix C). The actual swirl number, if physically measured, would be less because of the temperature increase from combustion and the friction losses from the wall and the sudden contraction at the choke. However, a high swirl number is consistent with both tangential entry swirl burners and cyclone combustors. Additionally, the divergent exit into the burnout section enlarges the recirculating zone in zone 4 (Syred and Beér, 1974). After this zone, the flow should approximate PFR behavior similar to the behavior documented by Beér and Chigier (1972). Figure 4.3 summarizes this analysis with a postulated model that will be the baseline case – model 0.

In Figure 4.3, each box represents an ideal reactor. Each reactor has two parameters, the mean residence time, τ, and the reactor temperature, T (or entrance temperature for the PFR). With θ_1 and θ_2 depicting the fractions of the flame gas stream and kiln effluent stream mixing prior to the choke, the schematic becomes a 12 parameter model. Even though this seems excessive, it represents the simplest possible model when geometry and observed temperature profiles are considered. Since the SCC may contain dead space, the total volume can not be used to eliminate a degree of freedom. The dead volume is not included on the schematic because its parameter is found through physical closure and does not add to the parameter count. Specifically, it is found by subtracting the identified ideal reactor volumes from the physical volumes of the mixing and/or burnout section. The analysis of the RTD moments (Chapter 3) further supports this postulated model. The next step is to identify the model's parameters.

4.3 Parameter Identification

4.3.1 Approach

The model parameters were determined by directly fitting the model to the normalized tracer step response rather than to the reconstructed RTDs. This method has the advantage of fitting the model to original data. The process uses an iterative approach to minimize the error defined by the least squares criterion. The Nelder-Meade simplex search routine (Nelder and Meade, 1965), provided in the standard MATLAB® library, performs the parameter adjustment to minimize the squares of the errors.

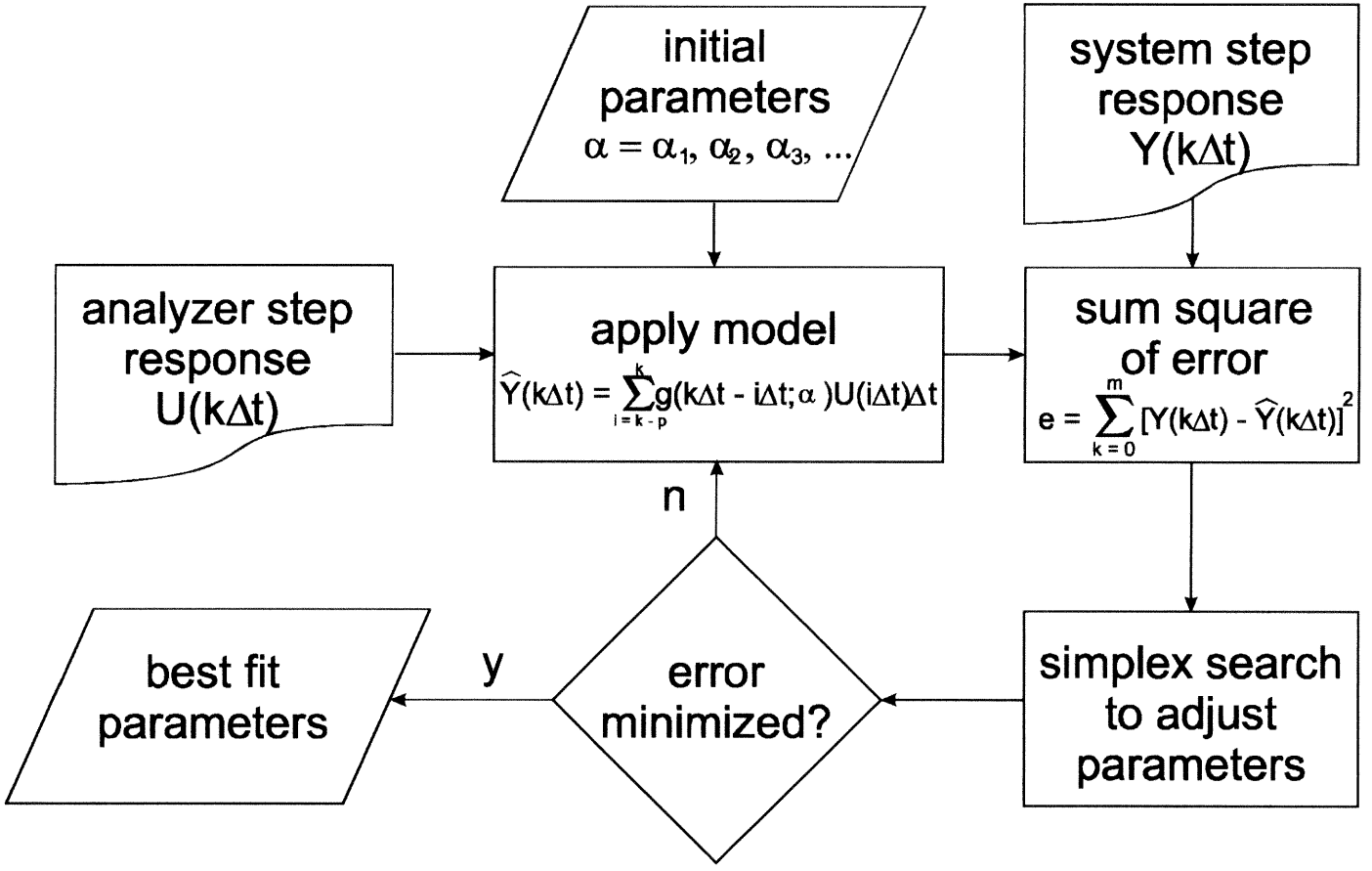


Figure 4.4 Parameter Search Schematic

Figure 4.4 shows the algorithm of this technique. The model uses the analyzer step response, $U(k\Delta t)$ (defined in Chapter 3, Section 3.3.1), as input along with an initial guess of the parameters and gives the model step response or estimated step response, $\hat{Y}(k\Delta t)$, using Equation 3.11 in step form. The total error is then calculated as the sum of the square differences between the system step response and the model step response and weighted with Boolean operators to prevent forbidden (negative) values. The simplex search continues to improve the solution for the parameters and terminates the loop once the total error has been minimized.

The model equation in Figure 4.4 shows the discrete form of the convolution equation with $g(k\Delta t)$ representing the transfer function. The actual convolutions were performed on MATLAB® using the LSIM function from the control toolbox. LSIM accepts the transfer function as a Laplace transform in the form of two polynomial coefficient vectors, one for the numerator and one for the denominator. The Laplace transform is a convenient form for ideal reactor networks. The basic building blocks, the PSR and PRF, are easily represented. Equation 4.5 shows the Laplace transform for the a PSR.

$$G(s) = \frac{a_r}{s + a_r} \tag{4.5}$$

Here, a_r is $1/\tau_r$, the reciprocal of mean residence time, $G(s)$ is the transfer function or the RTD of the PSR and s in the independent variable in Laplace domain. For a PFR the equation is an exponential function.

$$G(s) = e^{-\tau_d s} \tag{4.6}$$

Although the exponential form does not algebraically complicate the network transfer function in the Laplace domain, it does not give a convenient polynomial as required by LSIM. Rather than trying to approximate the exponential as a polynomial, the PFR was introduced as a time domain delay of τ_d after the LSIM calculation. However, the Laplace domain network transfer functions retain the exponential for consistent notation throughout this chapter.

Combining the reactors in the network proves the real convenience of Laplace transform notation. In the time domain, the combination of two reactors in series, $g_1(t)$ and $g_2(t)$, is the convolution,

$$g(t) = \int_0^t g_1(t-\tau)g_2(\tau)d\tau \tag{4.7}$$

but simply the product of their Laplace transformations.

$$G(s) = G_1(s)G_2(s) \tag{4.8}$$

The transfer function for parallel reactors, with stream fraction θ going to the first reactor, is the weighted sum of the reactor transfer functions, and is the same in both Laplace and time domains.

$$\begin{aligned} g(t) &= \theta g_1(t) + (1-\theta)g_2(t) \\ G(s) &= \theta G_1(s) + (1-\theta)G_2(s) \end{aligned} \tag{4.9 a \& b}$$

By using the Laplace transforms, the network model can be developed as a matter of algebraic manipulations that yield equations in a form compatible with the MATLAB ® control toolbox.

4.3.2 Models

Using the approach described above, the model depicted in Figure 4.2 was translated into formulas. Each formula represents a specific input to output combination corresponding to a data set (see Table 3.1). For the mixing section, this corresponds to the combinations with inputs at B1 and C and exiting at points 3 and 4 (as noted earlier the C to 3 combination had no measurable output). The resulting three combinations are given in the following equations:

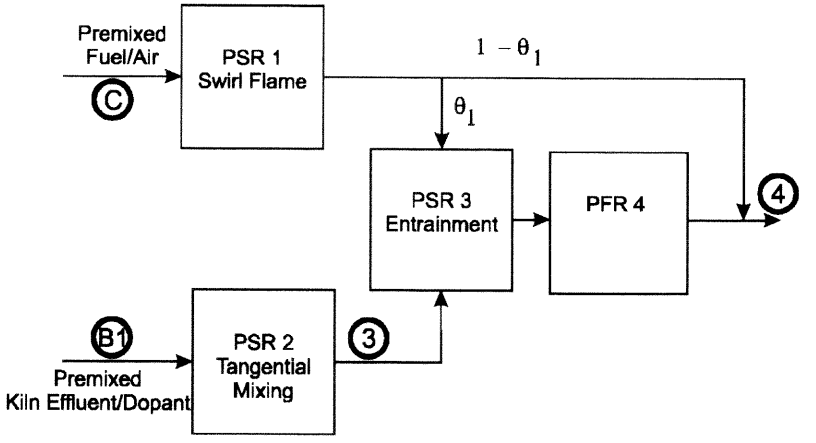
B1 to 4:
$$G(s) = \frac{\theta_2 a_2 a_3}{(s+a_2)(s+a_3)} + \frac{(1-\theta_2)a_2}{(s+a_2)}$$

C to 4:
$$G(s) = \frac{\theta_1 a_1 a_3}{(s+a_1)(s+a_3)} + \frac{(1-\theta_1)a_1}{s+a_1} \qquad \text{(4.10 a, b \& c)}$$

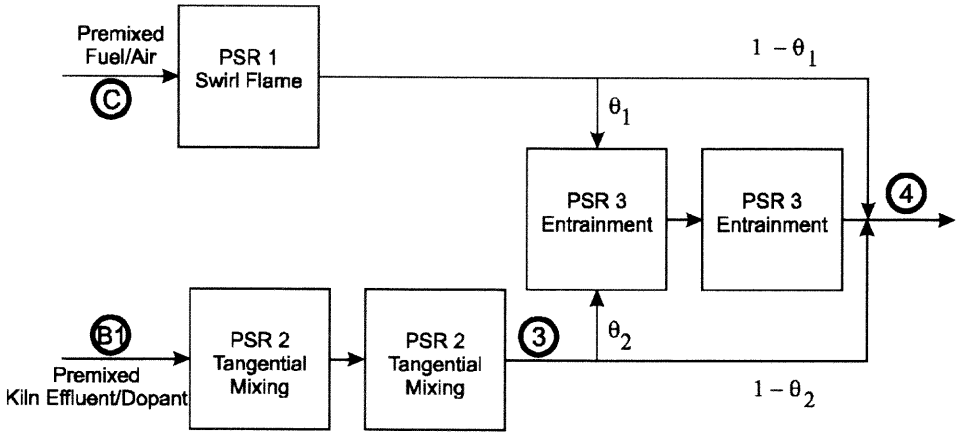
B1 to 3:
$$G(s) = \frac{a_2}{s+a_2}$$

where a_1, a_2, and a_3 are the respective reciprocals of the PSR mean residence times τ_1, τ_2, and τ_3, and θ_1 and θ_2 are the stream fractions shown in Figure 4.2. These parameters were fit by the algorithm illustrated in Figure 4.4.

Alternate Model 1



Alternate Model 2



Alternate Model 3

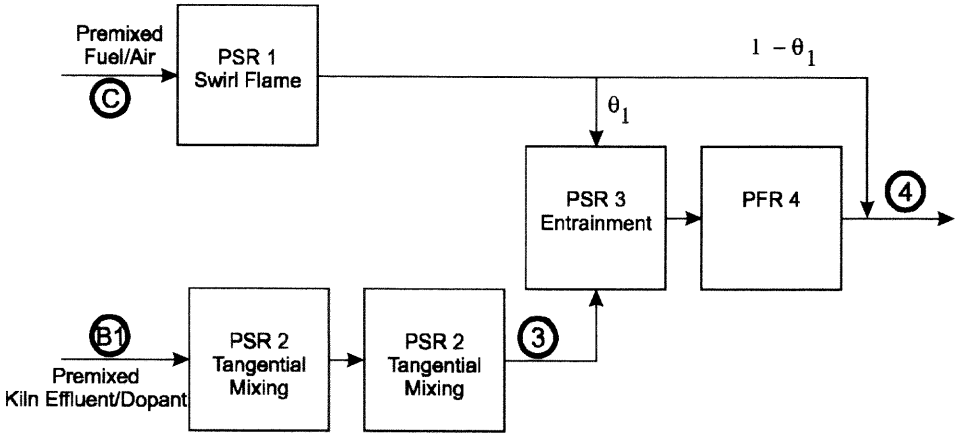


Figure 4.5 Candidate Ideal Reactor Mixing Section Models

Table 4.1 Competing Mixing Section Models

B1 → 4	C → 4	B1 → 3
alternative model 1:		
$G(s) = \frac{a_2 a_3 e^{-\tau_4 s}}{(s+a_2)(s+a_3)}$,	$G(s) = \frac{\theta_1 a_1 a_3 e^{-\tau_4 s}}{(s+a_1)(s+a_3)} + \frac{(1-\theta_1)a_1}{s+a_1}$,	$G(s) = \frac{a_2}{s+a_2}$
alternative model 2:		
$G(s) = \frac{\theta_2 a_2^2 a_3^2}{(s+a_2)^2(s+a_3)^2} + \frac{(1-\theta_2)a_2}{(s+a_2)^2}$,	$G(s) = \frac{\theta_1 a_1 a_3^2}{(s+a_1)(s+a_3)^2} + \frac{(1-\theta_1)a_1}{s+a_1}$,	$G(s) = \frac{a_2^2}{(s+a_2)^2}$
alternative model 3:		
$G(s) = \frac{a_2^2 a_3 e^{-\tau_4 s}}{(s+a_2)^2(s+a_3)}$,	$G(s) = \frac{\theta_1 a_1 a_3 e^{-\tau_4 s}}{(s+a_1)(s+a_3)} + \frac{(1-\theta_1)a_1}{s+a_1}$,	$G(s) = \frac{a_2^2}{(s+a_2)^2}$

An effort was made to improve the fit to the tracer data by considering alternative models that were consistent with the observed temperature profiles and SCC geometry. The three best fitting candidates are illustrated in Figure 4.5. These networks are described by the equations listed in Table 4.1. All these models describe the mixing section with five mixing parameters (mean residence time and bypass fractions, but not temperature). Each has minor variations from the base model (Figure 4.3), but maintain the same number of mixing parameters. The first model eliminates the bypass in the kiln gas stream by channeling all the effluent from PSR 2 to PSR 3, which fixes θ_2 to 1, and adds PFR 4 (with residence time τ_4) after PSR 3. This PSR to PFR order reflects a postulated dissipation of mixing energy after PSR 3 before the stream enters the choke and is consistent with the model suggested by Beer and Lee (1965). The second alternative model modifies the base model by using two equally sized PSRs in place of PSR 2 and two equally sized PSRs in place of PSR 3, which better

models a more complex mixing pattern. The third alternative is a hybrid approach between the first and the second in that it eliminates the bypass in the kiln gas stream, adds a PFR following PSR 3 and uses two equal PSRs in place of PSR 2.

The model downstream of the choke describes the stream as it expands from a fast moving fluid in the narrow choke to the slower moving fluid downstream. Translating the model in Figure 4.3 downstream of the choke (i.e. the burnout section) to a formula gives:

4 to 5:
$$G(s) = \frac{e^{-\tau_5 s} a_4}{s + a_4} \tag{4.11}$$

This assumes the confluence of the kiln gas and burner streams in the choke at point 4. Unlike the mixing section, this formula does not correspond to a measured tracer input/output combination. As an alternative, the 4 to 5 tracer run was synthesized by using the C to 5, B1 to 5, and B2 to 5 runs as the system response functions and the C to 4, B1 to 4, and B2 to 4 as the respective driver functions in place of the analyzer functions (see Appendix C for the point 4 to 5 RTDs). Adding an equally sized PSR to the model in Equation 4.11 improved the fit with the data, but the fitted parameters differed greatly depending on the tracer combination used to synthesized the 4 to 5 tracer run. This suggests exploring a more complex model.

Burnout section RTDs derived from tracers entering at tangential entry points (points B1 and B2) into the mixing section had mean times 0.6 to 0.7 seconds shorter than the RTD produced from the axial entry point (point C). Figure 4.6 shows a model proposed to

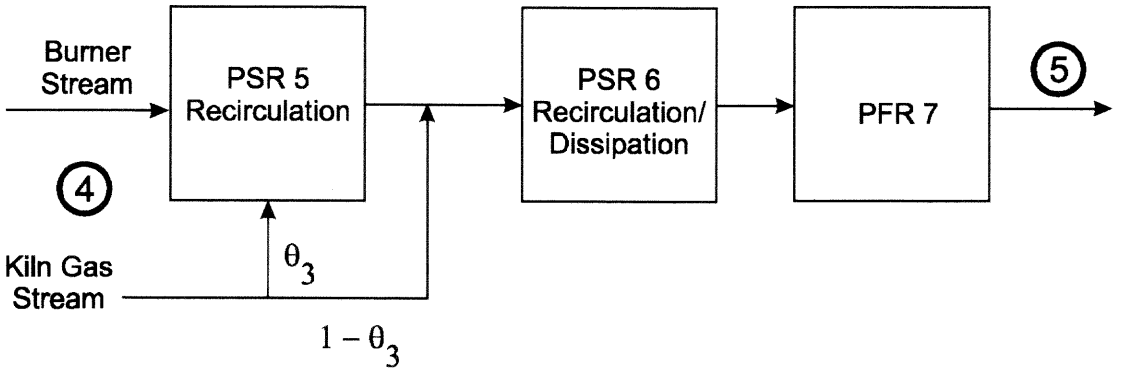


Figure 4.6 Candidate Ideal Reactor Burnout Section Model

reconcile these differences. It considered the kiln gas (tangential entry) and the burner gas (axial entry) streams as separate, which was consistent with the temperature profile in the choke (Figure 3.4). The model proposes that the burner gas stream, emerging from the center of the choke duct, forms a separate recirculating zone as the kiln gas bypasses this zone and mixes further downstream. Since other alterations to this model, such as additional downstream PSRs, significantly worsened the fit, no other candidates are presented for discussion. Equation 4.12 shows the modified burnout section model.

kiln gas stream 4 to 5:

$$G(s) = \frac{\theta_3 a_5 a_6 e^{-\tau_7 s}}{(s+a_5)(s+a_6)} + \frac{(1-\theta_3) a_6 e^{-\tau_7 s}}{(s+a_6)} \qquad (4.12\ \text{a\&b})$$

burner gas stream 4 to 5:

$$G(s) = \frac{a_5 a_6 e^{-\tau_7 s}}{(s+a_5)(s+a_6)}$$

4.3.3 Error Analysis

Once the parameter values are estimated, it is useful to determine the confidence bounds on them. This has several utilities. First, this allows testing the sensitivity of the objective model

(the combination of the mixing model with the thermo-kinetic mechanism and the thermodynamic data) to the errors of the mixing model parameters. Next, the limits are useful for refinement or selection of the preferred candidate networked ideal reactor models. Parameters with relatively high standard errors or high covariance often indicate overparameterization;, more parameters in the model than are needed to represent the data (Draper and Smith, 1998).

The networked ideal reactor model estimates the normalized system response of the tracer, $\hat{y}$. The i^{th} estimated point $\hat{y}_i$ is a function of the independent variable x_i (which correspond to the discrete time measurements, $i\Delta t$, for $i = 0, 1, 2, 3, \ldots m\text{-}1$) and n parameters ($\alpha_1, \alpha_2, \alpha_3 \ldots \alpha_n$). The sensitivity of the parameters can be deduced from the covariance matrix, $\boldsymbol{\Psi}$ (Press *et al.*, 1992) shown in matrix form:

$$\boldsymbol{\Psi} = \left[\frac{\partial \hat{y}(\boldsymbol{x};\boldsymbol{\alpha})^{/}}{\boldsymbol{\sigma}^2 \partial \boldsymbol{\alpha}} \frac{\partial \hat{y}(\boldsymbol{x};\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \right]^{-1} \tag{4.13}$$

where:

$$\frac{\partial \hat{y}(\boldsymbol{x};\boldsymbol{\alpha})}{\boldsymbol{\sigma}^2 \partial \boldsymbol{\alpha}} = \begin{bmatrix} \frac{\partial \hat{y}(x_1;\boldsymbol{\alpha})}{\sigma_1^2 \partial \alpha_1} & \frac{\partial \hat{y}(x_1;\boldsymbol{\alpha})}{\sigma_1^2 \partial \alpha_2} & \cdots & \frac{\partial \hat{y}(x_1;\boldsymbol{\alpha})}{\sigma_1^2 \partial \alpha_n} \\ \frac{\partial \hat{y}(x_2;\boldsymbol{\alpha})}{\sigma_2^2 \partial \alpha_1} & \frac{\partial \hat{y}(x_2;\boldsymbol{\alpha})}{\sigma_2^2 \partial \alpha_2} & \cdots & \frac{\partial \hat{y}(x_2;\boldsymbol{\alpha})}{\sigma_2^2 \partial \alpha_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \hat{y}(x_m;\boldsymbol{\alpha})}{\sigma_m^2 \partial \alpha_1} & \frac{\partial \hat{y}(x_m;\boldsymbol{\alpha})}{\sigma_m^2 \partial \alpha_2} & \cdots & \frac{\partial \hat{y}(x_m;\boldsymbol{\alpha})}{\sigma_m^2 \partial \alpha_n} \end{bmatrix} \tag{4.14}$$

and σ_i^2 is the local variance of the measurement of $\hat{y}_i$ (see Appendix C for the method used to deduce the local measurement variance). Each column of the matrix in Equation 4.14 shows the sensitivity of the system response $\hat{y}$ to a perturbation of parameter α_j. When applied to Equation 4.13, the diagonal elements of the covariance matrix gives the standard or probable parameter variance, and their square roots give the standard or probable error of the parameters.

$$\sigma(\alpha_j) = \sqrt{\Psi_{jj}} \tag{4.15}$$

The off-diagonal elements represent the covariance between parameters. Low values indicate a high degree of independence between parameters. An overparameterized model usually has some parameters with relatively high standard errors and a high covariance between them (Draper and Smith, 1998).

The standard errors do not give the confidence bounds. The confidence bound can be calculated independently using the t-distribution $f_\nu(1-p/2,\nu)$ (where p is the desired confidence level and ν represents the degree of freedom m - n) given by:

$$\alpha_j|\alpha_j = \alpha_j^* \pm f_\nu(1-p/2,m-n)\sigma_j \tag{4.16}$$

However, these confidence intervals are misleading and inaccurate since Equation 4.16 assumes that parameters are independent of each other and that they vary symmetrically

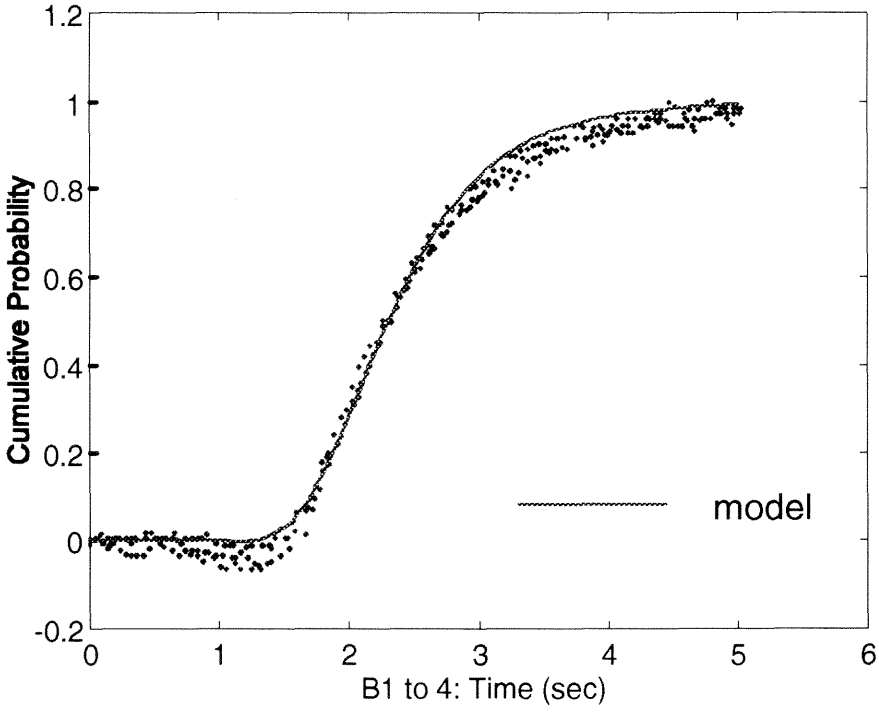


Figure 4.7 Comparison of Model 1 with Normalized Data: B1 to 4

around α_j, which is seldom the case for non-linear least squares estimation (Rooney and Biegler, 2001). Other methods that more accurately estimate the confidence region involve Monte Carlo techniques will not be explored in this work.

4.4 Results

In the mixing section, the models for B1/3, B1/4, and C/4 share five parameters. Rather than fitting each model individually, these models were fit simultaneously by adding the model-to-data errors into a single objective function. This reconciled the inevitable differences that would have emerged through separate treatment. The fitting was limited to those data points on the interval between 0 and 2.7 seconds. This had two effects: it weighted the interval of most rapid change, and it avoided the extended "tail" in the RTD that was a likely result from

dead space in the mixing section. This effect was most pronounced in the B1/4 tracer runs. Figure 4.7 shows a pronounced dead space effect between 3 and 4 seconds where the model overshoots the data. The model was insufficiently complex (by design) to account for the slow absorption or desorption of tracer from a dead volume. By fitting on the 0 to 2.7 second interval, the model parameters could be optimized for the portion they were meant to describe.

Table 4.2 Results of Competing Mixing Section Models

	Model 0 *(std error)*	Model 1 *(std error)*	Model 2 *(std error)*	Model 3 *(std error)*	Reconst RTD
standard deviation	0.040	0.028	0.029	0.027	
Theta 1	0.005 *(0.151)*	0.000 *(0.000)*	0.000 *(0.111)*	0.000 *(0.000)*	
Theta 2	1.000 *(0.113)*		1.000 *(0.038)*		
Tau 1(PSR)	0.330 *(0.083)*	0.333 *(0.007)*	0.333 *(0.061)*	0.333 *(0.007)*	
Tau 2 (PSR)	0.561 *(0.007)*	0.560 *(0.003)*	0.268 *(0.001)*	0.272 *(0.001)*	
Tau 3 (PSR)	0.596 *(0.091)*	0.003 *(0.594)*	0.277 *(0.012)*	0.003 *(0.942)*	
Tau 4 (PFR)		0.552 *(0.559)*		0.508 *(0.898)*	
B1 to 4					
mean	1.16	1.12	1.09	1.06	1.08
variance	0.67	0.31	0.30	0.15	0.40
skewness	1.42	2.00	1.00	1.41	1.46
C to 4					
mean	0.33	0.33	0.33	0.33	0.31
variance	0.11	0.11	0.11	0.11	0.17
skewness	1.97	2.00	2.00	2.00	2.03
B1 to 3					
mean	0.56	0.56	0.54	0.54	0.57
variance	0.31	0.31	0.14	0.15	0.32
skewness	2.00	2.00	1.41	1.41	2.57

All units in seconds except variance (seconds^2) and skewness (dimensionless).

Table 4.2 shows the fitted parameters for the four mixing section models, the standard errors of the parameters and compares the derived moments (mean, variance, and skewness) to the moments from the reconstructed RTDs. The table also gives the standard deviation between the data and each fitted model for a standard of comparison. In general, one can infer from the fitted parameters that little to no mixing takes place between the kiln gas and burner gas streams between the entrances and exit of the mixing section. All models show theta 1 as zero or near-zero which implies a complete bypass between the flame zone and choke. The flame and kiln gases essentially do not mix inside the so-called mixing section of the SCC. Additionally, all the kiln gas stays in a single stream, which the theta 2 value of one implies in models 0 and 2. Most likely some entrainment occurs at the interface between the two streams, but not sufficient enough to be distinguished as a separate mixing zone. The fitted parameters of all models show a great deal on consistency with the moments derived from the reconstructed RTDs.

The first moments (mean) derived from the model parameters show excellent agreement (within the measurement interval of 0.05 seconds) with the mean calculated from the reconstructed RTDs in Chapter 3. Higher order moments, however showed less agreement with some models. While the measurement of the variance from tracer data can be more problematic because it weights the data furthest from the mean (which is more effected by noise and hence less reliable), lack of agreement is a criterion for discriminating between models. For instance, second moments (variance) derived from the model parameters for models 2 and 3 show poor consistency with the second moment derived from the B1/3 RTD. Both these models use two equal PSRs for PSR 2 in the mixing section.

Using two equal PSRs, in place of one, halves the value of the variance calculated from the parameters (see Table 3.2, moments for model archetypes). Observing the same trend in Table 4.2 for the B1 to 3 RTD suggests that a single PSR here is a better model.

Comparison of the skewness coefficients derived from the model parameters with those derived from the reconstructed RTDs reveal several trends. All models have a single PSR between C and 4 which yields a skewness coefficient of 2 (the skewness coefficient from model 0 is slightly less that 2 due to a small amount of mixing with PSR 3) that agrees with the moment derived from the RTD. The single PSR between B1 and 3 for models 0 and 1 that gives a skewness coefficient to is closer to the skewness coefficient derived from the RTD of 2.57. The higher number from the reconstructed RTD could be indicative of long tails that result from dead space (Himmelblau and Bischoff, 1968) or simply an outlying data anomaly magnified by the higher moment. The skewness coefficient from the reconstructed RTD between B1 and 4 supports a model with two PSRs of approximately equal size. Models 0 and 3 conform to this postulate, but model 3 lacks good agreement with the variance.

As discussed earlier, models with high standard errors are indicative of overparameterization, which are models with more parameters than necessary to describe the data. The high standard errors for the bypass fractions in models 0 and 2 suggest a model that might eliminate one or both of these parameters. Models 1 and 3 have high standard errors for τ_3 and τ_4 which suggest a close correlation that might support eliminating one or the other. The relation between these two parameters will be further discusses in section 4.4.3.

Table 4.3 Results of Competing Burnout Section Models

	Model 0 (B2 input) *(std error)*	Model 0 (C input) *(std error)*	Model 1 *(std error)*	Reconst RTD
Theta 3			0.013 *(0.155)*	
Tau 5 (PSR)	0.782 *(0.014)*	0.898 *(0.011)*	0.765 *(0.117)*	
Tau 6 (PSR)			1.102 *(0.094)*	
Tau 7 (PFR)	0.811 *(0.029)*	1.345 *(0.018)*	1.287 *(0.032)*	
4 to 5 (B2 Input)				
mean	2.37		2.40	2.27
variance	1.22		1.23	1.05
skewness	1.41		1.98	0.75
4 to 5 (C Input)				
mean		3.14	3.15	2.99
variance		1.61	1.80	1.41
skewness		1.41	1.48	0.82

All units in seconds except variance (seconds^2) and skewness (dimensionless).

For the burnout section, model 0 (Figure 4.3) assumed a completely mixed flow emerging from the choke and model 1 assumed a complete segregation of the kiln gas and burner gas streams. The parameters for model 0 could be fit using any tracer run measured downstream of the choke and a tracer run originating at the same input and measured at the choke as the driving function (in the role of the analyzer function in Figure 4.4). Model 1 (Figure 4.6) required a simultaneous fit in the same manner as the mixing section. Model 0 was modified by replacing its single PSR with 2 equal PSRs in series (labeled PSR 5 for

consistency). This provided the best fit and its results are presented in Table 4.3. Results produced from the B2/5 and C/5 tracer runs on model 0 are compared to the simultaneous parameter fitting on model 1 using the same tracer runs.

The results produced for model 0 from the different tracer inputs (B2 and C) varied by 0.77 for total mean residence time. Even though each individually agrees with the reconstructed RTD, both cannot exist since different values are required for the same ideal reactors. Model 1, which assumes segregated streams, resolves the contradiction by giving the steams two separate paths initially and allowing the mixing to occur further downstream (see Figure 4.6). The resulting first and second moments derived from the model parameters show good agreement with those corresponding to the reconstructed RTDs.

4.4.1 Selecting a Model

Model 1 was selected for the mixing section. This choice was made for a number of reasons. First, it had a significantly better overall fit to the data than model 0, which is exhibited in the data-to-model standard deviation in Table 4.2. Next, the model parameters for a single PSR 2 produced a much better agreement with the variance derived from the reconstructed RTD from the B1 to 3 tracer run than the two PSRs used in models 2 and 3 (see Figure 4.4). Finally, the model is simple, which is consistent with the goals of the ideal reactor model.

Figure 4.8 shows excellent agreement between the differentiated normalized tracer data (the system response) with the model 1 system response. Figure 4.9 shows the same comparison when the effect of the analyzer is removed from the system response. The least square (LS) identification, described in Chapter 3, accounts for the large difference between

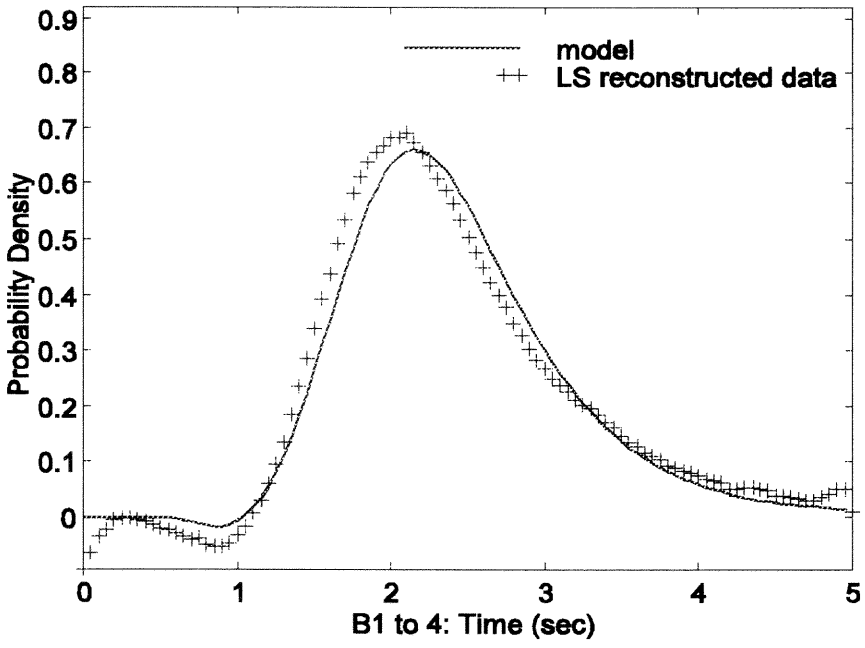


Figure 4.8 Comparison of Model 1 with System Response: B1 to 4

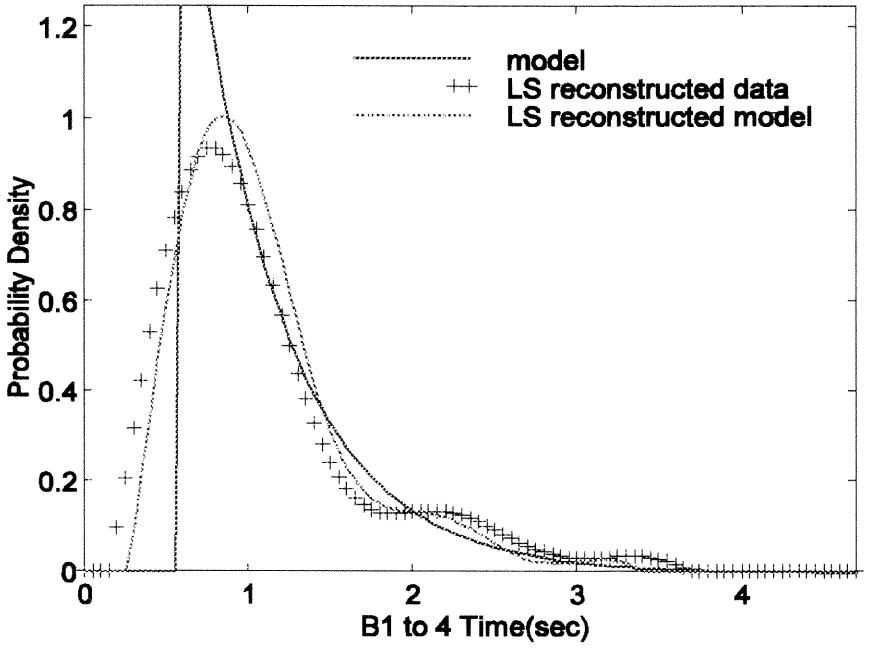


Figure 4.9 Comparison of Model 1 with LS Reconstructed RTD: B1 to 4

model 1 and the reconstructed RTD. In order to illustrate this, a system response for model 1 was produced, using the analyzer function as the driver, then reconstructed by the same process to produce the LS identified model. This shows excellent agreement with the LS reconstructed data and provides an better method of visually comparing data and model. The next section provides further analysis into this phenomenon.

The burnout section was best represented by model 1 since it accounted for the differences produced by tracer runs from different entry points on model 0. This assumes that the kiln gas and burner gas streams emerge segregated from the choke then mix in the next two PSRs. From the fitted parameters, most of the kiln gas bypasses the first mixing zone before completely mixing in the next mixing zone downstream (PSR 6). This implies that the added chlorocarbon dopant travels over half the length of the SCC before becoming completely mixed with the SCC burner effluent. The selected burnout model is compared to the differentiated and reconstructed data in Figures 4.10 and 4.11. Both show excellent agreement between data and model. In Figure 4.11, the LS reconstructed model is used as the basis of comparison.

4.4.2 Data Reconstruction and Analysis of the Power Spectrum

The analyzer has two effects on the step response of the reactor: it smears or broadens and delays the response, and it acts as a band limiting filter. The smearing effect was corrected by the LS reconstruction techniques covered in Chapter 3. The band limiting effect, however, limits the ability to reconstruct the underlying transfer functions. This effect creates the discrepancies between the reconstructed RTD and the fitted parameter model.

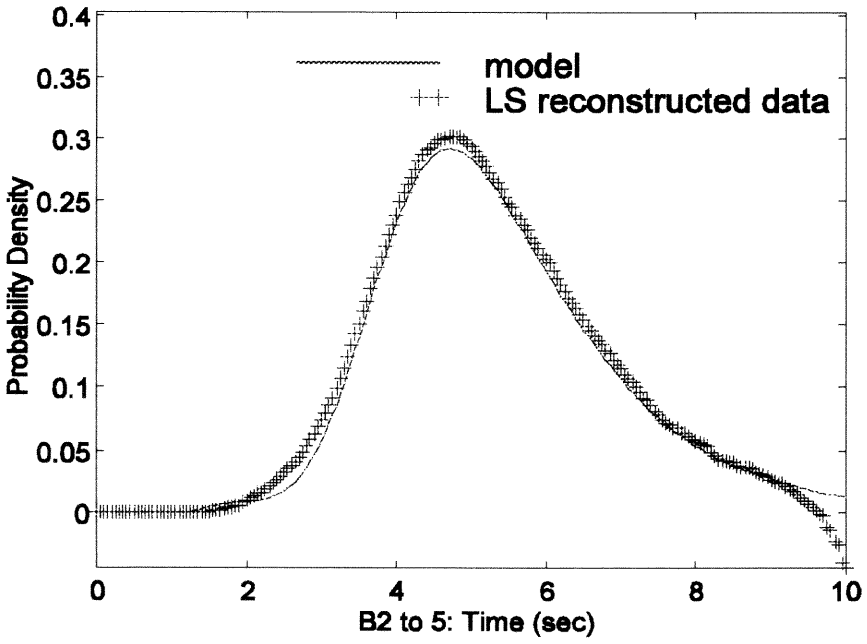


Figure 4.10 Comparison of Model with System Response: B2 to 5

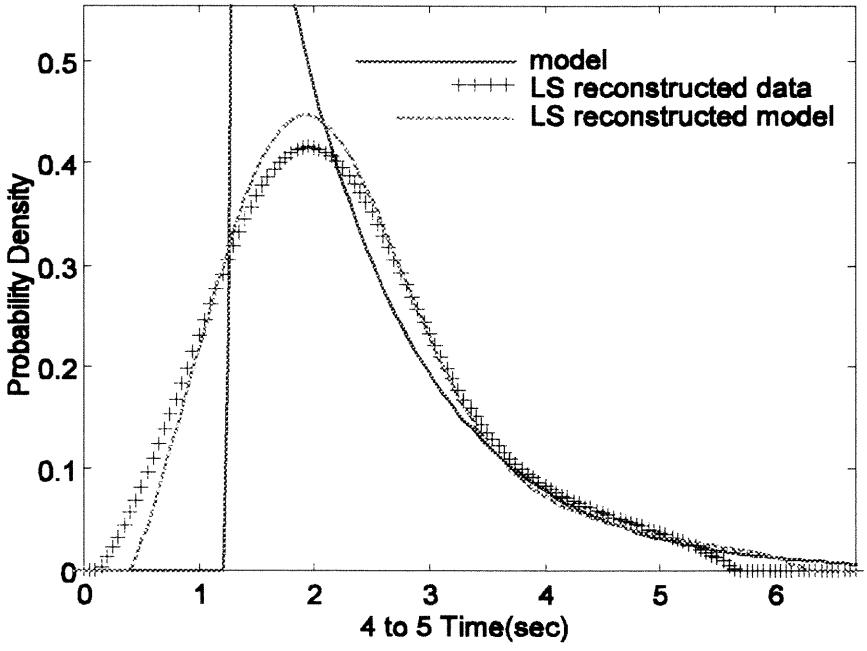


Figure 4.11 Comparison of Model with LS Reconstructed RTD: 4 to 5 w/ B2 Input

The reactor RTD is the impulse response of an underlying linear transfer function, $g(t)$. The impulse function, $\delta(t)$, transforms into frequency domain as a constant function, $\delta(\omega) = 1$ (where ω is angular frequency $2\pi f$), so it contains all frequencies at equal energy levels (O'Neil, 1991). The driven system will give a true response characteristic of all the frequencies in its transfer function. For discrete sampling, the Nyquist critical frequency f_c, a function of the sampling interval, imposes a limitation on the frequencies produced in the system response. Equation 4.16 shows the Nyquist critical frequency for a sampling interval, Δt, of 0.05 seconds (Press *et al.*, 1992). So, frequencies above 10 Hz cannot be measured by the analyzer.

$$f_c \equiv \frac{1}{2\Delta t} = 10 \text{ Hz} \tag{4.17}$$

Even assuming that the SO_2 tracer train can produce an impulse input (synthesized by taking the first derivative of a step input) the results are observed through an analyzer with its own characteristic linear response. As explained in section 3.3.3, the analyzer acts as the driver to the transfer function of the SCC to produce a system response (see Figure 3.5). The analyzer function in this construct, unlike an impulse function, produces a frequency spectrum that falls off far below the Nyquist critical frequency. Using the power spectral density of the analyzer response, which is its Fourier transformation multiplied by its complex conjugate, reveals a gain reduction of 100 dB above 1 Hz. Figure 4.12 shows the power spectrum density of the analyzer compared to model 1 for the B1 to 4 input/output combination in the mixing section. Since the analyzer function drives the response of the actual system, it dampens the frequencies above 1 Hz of the system response. As a result, the LS

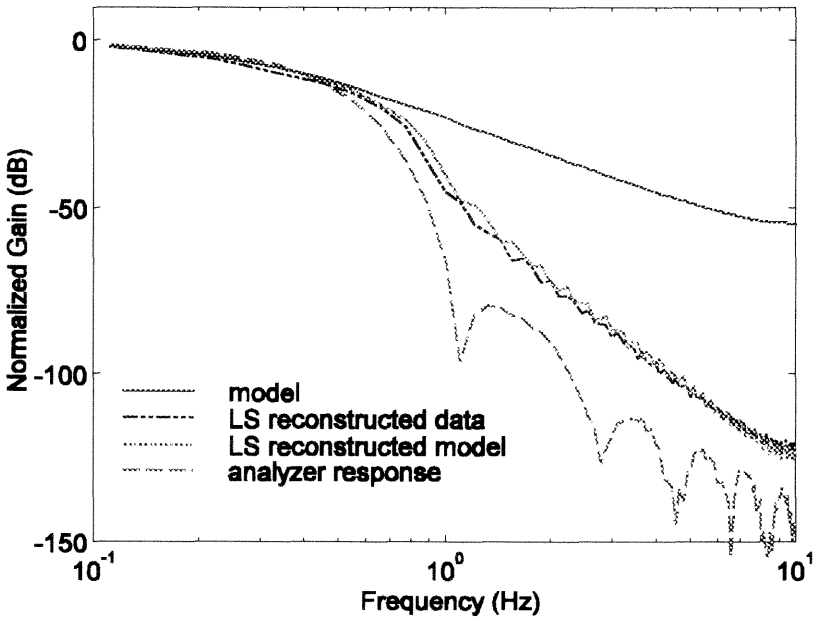


Figure 4.12 Power Spectrum Comparison of Model and LS Reconstructed RTD: B1 to 4

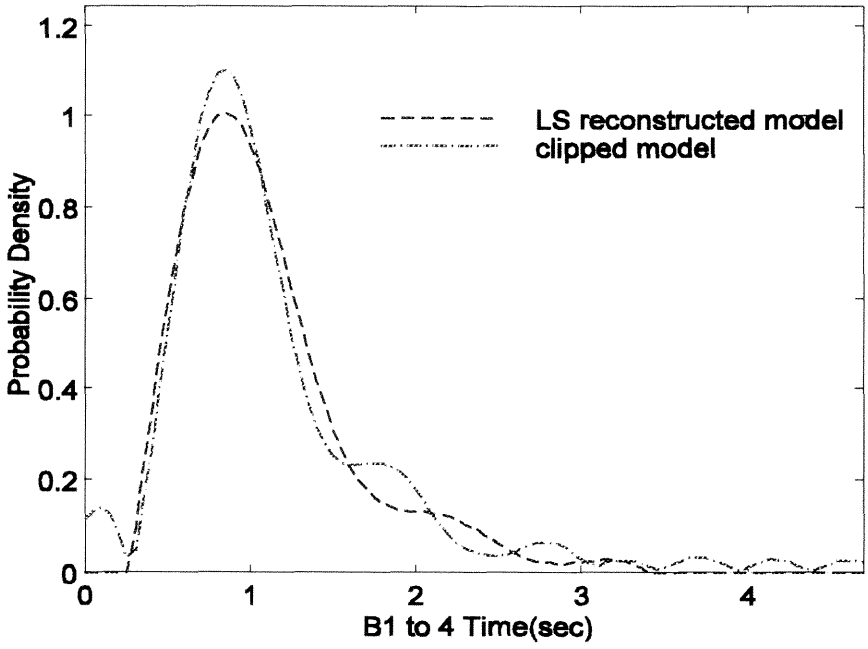


Figure 4.13 Comparison of Clipped and Reconstructed Model: B1 to 4

reconstructed data and LS reconstructed model (the model driven by the analyzer response then reconstructed again) fail to match the power densities of the model at the higher frequencies. This effectively limits the LS reconstruction of both data and model to frequencies below 1 Hz.

Figure 4.13 illustrates this point. The LS reconstructed model shows the selected B1 to 4 model transfer function driven by the analyzer response to produce a system response which was reconstructed by LS identification. The other line is the impulse response of the model transfer function with all frequencies above 1 Hz set to zero or "clipped" using a Fast Fourier Transformation. This results in a response that closely approximates the LS reconstructed model for B1 to 4, which closely resembles the LS reconstructed data (Figure 4.9). Therefore, the analyzer is acting as a limited bandwidth filter that dampens frequencies above 1 Hz.

This frequency band limitation produces non-unique solutions in the reconstruction process (Schafer *et al.*, 1981). For instance, the LS reconstructed data graphed in Figure 4.9 or 4.11 could have resulted from any one of a number of systems with power spectral densities equal below 1 Hz, but differing greatly above 1 Hz. This problem is more dominant in Figure 4.14 where the B2 to 4 tracer run, rather than the analyzer, is used as the driver function, and falls-off well below 1 Hz. The constrained iterative restoration algorithm (Schafer *et al.*, 1981) applied for the RTD reconstruction took a consistent approach by starting with the system response (model or real system driven by the analyzer function) as the initial solution. By doing this, nothing is assumed in the solution about the higher frequencies.

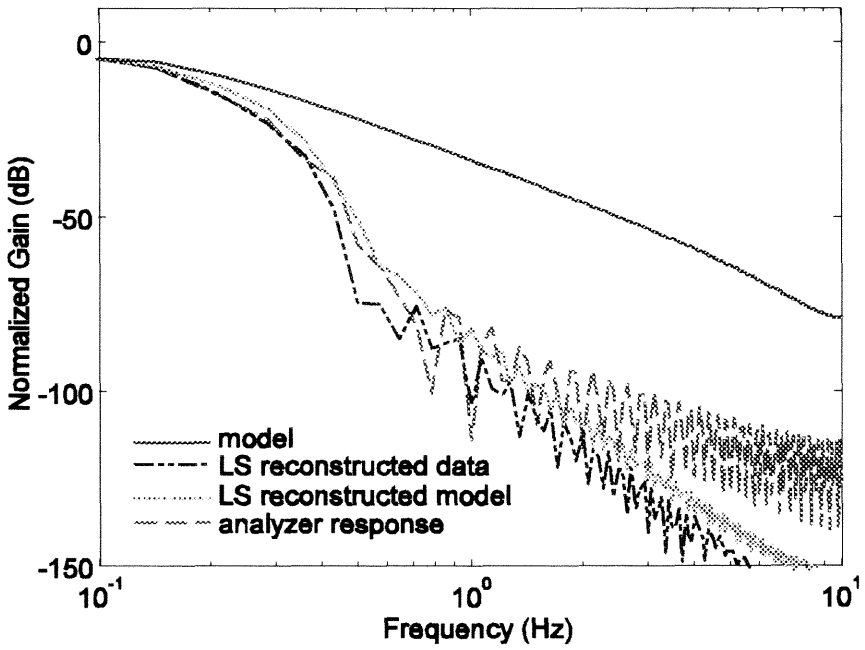


Figure 4.14 Power Spectrum Comparison of Model and LS Reconstructed RTD: 4 to 5 w/B2 input

4.4.3 Error Analysis

Table 4.2 shows the selected (preferred) mixing section model (model 1) with low standard errors for the first three parameters, θ_1, τ_1 and τ_2, but has relatively high standard errors for the last two parameters, τ_3 and τ_4. High standard errors indicate overparameterization; more parameters than can be supported by the model. At first glance, this appears to be the case. The parameters τ_3 and τ_4 are highly correlated. The correlation coefficient is a function of the parameter variances and the covariance between them as shown in Equation 4.18 (Draper and Smith, 1998),

$$\rho_{34} = \frac{\Psi_{34}}{(\Psi_{33}\Psi_{44})^{1/2}} = \frac{(-0.332)}{[(0.353)(0.313)]^{1/2}} = -1.00 \qquad (4.18)$$

which means that these parameters have a perfect negative correlation.

To understand this perfect correlation, recall that model 1 added PFR 4 following PSR 3 in the base model to achieve a better fit. Figure 4.15 shows the contour of the objective function, S, which is a summation of the squares of the differences between the system step response, Y, and the estimated (model) system response, $\hat{Y}$:

$$S(\tau_3,\tau_4) = \sum_{i=1}^{m} \left[Y_i(x_i) - \hat{Y}_i(x_i;\tau_3,\tau_4) \right]^2 \qquad (4.19)$$

The contour forms a "valley" from about 0.55 on the abscissa to the same value on the ordinate. This suggests the elongated shape of the confidence region. Since these two parameters compete in this region over a fixed total residence time for PSR 3 and PFR 4, it makes more sense to define new parameters: τ_{34} as the mean residence time for both PSR 3 and PFR 4, and θ_{34} as the fraction of that mean residence time in PSR 3. Defined in this manner, the parameter values become:

τ_{34}	0.555	*(0.014)*
θ_{34}	0.008	*(0.944)*

Thus, the residence time of PSR 3 and PSR 4 together is very certain, but it cannot be determined from the data where PSR 3 ends and PFR 4 begins with a high degree of certainty. This uncertainty must be explored when thermo-kinetic mechanisms are applied to the model.

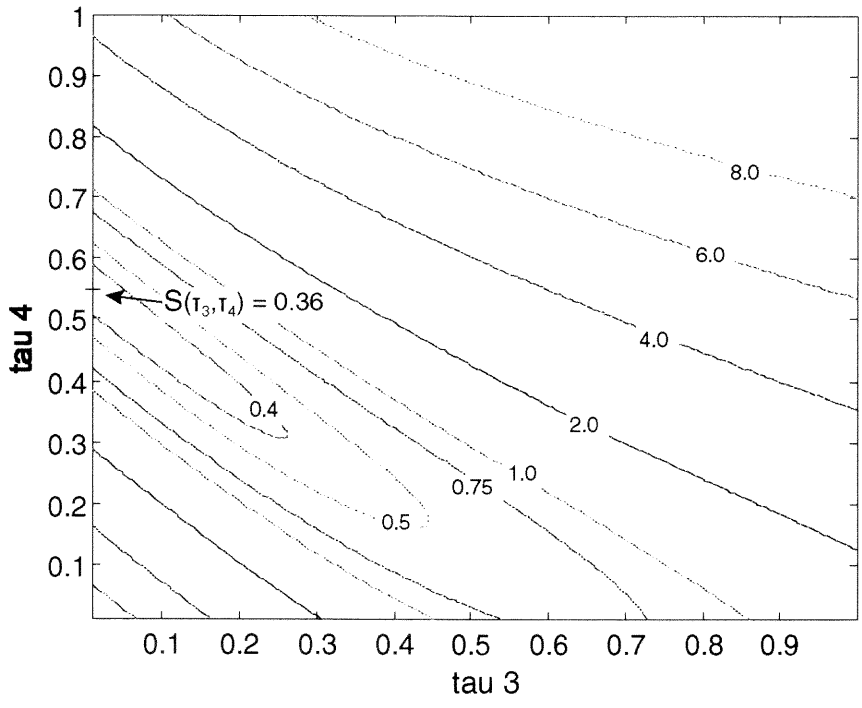


Figure 4.15 Objective Function Surface: τ_3 vs. τ_4

4.4.4 Temperature Parameters

The notional ideal reactors were assigned residence time parameters based on a best fit to the tracer step responses. This information is used in combination with physical temperatures observations to assign temperatures to each reactor. Cross-sectional temperature measurements from points B1, 4, and 5 are used to assign these temperatures. Figure 4.16 shows the relationship between the selected model and the physical SCC where these measurements were taken.

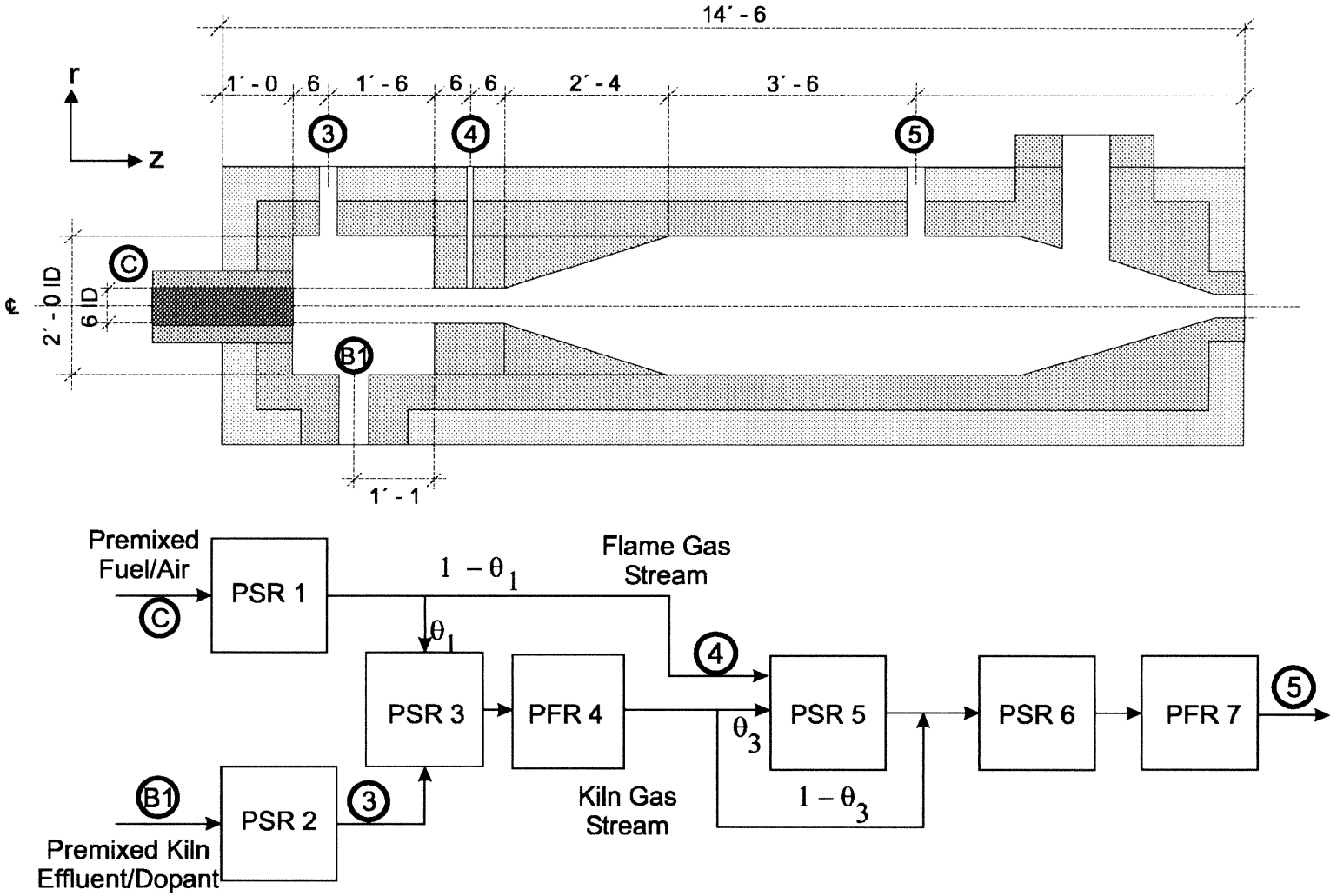


Figure 4.16 Selected Model Juxtaposed to Physical Secondary Combustion Chamber; Dimensions in Feet and Inches

At each point, the cross sectional temperature contour was estimated using the technique described in section 3.2.2. A weighted average of the generated cross-sectional mesh cells gave the average temperature:

$$\bar{\rho}\bar{C}_p\bar{T} = \sum_{i=1}^{n} \rho_i C_{p_i} T_i \qquad (4.20)$$

where n is the number of cells in a 0.1 inch grid. The average temperature increased from the mixing section (1031 K) to the choke (1450 K). The lower temperature in the mixing cross-section is due, in part, to the longer residence time of the cooler kiln gases (1.12 seconds from B1 to point 4) versus the faster moving flame gases (0.33 seconds from C to point 4). However, even when this is considered, the average temperature still increases. In contrast, the burnout section deceases to an average temperature of 1241 K as gas travels between points 4 and 5. The temperature increase from the mixing section to the choke and the temperature decrease in the burnout section suggests the methods for assigning temperatures to individual ideal reactors.

The cross section temperature contour (Figure 3.3) of the mixing chamber was used to determine PSR2. Averaging the temperatures that excluded the flame region, the area of rapid temperature rise inside the 1050 contour, resulted in a kiln gas stream average temperature of 1009 K, which was assigned as the temperature of PSR 2. The hotter zone (inside the 1050 K contour) had an average of 1344 K, but was not assigned to PSR 1 because of a higher average flame gas temperature found at point 4. The model shows two distinct

streams withing the choke, although the temperature profile (Figure 3.4) does not suggest a dividing point. Assuming a uniform velocity in the z direction in the choke section, the two gas streams were divided using the relative mass flow rate. A radial cross-section temperature contour was constructed and the cross sectional grid cells were divided out using the following relationship between velocity and mass flow:

$$\dot{m} = \rho A v_z = v_z \sum_{i=1}^{n} \rho_i A_i \tag{4.21}$$

where A_i the grid cell area and v_z is the uniform z direction velocity. This gave an average flame gas stream of 1538 K and a kiln gas stream temperature of 1404 K. Since the flame in the mixing section (PSR1) is modeled as a single PSR, only a single temperature may be assigned. The higher temperature was assigned to PSR 1. The other two ideal reactors temperatures were based on a uniform heat gain within the kiln gas stream to 1404 K.

The temperatures of the ideal reactors in the burnout section were assigned based on the model parameter mixing fraction, θ_3, and a uniform heat loss through the combustor walls to the ambient air. The heat loss, Q, can be calculated from either the enthalpy differences of the combined streams between the choke and point 5, or the heat transfer through the combustor walls using the reactor surface area, A, the overall heat transfer coefficient, U, and the log mean temperature difference, *LMTD*:

$$Q = \dot{m}(h_{out} - h_{in}) = UA\,(LMTD)$$

$$\text{where:} \quad LMTD = \frac{T_{out} - T_{in}}{\log\left(\frac{T_{out} - T_a}{T_{in} - T_a}\right)} \tag{4.22}$$

Using a measured ambient temperature, T_a, of 40 °C and mass specific enthalpy, h, based on complete combustion products resulted in an overall heat transfer coefficient, U, of 1550 ergs/s cm^2 °C (0.272 Btu/hr ft^2 °F). The ideal reactor temperature, T_R, is found for PSR 5 and PSR 6 using the energy balance for a stirred reactor:

$$\dot{m}(h_{in} - h_{out}) = UA\,(T_R - T_a) \tag{4.23}$$

The volume of the burnout section, V_B, is known from the physical geometry of the SCC, but the ideal reactor volumes are yet to be calculated. To do this, the volume specific heat transfer, UA/V_B, is used in combination with the ideal reactor volume expressed in terms of reactor mean residence time, τ_R, and the reactor temperature shown in the second group of terms on the right hand side of Equation 4.24:

$$\dot{m}[h_{in} - h_{out}(T_R)] = \frac{UA}{V_B}\,\frac{\tau_R \dot{m} R T_R}{\bar{W} P}\,(T_R - T_a) \tag{4.24}$$

where $\bar{W}$ is the mean molar mass and P is the reactor pressure. The enthalpy of the feed stream is fixed, but effluent stream enthalpy depends on the reactor temperature, and is described by the correlations in the *CHEMKIN Thermodynamic Database* (Kee, Rupley, Miller, 1992). Reactor temperature was solved for using a spreadsheet equation solver and

the results as shown in Table 4.4. The final ideal reactor is a plug flow reactor, which is assigned an input temperature from the output of PSR 6 and a output temperature as the average temperature at point 5.

Table 4.4 Model Parameters: Residence Times, Temperatures and Resulting Volumes

Ideal Reactor	Mean Residence Time (seconds)	Temperature (K)	Volume (Liters)
PSR 1	0.333	1538	24.0
PSR 2	0.560	1009	45.2
PSR 3	0.003	1009	0.2
PFR 4	0.552	1009 (*in*) 1404 (*out*) 1250 (*LMT*)	55.1
PSR 5	0.765	1451	53.1
PSR 6	1.102	1323	184.8
PFR 7	1.287	1323 (*in*) 1241 (*out*) 1281 (*LMT*)	209.0
Mixing Section			
(Physical System)	1.36	1031	178
(Active System)	1.12		125
Burnout Section			
(Physical System)	3.13	1342 (*LMT*)	410
(Active System)	2.39		447

The residence time of the physical system was calculated in the same manner that would have been used for a design calculation of a commercial SCC. It was based on the total flow, averaged measured temperature or log mean temperature, and the physical volume. The

active residence times considered only the path taken by the dopant laden kiln gas. These calculations show that the active path, identified in the networked ideal reactor model, is 1 second shorter than the superficial residence time from the design calculations. Differences are also apparent between physical volumes from the SCC geometry and active volumes from the sum of ideal reactor volumes. The difference between the active volume and physical volume in the mixing section is likely manifested as dead space or dead volume. Early investigations of probe placements indicated the presence of dead space in the mixing section adjacent to the choke wall. The gradual expansion from the choke into the burnout section prevents dead space caused by the abrupt diameter change between the mixing section and choke. The derived active volume for the burnout section is actually greater than the physical volume. This is a likely result of experimental error. Placed in perspective, the error reflects a 0.28 second error in a 3.13 second burnout section.

4.5 Summary

This chapter showed a mechanistic modeling approach to choose a network ideal reactor model. A model was chosen based on the best fit to SO_2 tracer data and consistency with physical geometry, resulting flow patterns, and temperature measurements. The chosen model is summarized in Figure 4.16 with parameters in Table 4.4. Frequency bandwidth limitation introduced by the SO_2 analyzer precludes distinguishing solutions that differ only in the higher frequencies. Thus, a best-fit model does not represent a unique solution and other considerations such as reactor geometry, analysis of flow patterns and swirl, and temperature cross section measurements also merited consideration in model selection.

The evaluation of standard errors for the fitted parameters quantified the uncertainty in the model. Within the mixing section, standard errors were generally low (less than 0.01 seconds). The exception was the parameters for the two in-series reactors PSR 3 and PFR 4 on the kiln gas stream between points 3 and 4. Further analysis revealed that the standard error for the combined mean residence times was very small, but the fraction apportioning that time between the two reactors was very high (0.944 for a 0 to 1 scale). Since the PSR and PFR represent different extremes of ideal macromixing, this indicated a large degree of uncertainty in quantifying the mixing characteristics between points 3 and 4 on the kiln gas stream. Parameters in the burnout section had larger standard errors, but were less than 0.12 second for parameters ranging from 0.77 to 1.29 seconds.. The standard error for the parameter describing the kiln and flame gas stream mixing in PSR 5 was the largest (0.155 for a 0 to 1 scale). Further analysis of model sensitivity to these mixing parameters will be covered with the application of thermo-kinetic mechanisms in Chapter 6.

Temperatures assigned using cross-section temperature profiles and energy balances enabled ideal reactor volumes to be determined. The difference between the active and physical volumes indicated the presence of 53 L of dead space in the 177 L mixing section, but found 37 L more active volume than the 410 L physical volume of the burnout section. As a percentage, this is much lower than the volume discrepancy in the mixing section. The parameter fitting process in the burnout section, which used the tracer runs in the mixing section rather than the analyzer as the driving functions, was more likely to have accentuated errors. Overall, there was good agreement between the identified model and the physical geometry of the SCC.

The discrepancy between the superficial residence time and residence time of the kiln gas reveals a potential fallacy in current design criteria. The identified networked ideal reactor model has a pathway for the kiln gas, which potentially carries unburned hazardous constituents, that takes 22 percent less time than the residence time from the design calculation. This suggests the need to incorporate some form of detailed flow analysis into design criteria. The classical chemical engineering method of identifying a multi-parameter ideal reactor model is one such approach.

CHAPTER 5

NON-IDEAL REACTOR MODELS

Mixing considerations dominate combustion modeling. The mixing and distribution of the reaction intermediates are especially important to the chain-reaction mechanism of combustion. First, the transient radicals that make up many of the intermediates in a combustion reaction promote additional reactions that either produce a radical (propagation), produce multiple radicals (chain-branching), or form stable products (termination). High temperature radicals mixing with incoming reactants sustain and stabilize the combustion process. The onset of blowout conditions is strongly effected by mixing (Barat, 1992). The second-order nature of the termination process strongly depends on the local concentration that is in part determined by the mixing process. Mixing conditions which favor high local concentrations of heavier radicals promote the carbon chain building reactions that form the larger and more complex products of incomplete combustion (PIC). This chapter will explore these aspects of mixing in the context of non-ideal models.

While the previous two chapters developed a model of mixing in the Secondary Combustion Chamber (SCC) based on a network of ideal reactors, this chapter will explore the middle ground between ideal mixing limits to assess the impact of non-ideal mixing on the combustion model. The ideal reactors used in the network, the perfectly stirred reactor (PSR) and the plug flow reactor (PFR), represent the two extremes of mixing: the PSR introduces the feed fluid which is then immediately and perfectly mixed with fluid already in the reactor,

and the PFR passes a non-mixing plug of fluid through the reactor. Another ideal limit can be described by further subdividing the mixing in the PSR into "macromixing" in the bulk fluid and "micromixing" at the molecular level; concepts that were notably advanced by Danckwerts (1957, 1958) and Zweitering (1959).

The bulk movement of fluid in a stirred reactor driven by agitation or recirculation is the mixing force often associated with macromixing while molecular diffusion is the mixing force often associated with micromixing. David (1994) disagrees with this description. He takes a stricter view by associating the types of mixing with the scale of the sample space rather than the associated force driving the mixing. He defines macromixing in stirred reactors, as "... the process leading to equal values of the average concentration in space," where micromixing involves "... processes governing the decay of physical segregation, characterized by local concentration fluctuations." The distinction is important since turbulent fluctuations and molecular diffusion contribute to both types of mixing. This definition links the residence time distribution (RTD), a bulk effluent concentration measured in time, with macromixing. However, the RTD provides no information about micromixing. Characterizing micromixing requires estimation of quantities such as the turbulent kinetic energy and dissipation rate. The literature is full of turbulent mixing models for addressing these issues, which will not be exhaustively reviewed here. This chapter, however, will focus on a simplified approach to an Interaction by Exchange with the Mean (IEM) model, known as the Partially Stirred Reactor (PaSR), for the purpose of exploring the impact of less than ideal micromixing.

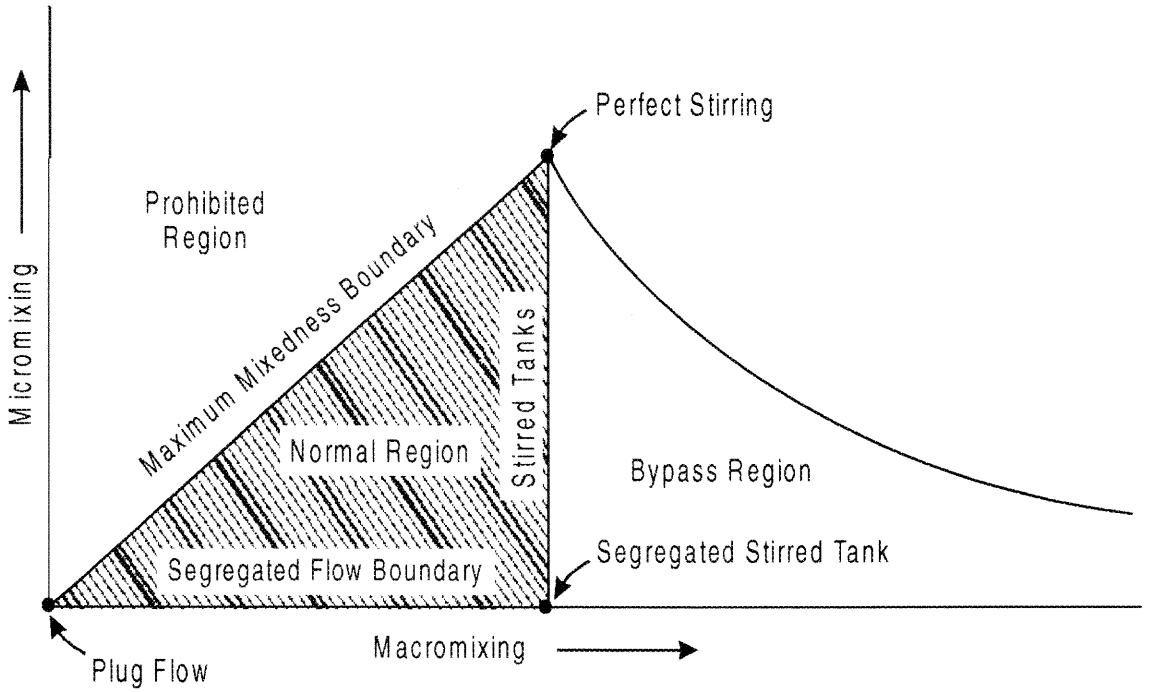


Figure 5.1 Schematic Representation of Mixing Space (Nauman and Buffham, 1983, 147)

5.1 Models Between Ideal Limits

Nauman and Buffham (1983) proposed a qualitative diagram (Figure 5.1) depicting the relationship between macromixing and micromixing and the associated reactor models. At the origin is the plug flow reactor model, a non-mixing plug of fluid moving through a reactor. As macromixing increases, micromixing is bounded by the maximum mixedness boundary until the limit is reached at the PSR model. Perfect mixing implies a homogeneous composition and temperature throughout the reactor at the macro scale (perfect macromixing) and the micro scale (perfect micromixing) and a characteristic exponential RTD. At the other extreme of micromixing, but the same degree of macromixing, is the segregated stirred tank model. It models a uniform composition when averaged at a macro scale, but can have wide fluctuations at the at the micro scale. It has the same RTD as a PSR since this is a

characteristic of macromixing. The cords that attach the PSR and the segregated stirred tank to the origin bound the normal region. Nauman and Buffham offer rigorous proof that for simple reaction systems – those with rate equations that can be reduced in terms of a single concentration variable – that maximum and minimum reactor performance occurs on these limits of the normal region. While the bounding of reactor performance is not assured for complex and multiple reactions, exploration of these bounds reveal the sensitivity of the system to micromixing.

The two models that form the cords bounding the normal region are the maximum mixedness model (Zwietering, 1959), and the segregated flow model (Dankwerts, 1958). In the segregated flow model, mixing only occurs between molecules with the same ages within the reactor until mixing with the bulk flow at the exit and thus as late as possible. Mixing in the maximum mixedness model is the reverse of the segregated flow model and occurs between molecules of different ages as early as possible. For each model, the macromixing, characterized by the RTD, is the independent variable, where the micromixing is intrinsic to the model and restricted to the bounds of Figure 5.1. The interior of the normal region may be explored by using other models with different intrinsic levels of micromixing such as the Axial Dispersion model, or using models with a independently adjusted level of micromixing. These include the coalescence and dispersion model (Curl, 1963) and its extension by Kridiotis *et al.* (1989). They allow the adjustment of the micromixing rate that requires the estimation of the coalescence rate. Another approach with a single adjustable parameter is to model the micromixing by the rate of relaxation of local conditions toward the mean

conditions of the reactor. This approach is known as Interaction by Exchange with the Mean (IEM). For a specified level of macromixing, the IEM model can explore the normal region between the maximum mixedness and segregated flow models.

5.1.1 The Segregated Flow and Maximum Mixedness Models

The concept of a segregated flow can be illustrated in a binary mixture, *A* and *B*. The local mole fractions, the values at a specific "point," for *A* and *B* are the values x_A and (1- x_A) respectively. The global mole fraction for *A* is the expected (weighted average) value of the local values, $\langle x_A \rangle$, and the value for *B* is $(1 - \langle x_A \rangle)$. Danckwerts (1958) defines the degree of segregation in terms of the mean square segregation:

$$I = \frac{\left\langle \left(\langle x_A \rangle - x_A \right)^2 \right\rangle}{\langle x_A \rangle (1 - \langle x_A \rangle)}. \qquad (5.1)$$

When the local concentration is the same as the global concentration (perfect micromixing), *I* has the value of 0, and when the mixture is perfectly segregated (i.e. $x_A = 0$ or 1) then *I* has a value of 1. Danckwerts refers to these local concentrations as the concentration at a "point," which is a small region of fluid large enough to contain many molecules, but small compared to the whole region. He then generalizes the concept by defining segregation in terms of an age rather than a concentration distribution in a "point." This concept parallels David's definitions of macromixing as process leading to equal values of concentration space irrespective of local physical segregation between "points" and micromixing as the process of decaying that segregation.

At one extreme, a stirred reactor can exhibit the same features of the macromixing found in a PSR (ideal macromixing), but with complete segregation on the micro scale. This extreme is the segregated stirred tank reactor. The model may be conceived of as "points" moving through a stirred reactor, each completely isolated and behaving as a batch reactor. Each "point" enters the reactor with the same composition, emerges with a composition dependent on the time spent in the reactor, and then mixes with other emerging "points" at the exit. The RTD weights the sum of these emerging "points." When the "points" are sufficiently small, this can be express as an integral:

$$Y_k = \int_0^{\infty} \int_0^{t'} \frac{\dot{w}_k W_k}{\rho} dt\, E(t')\, dt' \tag{5.2}$$

where $E(t') = \frac{1}{\tau} e^{-\frac{t'}{\tau}}$ and t' is the dummy time variable.

Here, the RTD reflects the stirred tank limit of complete macromixing, but the formula could use any arbitrary RTD between the stirred tank limit and the impulse or delta function for a plug flow, $E(t) = \delta(t - \tau)$. An alternate conceptualization for the same model is a PFR of infinite length with side streams drawn at the positions and volumetric flows consistent with the specified RTD (Figure 5.2a). This illustrates how the "points" remain segregated and mix only at the latest possible time as they emerge from the reactor. This model forms the segregated flow boundary in Figure 5.1.

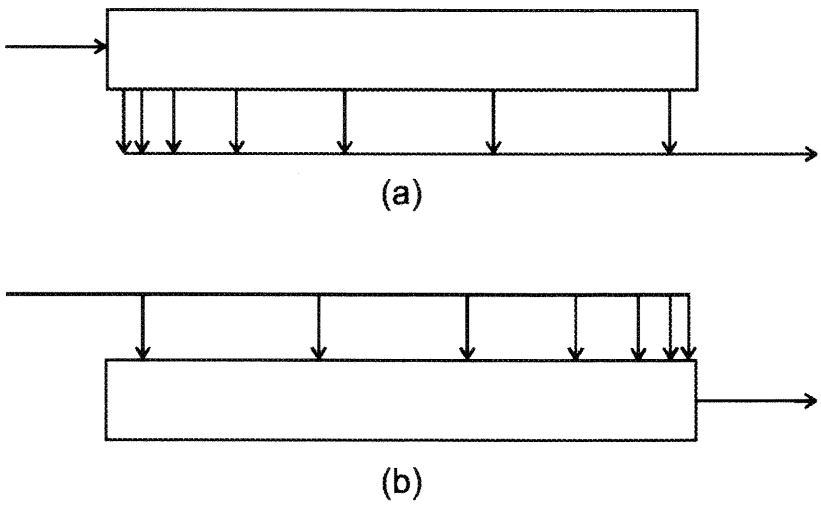


Figure 5.2 (a) Segregated Flow Model, (b) Maximum Mixedness Model

Zwietering developed a model that maximized the amount of mixing for an arbitrary RTD called maximum mixedness. Figure 5.2b shows the conceptual view of this model as the opposite of the segregated flow model. While the segregated flow model has mixing between "points" at the latest possible moment in the exit, the maximum mixedness model mixes the "points" at the earliest opportunity on entry into the reactor. The PFR analogy in Figure 5.2b shows the influent steams divided down the length of the reactor corresponding to infintesmal entry "points" determined by the RTD. At any given moment in the reactor a "point" will have an age of α and have life expectancy or residual life of λ. These quantities are related to the total time in the reactor by the equation:

$$t = \alpha + \lambda. \tag{5.3}$$

Again, using the Figure 5.2b as the conceptual guide, "points" entering the reactor mix with "points" having the same time remaining to the exit (i.e the same life expetency, λ). A mass

balance equation can be developed from this prespective. Using this view, the governing equation is (Zwietering, 1959):

$$\frac{dY_k}{d\lambda} = -\frac{\dot{w}_k W_k}{\rho} + \frac{E(\lambda)}{1 - F(\lambda)}(Y_k - Y_k^*) \qquad (5.4)$$

where Y_k^* is the mass fraction of species k in the feed. Using time as the independent variable, $F(t)$ is the cumulative distribution function that gives the probability a molecule will exit from the reactor by time t and relates to the RTD (a probability density function) by $E(t)=dF(t)/dt$. For a stirred tank, $F(t) = 1 - e^{-t/\tau}$, and with complete macromixing, $dY_k/d\lambda = 0$ for all λ, which reduces Equation 5.4 to the PSR formula (Equation 4.1). When macromixing is incomplete, values of the average concentration in space are no longer equal. There exists life expectancies, λ, where $dY_k/d\lambda \neq 0$. Integration requires the boundary conditions of $dY_k/d\lambda = 0$ for $\lambda = \infty$.

Zwietering uses this model as a means of analyzing the degree of segregation as a function of macromixing and micromixing. Following Danckwerts, he generalizes the degree of segregation using the age distribution of the molecules, $\varphi(\alpha)$, in the reactor. The age distribuion, $\varphi(\alpha)$ (a probability density function), relates to $F(t)$ by $\varphi(\alpha) = (1/\tau)[1 - F(\alpha)]$. Within a "point," the same distribution is given by $\varphi_p(\alpha)$. Using these distributions, the degree of segregation is defined as the ratio of the variance of ages between "points" to the variance of ages within the reactor:

$$J = \frac{\left\langle (\alpha_p - \langle\alpha\rangle)^2 \right\rangle}{\left\langle (\alpha - \langle\alpha\rangle)^2 \right\rangle} = \frac{\frac{1}{V}\int_V (\alpha_p - \bar{\alpha})^2 dv}{\int_0^\infty (\alpha - \bar{\alpha})^2 \varphi(\alpha) d\alpha} \qquad (5.5)$$

where, $\bar{\alpha} = \int_0^\infty \alpha\varphi(\alpha)d\alpha$ and $\alpha_p = \int_0^\infty \alpha\varphi_p(\alpha)d\alpha$.

The denominator is dependent only on the macromixing while the numerator is a function of micromixing. For instance, when the flow within the reactor is completely segregated each "point" has a uniform age. So the variance between "points" equals the variances of ages within the reactor and $J = 1$. When micromixing is complete as in a PSR each "point" has a distribution of ages the same as the reactor distribution of ages, or $\varphi(\alpha) = \varphi_p(\alpha)$. Thus, α_p equals α, which makes the variance between "points" and the degree of segregation, J, equal to 0.

Zwietering provides rigorous proof that the maximum mixedness model minimizes J. To illustrate this fact, a comparison is made between four competing models for a single stirred tank reactor with a RTD equivalent to two PSRs in series ($E(t) = 4t/\tau^2 e^{-2t/\tau}$). If this RTD provides the only characterization of the reactor, it can be modeled as with any one of four "ideal" models: a segregated flow (Equation 5.2 using $E(t) = 4t/\tau^2 e^{-2t/\tau}$), two segregated stirred tanks in series with each using $E(t) = (1/\tau) e^{-t/\tau}$, two PSRs in series, and the maxmixedness model using $E(t) = 4t/\tau^2 e^{-2t/\tau}$. All models have the macromixing characterized by the same RTD, but each exhibits different micromixing characteristics exhibited by different degrees of segregation, J (1.00, 0.714, 0.143, and 0.0275, respectively).

The different degrees of segregation have a significant impact for the fractional conversion of a notional second order reaction. Figure 5.3 illustrates this effect on fractional conversion of reactant A for these four models plotted against the reactor Damkohler number, $Da = k\tau C_{A0}$. A higher degree of segregation allows for "points" with higher concentrations, which favor the second order reaction and produce a larger fractional conversion.

The segregated flow and maximum mixedness models examine the bounds of micromixing. Particularly, they illustrate the effect of greater segregation on second or higher order reactions. This effect is especially important when modeling the formation of products of incomplete combustion. For example, carbon chain building reactions usually involve

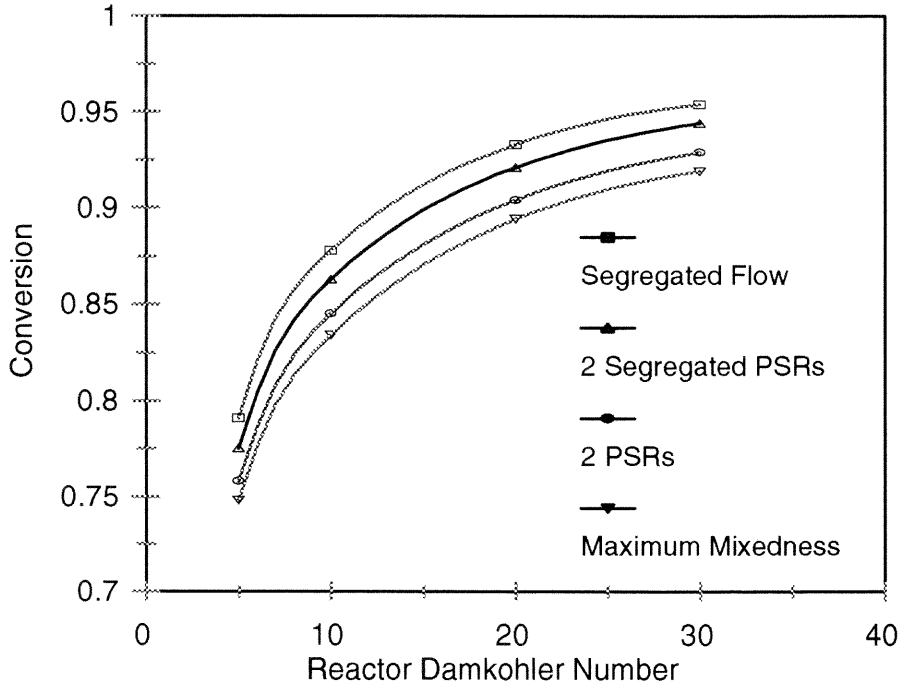


Figure 5.3 Comparison of Reactor Models for a Second Order Reaction in a Single Stirred Reactor with an Two PSR in Series RTD, $E(t) = 4t/\tau^2 e^{-2t/\tau}$ (spline fit on model points from Zwietering, 1959)

second order steps involving two radicals joining to form a stable compound. The next step is to explore this effect in the middle ground between these models. One such approach is the interaction by exchange with the mean (IEM) model.

5.1.2 Interaction by Exchange with the Mean (IEM) Model

The IEM model continues the application of the Danckwerts definition of the fluid "point." Consider a reactive flow with a composition vector that defines the concentrations of all species as a function of position and time, $\xi(x,t)$. The change in composition of a "point" as it moves through a reactive flow is given by (Pope, 1990):

$$\frac{D\xi}{Dt} = \Gamma\nabla^2\xi + S \tag{5.6}$$

where Γ is the vector of diffusion coefficients and S is the reaction rate vector, which is also a function of ξ. The substantial derivative, $\frac{D}{Dt}$, gives a Lagrangian view that follows the "point" in a flow and observes changes in composition with respect to time. Equation 5.6 shows that the change of composition of a "point" in time as it moves with the fluid is the sum of changes due to diffusion with its surroundings (the first term) and the reaction in that "point" (second term). The diffusion in a turbulent fluid is a complex process that lacks the required closure for an analytical solution. So assumptions or empirical equations are required to solve the problem. Pope proposes a simple first order closure by assuming a statistically homogeneous and constant density reactor. The closure term for the expected value of the diffusion term, conditional on a specific composition, ζ, is:

$$\langle \Gamma \nabla^2 \xi | \zeta \rangle = -\frac{1}{2} C_\xi \omega (\zeta - \langle \xi \rangle). \tag{5.7}$$

Diffusion has been simplified to the relaxation of the specific composition to the mean composition regulated by the mean turbulence rate, ω, and a constant C_ξ. Pope assigns a value of 2 to the constant and the mixing frequency, ω, may be derived from turbulent mixing parameters by the ratio of the turbulent dissipation, ε, to the turbulent kinetic energy, k. The spatial dependence may be removed by defining the composition vector as a function of only t, $\xi^+ = \xi(x^+[t], t)$ where x^+ denotes the position of the "point". From this, Pope proposes the following model:

$$\frac{d\xi^+}{dt} = -\frac{1}{2} C_\xi \omega (\xi^+ - \langle \xi \rangle) + S(\xi^+) \tag{5.8}$$

Equation 5.8 governs the material balance for the "point" moving through a stirred reactor. The underlying assumption of a statistically homogenous composition applies only when the macromixing produces these conditions throughout the reactor, which occurs when the RTD, $E(t) = (1/\tau) e^{-t/\tau}$. Because the equation governs the moving particle, Pope suggests a Monte Carlo approach for the solution. This model is the Partially Stirred Reactor Model.

5.2 The Partially Stirred Reactor Model

A partially stirred reactor (PaSR) model assumes a reactor with constant density, a homogeneous turbulence, and a statistically homogeneous composition. These conditions are reasonably approximated at the stirred tank boundary (Figure 5.1) between the PSR and Segregated Stirred Tank models. Correa and Braaten (1993) extended the concept of Equation 5.8 and developed and tested a Monte Carlo model shown in Equations 5.9 and 5.10. The model uses an array of N_p "points" (termed "particles" by the authors) each with compositions independent of position. At each step, "points" are added to the array and randomly selected for removal. The removed "points" mix to form the effluent stream. The governing mass balance for the n^{th} particle is:

$$\frac{dY_k^{(n)}}{dt} = \frac{\dot{w}_k W_k}{\rho} - C_\xi \omega \left(Y_k^{(n)} - \bar{Y}_k \right) \tag{5.9}$$

The energy balance equation (Equation 5.10) is the same as that for a PFR. The first term on the right is for reaction and diffusion, and the second term is for heat loss through walls of the reactor:

$$\bar{c}_p \frac{dT^{(n)}}{dt} = -\sum_{k=1}^{kk} h_k(T) \frac{dY_k^{(n)}}{dt} - Q_{loss} \tag{5.10}$$

The reactor average is updated at each time interval.

The Monte Carlo approach imposes the probability density function that approaches the RTD of a single PSR for a sufficiently large value of N_p (Correa & Braaten, 1993). While this method provides an intuitive approach to solving Equation 5.8, it is computationally inefficient. If a chemical kinetic mechanism with a large number of species, N_s, is used, the problem becomes computationally prohibitive to solve, since $(N_s + 1) \times N_p$ ODEs are generated. A study conducted by Correa and Braaten (1993) using a chemical mechanism for CH_4/air combustion with 27 species and 650 "points" (requiring the solution of 8,200 ODEs) required 10 hours on an Intel iPSC/860 Supercomputer. The inefficiency comes from solving multiple "points" with the same differential equations and subject to the same boundary conditions.

5.2.1 An Alternative Approach

If statistical homogeneity is held, the above scheme is inherently inefficient. When the system is at steady state, the composition of a "point" depends solely on the amount of time it spends in the reactor. Say two "points" entered the reactor operating at steady state and spent time t_1 and t_2 in the reactor with $t_1 < t_2$. At time t_1, both "points" would have the same composition. Point 2 would continue to react until it is withdrawn at time t_2. Up to time t_1, the same set of differential equations were solved twice. Since ODE solver algorithms exert the most computational effort when composition is changing the quickest (in the first microseconds), essentially twice the computational effort is required to gain redundant information.

An alternate approach uses the PFR conceptualization of the segregated flow model in Figure 5.2a as a starting point. At steady state, "points" with equal ages will have the same composition. As such, the integration can proceed like that of a PFR rather than an array of discrete batch reactors. The governing equation is the same as Equation 5.9, but with the superscript dropped:

$$\frac{dY_k}{dt} = \frac{\dot{w}_k W_k}{\rho} - C_\xi \omega \left(Y_k - \bar{Y}_k\right) \tag{5.11}$$

with the initial condition of $Y_k = Y_k(0)$. The average composition at steady state is the weighed by the probability density function for the age distribution, $\varphi(\alpha)$:

$$\bar{Y}_k = \int_0^\infty \varphi(\alpha) Y_k(\alpha) d\alpha. \tag{5.12}$$

Since $\varphi(\alpha)$ vanishes quickly, the integration may be stopped at a sufficiently large value of α. For this model, integration continues until 99.9999% of the mass is accounted for and the remainder is extrapolated from the last value of Y_k. In differential form, Equation 5.12 becomes:

$$\frac{d\bar{Y}_k}{d\alpha} = \varphi(\alpha) Y_k(\alpha). \tag{5.13}$$

Like the segregated flow model, the exit composition is weighted by the residence time distribution, $E(t)$. With the ideal macromixing required for the homogeneous composition,

the distributions are the same, $E(t) = \varphi(\alpha)$. This is not the case for other RTDs that will not produce a homogenous composition. The exit composition is given by:

$$\frac{dY_{k\ (out)}}{dt} = E(t)Y_k(t) \tag{5.14}$$

and is equal to the average composition in the reactor for the assumed conditions. As with composition, the segregated flow analogy applies to the calculation of local temperature. The temperature of a fluid "point" moving down the PFR analogy shown in Figure 5.2a is given by the PFR heat balance (Equation 4.4). The exit temperature is derived from the exit enthalpy, which is the average weighed by the RTD:

$$c_p\frac{dT}{dt} = -\sum_{k=1}^{kk} h_k\frac{dY_k}{dt} - \frac{Q}{\rho V}$$

$$\frac{d\bar{H}}{dt} = E(t)\sum_{k=1}^{kk} h_k Y_k \tag{5.15a\&b}$$

To integrate Equation 5.11, the average composition, $\bar{Y}_k$, must be known, but it cannot be since 5.13 must be integrated first. A solution is found through successive substitution starting with the solution to a perfectly stirred reactor (PSR) as the first guess. The process is continued until the solution converges. The convergence criteria is:

$$RTOL \geq \sqrt{\sum_{k=1}^{kk+1}\left(\frac{Z_k^{(n)} - Z_k^{(n-1)}}{Z_k^{(n)}}\right)^2} \tag{5.16}$$

where $Z^{(n)} = \left(T^{(n)}, Y_1^{(n)}, Y_2^{(n)}, \ldots, Y_{kk}^{(n)}\right)$.

Z_k is rejected from the tolerance calculation if $Z_k < ATOL$, an absolute tolerance. This prevents division by numbers outside the ODE solver's tolerance. This technique of successive substitution will be referred to as the direct integration and convergence method.

5.2.2 Modified Newton Convergence

For larger mechanisms, successive substitution becomes a computationally expensive technique. The convergence is ultimately stable, but only small steps are taken toward the solution of $\bar{Y}_k$. The multi-variable Newton's Method could be used as a convergence technique to improve efficiency.

The covergence routine solves for $\bar{Y}_k$ through a series of successive guesses, starting with the PSR solution as the initial guess, which eventually converge to the solution. Newton's method requires an objective that approaches zero at convergence. To create this, Equation 5.11 is placed in integral form and substituting it into Equation 5.12 to give:

$$\bar{Y}_k = \int_0^{\infty} \varphi(\alpha) \int_0^{\alpha} \left[\dot{w}_k W_k / \rho - C_\xi \omega \left(Y_k(t) - \bar{Y}_k \right) \right] dt\, d\alpha \tag{5.17}$$

From Equation 5.17, an objective equation is created for species k:

$$f_k = \bar{Y}_k - \int_0^{\infty} \varphi(\alpha) \int_0^{\alpha} \left[\dot{w}_k W_k / \rho - C_\xi \omega \left(Y_k(t) - \bar{Y}_k \right) \right] dt\, d\alpha = 0 \tag{5.18}$$

Successive iterations are made for $\bar{Y}_k$ until Equation 5.18 approaches zero. Newton's method finds each successive iteration in a single variable equation by finding the intersection

of the x-axis with a line tangent to the current guess. In multidimensional form, the next guess (superscript n+1) is found from the previous iteration using:

$$\begin{aligned} \boldsymbol{Y}^{(n+1)} &= \bar{\boldsymbol{Y}}^{(n)} - \boldsymbol{J}^{-1}\boldsymbol{F} \\ \text{where:} \quad \bar{\boldsymbol{Y}} &= [\,Y_1, Y_2, \ldots, Y_{kk}\,]^T \\ \text{and:} \quad \boldsymbol{F} &= [f_1, f_2, \ldots, f_{kk}]^T \\ \text{and:} \quad \boldsymbol{J} &\text{ is the Jacobian of } \boldsymbol{F} \end{aligned} \tag{5.19}$$

Finding the inverse of the Jacobian matrix is the most computationally demanding step of each iteration, but this can be reduced by using the same Jacobian for several iterations.

5.2.3 A Generalized PaSR Model for Non-Ideal Macromixing

The development of the PaSR model relied on ideal macromixing characterized by the exponential RTD. Ideal macromixing produces the statistically homogeneous composition within the reactor required for the closure model in Equation 5.7. Unfortunately, this qualification restricts the model to the "stirred tank" line in Figure 5.1. Models that describe the normal region of this figure must contend with degrees of macromixing where assumptions of statistically homogeneous compositions no longer apply. In this section, a generalized closure model is proposed that remains valid throughout the normal region. It still assumes a homogeneous and isotropic turbulence field that produces uniform micromixing conditions describable by a single parameter. Additionally, the closure model must remain valid as it converges to the limiting conditions on the boundaries of the normal region: maximum mixedness, segregated flow, and (partially) stirred tank.

As in section 5.1.2, the new closure term is based on the Lagrangian view of a "point" traveling through the reactor. Equation 5.20 proposes the closure term for the expected value of the diffusion term in Equation 5.6, conditional for a specific composition, ζ, as the relaxation with other "points" having the same life expectancy, λ (where the prime denotes a specific life expectancy), driven by the mixing frequency, ω.

$$\left\langle \Gamma \nabla^2 \xi \middle| \zeta \right\rangle = -\frac{1}{2} C_\xi \omega \left(\zeta - \left\langle \xi \middle| \lambda' \right\rangle \right) \tag{5.20}$$

Nauman and Buffham (1983) argue that limiting interactions to "points" with the same life expectancy is necessary for a model fundamentally consistent with the second law of thermodynamics. It is also consistent with the previous development of the closure model (Equation 5.7). Showing that, for ideal macromixing, "points" with same life expectancy have the same age distribution as that for all "points" in the reactor. This occurs when the age distribution, $\varphi(\alpha) = 1/\tau[1 - F(\alpha)]$, and residual life distribution, $\psi(\lambda) = 1/\tau[1 - F(\lambda)]$, are independent. Given the exponential RTD for ideal macromixing, $E(t) = (1/\tau)e^{-t/\tau}$, independence may be demonstrated by showing that the joint probability density (defined by Nauman and Buffham, 1983, 149) is the product of the two component pdfs:

$$j(\alpha,\lambda) = \frac{E(\alpha+\lambda)}{\tau} = \frac{1}{\tau}\left(\frac{1}{\tau}e^{-(\alpha+\lambda)/\tau}\right) = \left(\frac{1}{\tau}e^{-\alpha/\tau}\right)\left(\frac{1}{\tau}e^{-\alpha/\tau}\right) = \varphi(\alpha)\psi(\lambda). \tag{5.21}$$

This independence occurs only at the "stirred tank" line in Figure 5.1 and shows the consistency between Equation 5.7 and 5.20.

As with the development of the PaSR model, the governing equations for the generalized PaSR model considers a single "point" moving through the reactor. Like the maximum mixedness model where mixing only occurs between "points" (Zwietering, 1959) with the same life expectancy, Figure 5.2b as a stirred reactor analog provides a useful conceptual tool. "Points" enter a tubular reactor along its length in a manner consistent with the RTD. After a "point" enters the reactor, it moves toward the exit with a decreasing life expectancy as it interacts with other "points" entering the reactor and moving with it toward the exit. This can be described by:

$$\frac{dY_k^{(n)}}{d\lambda} = -\frac{\dot{w}_k W_k}{\rho} + \frac{1}{2} C_\xi \omega \left(Y_k^{(n)} - \langle Y_k | \lambda \rangle \right). \tag{5.22}$$

The conditional expected value of the composition considers only the "points" with the same life expectancy. In integral form, this is:

$$\langle Y_k | \lambda' \rangle = \int_0^\infty \varphi(\alpha|\lambda') Y_k(\alpha,\lambda') d\alpha$$
$$\text{where: } \varphi(\alpha|\lambda) = \frac{j(\alpha,\lambda)}{\psi(\lambda)} = \frac{E(\alpha+\lambda)}{1-F(\lambda)}. \tag{5.23}$$

The composition function, $Y_k(\alpha,\lambda)$, is more difficult to define as a continuous function since composition of the "point" is a function of age and life expectancy. Discrete methods for solution are easier to imagine. For instance, consider a large number of "points," N_p, exiting the reactor at the same time. Each entered the reactor at a separate time consistent with the

RTD, but moved through the reactor together after entry. This requires the solution of N_p distinct batch reactors interacting as they move through the reactor. The mean is calculated by a discrete average of the "points," but the batch reactors are solved continuously.

The two limiting cases occur at the bounds of the mixing frequency, ω. When $\omega = 0$ the second term, the mixing term, equals zero, and Equation 5.22 describes a segregated "point" moving through the reactor and reduces to Equation 5.2, but with an arbitrary RTD. When $\omega = \infty$ the mixing between "points" of equal life expectancy is instantaneous, so difference between the composition of the "point" and the conditional expected value is zero. The mass balance using the composition of the feed and the composition as a function of life expectancy follows the same development as the Zwietering maximum mixedness model (Equation 5.4). Thus, the generalized PaSR model moves between the segregated flow and maximum mixedness boundaries depending the mixing frequency. Thus, Equation 5.22 provides a reasonable candidate for a single parameter micromixing model to describe the normal region of Figure 5.1.

5.3 Comparison to Literature Results

A computer program was developed to solve the governing equations (5.11, 5.13, and 5.15) of the PaSR model. Both sucessive substitution and a modified Newton's convergence routines were employed to determine the steady state averages of the composition vector. To validate the computer model, a comparison was made with the results found in the literature of a PaSR model that used a Monte Carlo method for solution.

Four test systems were used. The first three attempted to reproduce results found by the Correa (1993, 1995), and Correa and Braaten (1993). These included: a CO/H_2 mechanism with 11 species and 23 elementary reaction steps (Correa, 1995) , a 18 species - 43 step CO/H_2 mechanism including nitrogen oxidation (Correa, 1993), and a 27 species - 77 step CH_4 oxidation mechanism including nitrogen oxidation (Correa and Braaten, 1993).

The 11 species CO/H_2 mechanism simulated an input temperature of 900 K at 1 atm with a 50-50 mixture of CO/H_2, and with air to achieve a fuel/air equivalence ratio of 0.3159. The simulation was originally done on a PC with a Intel 486DX2/66 microprocessor. Convergence for 17 successive runs for frequencies ranging between 10 to 10^5 took 56.5 minutes. The convergence time peaks at 5623 Hz, but took less than ten minutes at this frequency. The reactor conditions were set to agree with those specified for the Monte Carlo

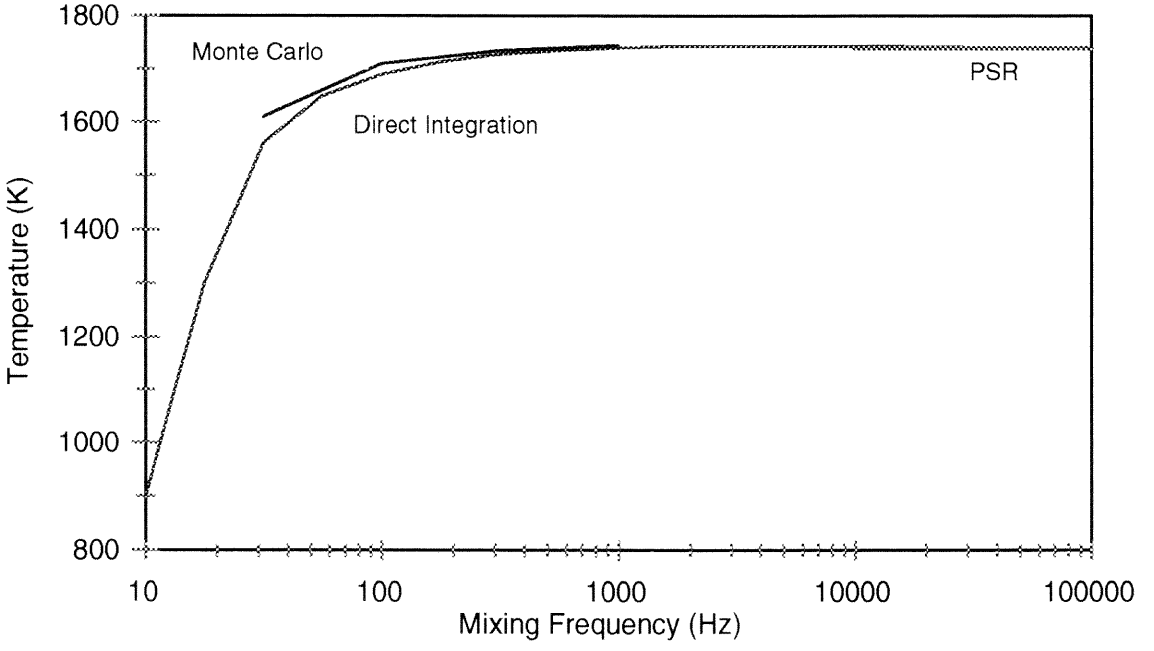


Figure 5.4 Effect of Mixing Frequency on Reactor Temperature for CO/H_2 and Air Combustion, Convergence Technique Compared to Monte Carlo Solution (Correa, 1995)

methods in by Correa (1995). Results from the convergence technique show excellent agreement with the published results from the Monte Carlo method for temperature (Figure 5.4), CO concentration (Figure 5.5), and OH concentration (Figure 5.6). The differences might be attributed to using slightly different reactor conditions, differences in the integration method (Monte Carlo vs. direct integration), or having different thermodynamic files.

The results for a relatively simple CO/H_2 mechanism reveal some basic relationships between micromixing – characterized by the mixing frequency – the reactor conditions, and the products of combustion. Figure 5.4 shows a strong relationship between temperature and mixing frequency. Reactor temperature depends on the completeness of reaction. The lower mixing frequencies (or greater segregation) inhibit the overall reaction. The fuel-lean mixture reacting near the combustion limit magnifies this effect and goes to a blowout state as the

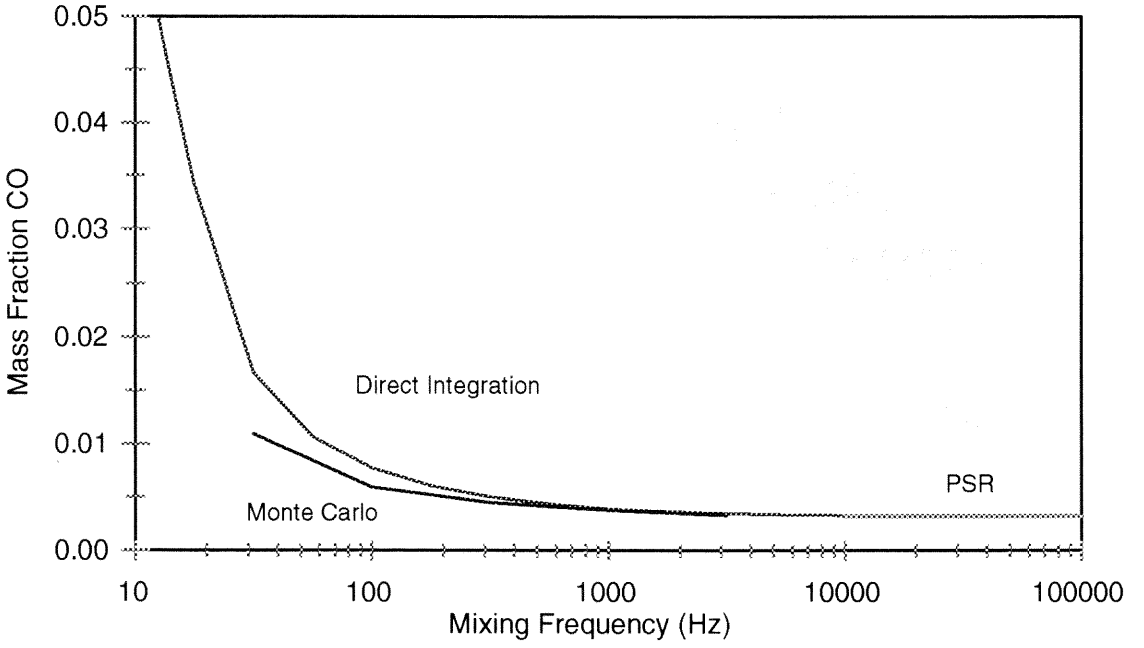


Figure 5.5 Effect of Mixing Frequency on CO Mass Fraction for CO/H_2 and Air Combustion, Convergence Technique Compared to Monte Carlo Solution (Correa, 1995)

mixing frequency approaches zero. Figure 5.5 reflects the same relationship between mixing frequency and completeness of reaction; lower mixing frequencies correspond to higher concentrations of CO, a reactant. The dependence of OH concentration on mixing frequency (Figure 5.6) again shows this relationship to completeness of reaction, since the hydroxyl radical is an intermediate between the reactant, hydrogen, and the product, water vapor. Additionally, the lower temperatures at the lower mixing frequencies shift the equilibrium with the hydroxyl radical toward the less reactive peroxyl radical (HO_2).

The 18 species - 43 step CO/H_2 mechanism including nitrogen oxidation (Correa, 1993) took longer to converge. Correa and Braaten use this system as their bench mark, which took 166 hours to solve on a Sun SPARC II workstation. The convergence method solved the problem much quicker and took no more than 1.4 hours for any mixing frequency. Finally, use of the direct integration method on the 27 species - 77 step CH_4 oxidation mechanism including nitrogen oxidation showed comparable results to those found in literature by the monte carlo method (Correa and Braaten, 1993).

Better computational performance allowed for the use of more complex mechanisms and analysis of a greater range of conditions. These comparisons were originally made using a PC with a Intel 486 DX2/ 66 MHz processor, which has the same computational efficiency as a Sun SPARC II workstation (MATLAB benchmark test). More modern processors such as a Pentium III / 800 MHz run at approximately 27 times that speed (as per the MATLAB benchmark test). Combining greater computational power and this more efficient algorithm provides the capability to rapidly solve detailed thermo-kinetic mechanisms while applying mixing conditions that more closely resemble real systems.

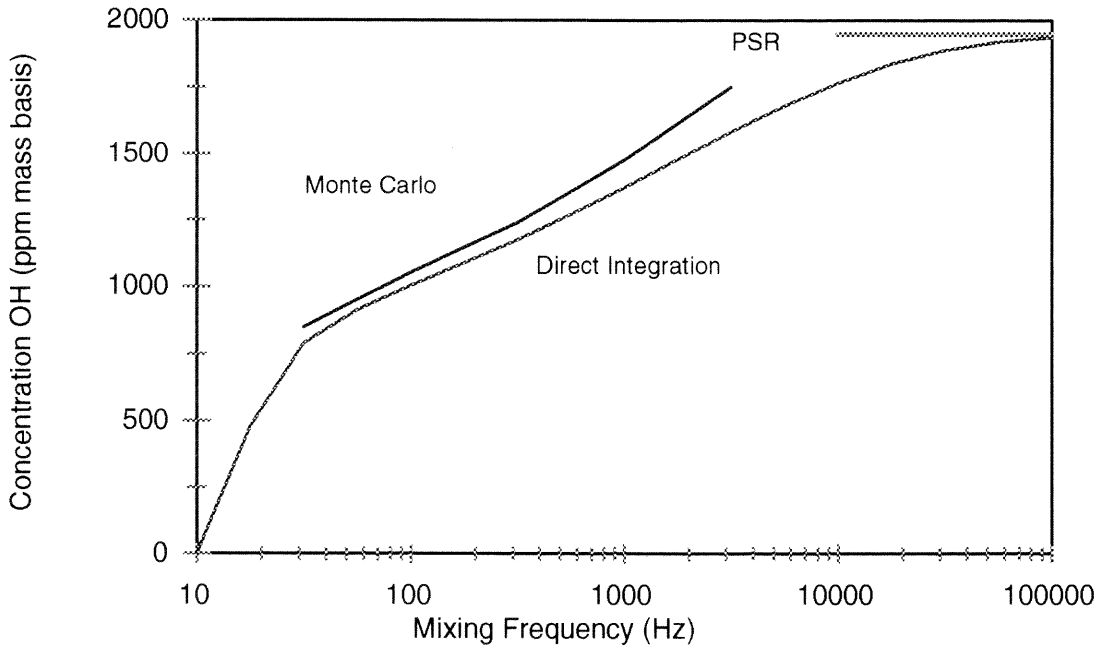


Figure 5.6 Effect of Mixing Frequency on OH Concentration for CO/H_2 and Air Combustion, Convergence Technique Compared to Monte Carlo Solution (Correa, 1995)

5.4 Comparison to Experimental Results: Toroidal Jet Stirred Combustor (TJSC) Near Blowout

The Toroidal Jet Stirred Combustor (TJSC) is a convenient system to use for comparing experimental data with the Partially Stirred Reactor (PaSR) model. The design of the combustor promotes a homogeneous composition throughout the reaction volume. This corresponds to the exponential RTD characteristic of the stirred reactor line in Figure 5.1 that can be modeled by the PaSR model (Equations 5.11 through 5.14). Additionally, analysis has been performed on the micromixing characteristics of the combustor that can be used to estimate the mixing parameter ω. This section compares the PaSR model to a series of

experiments conducted using a TJSC in fuel lean conditions near blowout. A favorable comparison is found between data and model, but an additional model that describes the quenching process in the sample probe was required to get this agreement.

5.4.1 System Description

Nenniger, Kridiotis, Chomiak, Longwell, and Sarofim (1984) introduced the TJSR to improve mixing performance of spherical research reactors. The 32 jets feed a fuel/air mixture from the manifold at high velocity into the torus (Figure 5.7). The shear from the evenly distributed jets entering the torus produces a highly turbulent mixing environment within the torus that gives the homogeneous conditions of a well stirred reactor and (assumed) isotropic turbulence. From isotropic turbulence relations they estimated the turbulent mixing time, τ_e:

$$\tau_e = \left(\frac{L_0^2}{\varepsilon} \right)^{1/3} . \tag{5.24}$$

The turbulent macro scale, L_0, was estimated from the vessel geometry and the macro fluctuations of the radial cross section temperature measurements. Turbulent dissipation rate, ε, was calculated based on the assumption of complete turbulent dissipation within the vessel using the jet exit velocity and reactor mean residence time. The mixing frequency, ω, is the reciprocal of the turbulent mixing time. The calculations made by Nenniger *et al.* correspond to a mixing frequency on the order of 10,000 Hz. If turbulent dissipation is not complete within the torus, as assumed, the actual mixing frequency would be lower.

Barat (1990) used the TJSR to explore well stirred reactor conditions near blowout. Using C_2H_4 / air combustion under fuel lean conditions ($\Phi = 0.54$) a series of experiments were conducted that increased the dilution ratio, where:

$$\text{Dilution Ratio} = \frac{\dot{n}_{added\ N_2}\ (\text{molar flow added } N_2)}{\dot{n}_T\ (\text{total molar flow})}. \tag{5.25}$$

A water cooled sample probe extracted exit samples to measure products of incomplete combustion (PIC): CO, CH_4, C_2H_6, C_2H_4, C_2H_2. Additionally, Rayleigh scattering using a frequency-doubled Nd:YAG laser (532 nm), was used to determine the temperature distribution within the torus. Barat demonstrated the presence of a temperature distribution

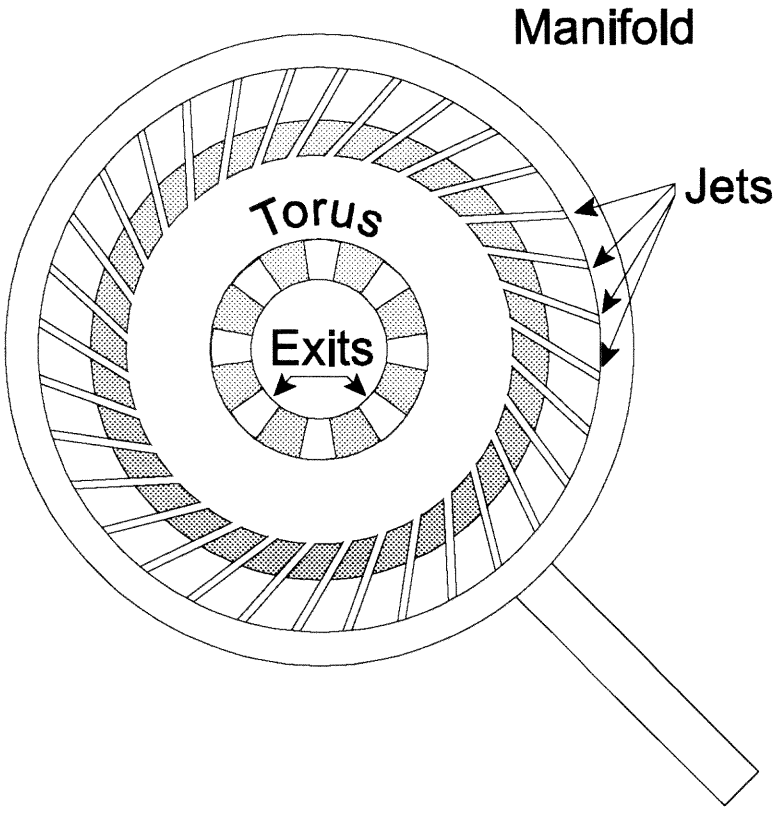


Figure 5.7 Toroidal Jet Stirred Combustor (TJSC) Radial View Cross Section (Barat 1990, 14)

that became increasingly bimodal with increasing dilution as the system approached blowout. This distribution was at the microscale between individual packets or "points" (as defined earlier in this chapter). A degree of microscale segregation existed despite the homogeneous macroscale conditions demonstrated by Nenniger *et al.* Additionally, the use of a PSR model, even when combined with a probe quenching model, consistently under predicted measured PICs. The following sub-sections show the comparison between these experiments and the PaSR model.

5.4.2 Modeling Approach

The PaSR model (Equations 5.11 through 5.15) with the exponential RTD characteristic of a well-stirred reactor was used to simulate the bench scale system. All simulations were run for adiabatic reactor conditions. A series of mixing frequencies – 316, 1000, 3162 and 10000 Hz (an exponential progression) – were used along with the PSR (the limiting condition) the show the effect of frequency. Model points were calculated at 0.02 intervals for dilution ratios ranging from 0.0 to 0.22.

In addition to solving the PaSR model by the methods described in section 5.2.2, a probe quench model was added to simulate the cooling process in the water jacketed sampling probe. This imposes a temperature profile on a PFR model that runs after the main PaSR

simulation. The profile was based on thermocouple measurements taken from a similar quenched probe set-up (Vaughn, 1988). The profile in differential form is:

$$\frac{dT}{dt} = -\frac{1}{2} bt^{-1/2} \left(T_0 - T_C\right) e^{-bt^{1/2}} \tag{5.26}$$

where T_0 is the input temperature, T_C is the coolant temperature, and b is a constant. The coolant temperature is 315 K and the constant, b, has a fitted value of 55.0 $s^{1/2}$.

For this simulation, the PaSR model used version 3.0 of the GRI Mechanism. The GRI Mechanism, a product of research sponsored by the Gas Research Institute, is optimized for the combustion of natural gas that includes C_1 and C_2 hydrocarbon oxidation, a limited set of C_3 (propane) oxidation reactions, and NO formation and re-burn chemistry. The GRI Mechanism uses 325 reactions with 53 species.

5.4.3 Comparison of Model to Experimental Data

Figures 5.8 through 5.11 show a comparison between experimental data and PaSR and PSR simulations given on a dry basis and corrected for dilution. The hollow triangles show the data points and filled markers show the model points that are included to help distinguish the multiple lines. Figure 5.8 shows the effect of dilution and mixing frequency on temperature. Increasing the mixing frequency, an intrinsic parameter, increases temperature toward the upper limit found from the PSR model. Increasing the dilution ratio, a system variable, decreases temperature as expected. The data points (hollow triangles) show good agreement between the experimental data and model for a mixing frequency between 3162 and 10,000 Hz.

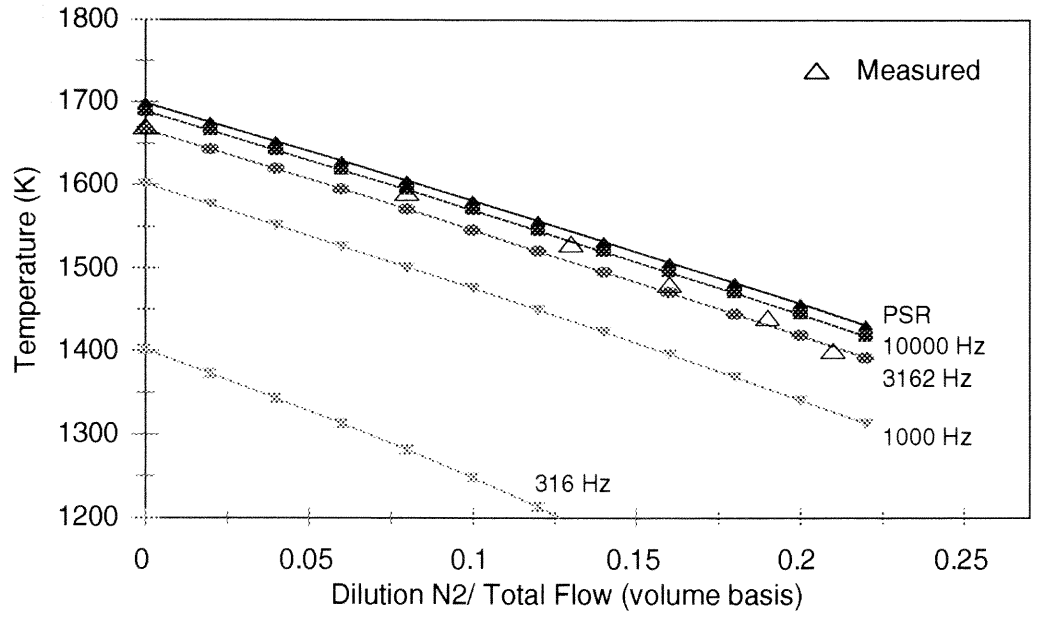


Figure 5.8 Reactor Temperature Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models

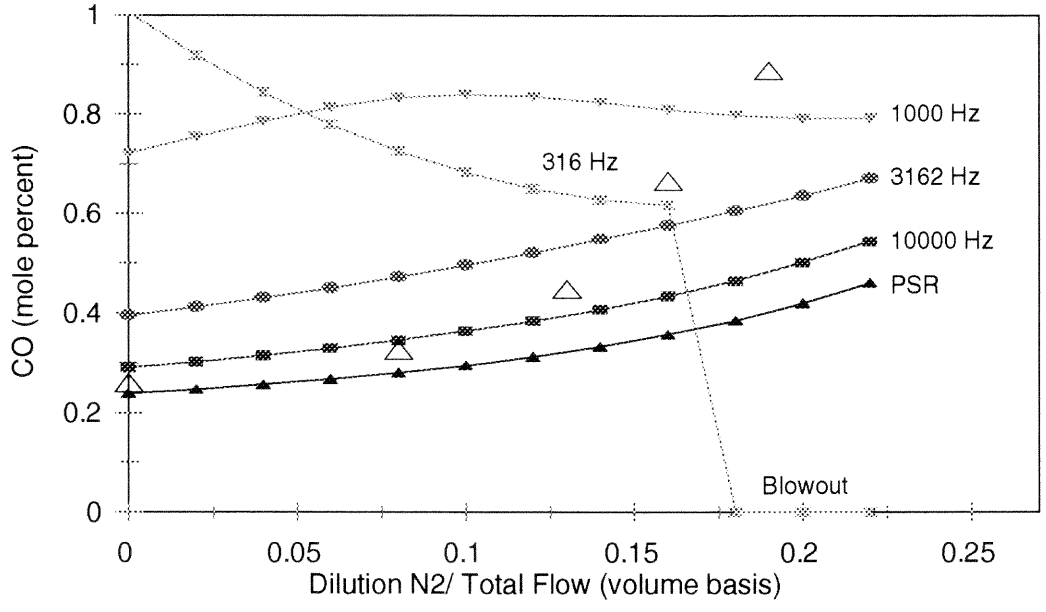


Figure 5.9 CO Concentration Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models with Probe Quench Model

The next three graphs include the probe quench model as a post process operation to the PaSR and PSR models. This corresponds to the samples taken from the TJSC using a quenched probe during the experiment. Figure 5.9 shows that lower frequencies and higher dilution ratios tend to increase CO concentration in the effluent. The CO concentration decreases for the second half of the 1000 Hz model run and the 316 Hz run until it drops to zero when blowout conditions are reached. The next section will show that the effect of mixing frequency on CO concentration is manifested only after the probe quench model and affected principally by the concentration of unburnt fuel and initial temperatures entering the probe. The experimental data showed no particular agreement with any frequency, but matched higher mixing frequencies at lower dilutions and lower mixing frequencies at higher dilution ratios.

The next two graphs show a stronger correlation between data and model. Figure 5.10 illustrates the effect of dilution and frequency on the concentration of unburnt fuel in the effluent. Decreasing the mixing frequency, which increases the degree of segregation at the microscale, significantly increases the amount of unburned fuel in the effluent. The increasing the amount of dilution, which lowers the reactor temperature, also increases the amount of unburnt fuel. The effect of dilution on unburnt fuel, as will be show later on, becomes more pronounced as a result of the probe quench model.

Figure 5.11 sums all measured hydrocarbons (C_2H_6, C_2H_4, C_2H_2, and CH_4) weighted according to the number or carbon atoms in the species. This is similar to a total hydrocarbon measurement, but ignores stable oxygenated hydrocarbons such as ketenes, aldehydes, and alcohols and larger chained hydrocarbons (all present in the GRI mechanism, but not

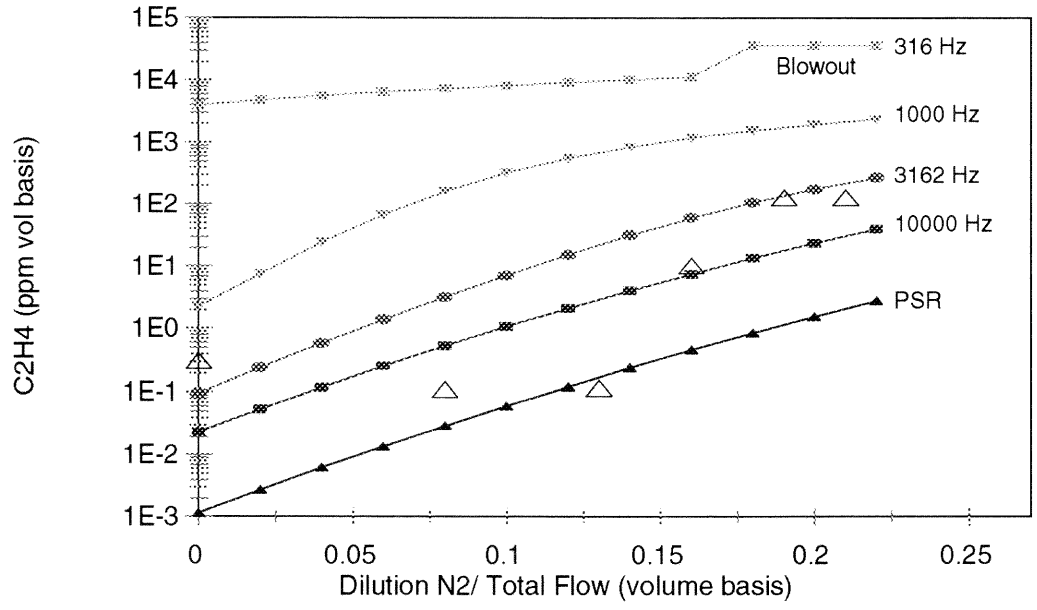


Figure 5.10 C_2H_4 (Unburned Fuel) Concentration Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models with Probe Quench Model

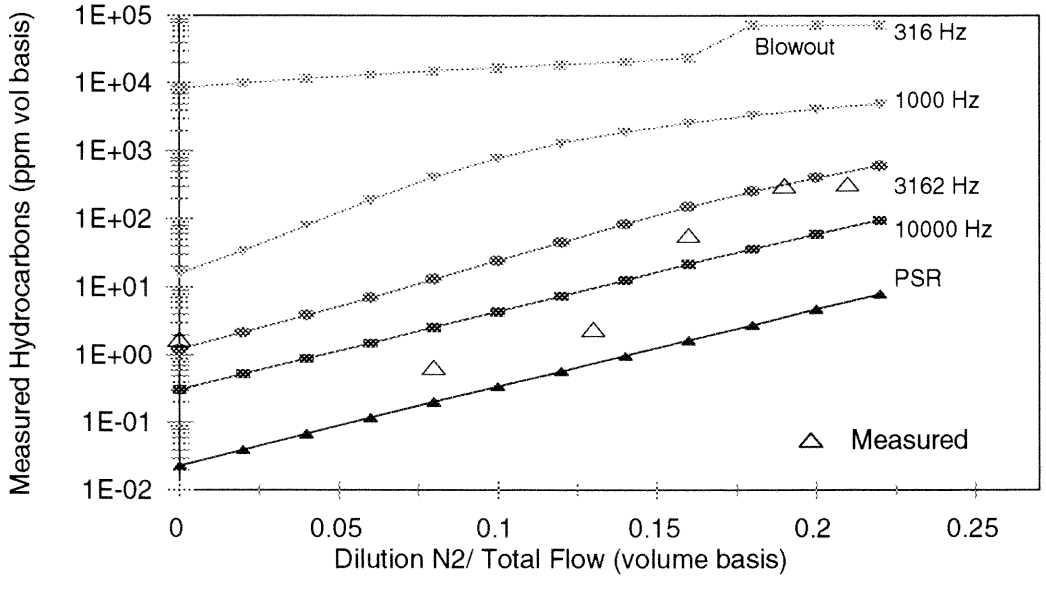


Figure 5.11 Total Measured Hydrocarbon Concentration Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models with Probe Quench Model

measured in the experiment), that can account for up to 30 percent of the effluent hydrocarbons calculated in the computer simulation. Although unburnt fuel dominates the total measured hydrocarbons at lower frequencies, other measured species, especially methane, make a larger relative contribution at the higher frequencies (see Appendix C Figures C.5.1 through C.5.3). The experimental data showed good agreement with the model in both Figures 5.10 and 5.11 between 3162 and 10,000 Hz. This mixing frequency range is consistent with those predicted by Nenniger *et al.* (1984).

5.4.4 Mixing Model Analysis

To the extent that the GRI mechanism provides a valid fundamental reaction model, analysis of the mixing model can provide insights to the relationship between mixing and combustion. This section will analyze the mixing model by examining its components. The PSR and PaSR models both share the same age distribution of "points" throughout the reactor; the state of those "points" is where they differ. The uniform temperature and composition throughout a PSR produces a uniform set of reaction rates. The degree of segregation characteristic of the "points" in a PaSR produces different reaction regimes based on "point" age. "Points" entering a PaSR take a finite time to mix to the average conditions of the reactor. Figures 5.12 and 5.13 show the reaction pathways for an infinitesimal "point" at different times $t^* =$ 0.0276 and 0.0283 (where dimensionless time $t^* = t/\tau$ and $\tau = 6.5$ ms). The diagrams show the major reaction pathways and displays the smaller reactant(s) along the path. The width of the arrow is roughly proportional to the reaction rate ranging from 10^{-8} to 10^{-3} moles/(cm^3 s). Even though Figures 5.12 and 5.13 represent conditions separated by less than 5 μs,

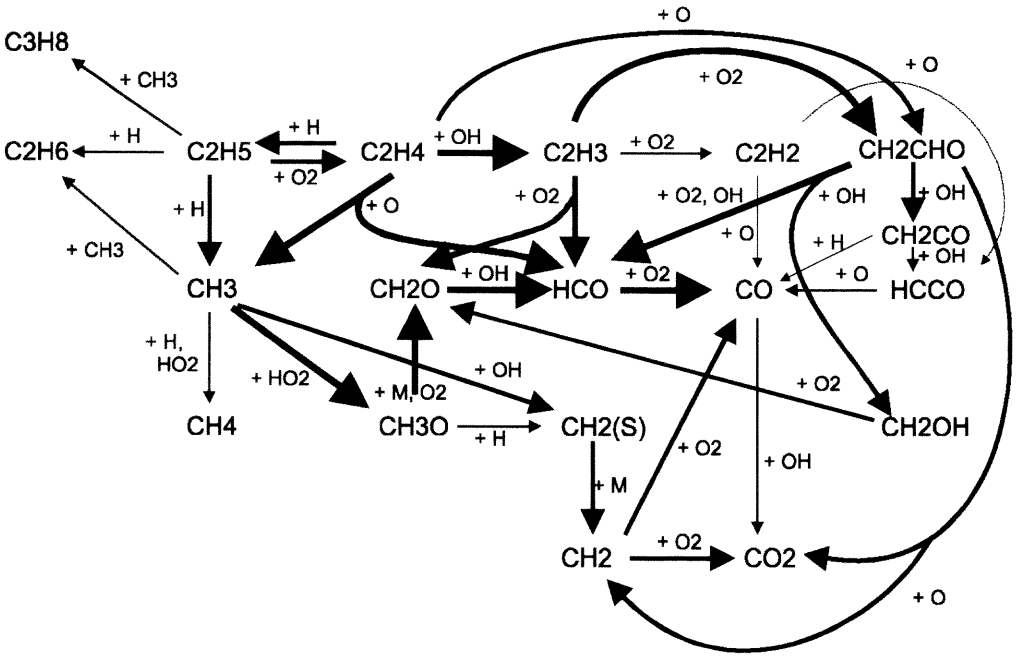


Figure 5.12 Reaction Pathway Analysis of PaSR, ω = 10000 Hz, $t^* = 0.0276$, T = 839 K, for C_2H_4 / Air Combustion using the GRI Mechanism

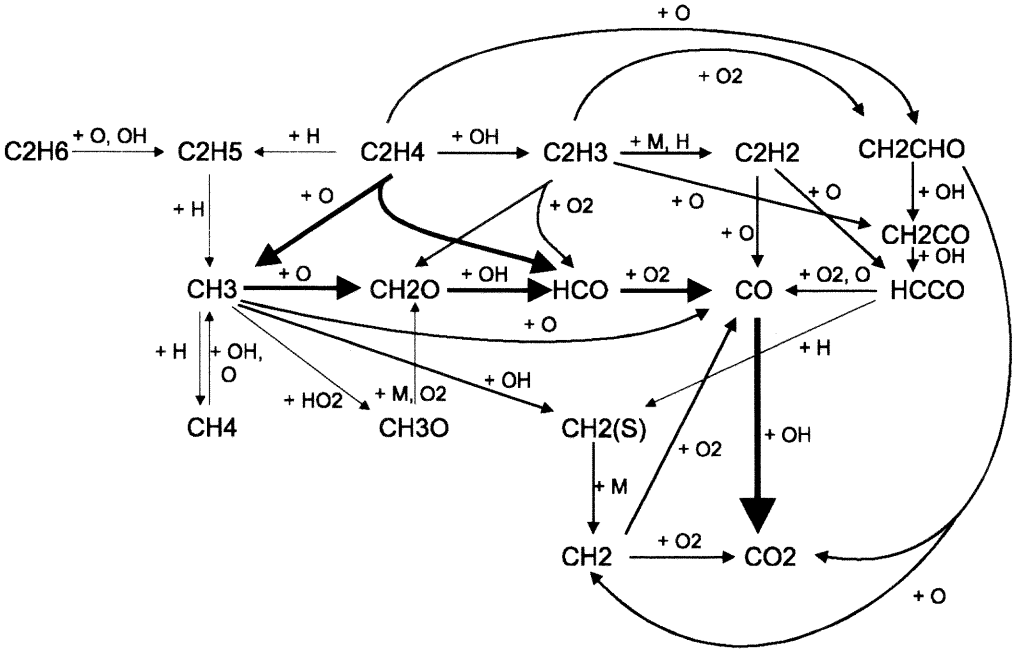


Figure 5.13 Reaction Pathway Analysis of PaSR, ω = 10000 Hz, $t^* = 0.0283$, T = 1631 K, for C_2H_4 / Air Combustion using the GRI Mechanism

the reaction regimes are very different. Initially, the lower temperature and lower concentration of oxidant radicals (Y_{OH} = 6.12×10^{-5}) favor the accumulation of the stable intermediates carbon monoxide (CO), propane (C_3H_8), ethane (C_2H_6), methane (CH_4), ethyne or acetylene (C_2H_2), and ketene (CH_2CO). As temperature and oxidant radical concentration (Y_{OH} = 1.66×10^{-3}) rise the regime shifts to a single predominate pathway between the fuel and its complete combustion product, CO_2. Stable intermediates are consumed and their concentration drops. The higher initial concentrations of stable intermediates go into the effluent because of the exponential distribution of "point" ages in the exit. When the probe quench model is added after the PaSR, concentrations of the stable intermediates can increase or decrease based on the specified probe conditions.

Separating the probe quench model from the PaSR reveals the contributions made by each. The effect on three species (CO, OH, and C_2H_4) illustrates some of these differences. Figure 5.14 shows a negligible dependence of average CO concentration (dry basis corrected for nitrogen dilution) on mixing frequency and steady increase in CO with an increasing dilution ration (except in the case of blowout for 316 Hz). This strongly contrasts with the PaSR plus probe quench model (Figure 5.9) that shows a strong dependence between mixing frequency and effluent CO concentration. Additionally, it shows a general increase in CO concentration as a result of the probe quench model suggesting significant continued reactions in the probe. Comparing the unburned fuel concentration in the PaSR model (Figure 5.15) with the PaSR plus probe quench model (Figure 5.10) also indicates continued reaction in the probe. In this case, the cooling effect of dilution and decreased mixing frequency appears to reduce the continued conversion in the probe. Without the probe model, the unburned fuel

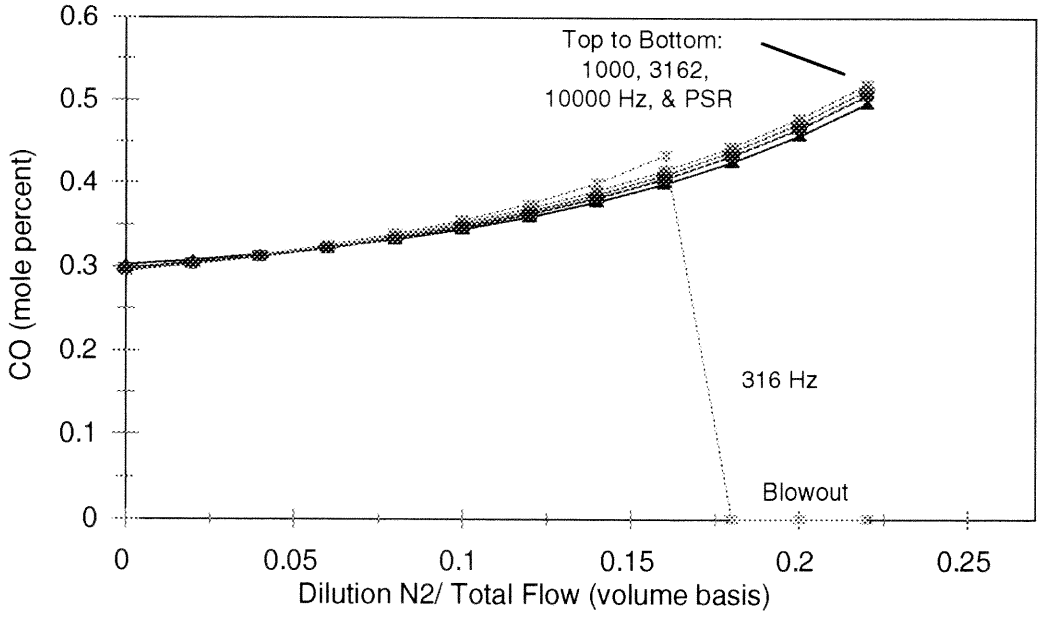


Figure 5.14 Carbon Monoxide, CO, Concentration Dependence on Dilution Ratio for PSR and PaSR Models

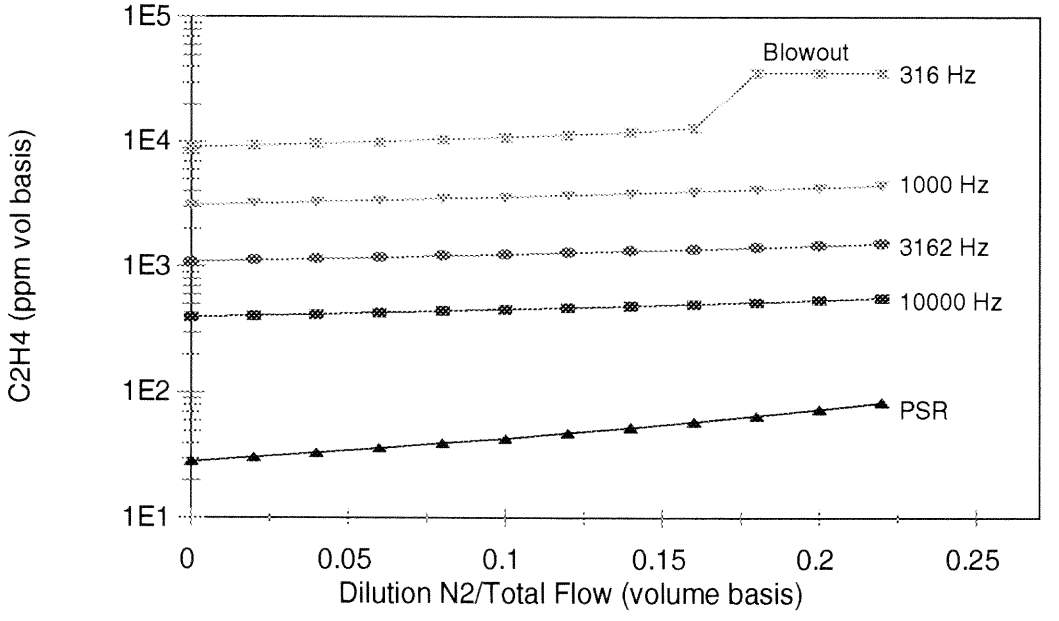


Figure 5.15 Ethylene (Unburned Fuel), C_2H_4, Concentration Dependence on Dilution Ratio for PSR and PaSR Models

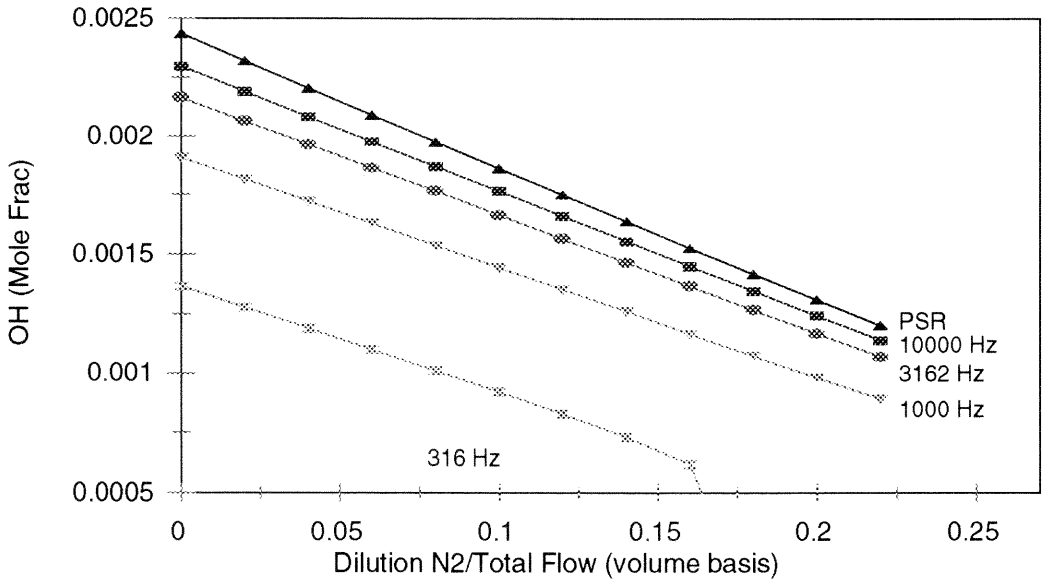


Figure 5.16 Hydroxyl Radical, OH, Concentration Dependence on Dilution Ratio for PSR and PaSR Models

concentration in the effluent is mostly affected by the mixing frequency. Both mixing frequency and dilution affect the concentration of the hydroxyl radical (Figure 5.16) in a fashion that is remarkably similar to average temperature (Figure 5.8). Because of this apparent temperature dependence, the OH concentration is negligible at the exit of the probe quench model. The next three subsections will explore the effects of mixing frequency, dilution, and the probe quench model with the GRI mechanism.

5.4.4.1 Effect of Mixing Frequency. Lowering mixing frequency delays the onset of combustion. Figure 5.17 clearly illustrates this view. This graph shows the temperature profile of a "point" traveling through the reactor. Reactants enter the reactor at 373 K and ultimately reach the same flame temperature. For high mixing frequencies, combustion occurs

rapidly. The 10,000 Hz line shows a rapid rise to the flame temperature at about $t^* = 0.03$, where the 316 Hz line takes a more gradual rise until $t^* = 0.5$ to reach the same level. This graph gives only part of the story. One must also consider the distribution of ages within the reactor.

Two additional observations regarding Figure 5.17 should be discussed. The first observation is that the sharper rise of the higher mixing frequency is counterintuitive. One would expect that the higher frequency to smooth-out abrupt changes. However, this behavior makes sense when considering the upper limiting case of the PSR that has a complete discontinuity at the reactor entrance. The second observation is that the nearly equal temperature both high and low frequencies reach for "points" past the initiation of combustion. This reflects the ultimate flame temperature of a premixed feed. Even with the same ultimate flame temperatures as the higher frequency reactor, the lower frequency reactor has a lower weighted average temperature in the effluent.

Within the reactor is an exponential distribution of "point" ages shown in Figure 5.18. This graph is a survival curve which is a reverse of cumulative distribution $[1 - F(t^*)]$ (see Equation 5.4). It shows the fraction of "points" remaining in the reactor after a given time. This distribution is important when considering any of the temperature-time or concentration-time profiles in that it show the fraction of "points" that reached those conditions. The age distribution, $\varphi(\alpha)$, ("point" age defined by Equation 5.3), which is related to F by $\varphi(\alpha) = (1/\tau)[1 - F(\alpha)]$, weights the local conditions at each infinitesimal "point" to get the average conditions (Equation 5.12) that the local conditions relax toward at a rate given by

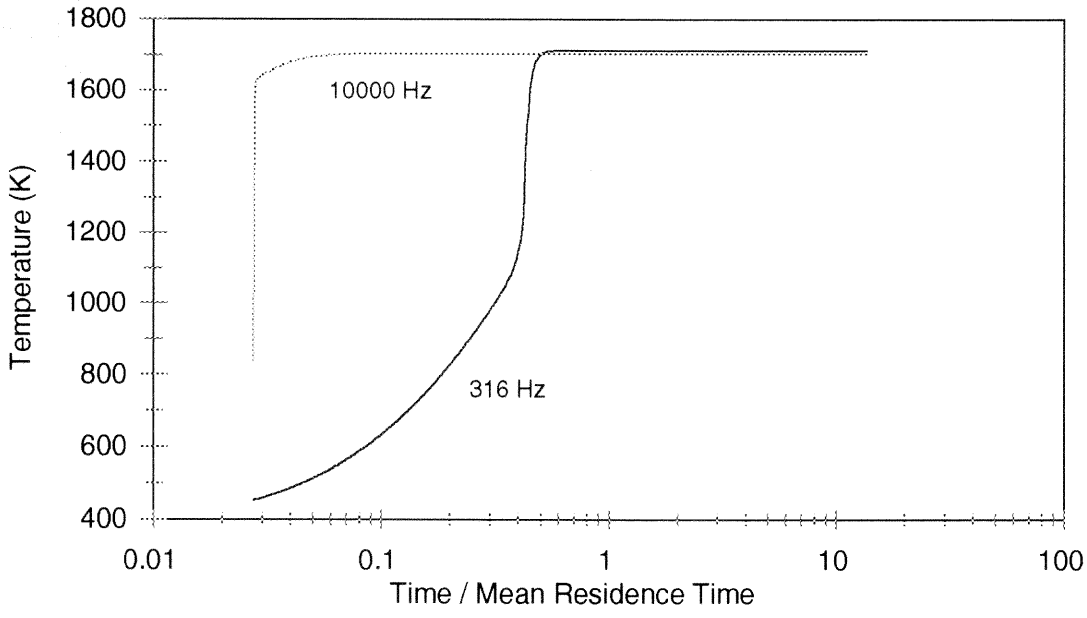


Figure 5.17 Effect of Mixing Frequency on Temperature Profile with Respect to Age

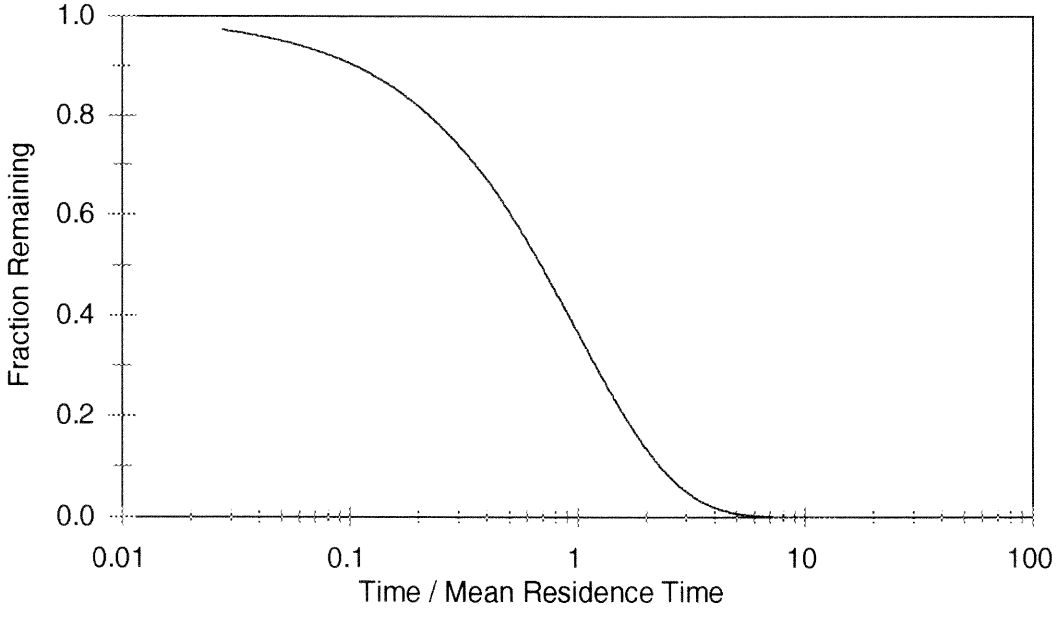


Figure 5.18 Survival Curve [1 - $F(t^*)$] Shows Fraction of "Points" Remaining in Reactor at a Given Time

the mixing frequency. When the age distribution is used as the weighting function for the temperature profile, a temperature probability distribution (Figure 5.19) can be derived for temperature.

Figure 5.19 shows the density of "points" at a given temperature on the abscissa. The probabilities are taken on a 30 K interval, so the ordinate shows the probability of any "point," or fraction of all "points," being within the given interval about that temperature. Both the simulations at 316 Hz and 10,000 Hz show the same mode at roughly 1700 K, but where 95.4 percent of the "points" in the higher frequency system are in the 30 K interval about 1700, only 61.2 percent of the 316 Hz system "points" fall into that interval. "Points" in the lower frequency simulation have a higher probability (0.31) of falling within a range between 400 and 1200 K. This graphic is another way of looking at the combination of Figures 5.17 and 5.18. It illustrates the greater range of temperatures at lower mixing frequencies. These lower temperatures promote the accumulation of stable intermediate compounds shown (see Figure 5.12).

Figure 5.20 shows a time composition profile similar to Figure 5.17. Again, the results illustrate the delaying effect of reduced mixing frequencies by the delayed peak of OH and CO concentration. Although not readily apparent in the figure, CO concentration peaks prior to the hydroxyl radical peak concentration. The rapid rise in OH concentration promotes the CO burnout reaction:

$$CO + OH \rightleftharpoons CO_2 + H \tag{5.27}$$

that rapidly reduces the accumulating carbon monxide.

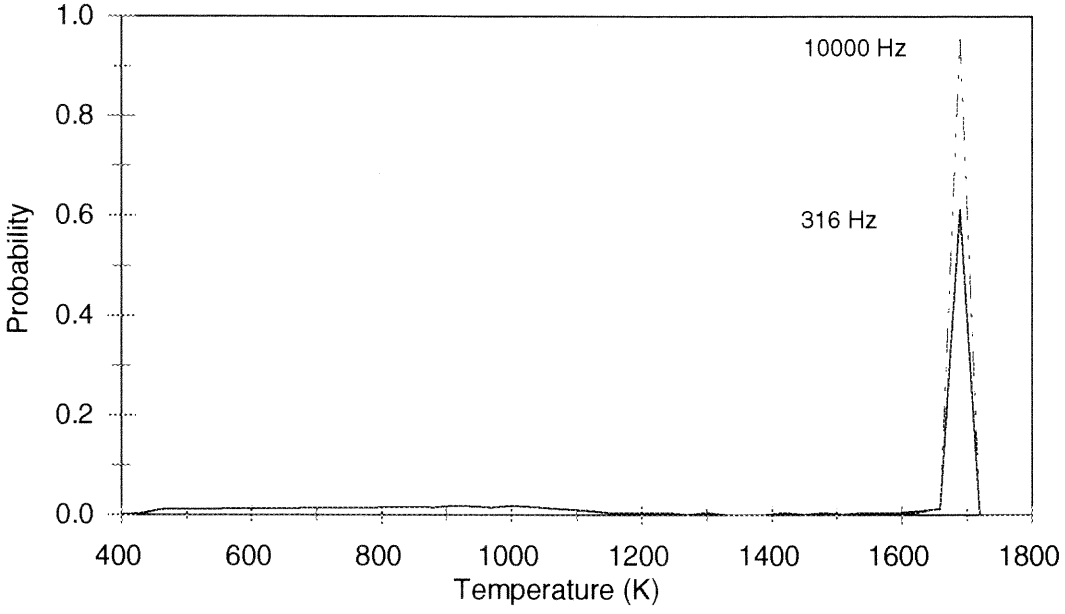


Figure 5.19 Effect of Mixing Frequency on the Temperature Distribution (Probability is on a 30 K Interval)

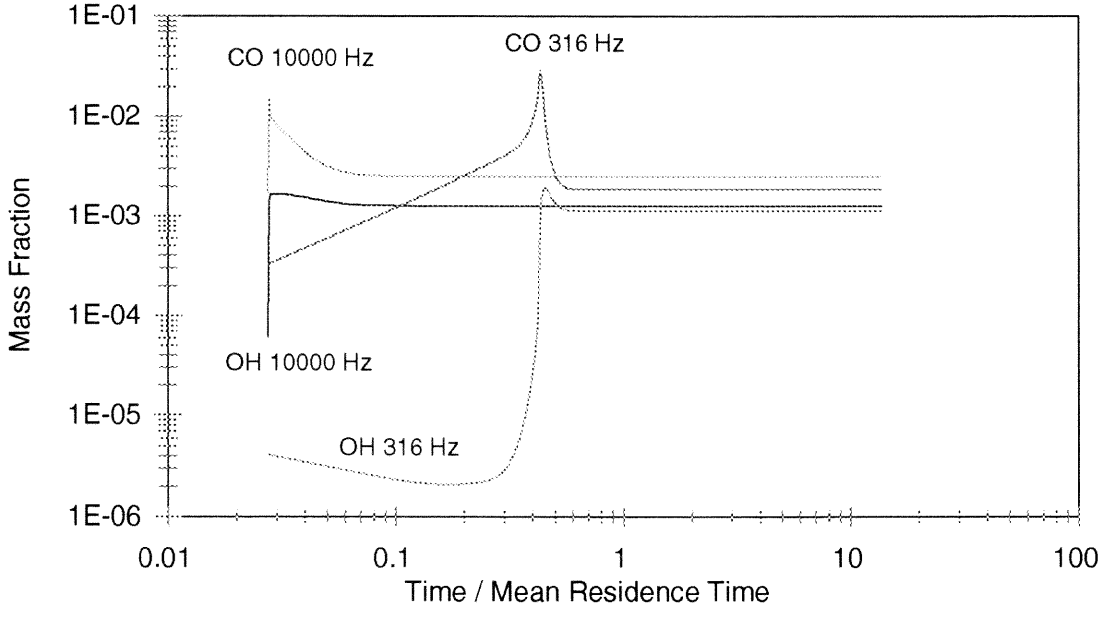


Figure 5.20 Effect of Mixing Frequency on Carbon Monoxide (CO) and Hydroxyl Radical (OH) Concentration Profile with Respect to Age

The 316 Hz lines contrast the behavior of OH and CO in the PaSR model. Carbon monoxide concentration gradually rises; CO is stable at both low and high temperatures. In contrast, the hydroxyl radical rises rapidly with temperature. Recall that Figure 5.16 shows the average OH concentration in the reactor closely follows the response of the average temperature in the reactor (Figure 5.8). Average CO concentration was almost independent of the effect of mixing frequency (Figure 5.14), but the distribution of CO varies widely with mixing frequency. Figure 5.21 shows the probability distribution of CO taken on a 2.0×10^{-4} interval. Even though the average concentration in the reactor is the same for both frequencies the lower frequency has a lower mode, but a wider variance; the 10000 Hz peak contains 93.2 percent of the "points" where the 316 Hz peak contains 59.1 percent of them.

Figure 5.22 shows the concentration profile for the methyl radical (CH_3). CH_3 is an important intermediate in the formation of other stable intermediate hydrocarbons, especially larger chain compounds as illustrated in Figure 5.12. Once temperature rises and the hydroxyl concentration increases, the concentration rapidly drops. This shows how the PaSR model can significantly depart from the PSR in an ability to model the formation of hydrocarbon PICs. The high concentration of methyl radicals promote the second order reactions in PIC formation, which increase as mixing frequency goes down. The lower homogenous distribution of this radicals in a PSR model does not promote these second order reactions under fuel-lean conditions.

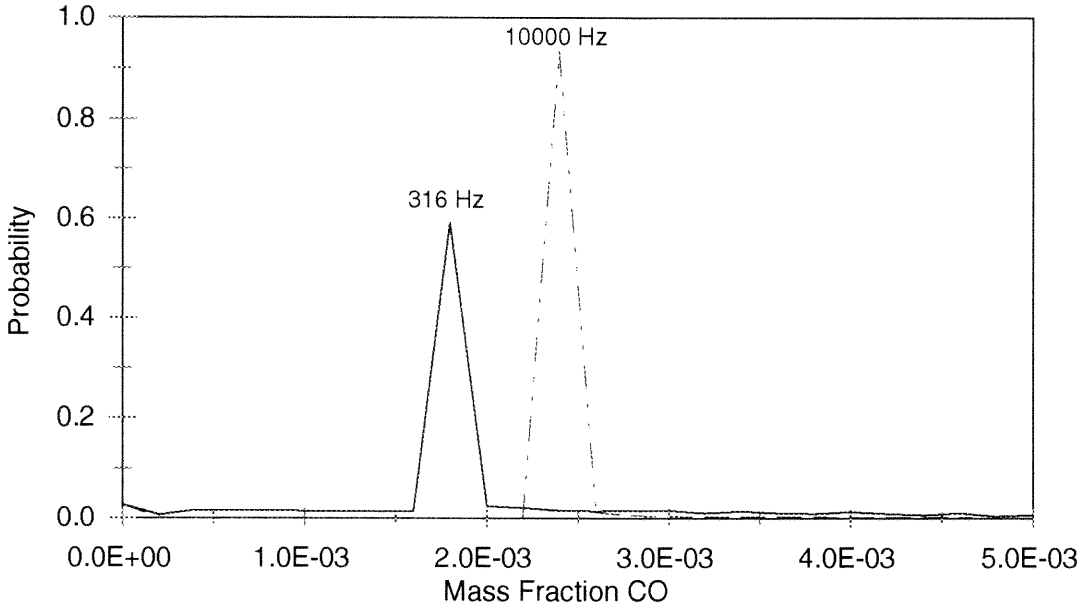


Figure 5.21 Effect of Mixing Frequency on the Carbon Monoxide (CO) Concentration Distribution (Probability is on a 2.00×10^{-4} Interval)

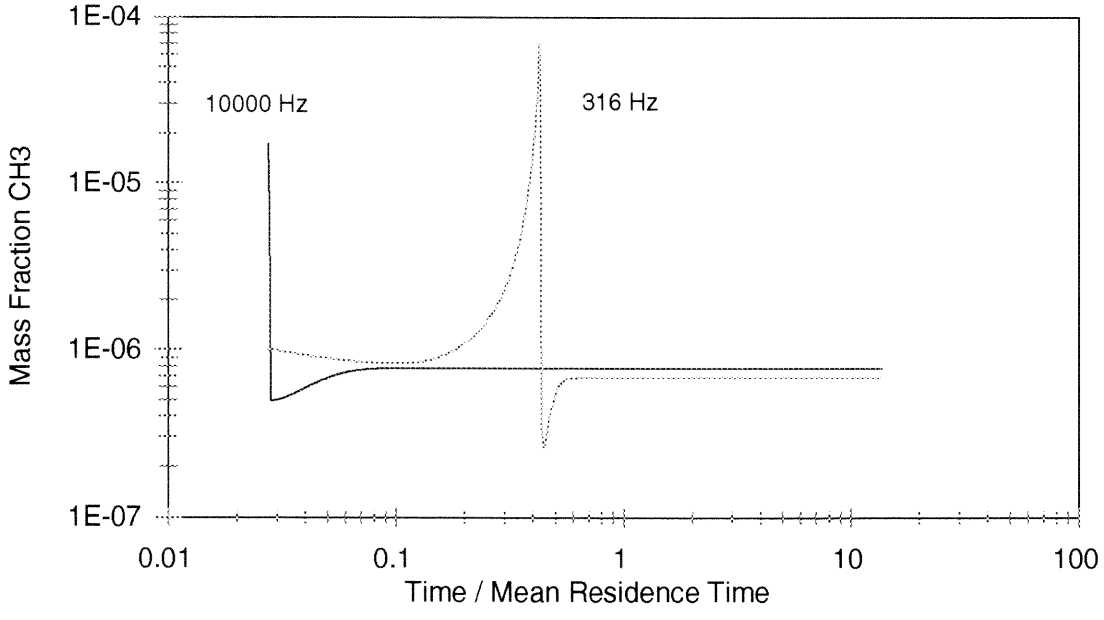


Figure 5.22 Effect of Mixing Frequency on Methyl Radical (CH_3) Concentration Profile with Respect to Age

5.4.4.2 Effect of Dilution. Adding a diluting gas such as nitrogen cools the reaction. The inert gases carry off the energy released by the reaction. The lower temperatures and the effect of diluting concentrations slows the individual reaction rates. Increased stability of the methyl radical and lower concentrations of the hydroxyl radical promotes the production of more stable products of incomplete combustion. As more diluting gas is added, the reactor eventually reaches the blowout point and combustion stops. Similar to the effect of decreasing mixing frequency; dilution also delays combustion in the reactor model. Two cases at a mixing frequency of 316 Hz are compared: no dilution and dilution near the blowout point.

Figure 5.23 illustrates two of the effects of dilution on the combustion model. The second line, which shows the age– temperature profile for a dilution near blowout, has a delayed rise in temperature and plateaus at a lower temperature. Applying the age distribution of "points" within the reactor to Figure 5.23 yields the temperature distribution in Figure 5.24. This graph reflects the spread in the temperature distribution measured by Barat (1990) as dilution increases. Barat noted that the variance of the measured temperature distribution increased with dilution and became bimodal at higher levels of dilution. In similar fashion, the temperature distribution of the PaSR model becomes increasingly bimodal. However, since the peak temperature decreases, the larger peak moves to the left decreasing the distance between it and the lower peak. This causes the variance to go down. This disagreement is partly an artifact of the model verus an experimental measurement. The low probability temperatures would not rise above the noise in an experimental measurement. If such a threshold were applied to the model, variances would go up with dilution also.

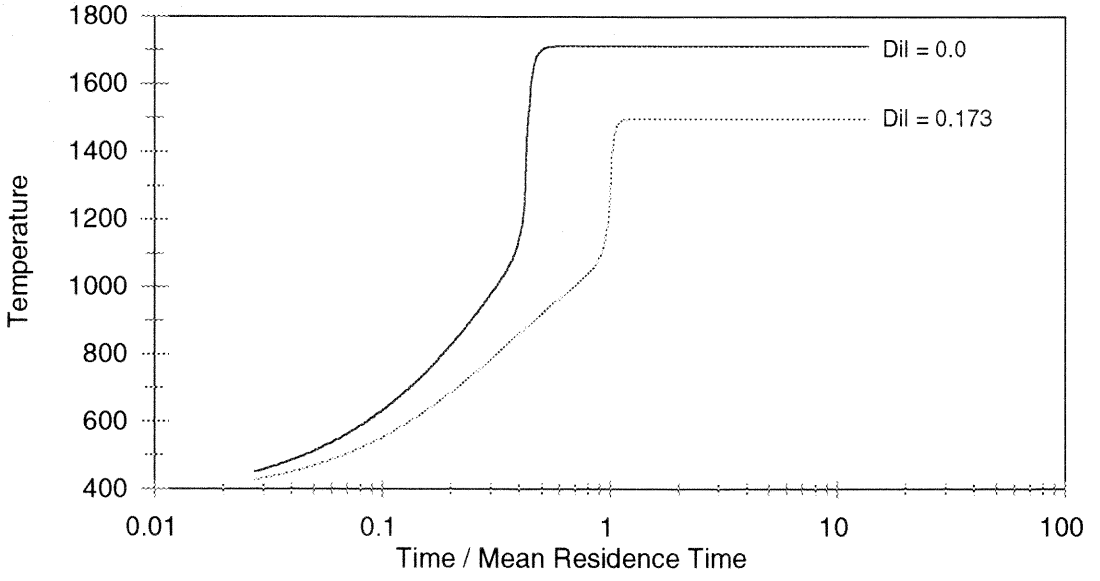


Figure 5.23 Effect of Dilution on Temperature Profile with Respect to Age, Dilution Ration Value is (Volumetric Flow Added N_2 / Total Volumetric Flow); ω = 316 Hz

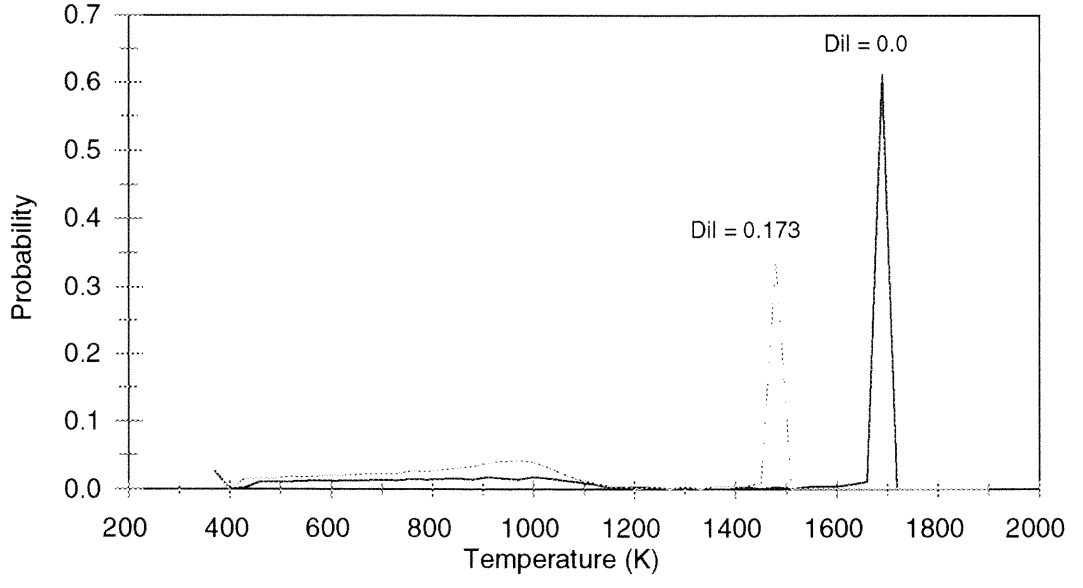


Figure 5.24 Effect of Dilution on the Temperature Distribution (Probability is on a 30 K Interval); ω = 316 Hz

Figure 5.25 shows the dilution effect on the concentration profile with respect to the age of the "points." The lower temperatures that result from dilution lowers the final concentration of the hydroxyl radical. Since OH reacts with CO to produce CO_2, the final concentration of CO raises as a result. Dilution also delays the rapid climb in OH concentration marking the onset of sustainable combustion. It occurs at a reduced age of 0.4 for the undiluted feed, but at 1.0 for the diluted feed. This has a significant effect on the effluent composition in the PaSR model. Recalling the survival curve (Figure 5.18), 67 percent of the "points" remain in the reactor at a reduced age of 0.4, but only 37 percent remain at 1.0. For the diluted case, the large percentage of "points" not past the onset of sustainable combustion would appear at areas of "local blowout," which is how Barat described the high dilution experiments. These cases were accompanied by higher CO concentrations in the effluent.

The probability distribution of the CO concentration taken on a 2.00×10^{-4} interval (Figure 5.26) better illustrates the effect of dilution on CO concentration in the model. As dilution pushes the simulation to the point of blowout, not only does the mode value increase, but the so do the probabilities at the higher concentrations. The mode peak for the dilution simulation decreases from 59 percent to 32 percent and the tail to the right increases from a sum total of 24 percent to 34 percent. This suggests that if one could measure "local" concentrations of CO that the higher density of high local concentrations of CO could be used as an prelude indicator to blowout, as suggested by Barat (1992).

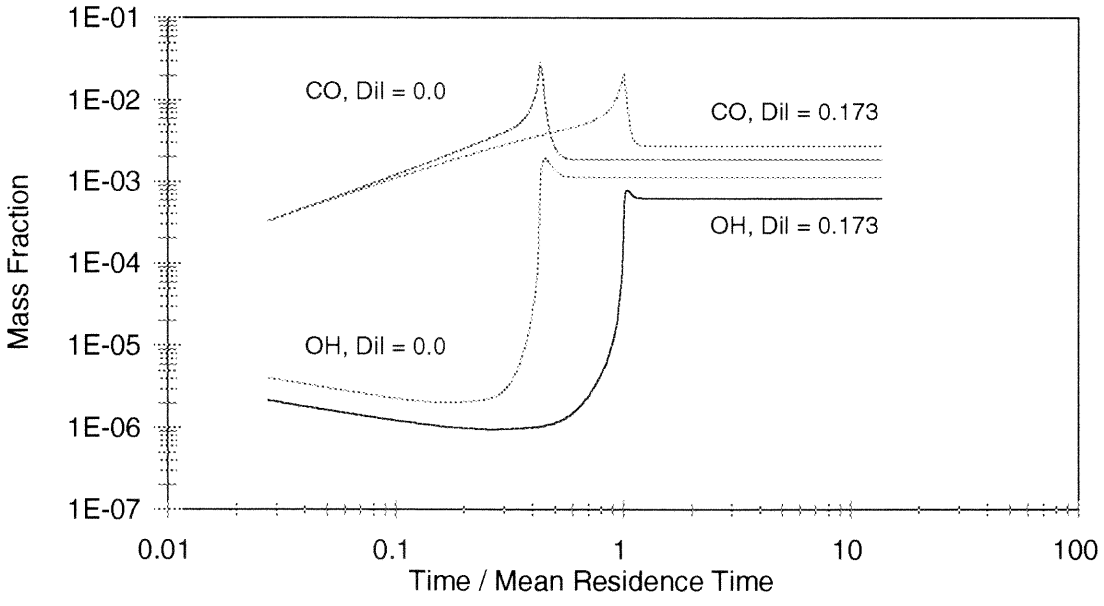


Figure 5.25 Effect of Dilution on Carbon Monoxide (CO) and Hydroxyl Radical (OH) Concentration Profile with Respect to Age; ω = 316 Hz

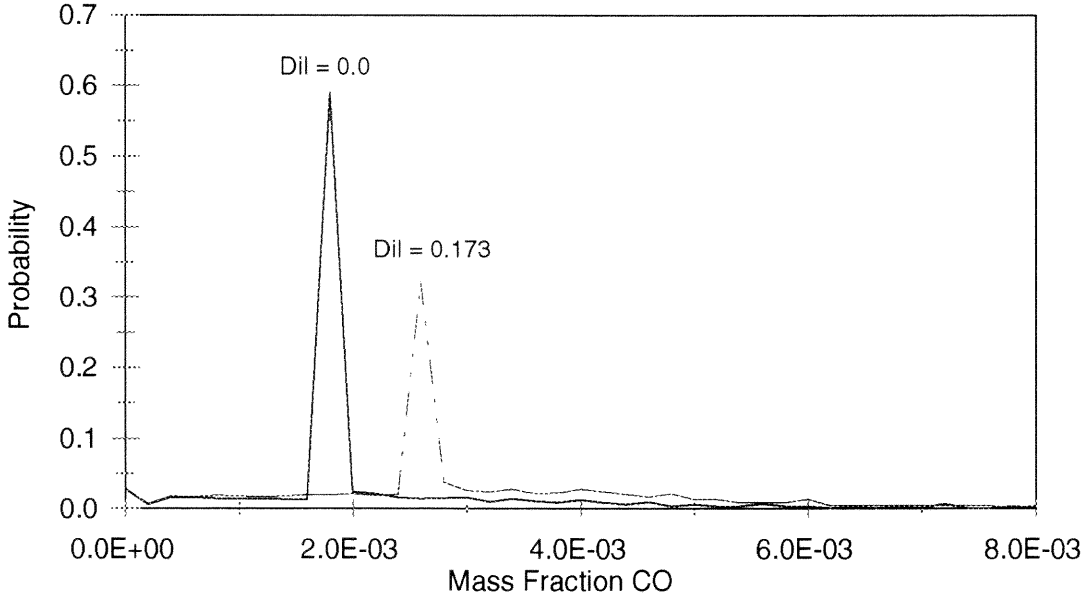


Figure 5.26 Effect of Dilution on Carbon Monoxide (CO) Concentration Distribution (Probability is on a 2.00 $\times 10^{-4}$ Interval); ω = 316 Hz

5.4.4.3 Probe Quench Effect. Analysis of a simple quenched probe model shows some of the difficulties in obtaining accurate measurements of the composition within the reactor. Even with the jacketed probe, significant reactions continue until the sample is sufficiently cooled. The computer simulation uses a simple PFR model with a imposed temperature profile given in differential form in Equation 5.26 and shown in Figure 5.27. The x-axis is the reduced probe length that has a residence time of 3 milliseconds. The temperature profile depends on the entrance temperature, which varies with mixing frequency and dilution (at a given fuel/air equivalence ratio) in the stirred reactor. The reaction paths are slowed directly by the dropping temperature and indirectly by the lowering concentrations of the unstable radical intermediates.

The second major factor effecting probe reactions is the amount of unreacted hydrocarbons in the PaSR sample. This hydrocarbon concentrated is dominated by unburnt fuel (C_2H_4) that continues to react in the presence of excess oxygen. The high temperature reaction pathways resemble those in Figure 5.13 and the low temperature reactions follow a similar path to Figure 5.12, which is the reverse order of the reactions in the PaSR. As a result, higher temperature PaSR samples tend to consume more of the stable intermediates. Since simulations with higher temperature PaSR samples are a result of higher mixing frequencies and tend to have less stable hydrocarbons in the effluents, those present tend to be rapidly consumed to complete combustion products. PaSR effluents at low temperatures that result from lower mixing frequencies have higher concentrations of unburnt fuel which enter the low temperature reactions that produce the stable products of incomplete combustion. Thus, the reactions in the probe tend to amplify the effects of the PaSR seen

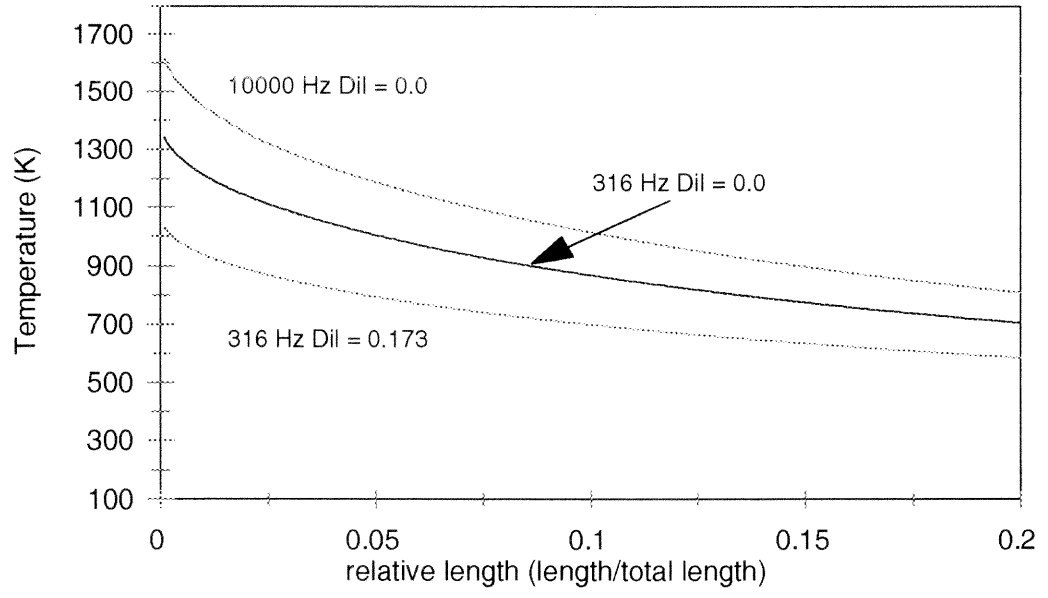


Figure 5.27 Temperature Profile in Probe Model for PaSR Inputs: 1000 Hz w/o Dilution; 316 Hz w/o Dilution; and 316 Hz Dilution Ratio of 0.173; Probe Residence Time = 3 ms

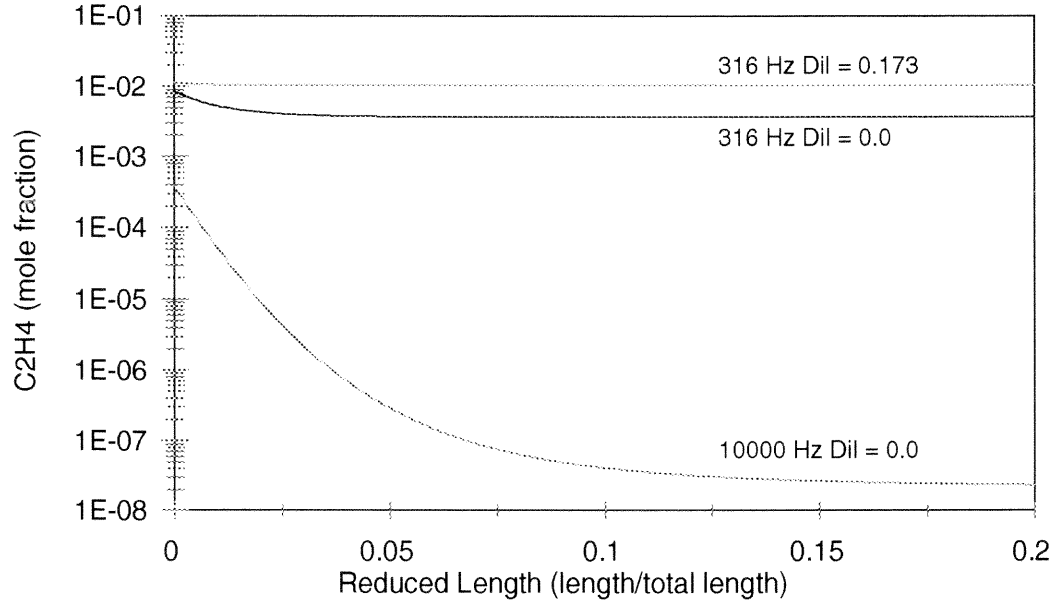


Figure 5.28 C_2H_4 Concentration Profile in Probe Model for PaSR Inputs: 1000 Hz w/o Dilution; 316 Hz w/o Dilution; and 316 Hz Dilution Ratio of 0.173; Probe Residence Time = 3 ms

when comparing CO in the effluent of the PaSR (Figure 5.14) to the CO concentrations after adding the probe quench model (Figure 5.9), and the comparisons of C_2H_4 before and after the probe quench model (Figures 5.15 and 5.10 respectively).

Figure 5.28 shows the profile of unburnt fuel concentration through the length of the probe quench model. As expected, the effluent from the 10,000 Hz, zero dilution, PaSR model, which has the highest effluent temperature, has largest percentage consumption of fuel. However, the logarithmic scale de-emphasizes the result of 316 Hz, zero dilution model, having the largest total conversion of the unburnt fuel. The combination of a large consumption of fuel and low temperature produces the greatest amounts of hydrocarbon PICs in the probe quench model. The probe quench model triples the concentration of CO (Figure 5.29) and increases the concentration of C_2H_2 by an order of magnitude (Figure 5.30) In contrast, the 10,000 Hz, zero dilution, run breaks-even on CO production and consumption and has a net consumption of C_2H_2 lowering it 3 to 4 orders of magnitude. The low entrance temperature of the 316 Hz run with a dilution ratio of 0.173 results in little reaction within the probe. Unlike the PaSR model, the PFR model used for the probe quench does not include mixing with average concentrations of the unstable intermediates that would continue to promote reaction. It results in the reactions quenching at higher temperatures when compared to the reactions occurring at T = 839 K at a mixing frequency of 10,000 Hz in Figure 5.12.

The impact of the probe quench model on the results of the PaSR model begs the question on the sensitivity of the probe quench model parameters. How accurately must we measure these parameters to get a model that reliably simulates the experimental system?

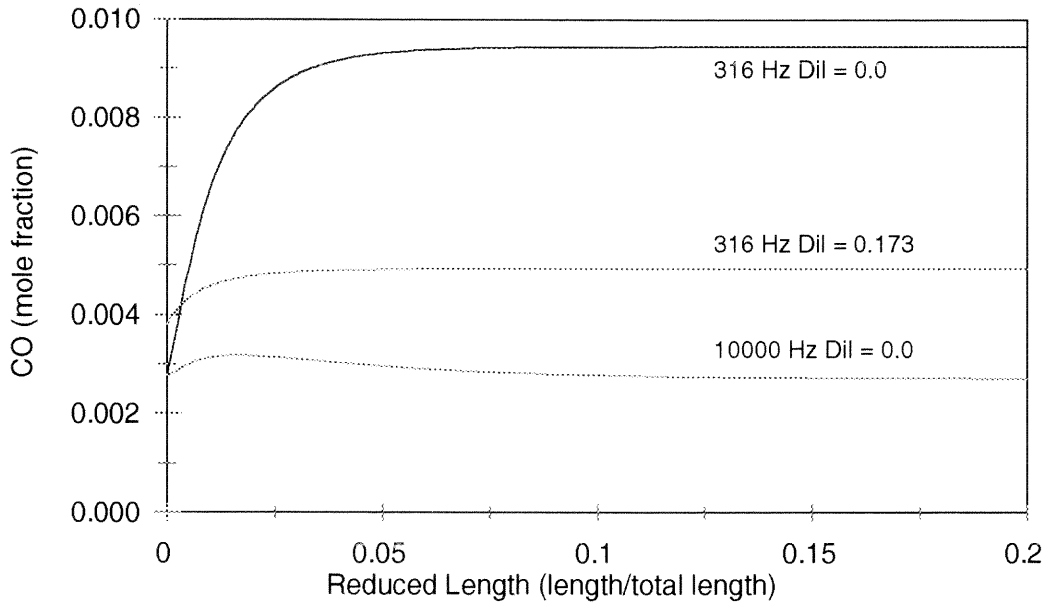


Figure 5.29 CO Concentration Profile in Probe Model for PaSR Inputs: 1000 Hz w/o Dilution; 316 Hz w/o Dilution; and 316 Hz Dilution Ratio of 0.173; Total Residence Time = 3 ms

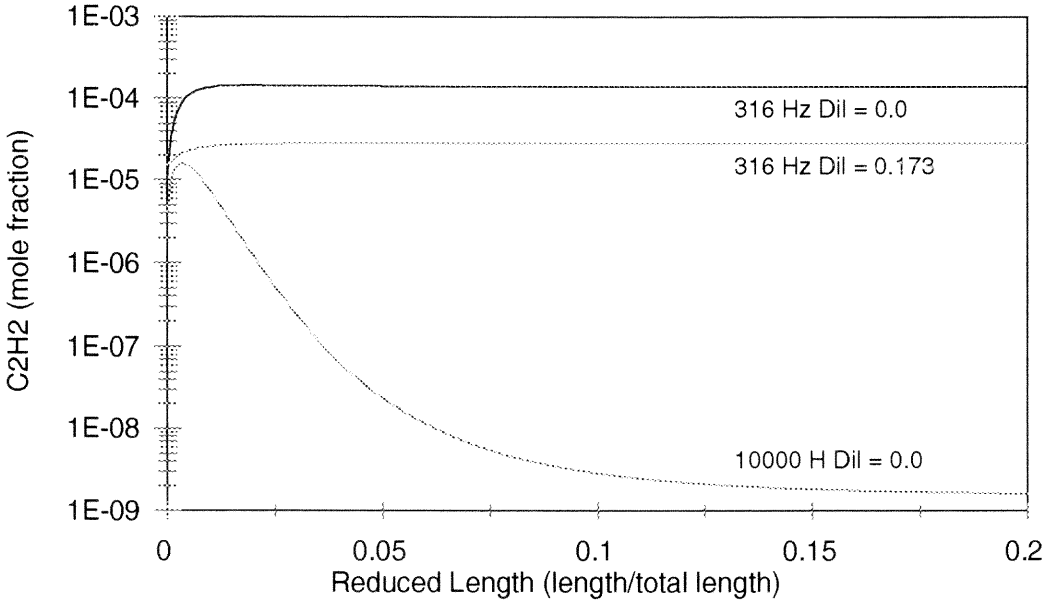


Figure 5.30 C_2H_2 Concentration Profile in Probe Model for PaSR Inputs: 1000 Hz w/o Dilution; 316 Hz w/o Dilution; and 316 Hz Dilution Ratio of 0.173; Total Residence Time = 3 ms

The probe quench model, in essence, uses a single parameter, b, while the other parameters, T_0 and T_c, represent the boundary conditions. The sensitivity coefficient of species i with respect the parameter b is:

$$S_i = \frac{b}{Y_i}\frac{\partial Y_i}{\partial b} \tag{5.28}$$

From this coefficient, the fractional uncertainty of the exit concentration of species i can be estimated from the fractional uncertainty of b:

$$\frac{\Delta Y_i}{Y_i} \approx S_i \frac{\Delta b}{b} \tag{5.29}$$

When $|S_i| << 1$, uncertainties in b do not significantly add to the uncertainties in Y_i, but when $|S_i| >> 1$ the uncertainties are amplified. Table 5.1 shows the sensitivity coefficients for selected species:

Table 5.1 Probe Quench Model Sensitivity Coefficients, S_i

ω (Hz)	Dilution Ratio	T_0 (K)	C_2H_4 in (mole frac)	C2H4	C_2H_6	C_2H_2	CO
316	0.173	1070	1.14e-02	0.00	0.00	0.00	-1.12
316	0.0	1401	8.60e-03	0.00	0.00	0.00	-0.58
3162	0.22	1405	1.10e-03	3.44	3.30	2.10	0.00
3162	0.0	1666	1.01e-03	13.22	15.65	13.43	0.00
10000	0.0	1688	3.67e-04	12.73	12.99	14.45	2.04

Higher probe inlet temperatures and unburnt fuel concentrations magnify the uncertainties of the probe quench parameter. These results suggest that, for measuring hydrocarbon PICs, quenched probe parameters should be carefully measured for condition that involve high temperatures and excess oxygen. Measurements of CO seem to be less dependent on that parameter.

5.5 Comparison to Experimental Results: Toroidal Jet Stirred Combustor (TJSC) Fuel Rich Conditions

Another comparison was made with a separate set of experiments conducted by Brukh, *et al.* (2001). These experiments explored ethylene/air combustion in fuel-rich conditions in order to measure and model the formation of aromatic compounds. Modeling efforts used a Polyaromatic Hydrocarbon (PAH) mechanism developed by Marinov *et al.* (1998). This mechanism, hence referenced as the Marinov mechanism, was designed against data from a pre-mixed laminar *n*-butane flame, but used earlier PAH models developed against fuel rich methane, ethane, ethylene and propane combustion as a foundation. This mechanism, which consists of 153 molecular species in 689 reactions, provided a good test case for the viability of the direct integration PaSR approach to detailed thermo-kinetic mechanisms.

5.5.1 Experimental Set-up

The experimental set up used a TJSC (Figure 5.7) described in section 5.4 with a second stage burnout section placed downstream of the TJSC. Figure 5.31 shows an axial view of the of the TJSC with burnout section. A jacketed probe sampled the effluent gases in the burnout section 21 milliseconds downstream of the TJSC exit. The effluent was concentrated

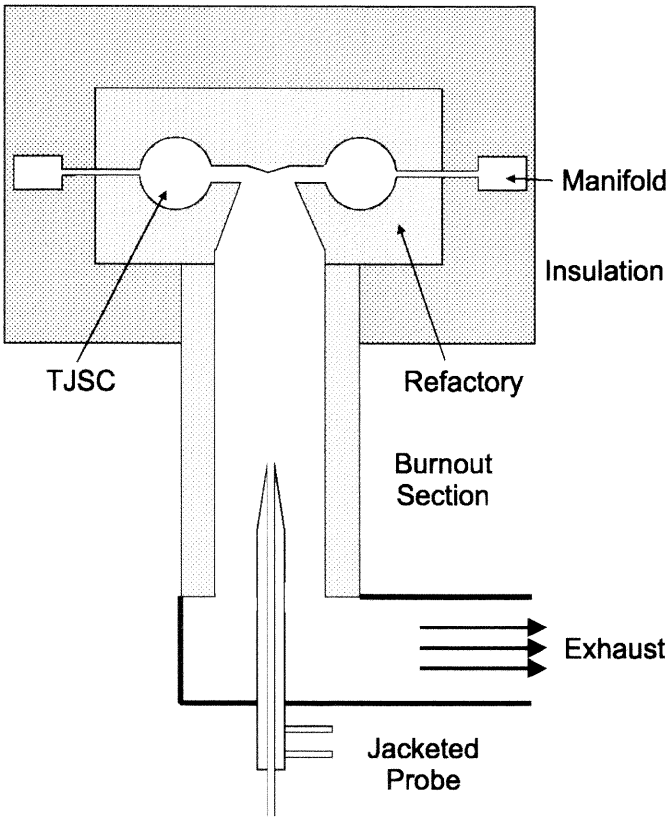


Figure 5.31 TJSC with Burnout Section Axial Cross-Section

using a technique developed by Mitra (1990) that adsorbs Volatile Organic Compounds (VOC) in a short adsorbent filled metal tube, and uses a electric current to desorb them into a GC. The concentrated pulse of VOCs allowed for better resolution of benzene down to the 100 ppb range using an on-line configuration.

Six combustion conditions were selected for measurement with fuel air equivalence ratio, Φ, ranging from 1.38 to 1.93. Additional nitrogen diluted the premixed ethylene and air mixture to control the temperature in the TJSC. Adjustment of the added nitrogen

maintained a constant 1623 K temperature in the TJSC for all combustion conditions. The gases exiting from the TJSC cooled to 1380 K at the sample point near the end of the burnout section.

5.5.2 Modeling Approach

The PaSR computer model was modified to incorporate a PFR to model the burnout section rather than the probe. A exponential cooling model was imposed by estimating the enthalpy loss between the TJSC effluent at 1623 K and the burnout section effluent at 1380 K, calculating the overall heat transfer coefficient from Equation 4.22, and using it as a constant temperature loss in the PFR heat balance (Equation 4.4). Since the PFR exit temperature was sufficiently low and combustion effluent sufficiently oxygen depleted, an additional probe quench model was not considered necessary. The modeled PaSR, which produced temperatures in excess of 1623 K for adiabatic conditions, required an estimated heat loss term to produce the required average reactor temperature. The PaSR used a mixing frequency of 3162 Hz, which was chosen for its fit with the fuel lean data and proximity of the calculated turbulent parameters by Nenniger *et al.* (1984).

The Marinov mechanism and its associated thermodynamic parameters were used in the simulation. A PaSR and PSR simulation was run using the feed conditions for each data point. PaSR simulation runs took between 20 to 36 hours using a Pentium III 850 MHz microprocessor with a Windows ME® operating system. This was a factor of 60 times longer than the average simulation run using the GRI 3.0 mechanism for fuel-lean conditions. The larger number of molecular species (153 vs 53) and reactions (689 vs. 325) of the

Marinov mechanism, as well as application of a heat loss term, accounts for this disparity. The large variance in scales of the concentrations of individual species makes the initial value differential equations "stiff" and requires smaller timesteps to solve. The exchange-with-the-mean term in Equation 5.11 along with the external heat loss term of Equation 5.15a exacerbate the timestep problem and frequently require computations near the limits of double precision floating point arithmetic. Convergence to a mean using successive substitution took up to 120 iterations and increased with Φ. The temperature specified PSR simulations, with subsequent PFR, took considerably less time and averaged between 5 to 15 minutes per run.

5.5.3 Comparison of Model to Experimental Results and Analysis

The results of both the two-stage PSR/PFR and PaSR/PFR models, corrected for nitrogen dilution and on a dry basis, are compared against the experimental data in Figure 5.32. Although both models show excellent agreement with experimental data, the competing approaches show less contrast than expected. At fuel/air equivalence ratios higher than 1.6, the models produce virtually identical results. Divergence occurs at lower values of Φ, but the PaSR predicts lower concentrations contrary to the hydrocarbon concentration predictions in fuel-lean conditions using the GRI mechanism. When the effect of the second stage PFR is removed, the divergence reverses with the PaSR predicting the higher concentrations as shown in Figure 5.33. These effects are consistent with the probe quench PFR model analyzed in section 5.4.4.3. The PaSR effluent has higher concentrations of both fuel and oxygen that contribute to the larger concentration changes in the modeled second stage.

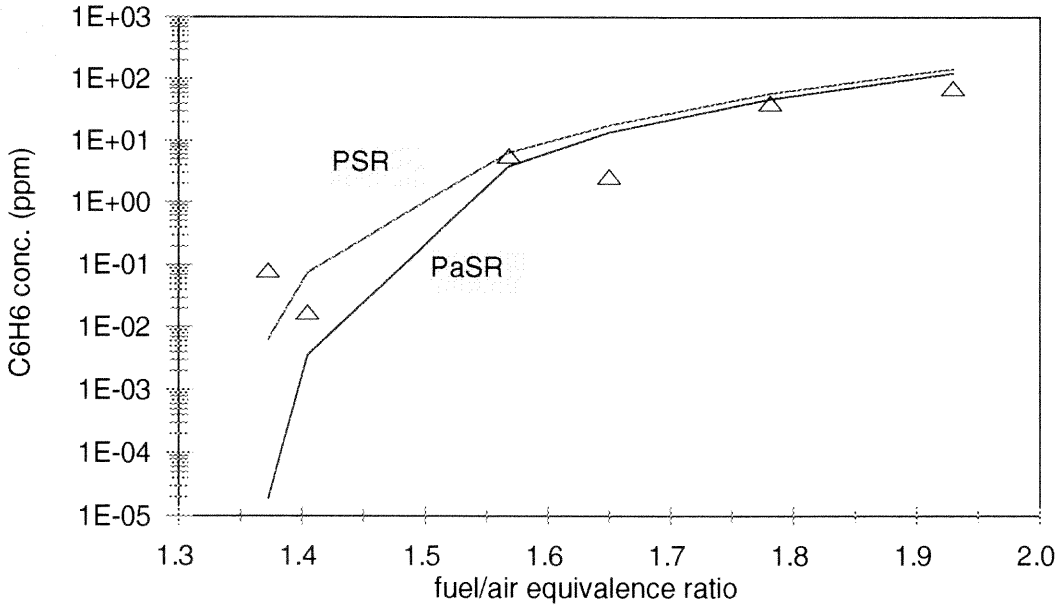


Figure 5.32 Benzene (C_6H_6) Effluent Concentration Dependence on Fuel/Air Equivalence Ratio, Comparison of Experimental Data (△) with PSR and PaSR (3162 Hz) Models with PFR Second Stage

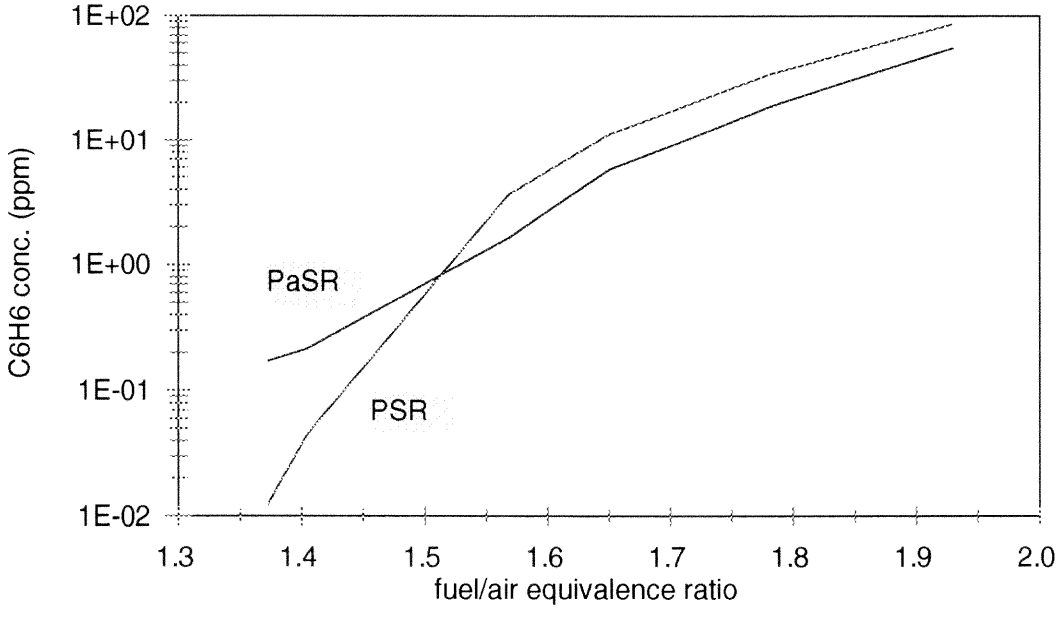


Figure 5.33 Benzene (C_6H_6) Effluent Concentration Dependence on Fuel/Air Equivalence Ratio, Comparison of PSR and PaSR (3162 Hz) Models without PFR Second Stage

The similarity between the PSR and PaSR results at higher values of Φ can be attributed to the resulting lower concentrations of oxidizing radicals such as the hydroxyl radical (OH). Figure 5.34 shows that at Φ = 1.38, the concentrations of OH are generally 20 times greater than those at Φ = 1.93. As a result, the hydrocarbon radical spike that precedes the rapid temperature rise, modeled by the PaSR for the fuel-lean systems (Figure 5.22 juxtaposed to Figure 5.17), is no longer present. Figure 5.35 shows the methyl radical age profile for fuel/air equivalence ratios of 1.38 and 1.93. The more fuel-rich condition has a flat methyl radical concentration profile, which does not differ greatly from the flat age/concentration profiles expected for perfect micromixing.

Further analysis was performed by exploring the major reaction pathways to benzene at the lower and higher fuel/air equivalence ratios. The major benzene production step was found to be the combination reaction of the propargyl radical (CH_2CCH) with itself. This was consistent with the finding of Marinov *et al.* who found this reaction to account for 80 percent of benzene production in fuel rich butane flames (1998), with the pathway through fulvene (the 5 carbon ringed compound in Figures 5.36 and 5.37) accounting for the balance. The model pathway to the propargyl radical varied depending on the fuel/air equivalence ratio used and the "point" or local age chosen. Four conditions were explored: two at Φ = 1.38 and two at Φ = 1.93. The local ages chosen corresponded to peak vinyl radical (C_2H_3) concentration and peak hydroxyl radical concentration. Figure 5.36 corresponds to the major pathways from ethylene to benzene at Φ = 1.38 at peak vinyl concentration. At this point, the vinyl radical lies on the major pathway involving a combination reaction with the methyl radical (CH_3) to produce the allyl radical (CH_2CHCH_2) followed by a hydrogen radical

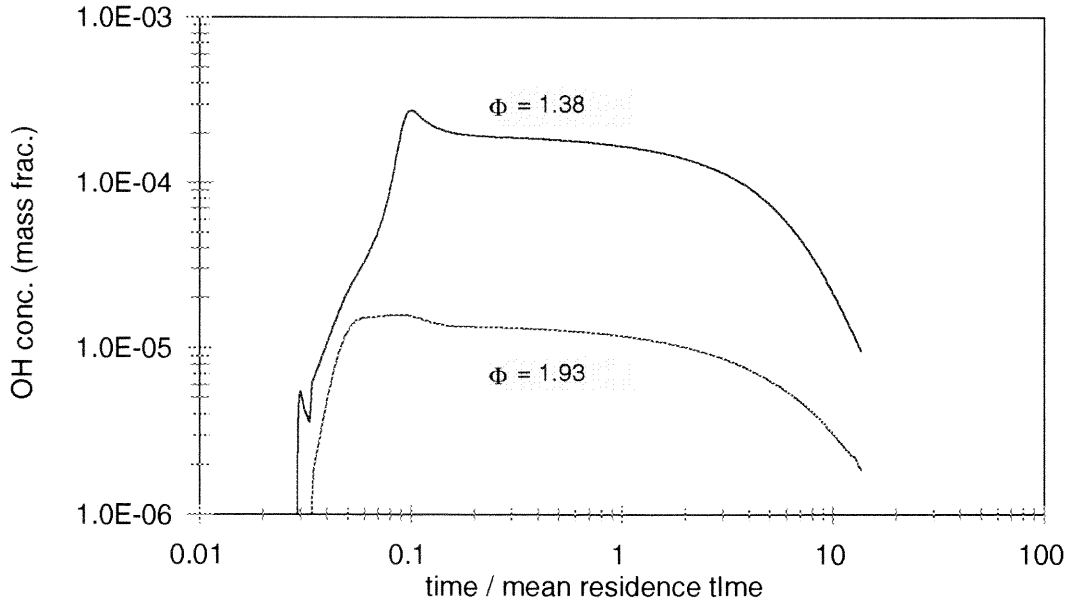


Figure 5.34 Hydroxyl Radical (OH) Age Profile in PaSR (3162 Hz) Model for Fuel-Rich Ethylene/Air Combustion

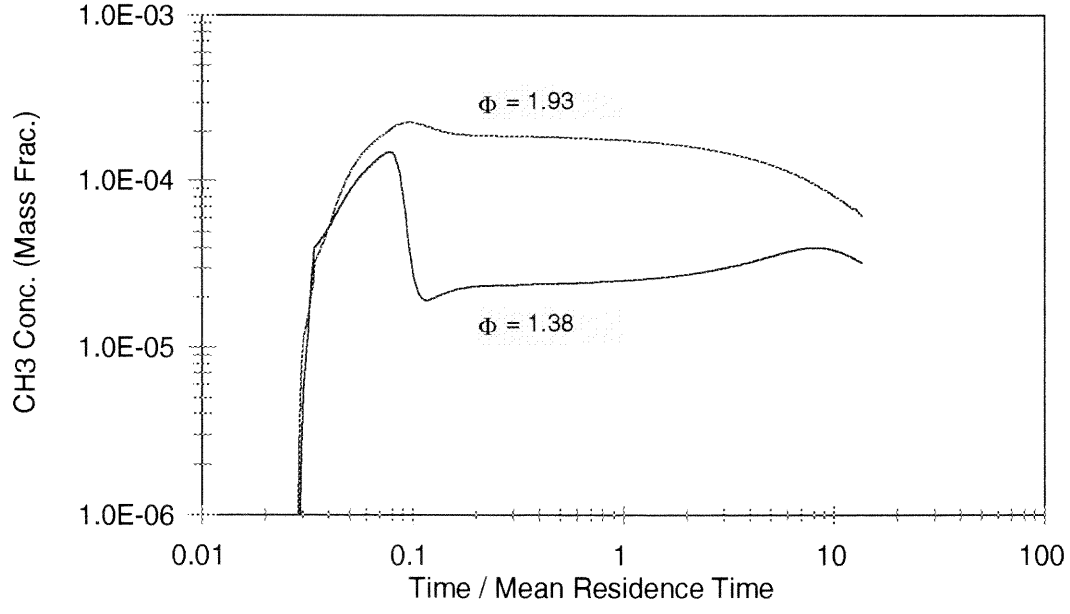


Figure 5.35 Methyl Radical (CH_3) Age Profile in PaSR (3162 Hz) Model for Fuel-Rich Ethylene/Air Combustion

abstraction to allene (CH_2CCH_2). Another hydrogen radical abstraction produces the propargyl radial that is stabilized by it resonance structure (Marinov, 1998). When the local age has advanced less than 300 microseconds from 0.063τ to 0.101τ (where the mean residence time, τ = 7.5 milliseconds), OH concentration peaks and the "point" temperatures rises from 1460 K to 1688 K. At these conditions, reactions of benzene and major C3 and C4 intermediates shift to net consumption. However, at a higher fuel/air equivalence ratio the regime become one of net production. Figure 5.37 shows major reaction pathways to benzene at peak OH concentration for Φ = 1.93 at a local temperature of 1629 K. Several differences become apparent. The two major paths to the propargyl radical are a singlet methylene radical ($CH_2(S)$ or $^1CH_2$) addition with acetylene (C_2H_2) and a hydrogen abstraction of propyne (CH_3CCH). Propyne is arrived at through a combination reaction between the methyl and vinyl radicals to form allene followed by an isomerization reaction.

The fuel-rich conditions with low levels of oxidizing radicals that result in a high sustained level of hydrocarbon radicals makes the micromixing modeled by the PaSR of little consequence in predicting benzene concentrations. However, the rapid OH radical rises seen in fuel-lean combustion, studied earlier in this chapter, that result in brief periods of low temperature oxidation provides a better demonstration of the value of this model. In the Marinov mechanism the formation of benzene, followed by hydrogen abstraction to the phenyl radical is the pathway to PAH formation. In premixed fuel-lean well stirred combustion, most of these reactions occur at an early local age before the temperature and OH radical concentration peaks. The PaSR model may be more successful in predicting PAH formation in these systems.

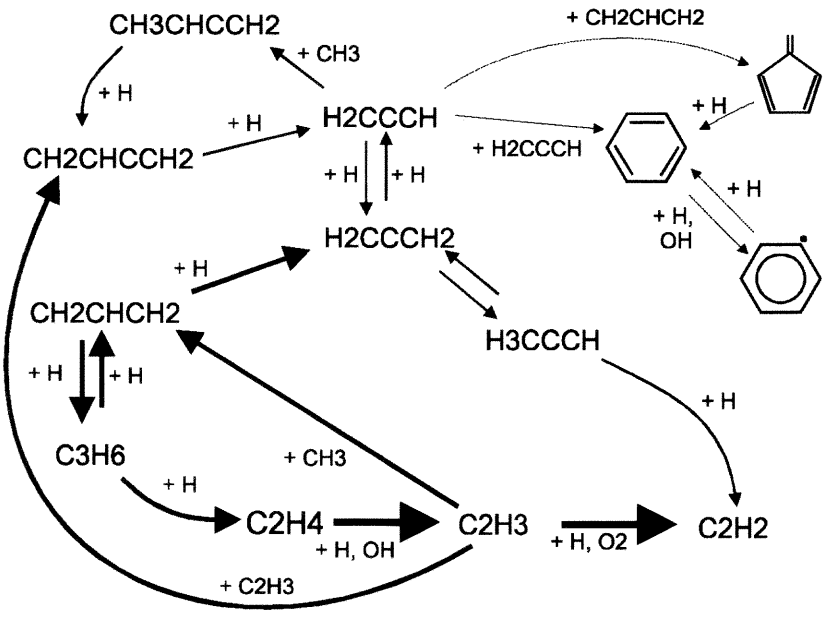


Figure 5.36 Major Reaction Pathways to Benzene (C_6H_6), Φ= 1.38 at peak C_2H_3 Concentration (t = 0.063τ and T = 1460 K)

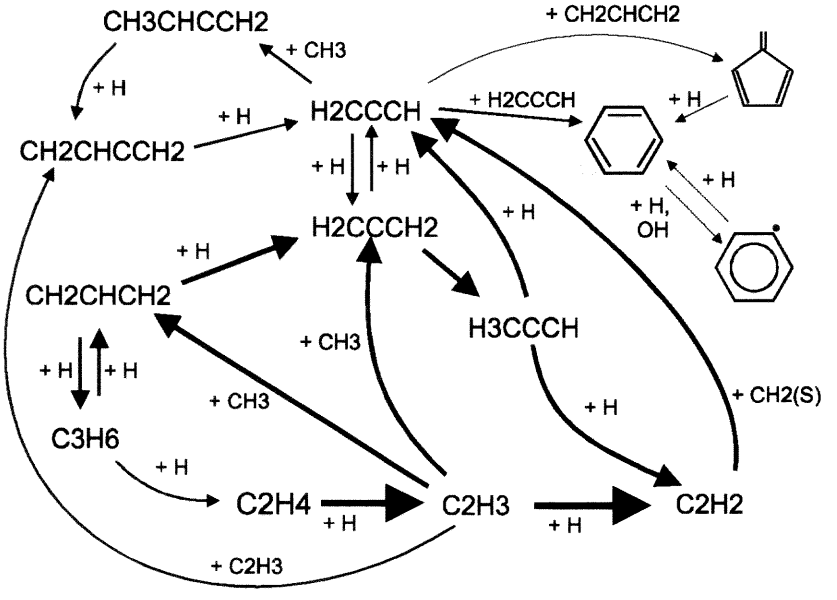


Figure 5.37 Major Reaction Pathways to Benzene (C_6H_6), Φ= 1.93 at peak OH Concentration (t = 0.088τ and T = 1629 K)

5.6 Summary

In this chapter, a PaSR model was used to simulate ethylene/air combustion in a TJSC and compare results to experimental data from fuel-lean and fuel-rich reactor conditions. The PaSR model uses an interaction by exchange with the mean (IEM) as a turbulent moment closure to simulate finite time mixing at the microscale. Local conditions are relaxed toward the mean at a rate defined by the mixing frequency, ω. Comparisons between model and experiment for fuel-lean conditions explored the reactors at near-blowout. These were modeled using the GRI 3.0 mechanism (Smith *et al.*,1999). The fuel-rich condition experiments measured the formation of aromatic hydrocarbons, specifically benzene, and important aromatic formation intermediates such as acetylene. These conditions are modeled using an *n*-butane mechanism (Marinov *et al.*, 1998).

The PaSR, and similar Probability Distribution Function (PDF) models, previously solved using a Monte Carlo simulation of a large array of fluid particles, are computationally intractable for large mechanisms such as the GRI 3.0 mechanism. An alternate solution technique is employed in this work that approximates mean conditions and solves the deterministic model to refine the approximation and eventually converge on a solution. The approximation, direct integration, and convergence technique compared favorably with the Monte Carlo modeling calculations presented by Correa (1993 and 1995) for smaller mechanisms, but using, on average, less than $1/200^{th}$ of the CPU time. This new technique allows use of considerably larger detailed mechanisms.

The comparison of PaSR model predictions to the experimental data from a TJSC operating in both fuel-lean and fuel-rich conditions showed good agreement between model and data. Analysis of the fuel-lean data near blowout showed best agreement with the PaSR simulations at a mixing frequency of 3162 Hz, a plausible frequency based on analysis by Nenniger *et al.* (1984), against the measurements of hydrocarbon PICs and unburned fuel (ethylene). Decreasing the mixing frequency had the effect of delaying combustion. It produced a greater range of local temperatures and local concentrations. Likewise, increasing the dilution of the feed also delayed combustion. This had the effect of increasing the bimodal nature of the temperature distribution, similar to the effect observed by Barat (1990) from Rayleigh scattering measurements. The PaSR model also simulated a broader distribution of CO concentrations near blowout that showed a "local blowout" effect. For fuel-rich conditions, PaSR model predictions had excellent agreement with experimental results, but the results of the PaSR model converged with the results of PSR model as the fuel/air equivalence ratio increased. Analysis showed that the reduced concentrations of the hydroxyl and other oxidizing radicals at the higher fuel/air equivalence ratios produced a rather constant age profile of hydrocarbon radicals similar to the uniform composition expected in a PSR. Overall, the PaSR model gave better predictions than the PSR model, and simulated other effects, such as the temperature distribution, that cannot be modeled by a PSR.

Finally, this chapter presented a generalized PaSR model applicable to non-ideal macromixing that would be characterized by an RTD reflecting multiple PSRs in series. The idea of developing a hybrid micromixing model between the Segregated Flow and Maximum Mixedness models is not new. Spencer *et al.* (1980) reviewed several concepts of an

approach to the Maximum Mixedness model, but with delayed mixing. One mixing rule kept fluid "points" segregated until a fixed age and completely mixed thereafter, and the other rule kept fluid "points" segregated until a fixed fraction of the total time in the reactor and completely mixed thereafter. Neither mixing rule produces a continuous PDF. Instead, this chapter proposed a mixing rule that relaxes the composition between individual fluid "points" with the mean of other "points" with the same life expectancy in the reactor. This mixing rule extends the Interaction-by-Exchange-with-the-Mean (IEM) approach, but uses a localized mean of the group of fluid "points" that exit at the same time.

Caution is warranted when applying any IEM model. IEM reduces complex geometric effects to one or two parameters and fails to adequately address local effects. These limitations can be overcome. Applications to simple geometries in conditions with high turbulent kinetic energies and sufficiently small integral time scales (Equation 5.24) are most likely to produce good results. The IEM approach offers the advantage of using detailed thermo-kinetic mechanisms in a micromixing model. This is a potential tool for analyzing practical combustion systems.

CHAPTER 6

APPLICATION AND ANALYSIS OF THE NETWORKED IDEAL REACTOR MODEL

In this chapter, the model developed in Chapters 3 and 4 is applied with detailed thermo-kinetic mechanisms. Use of a networked ideal reactor model allowed for the application of the large mechanisms needed to describe combustion of chlorinated hydrocarbons (CHC). Comparisons are made between model predictions and experimental data at various sample points for different measured PICs. In general, the model results do not agree well with the data. However, the trends exhibited in both data and model provide insights about SCC design.

6.1 Experimental

Chapter 3 provides a complete description of the pilot scale Rotary Kiln Incinerator Simulator (RKIS). Throughout all tests, the Eclipse 82 MTVA primary natural gas burner in the rotary kiln section (see Figure 3.1) was operated at 73 kW with a slight excess of oxygen. The afterburner operated at a constant fuel rate of 51 kW. The SCC was varied between three conditions – lean, stoichiometric, and rich – by varying the air flow to the afterburner. Test burn runs compared one of two surrogate wastes, carbon tetrachloride (CCl_4) or methylene chloride (CH_2Cl_2), to blank runs that were used as a baseline. Nitrogen pressurized liquid

surrogate waste was injected at 80 mL/min into point B2 (see Figure 3.1) in an air atomized nozzle. The droplet size of the spray was altered by using one of two pressure settings. Table 6.1 shows a summary of these conditions.

Table 6.1 Reactor Conditions

	Φ	Fuel/Dopant (mol/hr)	Air (mol/hr)	Excess O_2 (%)*
Fuel Lean				
Main Burner	0.85	311	3570	2.9
Afterburner	0.91	201	2140	1.7
SCC Overall	0.72	512	5710	2.5
CCl_4	0.72	49.6		2.5
CH_2Cl_2	0.86	74.9		1.2
Stoichiometric				
Main Burner	0.85	311	3570	2.9
Afterburner	1.05	201	1850	n/a
SCC Overall	0.82	512	5420	1.5
CCl_4	0.82	49.6		1.5
CH_2Cl_2	0.96	74.9		0.3
Fuel Rich				
Main Burner	0.85	311	3570	2.9
Afterburner	1.25	201	1560	n/a
SCC Overall	0.93	512	5130	0.6
CCl_4	0.93	49.6		0.6
CH_2Cl_2	1.10	74.9		n/a

* mole percentage of oxygen in product stream assuming complete combustion

Samples were collected from preliminary runs using an EPA standard Volatile Organic Sample Train (VOST) and analyzed in a GC/ Mass Spectrometer. Samples were made in the burnout section of the SCC using reference samples of each surrogate waste or dopant. Based on these VOST samples a list of target analytes in Table 6.2 were developed. Although this list is extensive, concentrations of chloromethane (CH_3Cl), perchloroethylene

(C_2Cl_4), trichloroethylene (C_2HCl_3), chlorobenzene (C_6H_5Cl), and chloroform ($CHCl_3$), as well as the dopants, dominated the chlorinated hydrocarbon (CHC) PICs, and benzene (C_6H_6), and to a limited extent, toluene ($C_6H_5CH_3$), dominated the hydrocarbon PICs. The lighter C_1 and C_2 hydrocarbons were not measured.

Table 6.2 Target Analytes

chloroform	1,2, dichlorobenzene	ethyl benzene
carbon tetrachloride	methylene chloride	m,p xylenes
benzene	trans 1,2 dichloroethane	o xylene
toluene	methyl ethyl ketone	styrene
trichlorethylene	1,1,1 trichloroethane	1,3 dichlorobenzene
perchloroethylene	1,2 dichloroethane	1,4 dichlorobenzene
chlorobenzene		

Sample runs consisted of combinations of reactor condition (lean, stoichiometric, or rich), dopant (carbon tetrachloride, methylene chloride, or blank), injection pressure (high or low), and sample point (3, 4, or 5). Regression analysis performed by Lemieux (Bass, Barat, Sacchi, and Lemieux, 1995) analyzed the on-line GC results for trends based on choice of dopant, residence time (based on port location), and equivalence ratio. Results showed a strong correlation ($R^2 = 0.9195$) for benzene formation with fuel/air equivalence ratio, but independent of the dopant used. Strong relationships with all factors were found with toluene ($R^2 = 0.9362$), methylene chloride ($R^2 = 0.7376$), perchloroethylene ($R^2 = 0.6598$), and trichloroethylene ($R^2 = 0.6277$). Unfortunately, comparable levels of trichloroethylene were also detected in the blank runs, so it is a possible low-level contaminant in all runs and makes the data suspect. GC and continuous emission monitor data are listed in Appendix B.

6.2 Modeling Approach

A computer code, titled *AFTERBURNER* Version 2.4 (Appendix D), was developed to solve the network of ideal reactors modeled after the physical SCC in Chapter 4 and summarized in Figure 4.16 and Table 4.4. The code uses the *CHEMKIN-II* library (Kee *et al.*, 1994) with *PSR* (Glarborg *et al.*, 1986). *AFTERBURNER* was designed with features to:

- read user input for reactor parameters and multiple input streams.
- calculate afterburner and overall fuel/air equivalence ratios and adjust the air stream to a specified value of Φ.
- accept reactor size as a volume or mean residence time.
- use heat transfer coefficients for calculating heat loss or specify a PSR operating temperature, and estimate heat transfer coefficients for PFR sections based on a desired output temperature.
- mix streams according to the network model.
- save output corresponding to the sample points in a formatted files for selected species.
- run sensitivity analysis of the model parameters.

The on-line GC sample train was simulated as a fixed temperature PSR with an elevated water vapor content that simulated the rapid quench of the impinger. Since temperature and time measurements of the sample lines were not taken, this was not developed as part of the model.

Several approaches were available for determining the energy balance within the reactor applicable to all reactor conditions. Since the one set of ideal reactor temperatures deduced from physical temperature measurements applied only to stoichiometric conditions,

a method was needed to generalize these findings to all modeled conditions. The simplest approach would have been a fixed temperature approximation for all ideal reactors for all conditions. This, of course, ignores the temperature dependence on fuel/air equivalence ratio. To capture this effect, a heat transfer model was established and calibrated with the single set of temperature data. The initial approach used for all reactors was an overall wall conduction model. Specifically:

$$Q_{loss} = UA(T - T_a) \tag{6.1}$$

where T is the reactor temperature, T_a is the temperature of the surroundings and UA is the product of the surface area and the overall heat transfer coefficient. For the burnout section, T_a is the ambient temperature of the room which was approximately 40 °C. The heat transfer in the mixing section was more complex. The hotter flame is separated from the cooler wall by the cooler kiln gas stream. If the temperature of the kiln gas is used as the value for T_a in Equation 6.1, the resulting value of UA becomes too large to maintain flame stability under fuel-rich conditions. This results in model predictions of flame blow-out.

Radiation heat transfer is a more appropriate heat transfer model for the flame in the mixing section, and is the typical heat transfer mechanism from the flame in boilers and incinerators. The radiation from a flames is given by (Hottel and Sarofim, 1967):

$$Q = A_s \varepsilon_s \varepsilon_G \sigma (T_G^4 - T_s^4 \alpha_G / \varepsilon_G) \tag{6.2}$$

where ε is the emissivity, α is the absorptivity, σ is the Stefan-Boltzmann constant, and the subscripts s and G stand for the wall surface and the gas respectively. The gas emissivity is

not likely to remain constant over the range of conditions since it is dependent on H_2O and CO_2 concentration. The soot induced luminosity of a fuel rich flame also increases this value. If these complications are ignored and the ratio of absorptivity to emissivity considered unity which is a reasonable approximation when the ratio of T_G to T_S is not far from unity, then radiative heat loss is given by (Perry and Green, 1984):

$$Q = (\bar{GS})\sigma(T_G^4 - T_s^4) \tag{6.3}$$

where $\bar{GS}$ is the pseudo total exchange area. This is approximated from an enthalpy balance using a computer run with the specified temperatures from Table 4.4. Not only does the radiation heat loss model for PSR 1 make better physical sense, the more rapid fall-off of heat loss, as the equivalence ratio drives the flame temperature down, allows for stable model solutions in fuel-rich conditions.

Equation 6.1 was retained as the heat transfer model for PSRs 2 and 3, and PFR 4. However, the flame temperature was substituted for the ambient temperature since the flame heated the kiln gas in the mixing section. Table 6.3 summarizes the heat transfer models and parameters used. With the exception of PSR 1, the computer model determined the ideal reactor temperatures during execution. The temperature of PSR 1 was determined using the GRI 3.0 mechanism without the nitrogen chemistry to speed execution. The temperature was found based on a iterative solution using energy balance between the heat of combustion and radiative heat loss. The GRI 3.0 mechanism was also used to determine the initial guess for each ideal reactor temperature for the application of the more complex CHC chemistry mechanisms.

Table 6.3 Heat Transfer Parameters and Energy Balance

Reactor	T in (K)	Tout (K)	ΔH^1 (kcal/s)	Model[2]	Parameter	Units	Volume (L)
PSR1	313	1538	-4.21	$(\bar{GS})\sigma(T^4 - T_s^4)$	640.	cm^2	24.0
PSR2	977	1009	0.287	$UA(T - T_f)$	0.542	cal/(s·K)	45.2
PSR3[3]	1009	1009	0.00	$UA(T - T_f)$	0.048	cal/(s·K)	0.20
PSR4	1009	1404	3.78	$UA(T - T_f)$	13.1	cal/(s·K)	55.1
PSR5	1537	1451	-0.598	$UA(T - T_a)$	0.524	cal/(s·K)	53.1
PSR6	1419	1323	-2.13	$UA(T - T_a)$	2.10	cal/(s·K)	185.
PFR7	1323	1241	-1.58	$UA(T - T_a)$	1.62	cal/(s·K)	220.

1. Total heat loss through the SCC walls account for approximately 25 percent of energy entering the reactor
2. T is the reactor temperature; T_f is flame temperature (PSR 1); T_a is ambient temperature (40 K)
3. Heat transfer parameter for PSR 3 was calculated from PFR 4 on the per unit volume basis

Figure 6.1 shows the effect of fuel/air equivalence ratio, Φ, on the ideal reactor temperatures of PSRs1, 5, and 6, and the exit temperatures of PFRs 4 and 7. The temperatures of these reactors varied widely with Φ, while the entering kiln gas in PSRs 2 and 3 held a fairly narrow ranges from 1003 to 1012 K. The excess oxygen in the kiln gas mixing with the afterburner flame stream in the burnout section produce increasing SCC exit temperatures as the fuel-rich conditions decreased temperatures in the mixing section and choke. Since the kiln gas stream does not completely mix with the flame gas until further down the burnout section, the final combustion of the flame products occurs near the end of the SCC causing the temperature to rise in that region of the reactor.

Several chlorine mechanisms were available for use in the networked ideal reactor model. The most extensive available was developed by Chiang, Park, and Bozzelli (1994) and Chiang (1995) and includes 233 species and 727 reactions. This thermo-kinetic mechanism models the pyrolysis and oxidation of mixtures of CHCs and hydrocarbons in air. The mechanism includes the formation up to C_6 ringed compounds (benzene and chlorobenzenes). This mechanism was developed based on flow reactor experiments featuring a relatively high CHC to hydrocarbon ratio, dilute concentrations, and relatively low temperatures. This may limit its utility in systems without dilution that predominately feature hydrocarbons reacting at higher (combustion) temperatures. As large as this mechanism is, it lacks some reaction paths needed to provide an adequate simulation of benzene formation. For instance, the mechanism features only two reactions for benzene formation:

$$\begin{aligned} \cdot CH{=}CHC{\equiv}CH + CH{\equiv}CH &\rightleftharpoons C_6H_6 + H \\ CHCl{=}CHCH{=}\dot{C}H + CH{\equiv}CH &\rightleftharpoons C_6H_6 + Cl \end{aligned} \qquad (6.4)$$

Neither of these reflect more recent evidence in C_6 ring formation going primarily through the C_3 route with the propargyl radical (Marinov *et al.*, 1998) shown in Figures 5.36 and 5.37.

A smaller mechanism developed by Ho and Bozzelli (Ho and Bozzelli, 1992b, and Bozzelli, 1996) uses 62 species and 268 reactions to model CHC and hydrocarbon combustion up to C_2 carbons. This mechanism models the oxidation and pyrolysis of methylene chloride, but does not include carbon tetrachloride, which limits its utility for application to this set of experiments. Like the Chiang mechanism, the Ho mechanism was developed against data from dilute flow reactor experiments and lack some of the pathways

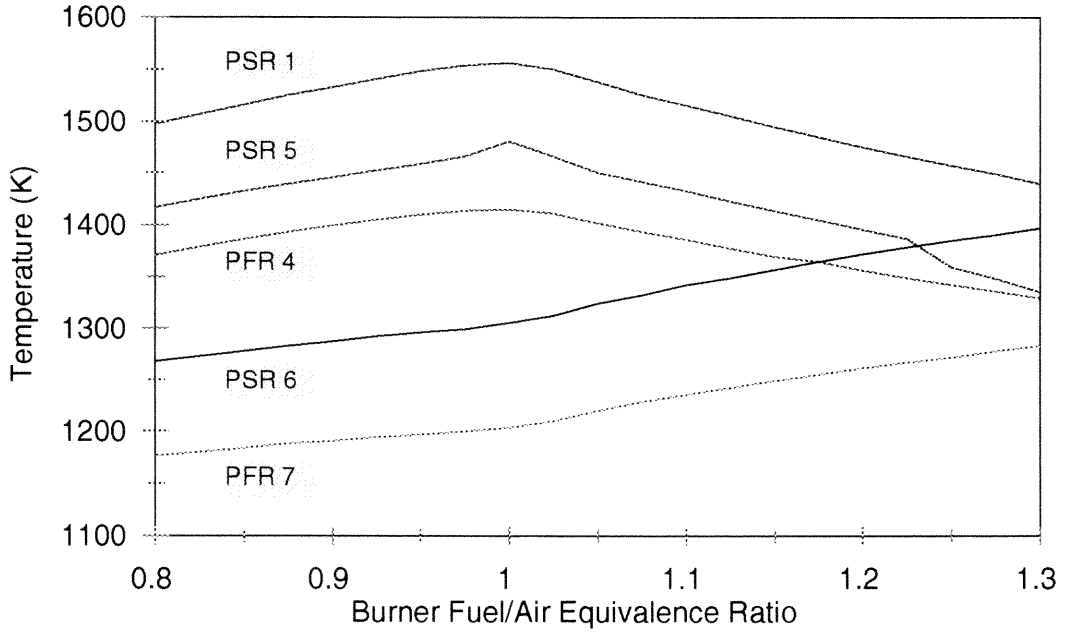


Figure 6.1 Ideal Reactor Temperatures Dependence on Afterburner Fuel/Air Equivalence Ratio (Dopant = Blank; GRI 3.0 Mechanism w/o Nitrogen Chemistry)

that are necessary for higher concentrations. For instance, on trial simulation runs, application of the Ho mechanism resulted in the production of dichloroacetylene (C_2Cl_2), an unstable compound, as the dominate CHC PIC with mole fractions on the order of 10^{-3}, but experimentally C_2Cl_2 was not detected in the VOST method. The Ho mechanism provides a single molecular elimination reaction for formation:

$$CHCl{=}CCl_2 \rightleftharpoons CCl{\equiv}CCl + HCl \qquad (6.5)$$

but not "outlets" for oxidation or pyrolysis. Chloroacetylene (C_2HCl) also has unusually high concentrations in the model runs (mole fractions on the order of 10^{-5}), but again, it was not

detected in the VOST method. Two molecular elimination reactions account for formation of C_2HCl:

$$\begin{aligned} CHCl{=}CHCl &\rightleftharpoons CH{\equiv}CCl + HCl \\ CH_2{=}CCl_2 &\rightleftharpoons CH{\equiv}CCl + HCl \end{aligned} \tag{6.6}$$

but only two reactions provide destruction pathways:

$$\begin{aligned} CH{\equiv}CCl + H &\rightleftharpoons \cdot C{\equiv}CH + HCl \\ CH{\equiv}CCl + H &\rightleftharpoons CH{\equiv}CH + Cl \end{aligned} \tag{6.7}$$

Pathway analysis of methane destruction showed other anomalies in the mechanism. Analysis under nominally fuel-rich conditions where the kiln and flame streams mixed at the entrance to PSR 6 showed that the primary path from CH_4 to CO going through the C_2 route despite relatively high mole fractions of CH_4 (4×10^{-3}) and O_2 (3×10^{-4}). These problems could be easily corrected.

6.3 Comparison Between Modeling and Experimental Results

Using the Chiang mechanism two computer simulation runs were performed, one for each dopant. Each simulation ranged between SCC burner fuel/air equivalence ratios between 0.7 and 1.3 incremented 0.25 for each step. Each run took approximately 2 hours (for 21 sets of conditions). The results are shown in Figures 6.2 through 6.9.

Figures 6.2 and 6.3 show the temperature at the end of the SCC burnout section (point 5) as a function of Φ. Temperature increases with Φ for both dopants, but peaks for methylene chloride at Φ = 1.05. Figure 6.2 shows a difference between the dopants for the model runs as much as 50 K, but when view in context of overall SCC fuel/air equivalence ratio (Figure 6.3), which includes the excess air from the kiln gas, the difference is less significant. The complex heat transfer model applied to five PSRs, which base their heat transfer ΔT on the initial approximation of reactor temperature, accounts for some of the jaggedness of the model lines and is the likely cause of the difference between the dopants. The trends of the model do not predict the results of the actual experimental measurements. The experimental temperature measurements show no dependence on Φ and average 1242 K with no statistically significant difference between the dopants. The temperatures have a wide range of variability. Some fluctuations in the excess oxygen in either the main burner or could be in the sample line. Recall in Chapter 5 how the differences in CO concentration SCC burner could account for this. This is significant because PIC concentrations are very sensitive to temperature.

Figure 6.4 shows the comparison between model runs for both dopants and the experimental data for CO concentration at point 5. The model and the data consistently show no significant difference between dopants and the exponential rise of CO with increasing Φ. However, the model consistently under predicts the experimental results. In Chapter 5, the prediction of elevated CO concentrations for poor mixing (lower mixing frequency) was a function of reactions in the sample probe and not the modeled reactor (see Figures 5.9 and 5.14). The long, unquenched sample line for the CO monitor could account for additional

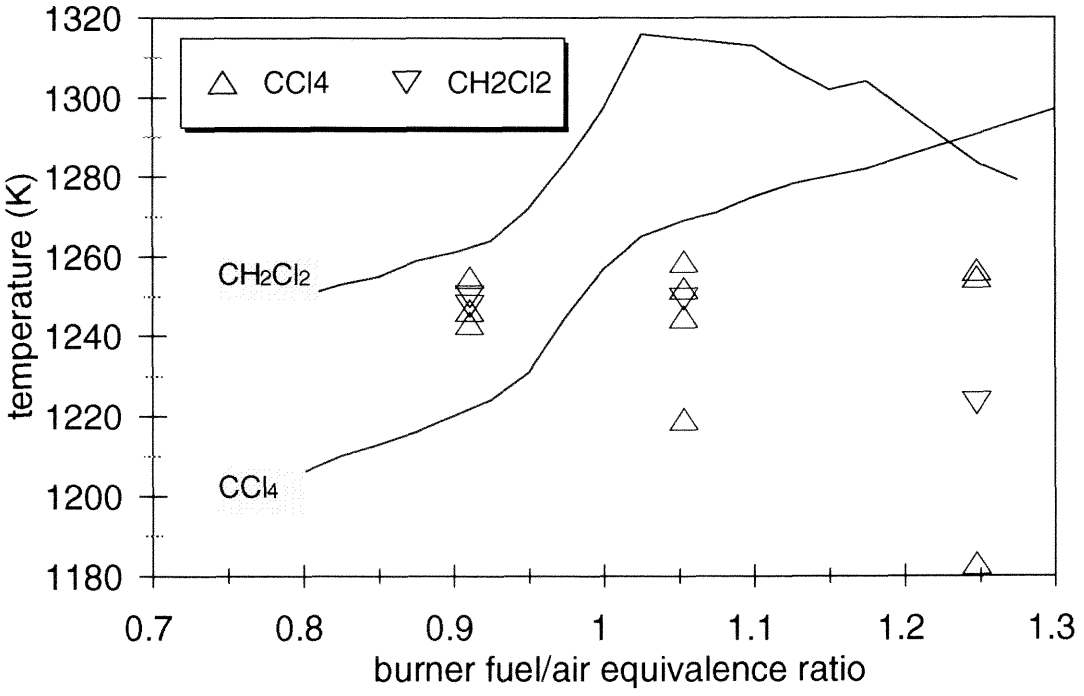


Figure 6.2 Effect of Burner Φon Temperature at Point 5, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

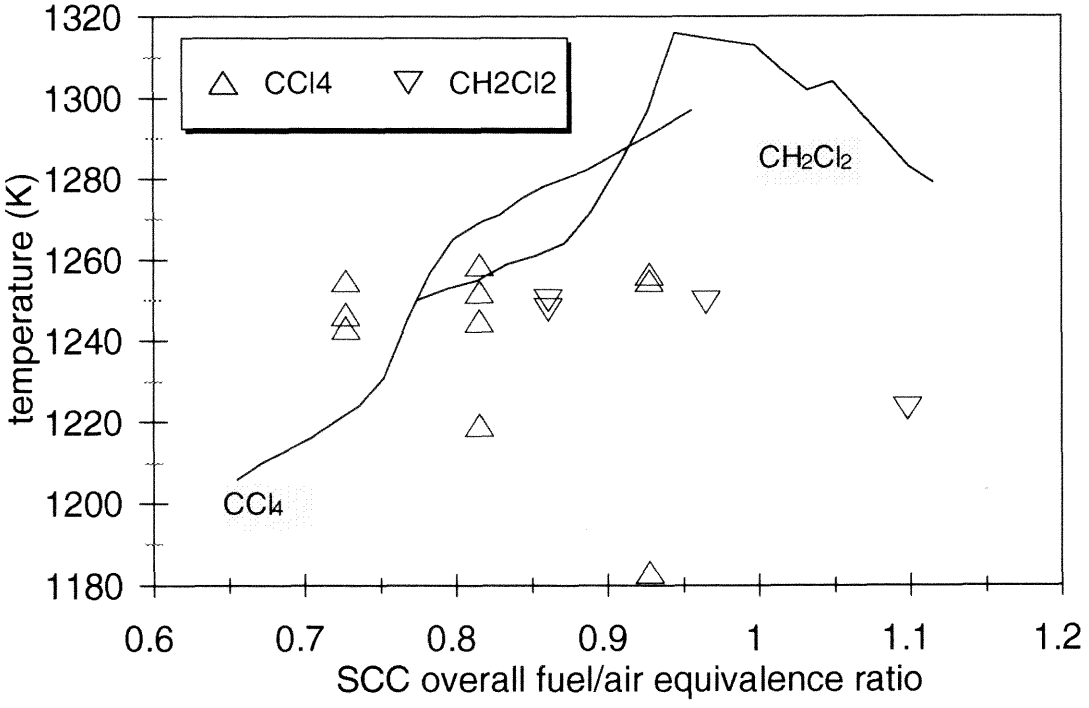


Figure 6.3 Effect of Overall SCC Φon Temperature at Point 5, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

reactions with unburnt fuel and hydrocarbon PICs raising the concentration of CO. The model consistently predicted Total Hydrocarbon (THC) concentrations on the order of 10 ppm at fuel lean conditions to 5000 ppm at fuel-rich conditions, but even complete conversion to CO would not completely account for the difference. Next, lack of sufficient mixing of fuel and air in the flame, and lack of sufficient mixing of the dopant in the mixing section could have caused the higher measured concentrations of CO. An ideal reactor, as well as a PaSR, model assumes premixed conditions, but actual conditions were non-premixed. Non-premixed fuels can produce fuel-rich zones that promote CO production. Experiments done by Sacchi (Bass *et al.*, 1995) showed that intentionally produced poor mixing regions with fuel (or dopant) elevated the levels of CO and also increased benzene production.

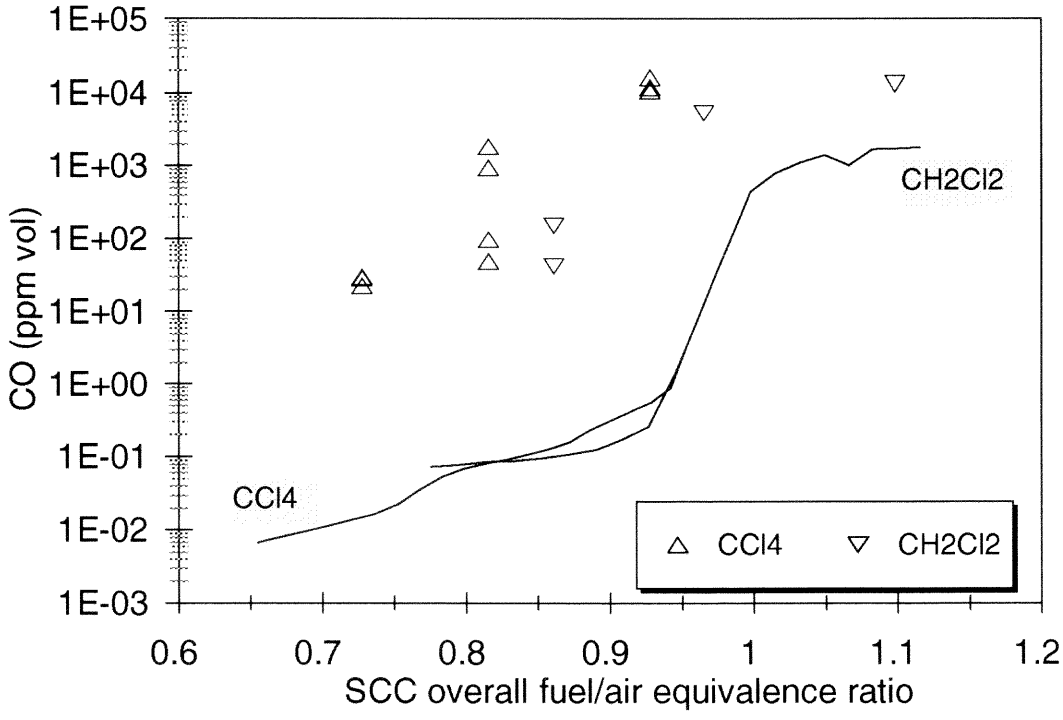


Figure 6.4 Effect of Overall SCC Φ on CO Concentration at Point 5, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

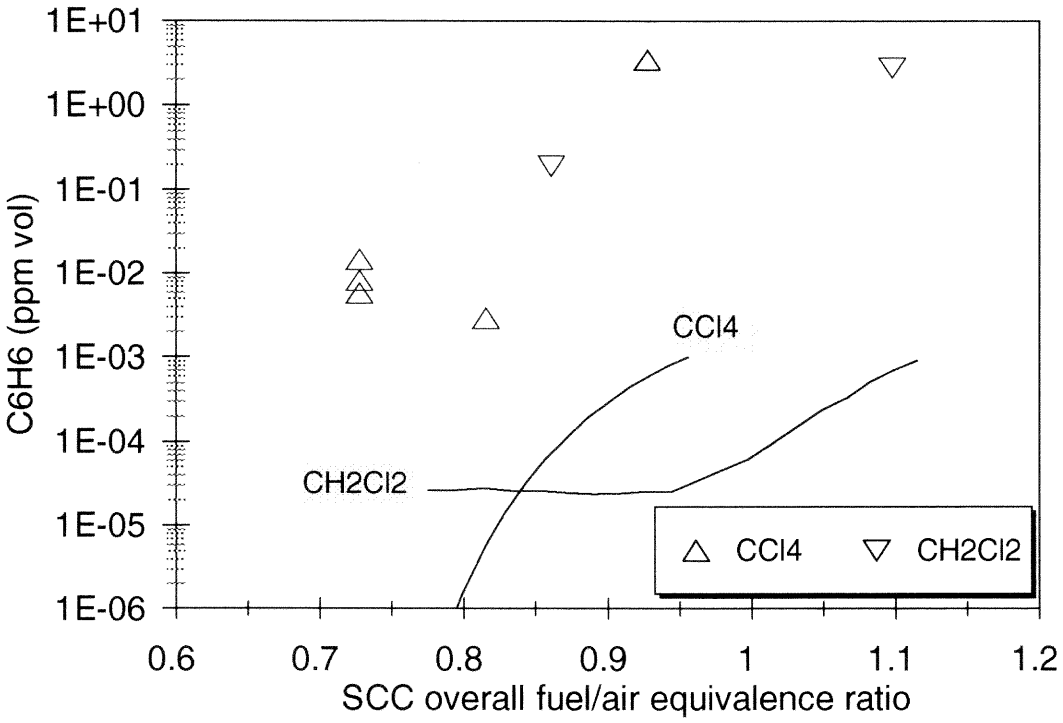


Figure 6.5 Effect of Overall SCC Φ on Benzene Concentration at Point 5, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

Figure 6.5 compares model predictions with measured benzene at point 5. The y-axis clips off the lower range of the model prediction with CCl_4 as the dopant in order to show the dependence of the experimental measurements on Φ. The model underpredicts the benzene concentration by 4 orders of magnitude, but shows the trend of increased benzene production with increasing Φ. The initial regression analysis performed by Lemieux (Bass *et al.*, 1995) showed no difference between dopants, but this was based on a independent variable of burner fuel/air equivalence ratio. When the oxygen demand of the dopant is considered, the model predicts a difference that may be reflected in the experimental data, but there are insufficient data points for the methylene chloride dopant to validate this difference. As

discussed earlier, the increased benzene concentrations above model predictions could be indicative of poor mixing. Other mechanisms such as heterogenous reactions linked to soot growth, not accounted for in a premixed homogeneous model, could also account for the difference.

Figures 6.6 and 6.7 compare model predictions with experimental measurements of chloroform ($CHCl_3$) at SCC choke (point 4) and point 5. Comparing the two figures shows that chloroform is produced earlier in the SCC and not effected by burner fuel/air equivalence ratio. Pathway analysis at points 3, 4 and 5 shows that almost all production occurs in the mixing section in PSR 2. The mechanism produces different pathways for each dopant. For methylene chloride the path is:

$$CH_2Cl_2 \xrightarrow{+Cl} \dot{C}HCl_2 \xrightarrow{+Cl,Cl_2} CHCl_3 \tag{6.8}$$

For carbon tetra chloride the pathway is:

$$CCl_4 \xrightarrow{+Cl} \dot{C}Cl_3 \xrightarrow{+H_2O} CHCl_3 \tag{6.9}$$

Within the mixing section, the model predicts concentrations of chloroform two orders of magnitude higher for the carbon tetrachloride dopant, which follows in the choke, but not at the end of the burnout section. Both models under predict chloroform at that point. The data shows no apparent difference between dopants and chloroform concentration. The agreement between model and data that chloroform is formed early in the SCC is significant. In the networked ideal reactor model PSR 2 does not mix with the flame stream and has a relatively low temperature (1000K) consistent with experimental measurement. So, $CHCl_3$ formation depends on low temperatures and high concentrations of dopant.

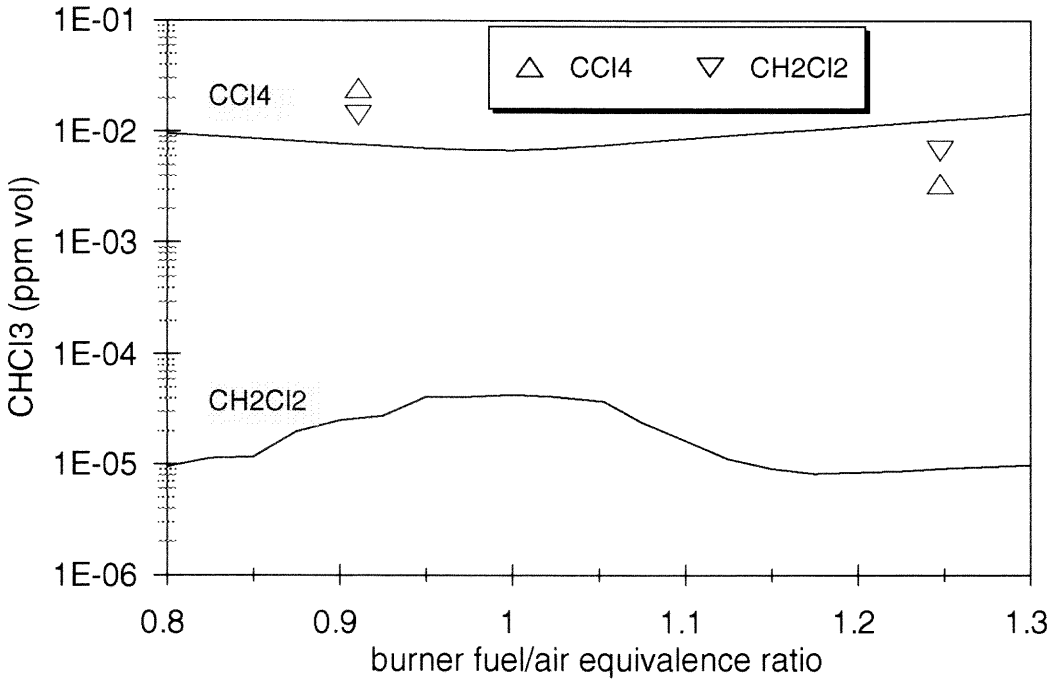


Figure 6.6 Effect of Burner Φ on Chloroform Concentration at Point 4, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

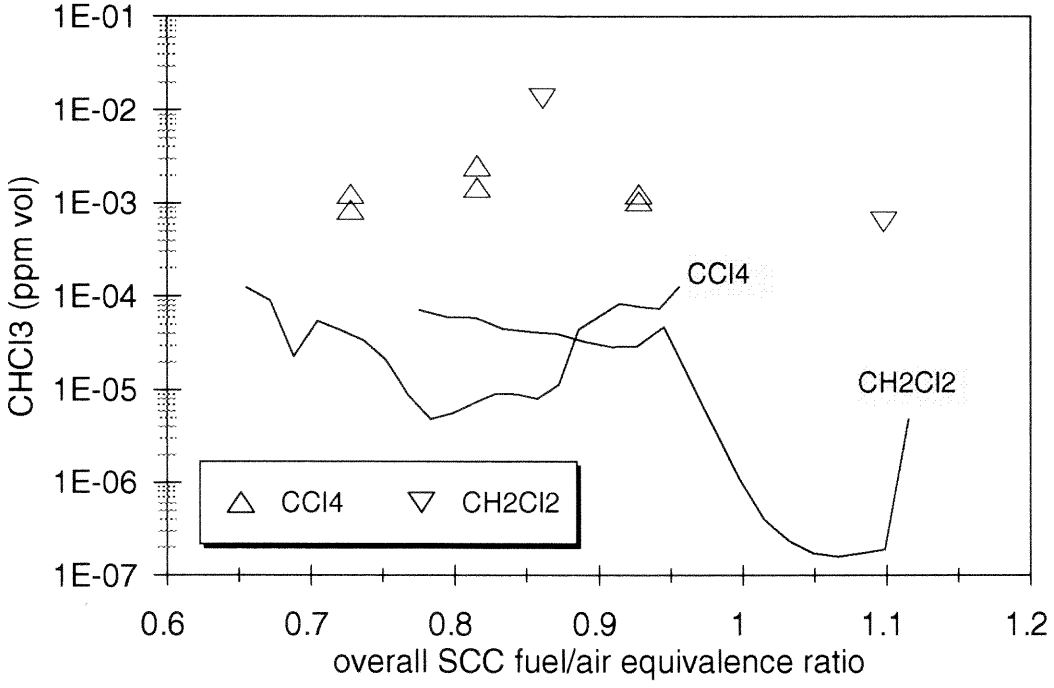


Figure 6.7 Effect of Overall SCC Φ on Chloroform Concentration at Point 5, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

The next set of graphs show the comparison perchloroethylene (C_2Cl_4) predictions and measurements at points 4 and 5. Like chloroform, perchloroethylene is formed early in the SCC and consumed in the burnout section. Also, like chloroform, burner fuel/air equivalence ratio has little impact on perchloroethylene concentration for both model and experimental data. The model tends to overpredict perchloroethylene concentration with CCl_4 as the dopant, but performs reasonably with methylene chloride. The mechanisms of formation are somewhat different between dopants. With carbon tetrachloride as the dopant, formation involves a 2 or 3 step process early in the mixing section:

$$CCl_4 \xrightarrow{+Cl} \dot{C}Cl_3 \xrightarrow{+\dot{C}Cl_3} C_2Cl_6 \xrightarrow{-Cl_2} C_2Cl_4 \tag{6.10}$$

with an almost equal alternate path between the trichloromethyl radical and perchloroethylene. The path with methylene chloride is longer and slower with the concentration of perchloroethylene in the mixing section (point 3) several orders of magnitude lower than the concentration in the choke. The pathway is:

$$\begin{aligned} &CH_2Cl_2 \xrightarrow{+Cl} \dot{C}HCl_2 \xrightarrow{+\dot{C}HCl_2} CHCl_2CHCl_2 \xrightarrow{-HCl} \\ &C_2HCl_3 \xrightarrow{+Cl} \dot{C}ClCCl_2 \xrightarrow{+HOCl} C_2Cl_4 \end{aligned} \tag{6.11}$$

Again, like chloroform production, the formation of chlorinated PICs early in the SCC suggests that control of PIC formation requires limiting low temperature areas with high POHC concentrations. Additional graphs that compare model and experimental data for chlorobenzene and carbon tetrachloride are in Appendix C.

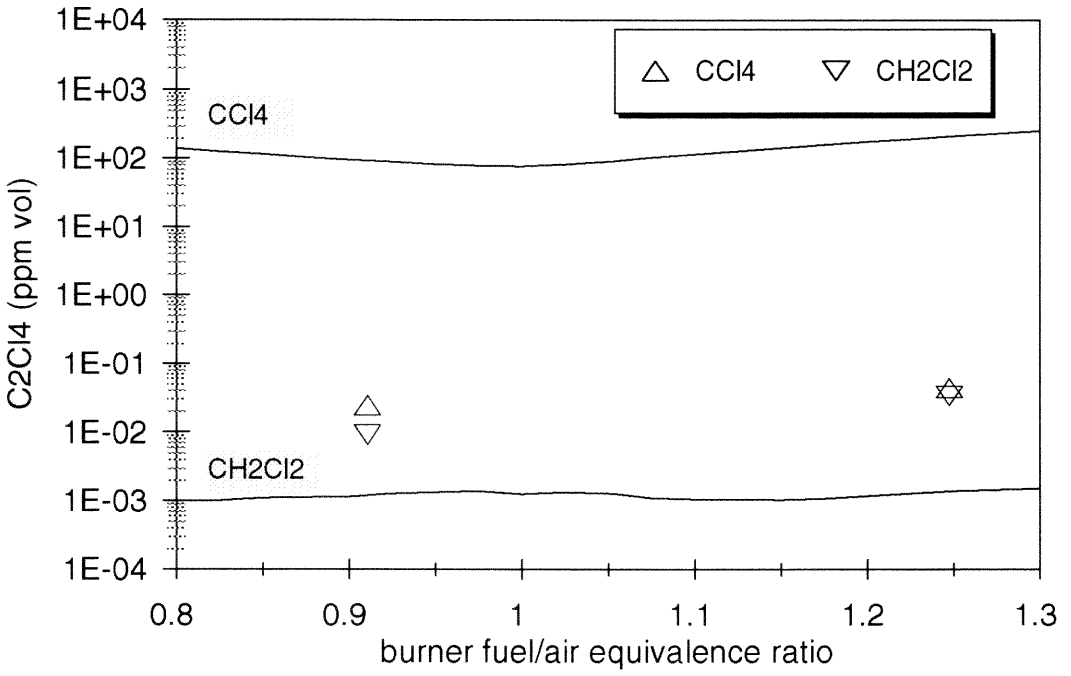


Figure 6.8 Effect of Burner Φ on Perchloroethylene Concentration at Point 4, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

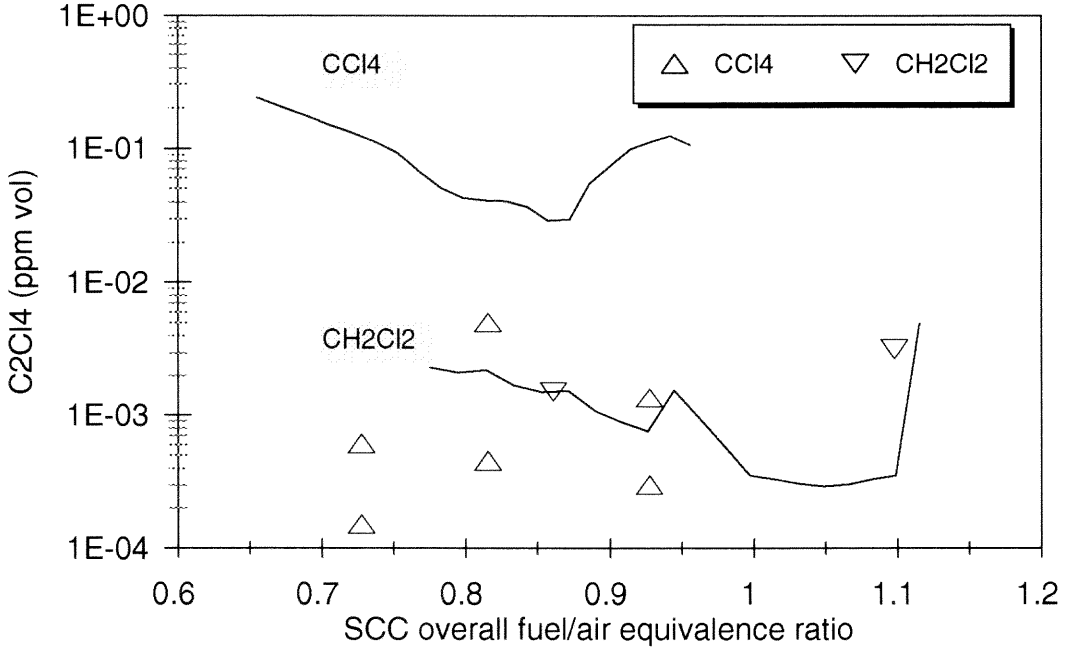


Figure 6.9 Effect of Overall SCC Φ on Perchloroethylene Concentration at Point 5, Comparison Between Model (using Chiang Mechanism) and Experimental Results for Different Dopants

6.4 Analysis of Results

Even though agreement between model and data is poor, the networked reactor model combined with the thermo-kinetic mechanism shows the same trends as the experimental data. Some of these trends can be analyzed in order to gain insight into the effect of reactor geometry and macromixing on performance. The sensitivity analysis of the model parameters is one method of exploring these trends. In so far that the parameters of the networked ideal reactor model reflect the basic macromixing characteristics of the SCC, an analysis of the sensitivity of those parameters to variations provide insight to the effect of modifying those characteristics on PIC formation.

In the context of the mass fractions of a system with *kk* components the normalized first order sensitivity coefficient is defined as:

$$\bar{S}_{ij} = \frac{\alpha_i}{Y_j}\frac{\partial Y_j}{\partial \alpha_i} \tag{6.12}$$

where α_i is the i^{th} parameter. On a mole basis the sensitivity coefficient becomes (Glarborg *et al.*, 1986):

$$\hat{S}_{ij} = \frac{\alpha_i}{X_j}\frac{\partial X_j}{\partial \alpha_i} = \bar{S}_{ij} - \alpha_i \bar{W}\sum_{j=1}^{kk}\frac{1}{W_j}\frac{\partial Y_j}{\partial \alpha_i} \tag{6.13}$$

In a system with *n* parameters, the perturbation of the resulting j^{th} mole fraction is the sum of the perturbation of each parameter times the respective molar sensitivity coefficient (Frank, 1978):

$$\frac{\Delta X_j}{X_j} = \sum_{i=1}^{n} \hat{S}_{ij} \frac{\Delta \alpha_i}{\alpha_i} \tag{6.14}$$

The uncertainty of the 7 volumetric parameters (of the SCC model) is quantified in the standard errors determined in Chapter 4. However, since the uncertainty of the temperature parameters is not quantified the sensitivity coefficients will not be used to quantify the uncertainty of the resulting mole fractions. The sensitivity coefficients are useful to analyze trends, but in a system with 233 chemical species and 15 model parameters that results in 3495 sensitivity coefficients for each set of reactor conditions, not all combinations will be considered in this analysis. Rather, this section will look at two PICs, benzene and perchloroethylene, and focus on the larger sensitivity coefficients.

Predicted concentrations of benzene at the SCC exit are highly dependent on temperature in PSR 1, the SCC burner flame. When the dopant is CCl_4, sensitivity coefficients range between -30 for fuel-lean to 1.2 for fuel-rich. Under fuel-lean and stoichiometric conditions, temperature increases in PSR 1 decrease benzene output at the SCC exit. This same trend is noted when methylene chloride is the dopant where sensitivity coefficients, $\hat{S}_{ij}$, range from -13 at fuel-lean conditions to -29 at fuel-rich conditions. With CH_2Cl_2 as the dopant, the heat transfer to PFR 4 is also important, especially under fuel-lean conditions where the sensitivity coefficient is -74. These findings suggest that increased burner temperature and better heating of the kiln gas stream will reduce benzene in the SCC exit.

The formation of perchloroethylene is also dependent on conditions in PSR 1. For the carbon tetrachloride dopant, higher temperatures in PSR 1 produce a significant downward trend in C_2Cl_4 at the SCC exit. At first glance, this does not make intuitive sense, because the CHC dopant does not mix into the SCC burner flame. However, the temperature of PSR 1 has a significant impact on the temperatures of PSR 2, 3, and PFR 4, which are all in the mixing section of the SCC. This effect is also reflected in the sensitivity of the heat transfer coefficient of PFR 4. The reactor size of PSR 1 also has an effect with the methylene chloride dopant system. Increases in the SCC burner flame size decrease perchloroethylene at the SCC exit, with the sensitivity coefficients ranging from -20 (fuel-lean) to -450 (fuel-rich). Since it also has the same effect on hydrocarbon PICs such as acetylene and ethylene, this effect suggests a relationship between downstream hydrocarbon concentration and the destruction of C_2Cl_4 downstream. These effects taken together suggest that a higher burner flame temperature and decreased hydrocarbon PIC production in the flame by better mixing and an increased recirculating zone would reduce the formation of CHCs downstream.

6.5 Summary

In this chapter, a networked ideal reactor model identified in Chapter 4 was applied to the SCC over a variety of conditions. A heat transfer model was used to apply the temperature parameters, developed from data at stoichiometric conditions, to a range of conditions. A thermo-kinetic mechanism developed by Chiang (1995) was applied to the model. Even though this mechanism has shortcomings when applied to combustion conditions, it contained sufficient detail to provide predictions for many of the target analytes measured by the on-line

GC. The model results did not agree well with the experimental data. However, it followed many of the underlying trends revealed by the data.

Several factors could have improved the modeling effort. More data were required on the operating conditions of the SCC, especially reactor temperatures using a suction pyrometry at multiple points during operation. This would have reduced the level of extrapolation required in developing a heat transfer model. Additionally, no data were available on the effect of the sample quench upstream of the on-line GC and how this might have affected results. Additional validation of the thermo-kinetic mechanism would have increased confidence in the model. This should include data taken at combustion temperatures in research devices such as a TJSC or a laminar flat flame. The aromatic chemistry of the Chiang mechanism requires additional work to provide for additional potential reaction pathways especially through the propargyl radical. These modifications would require additional validation on idealized bench scale systems.

The underlying trends mirrored by the reactor model provide a means to suggest SCC design modifications to improve performance. The trends found by parameter sensitivity analysis suggest that SCC performance could be improved by: (1) maintaining a high burner flame temperature; (2) reducing unburned hydrocarbons in the burner effluent by improving mixing and increasing the recirculating volume; and (3) achieving earlier heating of the kiln gas perhaps by using multiple tangential entries into the mixing section. Further investigation into this system should determine the effect of such improvements.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Conclusions

This research has explored the identification and application of an ideal reactor model to a Secondary Combustion Chamber (SCC). This was done to explore the middle ground between detailed thermo-kinetic mechanisms, which have been generally applied to well-defined mixing systems, and the application of detailed turbulent mixing models with reduced thermo-kinetic mechanisms to complex geometries. This section summarizes what has been learned about mixing in a practical combustion system and success in applying detailed thermo-kinetic mechanisms to it. It also, provides conclusions regarding the ability to depart from the concept of ideal mixing while still retaining detailed chemistry in the model. Finally, recommendations are made regarding the general design of an SCC, and future research in the area.

Analysis of the geometry of the SCC suggested the existence of up to four distinct mixing zones (flame zone, kiln gas zone, entrainment zone, and post-choke mixing zone), that is, four distinct zones which could be modeled as PSRs. Temperature profiles supported this conjecture by showing the presence of two distinct zones in the mixing chamber, and two emerging streams from the choke that rapidly mix as they move downstream. The RTDs, which were resolved from the tracer studies, further supported a multiple PSR model. The tracer results validated the supposition that the flame and the kiln gas maintain two distinct

zones in the mixing chamber. This has important consequences for secondary combustion chamber designs that rely on the tangential mixing of the kiln gas. Little mixing occurred between the two streams until further downstream in the burnout section.

A model was chosen based on the best fit to SO_2 tracer data and consistency with physical geometry, resulting flow patterns, and temperature measurements. The chosen model is summarized in figure 4.16 with parameters in table 4.4. A frequency bandwidth limitation introduced by the SO_2 analyzer precluded distinguishing solutions that differ only in the higher frequencies. Hence, a best-fit model does not represent a unique solution and other considerations such as reactor geometry, analysis of flow patterns and swirl, and temperature cross section measurements also merited equal consideration in model selection Temperatures assigned using cross-section temperature profiles and energy balances enabled ideal reactor volumes to be determined. The difference between the active and physical volumes indicated the presence of 53 L of dead space in the 177 L mixing section, but found 37 L more active volume than the 410 L physical volume of the burnout section. However, overall there was good agreement between the identified model and the physical geometry of the SCC.

A heat transfer model was used to apply the temperature parameters, developed from data obtained at stoichiometric conditions, to a range of conditions. A thermo-kinetic mechanism developed by Chiang (1995) was applied to the model. This mechanism had shortcomings, but it contained sufficient detail to provide predictions for many of the target analytes measured by the on-line GC. The model results did not agree well with the

experimental data. However, it followed many of the underlying trends revealed by the data. Sensitivity analysis of the parameters was used to further explore trends and recommend potential design improvements to reduce PIC formation.

A PaSR model was employed that used an interaction by exchange with the mean (IEM) as a turbulent moment closure to simulate finite time mixing at the microscale. Local conditions are relaxed toward the mean at a rate defined by the mixing frequency, ω. Comparisons between model and experiment for fuel-lean conditions explored a Toroidal Jet Stirred Combustor (TJSC) at near-blowout using the GRI 3.0 mechanism. An *n*-butane mechanism was used for fuel-rich condition simulations of experiments that measured the formation of aromatic hydrocarbons, specifically benzene in the TJSC. An alternate solution technique, developed in this work, approximated mean conditions and solved the deterministic model to refine the approximation and eventually converge on a solution. The approximation, direct integration, and convergence technique compared favorably with the Monte Carlo modeling calculations presented by Correa (1993 and 1995) for smaller mechanisms, but used, on average, less than $1/200^{th}$ of the CPU time. This new technique allowed use of considerably larger detailed mechanisms. Additionally, a generalized PaSR model was proposed to include the effects of non-ideal macromixing. The model takes a hybrid approach to the Zwietering Maximum Mixedness Model. The generalized model uses a mixing rule that relaxes the composition between individual fluid "points" with the mean of other "points" with the same life expectancy in the reactor.

The comparison of PaSR model predictions to the experimental data from a TJSC operating in both fuel-lean and fuel-rich conditions showed good agreement between model and data at a mixing frequency of 3162 Hz. Decreasing the mixing frequency had the effect of delaying combustion. It produced a greater range of local temperatures and local concentrations. Likewise, increasing the dilution of the feed also delayed combustion. This had the effect of increasing the bimodal nature of the temperature distribution, similar to the effect observed by Barat (1990) from Rayleigh scattering measurements.

7.2 Recommendations

The analysis of the SCC revealed that little mixing between the kiln gas stream and the flame occurs in the so-called mixing section. However, typical SCC designs lack a choke section, which forces additional mixing downstream. Designs without a choke section, are unlikely to meet requisite retention time - temperature requirements, even though averaged (assumed mixed) conditions are within regulatory standards. This analysis seem to suggest that some turbulence producing obstruction, like a choke or bluff body, is essential to ensue complete mixing.

The underlying trends mirrored by the reactor model suggested further SCC design modifications to improve performance. The trends found by parameter sensitivity analysis suggested that SCC performance could be improved by: maintaining a high burner flame temperature; reducing unburned hydrocarbons in the burner effluent by improving mixing and

increasing the recirculating volume; and achieve earlier heating of the kiln gas perhaps by using multiple tangential entries into the mixing section. Further investigation into this system should determine the effect of such improvements.

A better modeling effort is possible with several improvements. More data are required on the operating conditions of the SCC, especially reactor temperatures at multiple points during operation using suction pyrometry to maximize accuracy of measurements. Less extrapolation of heat transfer parameters would be required. The water quench system upstream of the on-line GC should be analyzed and modeled to determine specific effects on target analytes. Finally, additional validation of the thermo-kinetic mechanism will increase confidence in the SCC model. Validation of the mechanism should include data taken at combustion temperatures in research devices such as a TJSC or a laminar flat flame. The aromatic chemistry of the Chiang mechanism requires additional work to provide for additional potential reaction pathways especially through the propargyl radical.

Finally, additional work is needed to develop the generalized PaSR model. This model has the potential of applying detailed thermo-kinetic mechanisms to the complex mixing system in an SCC while considering the effects of non-ideal macro and micromixing. However, caution is warranted when applying a model that reduces complex geometric effects to one or two parameters. These limitations can be overcome. Applications to simple geometries in conditions with high turbulent kinetic energies and sufficiently small integral time scales should be tested first. Computational Fluid Dynamics (CFD) analysis of complex geometries will be necessary to understand the details of the turbulent mixing before applying the model.

APPENDIX A

TERMS AND SYMBOLS

A.1 Symbols

Nomenclature

a_n	inverse of mean residence time for reactor n
b	probe quench model parameter
c_p	mass weighted average specific heat
$\mathrm{Da_T}$	turbulent Damköhler number
$E(t)$	RTD
$F(t)$	cumulative distribution function
f_c	Nyquist critical frequency
$g(t)$	system transfer function
$G(s)$	Laplace transform of the system transfer function
G_x	axial flux
G_φ	angular flux
$\mathbf{h}$	weighting function vector
$\hat{\mathbf{h}}$	estimate of the weighting function vector
$h(k\Delta t)$	system weighting function
h_k	specific enthalpy of species k
$j(\alpha,\lambda)$	age, life expectancy, joint probability distribution
$LMTD$	log mean temperature difference
l	angular momentum
$\dot{m}$	mass flow rate
n	number of equal ideal reactors
$\dot{n}$	molar flow rate
P	pressure
p	an upper index value
Q	heat flow rate
s^3	skewness coefficient
S	swirl number
S	reaction rate vector
$\bar{S}_{ij}$	first order sensitivity coefficient
$\hat{S}_{ij}$	first order sensitivity coefficient, mole basis
s	Laplace-space independent variable
t	independent time variable
Δt	discrete time step
T	temperature

$\mathbf{U}$	input matrix
$u(k\Delta t)$	discrete input function
$U(k\Delta t)$	discrete step input function
$u(t)$	continuous input function
$U(t)$	continuous step input function
$\mathbf{U}$	transposed input matrix
$\mathbf{v}$	system noise vector
V	reactor volume
$v(t)$	system noise
W_k	molar mass of species k
$\dot{w}_k$	molar production rate of species k
X_k	mole fraction of species k
$\mathbf{y}$	system output vector
$y(k\Delta t)$	discrete system impulse response
$Y(k\Delta t)$	discrete system step response
$\hat{y}(k\Delta t)$	estimated (model) system impulse response
$\hat{Y}(k\Delta t)$	estimated (model) system step response
$y(t)$	continuous system impulse response
$Y(t)$	continuous system step response
Y_k	mass fraction of the k^{th} species
	expected value

Greek Symbols

$\boldsymbol{\alpha}$	parameter vector
α_i	parameter i
α	point age in the reactor
$\delta(t - \tau)$	impulse function with offset τ
$\boldsymbol{\Gamma}$	input/output cross correlation function vector
ζ	specific composition field
θ_n	bypass fraction at location n
λ	point life expectancy: time remaining the reactor
μ	mean
ξ	composition scalar field
ρ	density
$\sigma(\alpha)$	age distribution
σ^2	variance
$\boldsymbol{\sigma}^2$	vector of local variances
τ	dummy time variable
τ	mean residence time
Φ	fuel/air equivalence ratio
$\boldsymbol{\Phi}$	input autocorrelation function matrix
$\boldsymbol{\Psi}$	covariance matrix

Ψ_{ij}	covariance between i^{th} and j^{th} parameters
$\psi(\lambda)$	point life expectancy distribution
ω	mixing frequency
ω	angular velocity

Subscripts and Indicies

i	an index
j	an index
k	step index, $k = 0, 1, 2, 3, \ldots$
k	the index for a specific chemical species
n	an index

A.2 Terms

CDF	Cumulative Distribution Function
CFD	Computational Fluid Dynamics
CHEMKIN-II	(Kee *et al.*, 1989) Software Package for the formulation, solution, and interpretation of gas-phase chemical kinetics
CPU	Central Processing Unit
DNS	Direct Numeric Simulation, direct integration of Navier-Stokes and associated continuity equations
DRE	Destruction and Removal Efficiency, percentage of POHC converted in the incinerator
DVODE	(Brown, Hindmarsh, and Byrne, 1989) Double precision Variable coefficient ODE solver
ECD	Electron Capture Detector, a post column detector on a gas chromatograph
EPA	U.S. Environmental Protection Agency
FID	Flame Ionization Detector, a post column detector on a gas chromatograph
GRI	Gas Research Institute
IEM	Interaction by Exchange with the Mean, turbulent closure model
LS	Least Squares
ODE	Ordinary Different Equation
PAH	Polyaromatic Hydrocarbon
PaSR	Partially Stirred Reactor, mixing model using IEM closure
PCDD	Pentachlorodibenzodioxin
PCDF	Pentachlorodibenzofuran
PDF	Probability Distribution Function
PFR	Plug Flow Reactor

PIC	Product of Incomplete Combustion
POHC	Principle Organic Hazardous Component
PREMIX	(Kee *et al.*, 1993) A FORTRAN program for modeling steady laminar one-dimensional premixed flames
PSR	Perfectly Stirred Reactor
PSR	(Glarborg, *et al,* 1992) A Fortran program for modeling well-stirred reactors
QRRK	Quantum Rice-Ramperger-Kassel, methodology for determining kinetic parameters for unimolecular and bimolecular pressure dependent reactions
RKIS	Rotary Kiln Incinerator Simulator, a pilot scale rotary kiln and SCC located at EPA, Research Triangle Park, NC.
RTD	Residence Time Distribution
SCC	Secondary Combustion Chamber
SISO	Single Input Single Output
TSJC	Toroidal Jet Stirred Combustor, a bench scale premixed gas combustor that generates mixing energy from 32 jets
VOC	Volatile Organic Compound
VOST	Volatile Organic Sample Train

APPENDIX B

EXPERIMENTAL DATA

Listed below are experimental data from the Secondary Combustion Chamber (SCC) of the Rotary Kiln Simulator (RKIS) provided by the Air Pollution Technology Branch, of the National Risk Management Research Laboratory, Research Triangle Park, NC.

Table B.1 Temperature Measurements in the Mixing Section

X-position (in.)	Y-position (in.)	Temperature (°F)	X-position (in.)	Y-position (in.)	Temperature (°F)
0	1	2215	0	-1	1650
0	2	2286	0	-2	1355
0	3	2266	0	-3	1311
0	4	1865	0	-4	1310
0	5	1331	0	-5	1307
0	6	1316	0	-6	1310
0	7	1311	0	-7	1313
0	8	1312	0	-8	1317
0	9	1311	0	-9	1316
0	10	1310	0	-10	1312
0	11	1318	0	-11	1310
0	12	1331	0	-12	1307
1	0	1983	-1	0	1999
2	0	2309	-2	0	1378
3	0	2274	-3	0	1324
4	0	1891	-4	0	1322
5	0	1375	-5	0	1317
6	0	1319	-6	0	1317
7	0	1317	-7	0	1315
8	0	1314	-8	0	1321
9	0	1311	-9	0	1315
10	0	1317	-10	0	1310
11	0	1322	-11	0	1297
12	0	1322	-12	0	1216

Table B.2 RKIS PIC Concentrations (mg/m^3)

Run	Dopant	Aerosol Pressure	AB SR	Port	$CHCl_3$	CCl_4	C_6H_6	C_2HCl_3	C_2Cl_4	C_6H_5Cl
0622RG4	CCl4	high	0.85	4	0.016	0.012	14.735	0.273	0.287	2.582
0616RP5	CCl4	low	0.85	5	0.005	0.349	10.777	0.049	0.009	2.264
0615RG5	CCl4	high	0.85	5	0.006	0.585	11.025	0.078	0.002	1.059
0621SP3	CCl4	low	1	3	0.187	0.136	0.036	0.099	0.026	0.020
0621SG3	CCl4	high	1	3	0.082	0.149	0.031	0.098	0.027	0.082
0614SG5	CCl4	high	1	5	0.012	0.268	0.009	0.044	0.033	0.061
0615SP5	CCl4	low	1	5	0.007	0.311	0.009	0.080	0.003	0.054
0622LG4	CCl4	high	1.1	4	0.115	0.078	1.165	0.124	0.159	0.096
0620LG5	CCl4	high	1.1	5	0.004	0.111	0.018	0.059	0.001	0.006
0616LG5	CCl4	high	1.1	5	0.004	0.221	0.046	0.068	0.004	0.255
0620LP5	CCl4	low	1.1	5	0.006	0.088	0.026	0.048	0.001	0.012
0623MRG4	CH2Cl2	high	0.85	4	0.032	0.005	9.946	0.260	0.221	16.571
0623MRG5	CH2Cl2	high	0.85	5	0.003	0.002	8.806	0.122	0.021	2.165
0621MSG3	CH2Cl2	high	1	3	0.207	0.020	5.640	0.127	0.071	19.917
0623MLG4	CH2Cl2	high	1.1	4	0.066	0.040	0.306	0.097	0.060	0.018

Table B.2 RKIS PIC Concentrations, Continued (mg/m^3)

Run	Dopant	At. Pressure	AB SR	Port	$CHCl_3$	CCl_4	C_6H_6	C_2HCl_3	C_2Cl_4	C_6H_5Cl
0623MLG5	CH2Cl2	high	1.1	5	0.063	0.005	0.609	0.091	0.010	0.289
0622RB4	n/a	blank	0.85	4	0.000	0.001	9.376	0.029	0.001	0.017
0615RB5	n/a	blank	0.85	5	0.000	0.135	12.340	0.075	0.002	0.019
0620RB5	n/a	blank	0.85	5	0.000	0.001	10.482	0.037	0.001	0.000
0621SB3	n/a	blank	1	3	0.002	0.002	0.062	0.048	0.002	0.037
0621SB3	n/a	blank	1	3	0.006	0.001	0.203	0.057	0.002	0.760
0620SB5	n/a	blank	1	5	0.000	0.001	0.263	0.054	0.001	0.000
0615SB5	n/a	blank	1	5	0.000	0.000	0.007	0.052	0.002	0.059
0614SB5	n/a	blank	1	5	0.000	0.000	0.005	0.045	0.000	0.000
0622LB4	n/a	blank	1.1	4	0.003	0.001	0.306	0.013	0.006	0.009
0620LB5	n/a	blank	1.1	5	0.000	0.002	0.006	0.080	0.002	0.013
0616LB5	n/a	blank	1.1	5	0.000	0.022	0.567	0.075	0.002	0.102

Table B.3 RKIS Continuous Monitor Readings

Run	Kiln O2 (%)	Kiln CO2 (%)	Kiln CO (ppm)	Stack O2 (%)	Stack CO2 (%)	Stack CO (ppm)	Kiln T (°F)	SCC Mix T (°F)	SCC Mid T (°F)
0614SG5	5.4	9.3	391	1.9	11.7	48	1884	1306	1735
0614SB5	5.4	9.3	391	1.9	11.7	48	1884	1306	1735
0615SP5	N/A	N/A	N/A	2.2	12.5	97	1897	1359	1806
0615SB5	N/A	N/A	N/A	2.2	12.5	97	1897	1359	1806
0615RG5	N/A	N/A	N/A	0.5	13.1	10790	1900	1361	1799
0615RB5	N/A	N/A	N/A	0.4	12.4	9550	1895	1363	1775
0616RP5	0.0	12.5	2276	0.0	13.1	11830	1908	1339	1802
0616LB5	1.6	11.7	14	2.9	11.1	20	1922	1364	1797
0616LG5	0.6	12.4	436	1.6	12.2	22	1937	1368	1799
0620LB5	0.6	12.0	552	2.0	11.4	26	1811	1307	1748
0620RB5	0.0	11.9	5850	0.3	11.8	11130	1839	1320	1704
0620SB5	0.0	12.2	2310	0.5	12.3	638	1855	1332	1773
0620LP5	0.4	12.2	719	2.3	11.7	28	1882	1351	1778
0620LG5	0.8	11.9	735	2.7	11.3	29	1895	1361	1784

Table B.2 RKIS Continuous Monitor Readings

Run	Kiln O2 (%)	Kiln CO2 (%)	Kiln CO (ppm)	Stack O2 (%)	Stack CO2 (%)	Stack CO (ppm)	Kiln T (°F)	SCC Mix T (°F)	SCC Mid T (°F)
0621SG3	0.1	12.1	1858	0.5	12.6	925	1874	1321	1781
0621SB3	0.0	12.1	1677	0.5	12.2	1790	1901	1337	1801
0621SP3	0.0	12.0	2220	0.2	12.8	1850	1904	1343	1794
0621MSG3	0.0	12.0	2114	0.2	12.4	5350	1918	1347	1789
0621SB3-2	N/A	N/A	1165	0.3	12.2	1023	1892	1357	1809
0622LB4	0.5	12.1	796	2.2	11.8	23	1821	1268	1743
0622RB4	0.0	11.8	7400	0.3	12.1	13790	1856	1280	1665
0622RG4	0.0	11.9	8940	0.4	12.4	16230	1879	1295	1670
0622LG4	0.3	12.3	1100	2.0	12.4	29	1930	1339	1778
0623MLG4	1.0	11.6	44	1.4	12.1	151	1906	1349	1786
0623MRG4	0.0	11.8	4730	0.0	12.0	13160	1917	1341	1742
0623MLG5	0.9	11.6	48	1.7	12.0	41	1931	1374	1790

APPENDIX C

ADDITIONAL GRAPHICS, EQUATIONS, AND CALCULATIONS

C.1 Graphics

This section contains additional graphs that support various discussions in Chapters 3 through 5.

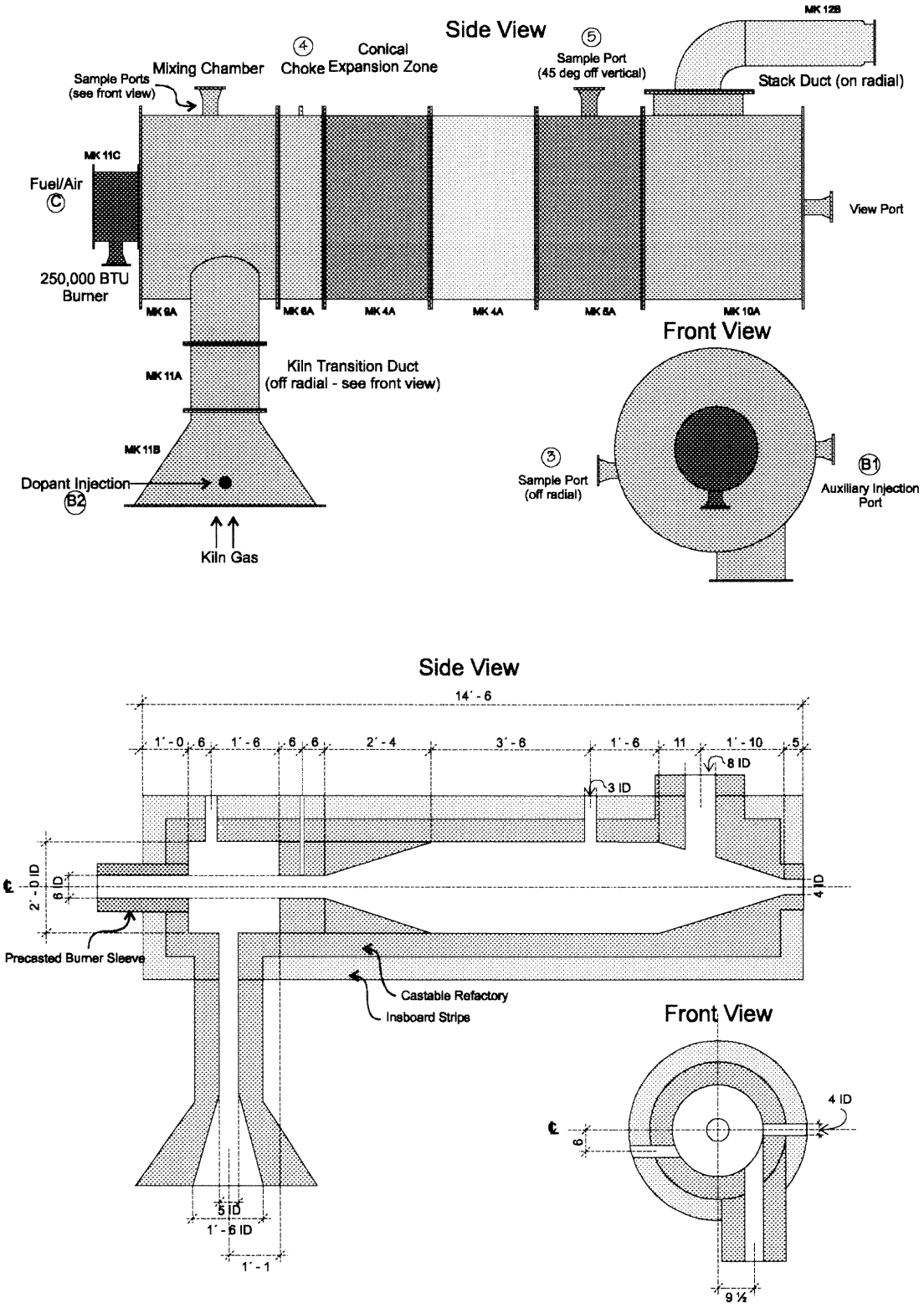


Figure C.3.1 Scale Diagram of the Secondary Combustion Chamber

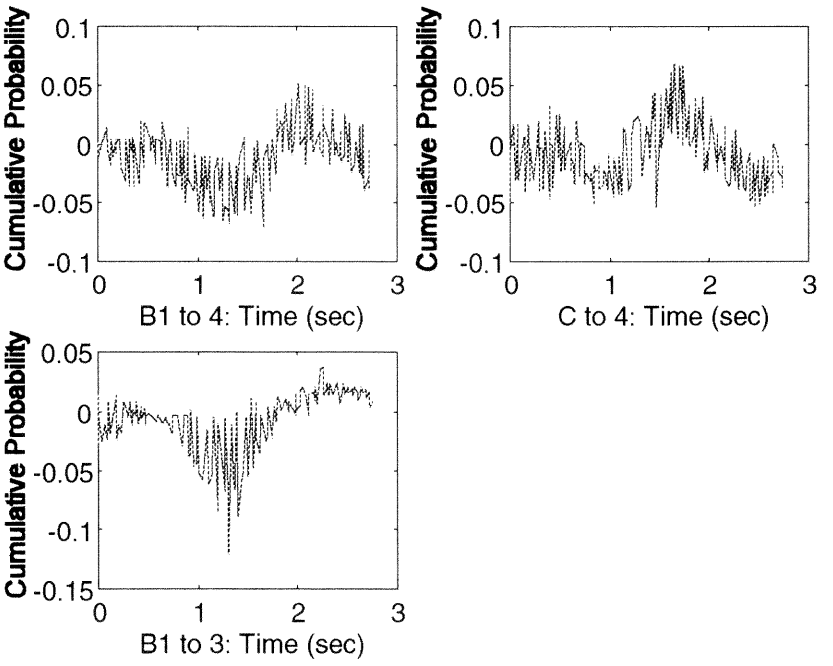


Figure C.4.1a Mixing Section Model Residuals (y-axis shows residuals on a scale from 0 to 1)

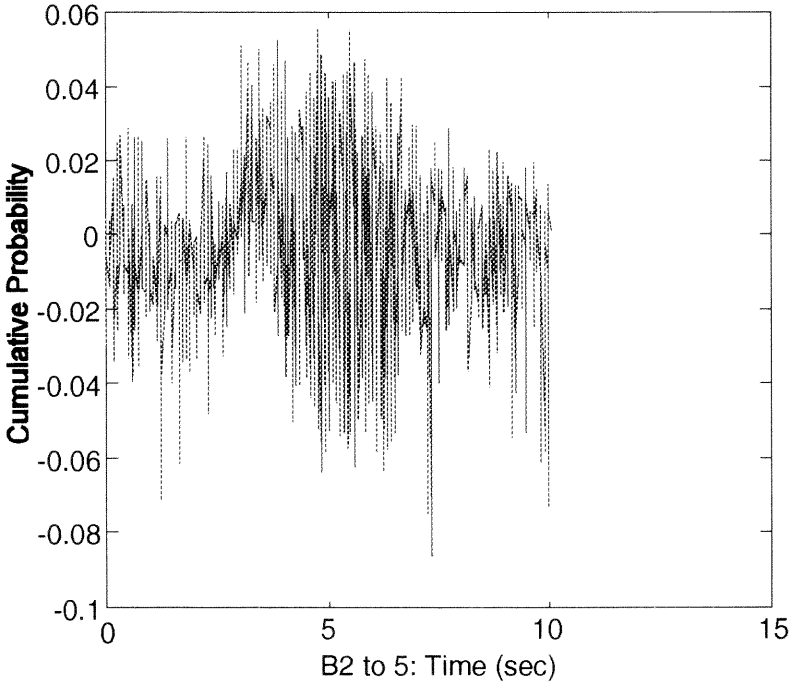


Figure C.4.1b Burnout Section Residuals (y-axis shows residuals on a scale from 0 to 1)

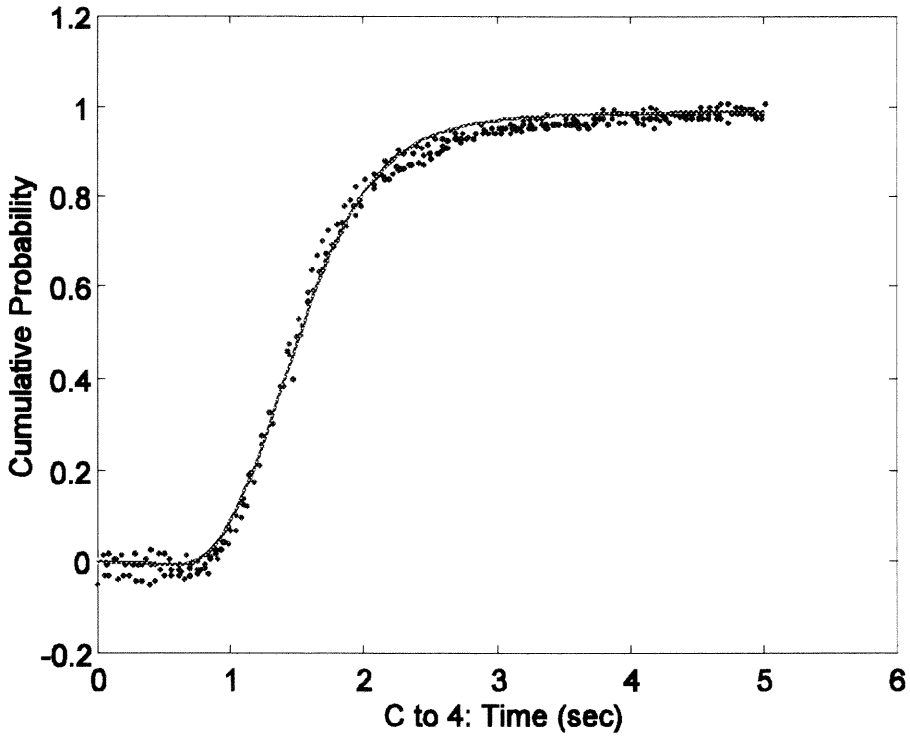


Figure C.4.2a Comparison of Model 1 with Normalized Data: C to 4

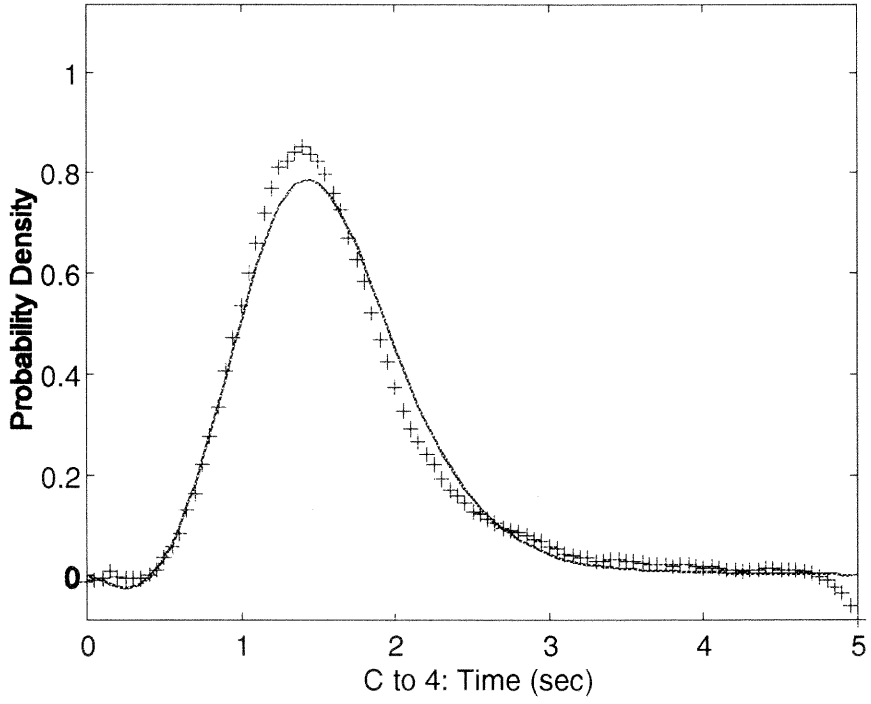


Figure C.4.2b Comparison of Model 1 with System Response: C to 4

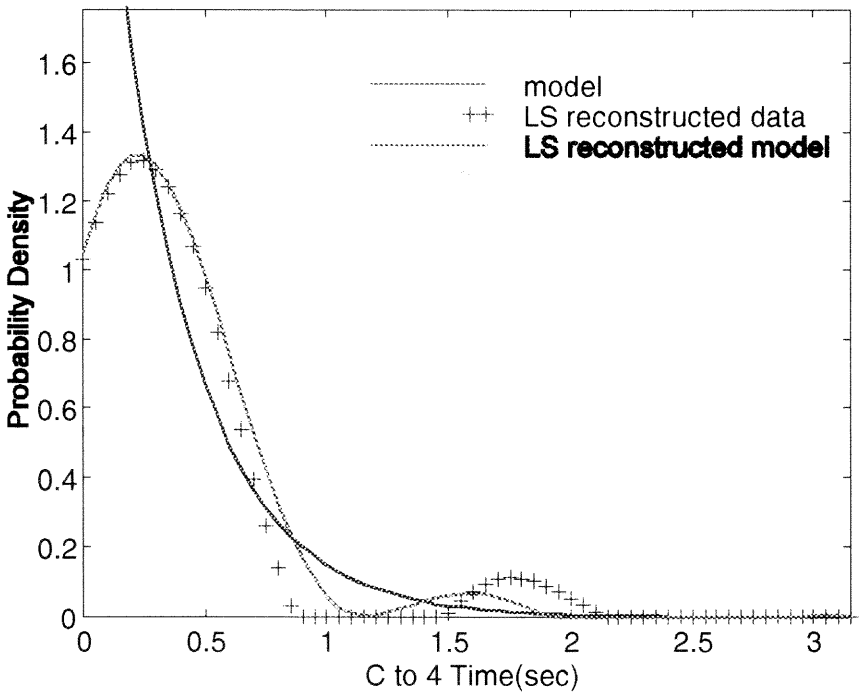


Figure C.4.2c Comparison of Model with LS Reconstructed RTD: C to 4

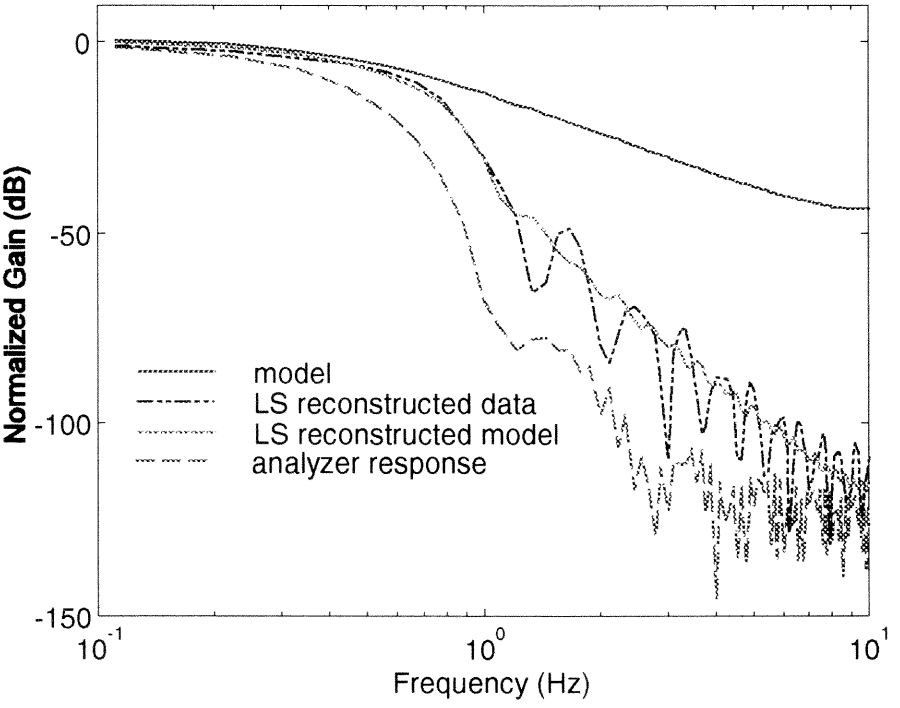


Figure C.4.2d Power Spectrum Comparison of Model and LS Reconstructed RTD: C to 4

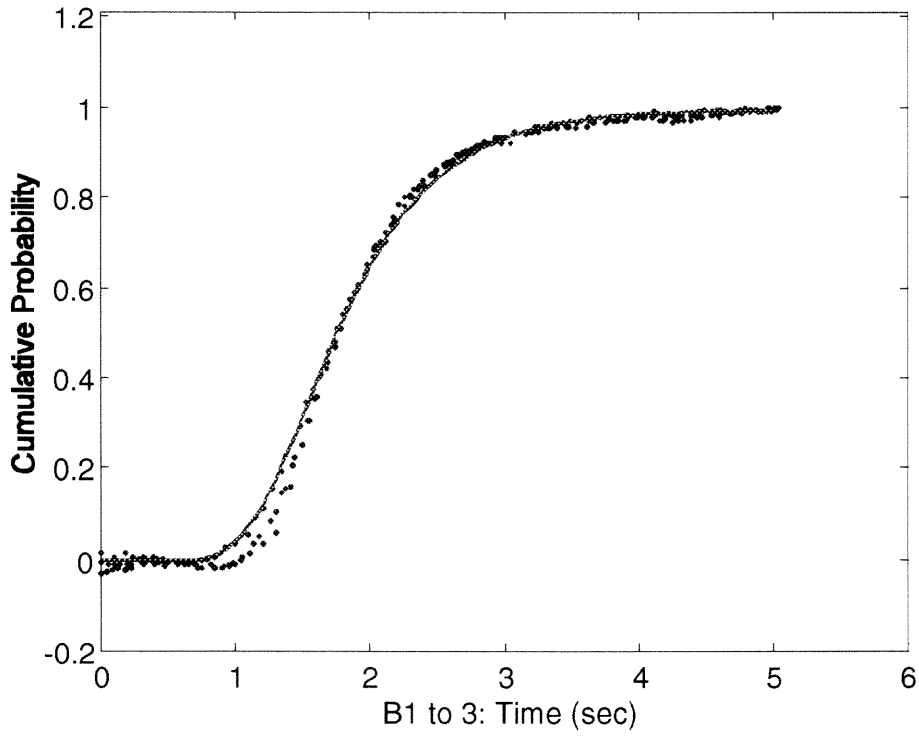


Figure C.4.3a Comparison of Model with Normalized Data: B1 to 3

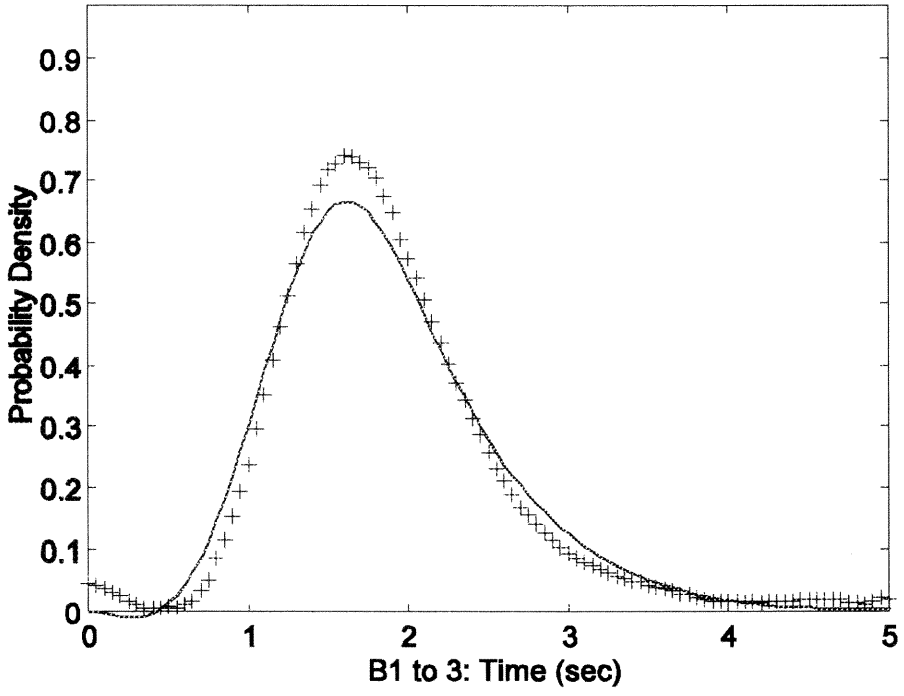


Figure C.4.3b Comparison of Model with System Response: B1 to 3

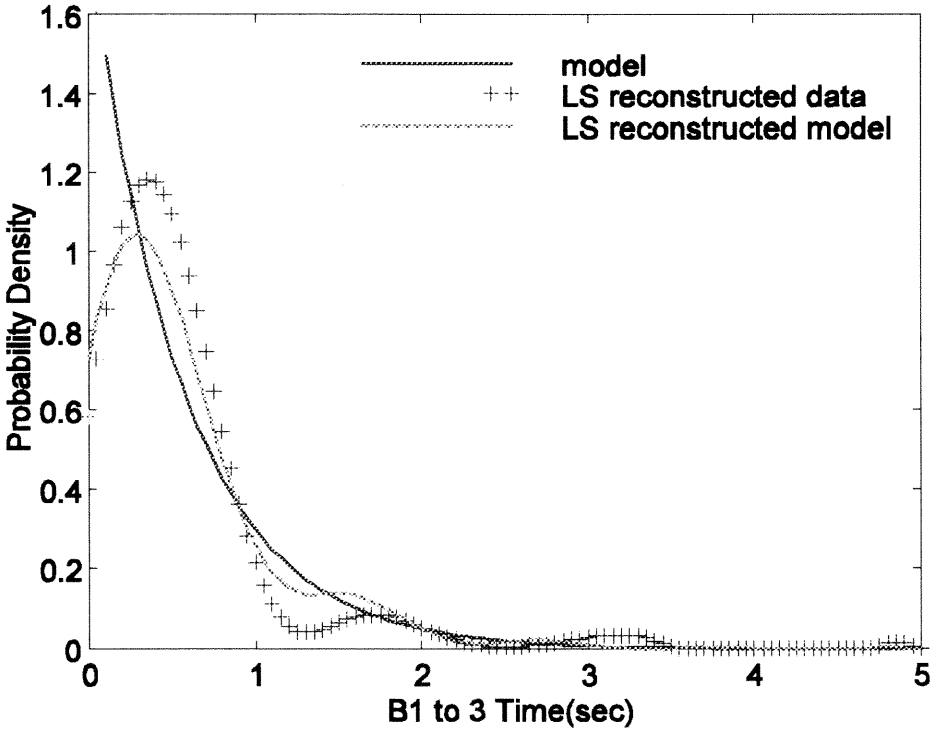


Figure C.4.3c Comparison of Model with LS Reconstructed RTD: B1 to 3

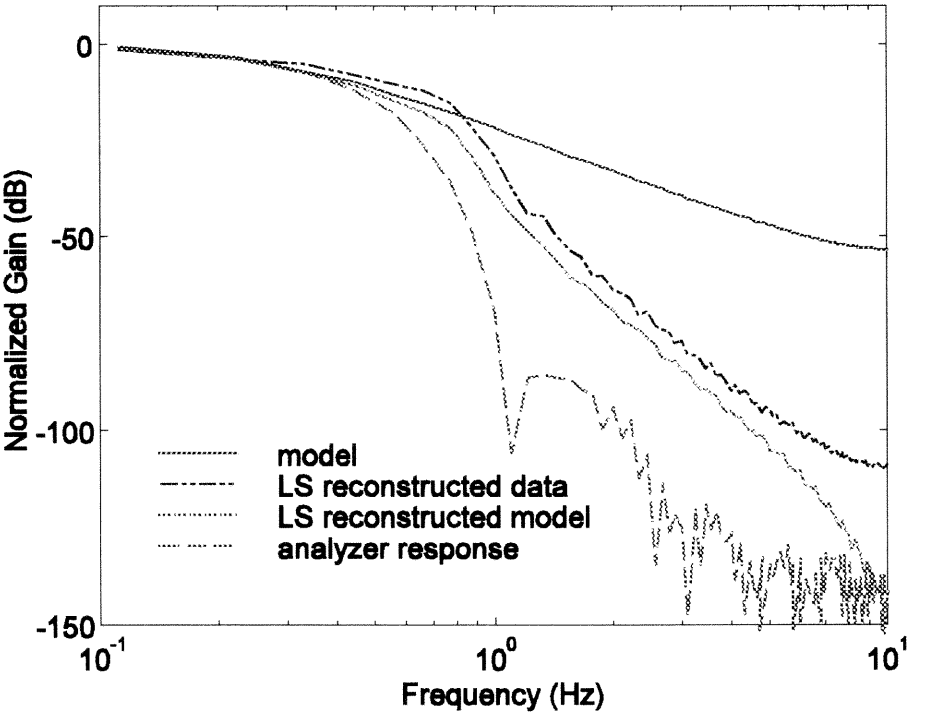


Figure C.4.3d Power Spectrum Comparison of Model and LS Reconstructed RTD: B1 to 3

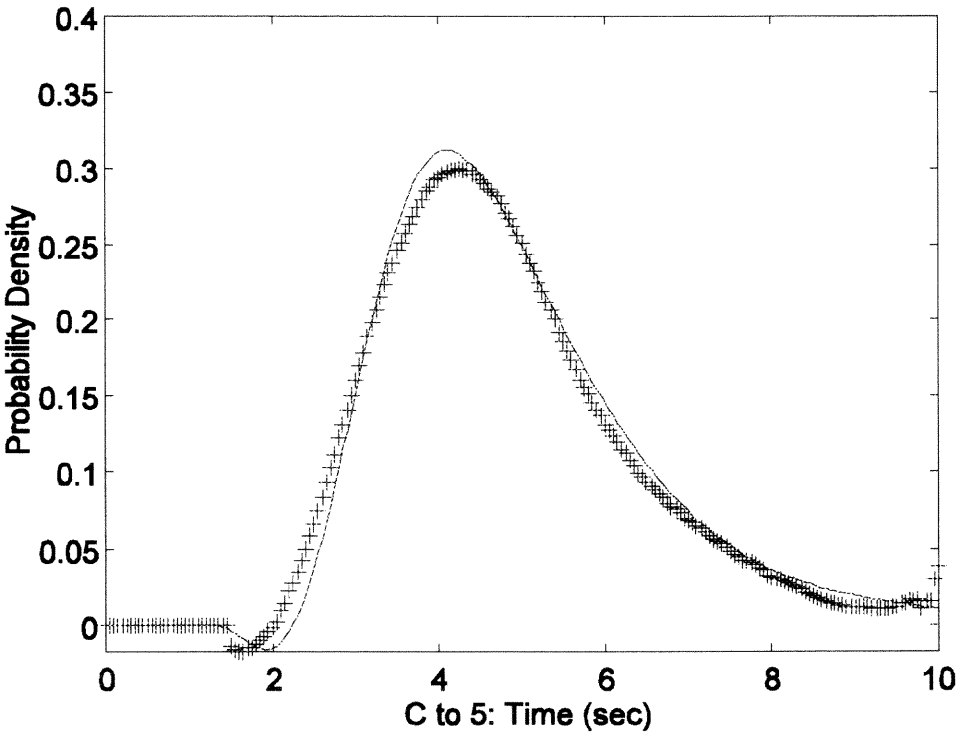


Figure C.4.4a Comparison of Model with System Response: C to 5

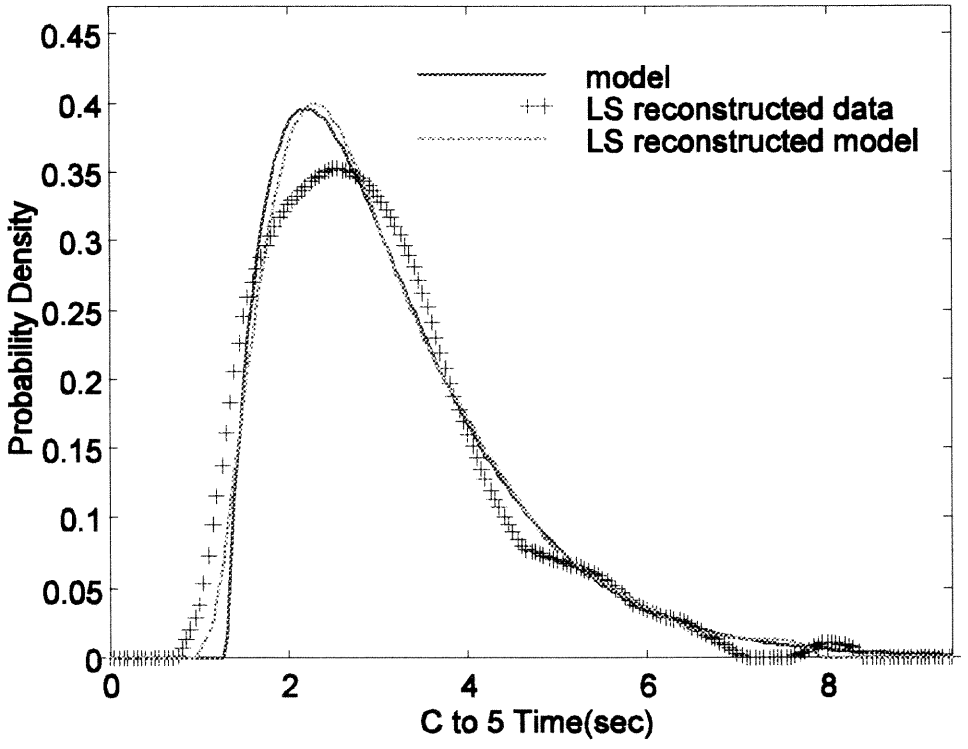


Figure C.4.4b Comparison of Model with LS Reconstructed RTD: 4 to 5 w/C input

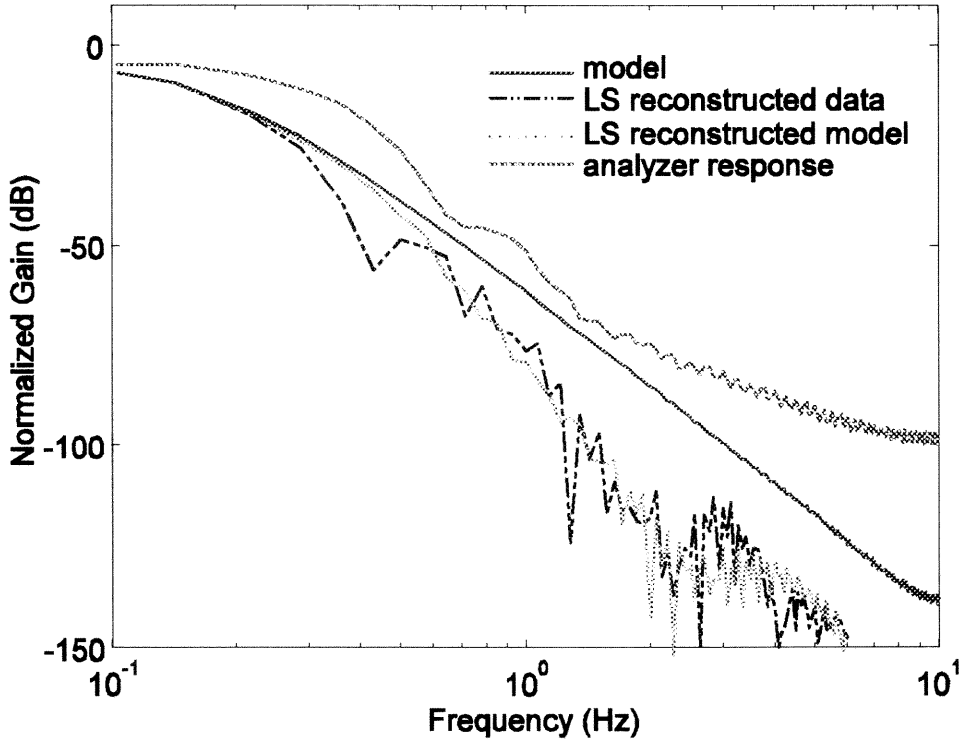


Figure C.4.4c Power Spectrum Comparison of Model and LS Reconstructed RTD: C to 5

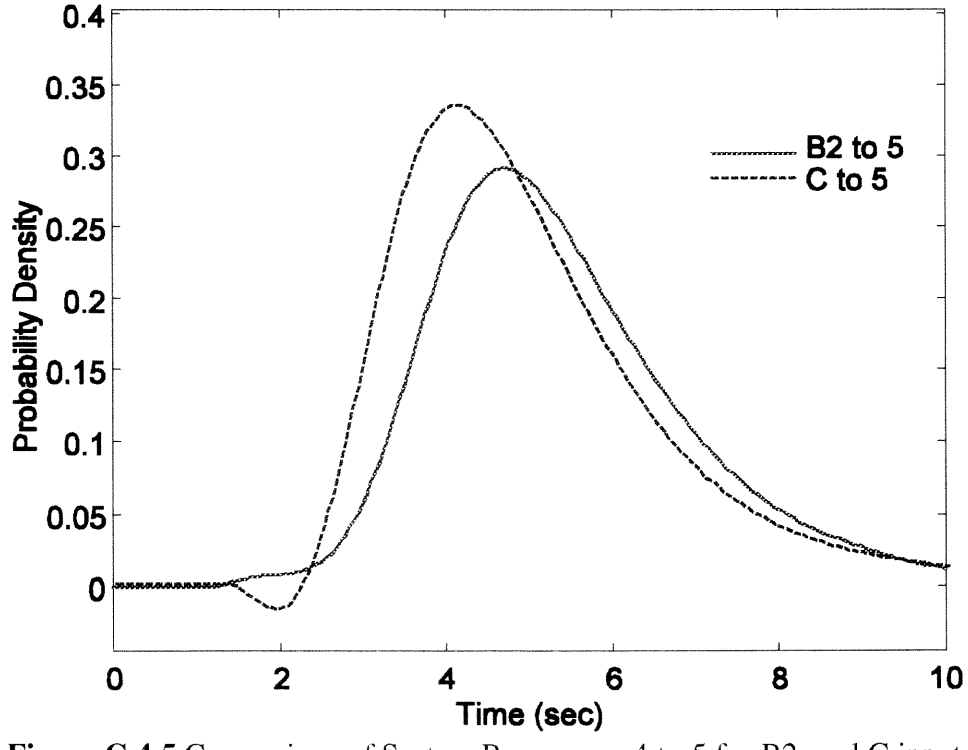
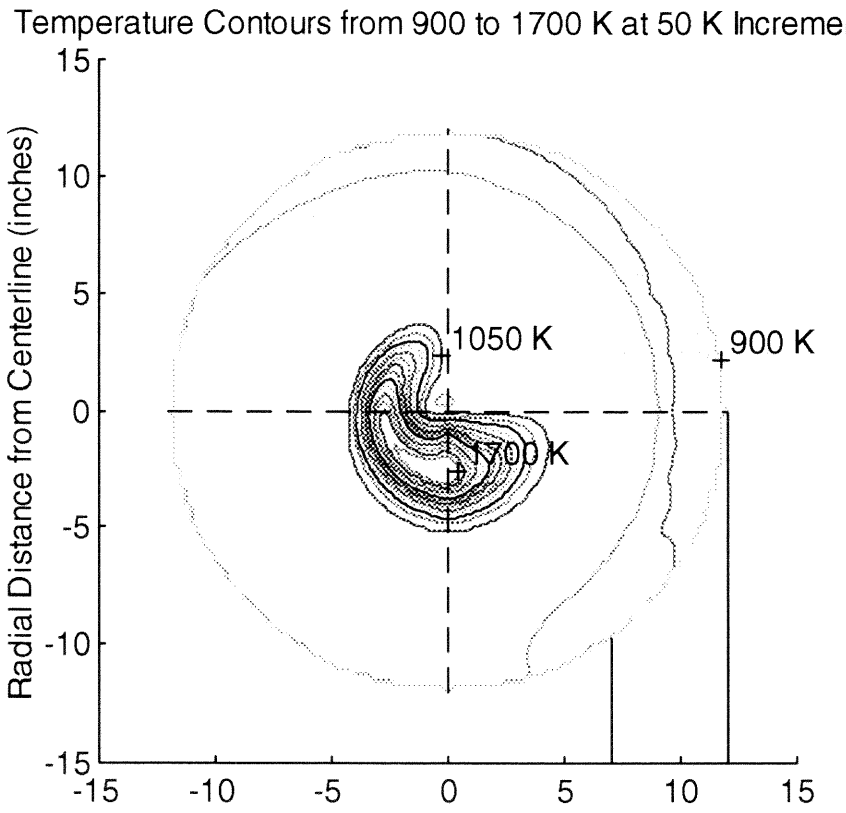
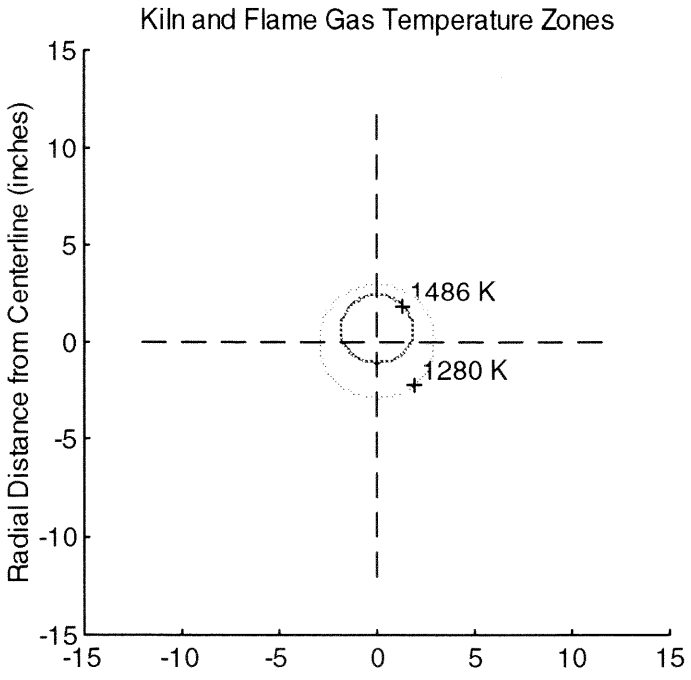


Figure C.4.5 Comparison of System Responses: 4 to 5 for B2, and C input



Average T	1038.8 K
Density Weighted Average T	1029.5 K
Enthalpy Density Weighted Average	1030.8 K
Flame Gas Density Weighted Average T	1339.1 K
Flame Gas Enthalpy Density Weighted Average	1343.5 K
Kiln Gas Density Weighted Average T	1008.8 K
Kiln Gas Enthalpy Density Weighted Average	1009.0 K
Zone Division Temperature	1101.4 K
Flame Area Fraction	0.1016

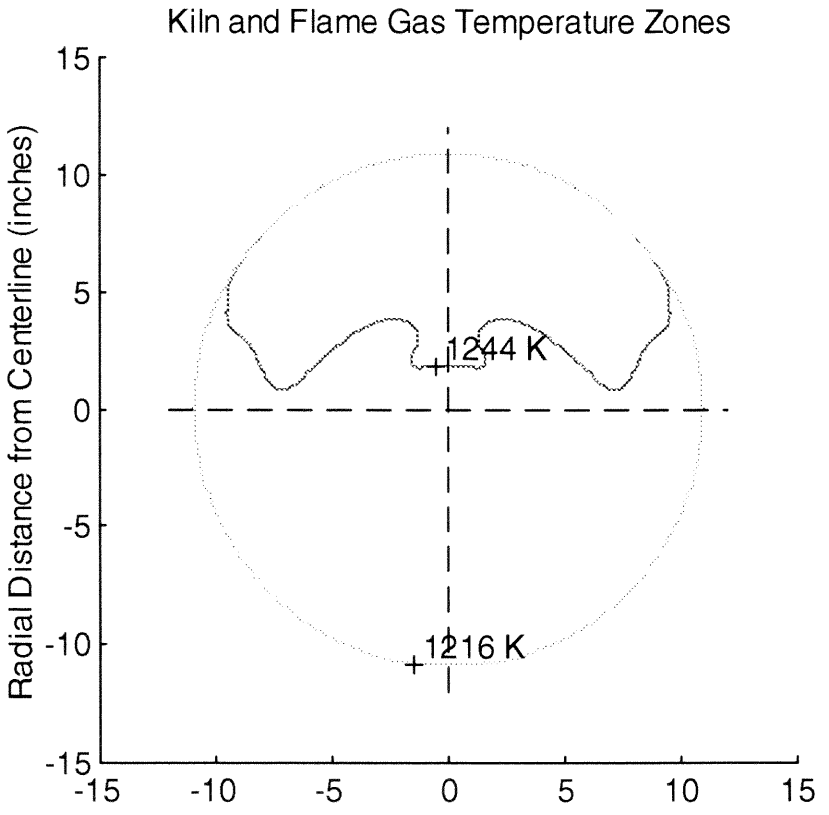
Figure C.4.6 Mixing Chamber Cross Section Temperature Analysis



Mass Flows:

Kiln Gas	108 kg/hr
Flame Gas	56.8 kg/hr
Average T	1453.9 K
Density Weighted Average T	1449.9 K
Enthalpy Density Weighted Average	1450.6 K
Flame Gas Density Weighted Average T	1537.8 K
Flame Gas Enthalpy Density Weighted Average	1537.9 K
Kiln Gas Density Weighted Average T	1403.7 K
Kiln Gas Enthalpy Density Weighted Average	1403.9 K
Zone Division Temperature	1486.3 K

Figure C.4.7 Choke Cross Section Temperature Analysis



Mass Flows:

Kiln Gas	108 kg/hr
Flame Gas	56.8 kg/hr
Average T	1240.6 K
Density Weighted Average T	1240.6 K
Enthalpy Density Weighted Average	1240.6 K
Flame Gas Density Weighted Average T	1248.6 K
Flame Gas Enthalpy Density Weighted Average	1248.6 K
Kiln Gas Density Weighted Average T	1236.4 K
Kiln Gas Enthalpy Density Weighted Average	1236.4 K
Zone Division Temperature	1243.8 K

Figure C.4.8 Burnout Section Cross Sectional Analysis

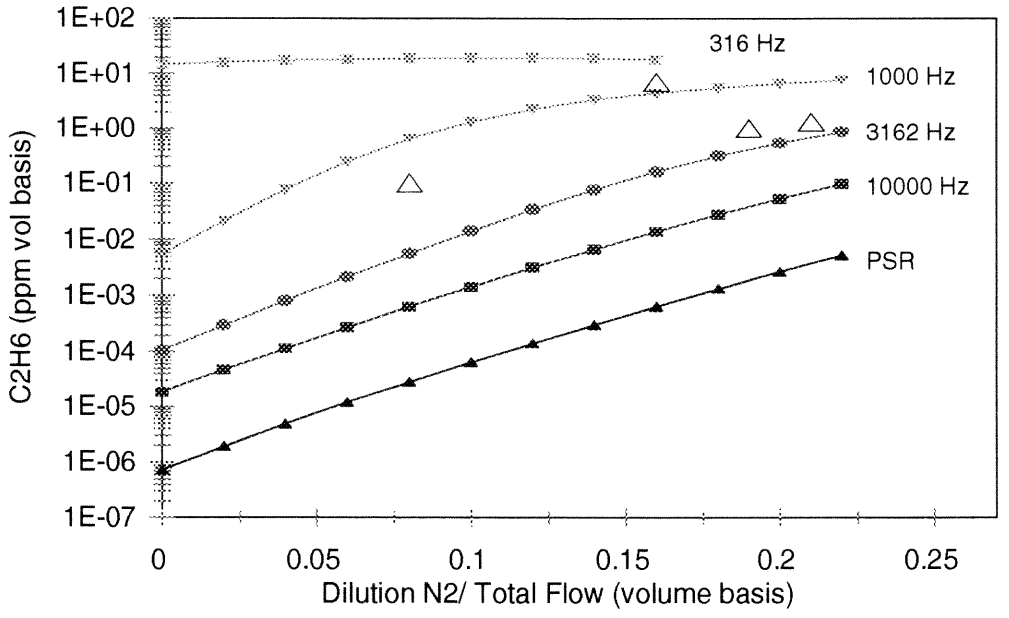


Figure C.5.1 C_2H_6 Concentration Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models with Probe Quench Model

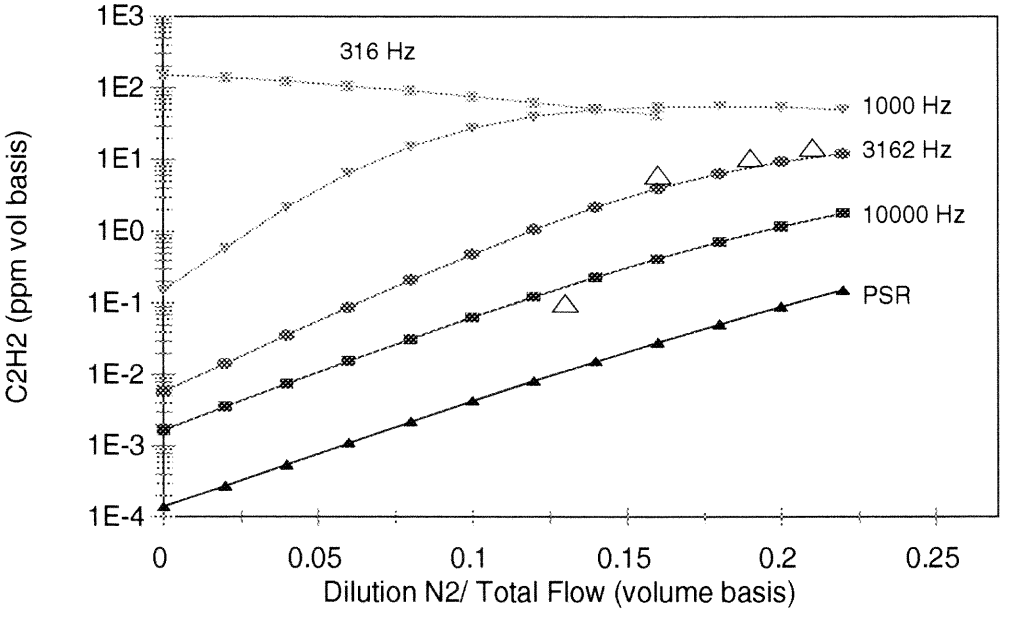


Figure C.5.2 C_2H_2 Concentration Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models with Probe Quench Model

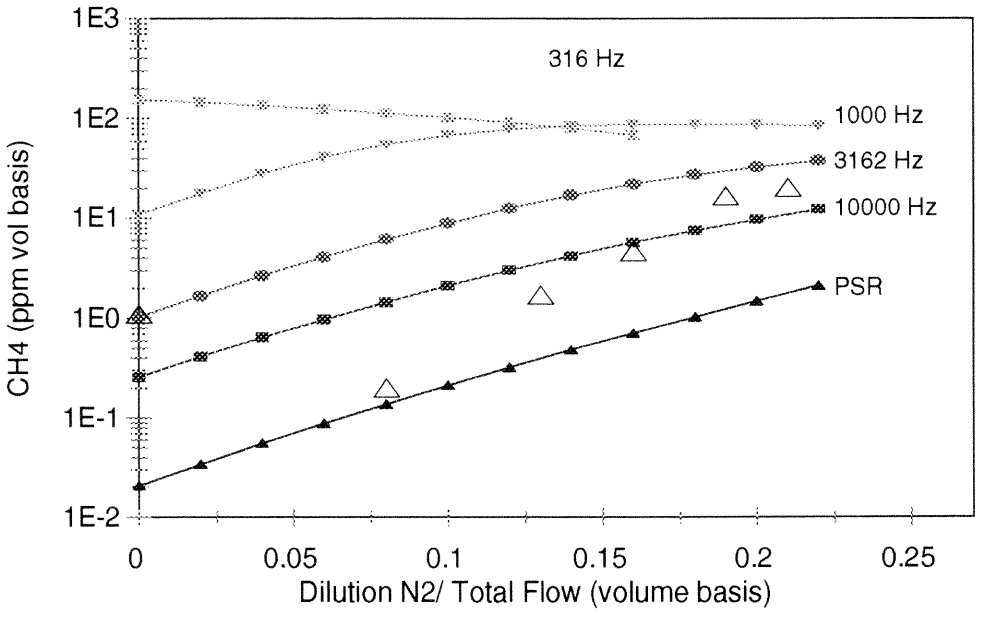


Figure C.5.3 CH_4 Concentration Dependence on Dilution Ratio, Comparison of Experimental Data (△) with PSR and PaSR Models with Probe Quench Model

C.2 Calculations

C.2.1 Calculation of Swirl Number

Swirl Number is calculated based on tangential flow. Consider an isothermal system with tangential mixing having an angular velocity of ω and a axial velocity of u. The radius is R_0. Swirl number is defined as (Beér and Chigier, 1972):

$$S = \frac{G_\varphi}{G_x R_0} \tag{C.1}$$

where angular momentum is:

$$G_\varphi = \int_0^R (\omega r)\rho u 2\pi r dr \tag{C.2}$$

and axial flux is:

$$G_x = \int_0^R u\rho u 2\pi r dr \tag{C.3}$$

For the mixing section of the SCC shown in figure C.3.1, the axial flux may be approximated by finding the average axial velocity, $\bar{u}$. Then equations C.2 reduces to:

$$G_x = \underbrace{(\bar{u}\rho)}_{\text{momentum}} \times \underbrace{(\bar{u}\pi R_o^2)}_{\text{flux}} = \rho\pi R_0^2 \bar{u}^2 \tag{C.4}$$

The average axial velocity is calculated from the axial mass flux, which is the sum of the mass flux entering the axially in the burner, $\dot{m}_a$, and the mass flux entering tangentially, $\dot{m}_t$:

$$\bar{u} = \frac{(\dot{m}_a + \dot{m}_t)}{\rho\pi R_0^2} \tag{C.5}$$

Substituting equation C.5 into C.4 gives:

$$G_x = \rho\pi R_0 \left[\frac{(\dot{m}_a + \dot{m}_t)}{(\rho\pi R_0^2)}\right]^2 = \frac{(\dot{m}_a + \dot{m}_t)^2}{\rho\pi R_0^2} \tag{C.6}$$

Assume a uniform axial velocity in the mixing section:

$$G_\varphi = l\bar{u}\pi R_0^2 \tag{C.7}$$

where l is the angular momentum.. Following an example provided by Hallett (1986) of swirl number calculations in a tangentially stirred reactor, the angular momentum may be calculated from the location, angle and velocity of the tangentially entering fluid. For the SCC mixing section shown in figure C.1, the fluid enters in a kiln transition duct with radius, R_t. So the angular momentum is given by:

$$l = \omega\rho(R_0 - R_t)\frac{\dot{m}_t}{\dot{m}_a + \dot{m}_t} \tag{C.8}$$

The angular velocity at the point of entry is the velocity in the kiln transition duct, which is:

$$\omega = \frac{\dot{m}_t}{\rho\pi R_t^2} \tag{C.9}$$

Substituting equations C.8 and C.9 into C.7 gives:

$$G_{\varphi} = \frac{\dot{m}_t^2 (R_0 - R_t)}{\rho \pi R_t^2} \tag{C.10}$$

Ignoring friction losses, both angular and axial flux is conserved (Beér and Chigier, 1972). So as the flow enters the choke both G_x and G_φ remain the same, but the radius reduces to R_c. So the swirl number at the choke, S_{choke}, is given by:

$$S_{choke} = \frac{G_{\varphi}}{G_x R_c} = \left[\frac{R_0 \dot{m}_t}{R_t(\dot{m}_a + \dot{m}_t)}\right]^2 \frac{(R_0 - R_t)}{R_c} \tag{C.11}$$

At stoichiometric conditions, the tangential mass flow is:

$$\dot{m}_t = \left[\left(3566 \frac{\text{mol air}}{\text{hr}}\right)\left(28.84 \frac{\text{g air}}{\text{mol air}}\right) + \left(311 \frac{\text{mol gas}}{\text{hr}}\right)\left(16.58 \frac{\text{g gas}}{\text{mol gas}}\right)\right]\left(3600 \frac{\text{s}}{\text{hr}}\right)$$

$$= 30.3 \frac{\text{g}}{\text{s}}$$

Likewise, the axial mass flow is

$$\dot{m}_a = \left[\left(1853 \frac{\text{mol air}}{\text{hr}}\right)\left(28.84 \frac{\text{g air}}{\text{mol air}}\right) + \left(201 \frac{\text{mol gas}}{\text{hr}}\right)\left(16.58 \frac{\text{g gas}}{\text{mol gas}}\right)\right]\left(3600 \frac{\text{s}}{\text{hr}}\right)$$

$$= 15.7 \frac{\text{g}}{\text{s}}$$

which gives a swirl number at the choke calculates as (units cancel):

$$S_{choke} = \left[\frac{30.3}{(30.3+15.7)}\right]^2\left(\frac{12}{2.5}\right)^2\left(\frac{9.5}{3}\right)$$

$$= 31.7$$

This neglects friction losses, but the value is in the typical range of cyclone combustors. The increase in temperature from mixing chamber to the choke can be accounted for by a simple formula provided by Syred and Beér (1974):

$$S = S_{isothermal}\frac{T_{inlet}}{T_{outlet}} = 31.7\left(\frac{1039\ \text{K}}{1458\ \text{K}}\right) \qquad \text{(C.12)}$$

$$= 22.6$$

The large swirl number would logically lead to a large recirculating zone in the burnout section. The divergent cone at the choke exit enlarges the recirculation zone, increases recirculation mass flow substantially, and reduces the pressure loss coefficient (Syred and Beér, 1974).

C.2.2 Model Moments

Chapter 3 presented three archetype ideal reactor models and the associated model moments: mean, variance, and skewness coefficient. These three models are:

Case 1. PSR 1 followed by PFR *d*

Case 2. two PSRs, 1 and 2, of unequal size in series

Case 3. *n* PSRs in series, each of size *i*

The RTD functions for these for cases are listed below. τ is the mean residence time of an ideal reactor in the network.:

$$\begin{aligned}
&\text{Case 1.} \quad E(t) = 1/\tau_1 e^{(\tau_d - t)/\tau_1} \text{ for } t \geq \tau_d, \text{ 0 otherwise} \\
&\text{Case 2.} \quad E(t) = \frac{1}{\tau_1 - \tau_2}\left(e^{-t/\tau_1} - e^{-t/\tau_2}\right) \\
&\text{Case 3.} \quad E(t) = \frac{t^{n-1}}{(n-1)!\tau_i^n} e^{-t/\tau_i}
\end{aligned} \tag{C.13}$$

C.2.2.1 Case 1 Moments. Applying the RTD to the definition of the mean:

$$\mu = \int_0^\infty tE(t)dt = \int_{\tau_d}^\infty 1/\tau_1 e^{(\tau_d - t)/\tau_1} dt \tag{C.14}$$

change variables to x, such that $x = t - \tau_d$:

$$\begin{aligned}
\mu &= \frac{1}{\tau_1}\int_0^\infty (x + \tau_d) e^{-x/\tau_1} dx = \frac{1}{\tau_1}\int_0^\infty x e^{-x/\tau_1} dx + \frac{\tau_d}{\tau_1}\int_0^\infty e^{-x/\tau_1} dx \\
&= \tau_1 + \tau_d
\end{aligned} \tag{C.15}$$

Using the definition of the variance:

$$\sigma^2 = \int_0^\infty (t - \mu)^2 E(t)dt = \int_0^\infty t^2 E(t)dt - \mu^2 \tag{C.16}$$

and applying the Case 1 RTD with the same change in variable as equation C.15:

$$\sigma^2 = \int_0^\infty \frac{(x+\tau_d)^2}{\tau_1} e^{-x/\tau_1} dx - (\tau_1+\tau_d)^2$$

$$= \int_0^\infty \left(\frac{x^2}{\tau_1} + \frac{2x\tau_d}{\tau_1} + \frac{\tau_d^2}{\tau_1} \right) e^{-x/\tau_1} dx - (\tau_1+\tau_d)^2 \qquad \text{(C.17)}$$

$$= 2\tau_1^2 + 2\tau_1\tau_d + \tau_d^2 - \tau_1^2 - 2\tau_1\tau_d - \tau_d^2$$

$$= \tau_1^2$$

Using the definition of the skewness coefficient:

$$s^3 = \frac{1}{\sigma^3}\int_0^\infty (t-\mu)^3 E(t)dt = \frac{1}{\sigma^3}\int_0^\infty t^3 E(t)dt - \frac{3\mu}{\sigma} - \frac{\mu^3}{\sigma^3} \qquad \text{(C.18)}$$

and applying the Case 1 RTD with the same change in variable as equation C.18:

$$s^3 = \frac{1}{\tau_1^3}\int_0^\infty \frac{(x+\tau_d)^3}{\tau_1} e^{-x/\tau_1} dx - \frac{3(\tau_d+\tau_1)}{\tau_1} - \frac{(\tau_d+\tau_1)^3}{\tau_1^3}$$

$$= \frac{1}{\tau_1^4}\int_0^\infty \left(x^3 + \tau_d x^2 + 3\tau_d^2 x + \tau_d^3 \right) e^{-x/\tau_1} dx - 3 - \frac{3\tau_d}{\tau_1} - \frac{\tau_d^3}{\tau_1^3} - \frac{3\tau_d^2}{\tau_1^2} - \frac{3\tau_d}{\tau_1} - 1 \qquad \text{(C.19)}$$

$$= 6 + \frac{6\tau_d}{\tau_1} + \frac{3\tau_d^2}{\tau_1^2} + \frac{\tau_d^3}{\tau_1^3} - 3 - \frac{3\tau_d}{\tau_1} - \frac{\tau_d^3}{\tau_1^3} - \frac{3\tau_d^2}{\tau_1^2} - \frac{3\tau_d}{\tau_1} - 1 = 2$$

C.2.2.2 Case 2 Moments. The RTD for Case 2 is substituted into equation C.14:

$$\mu = \frac{1}{(\tau_1-\tau_2)}\left(\int_0^\infty te^{-t/\tau_1} dt - \int_0^\infty te^{-t/\tau_2} dt \right) = \frac{(\tau_1^2-\tau_2^2)}{(\tau_1-\tau_2)} = \tau_1 + \tau_2 \qquad \text{(C.20)}$$

Likewise, the variance is found by substituting the Case 2 RTD in equation C.16:

$$\sigma^2 = \frac{1}{(\tau_1 - \tau_2)}\left(\int_0^\infty t^2 e^{-t/\tau_1} dt - \int_0^\infty t^2 e^{-t/\tau_2} dt\right) - (\tau_1 + \tau_2)^2$$

$$= \frac{2(\tau_1^3 - \tau_2^3)}{(\tau_1 - \tau_2)} - \tau_1^2 - 2\tau_1\tau_2 - \tau_2^2 \qquad \text{(C.21)}$$

$$= \frac{2(\tau_1 - \tau_2)(\tau_1^2 + \tau_1\tau_2 + \tau_2^2)}{(\tau_1 - \tau_2)} - \tau_1^2 - 2\tau_1\tau_2 - \tau_2^2 = \tau_1^2 + \tau_2^2$$

for the skewness coefficient:

$$s^3 = \frac{1}{(\tau_1^2 + \tau_2^2)^{3/2}(\tau_1 - \tau_2)}\left(\int_0^\infty t^3 e^{-t/\tau_1} dt - \int_0^\infty t^3 e^{-t/\tau_2} dt\right) - \frac{3(\tau_1 + \tau_2)}{(\tau_1^2 + \tau_2^2)^{1/2}} - \frac{(\tau_1 + \tau_2)^3}{(\tau_1^2 + \tau_2^2)^{3/2}} \qquad \text{(C.22)}$$

$$= \frac{1}{(\tau_1^2 + \tau_2^2)^{1/2}}\left[\frac{6(\tau_1^4 - \tau_2^4)}{(\tau_1^2 + \tau_2^2)(\tau_1 - \tau_2)} - 3(\tau_1 + \tau_2) - \frac{(\tau_1 + \tau_2)^3}{(\tau_1^2 + \tau_2^2)}\right] = \frac{2(\tau_1^3 + \tau_2^3)}{(\tau_1^2 + \tau_2^2)^{3/2}}$$

C.2.2.3 Case 3 Moments. The RTD for Case 3 is substituted into equation C.14:

$$\mu = \frac{1}{(n-1)!\tau_i^n}\int_0^\infty t^n e^{-t/\tau_i} dt = \frac{n!\tau_i^{(n+1)}}{(n-1)!\tau_i^n} = n\tau_i \qquad \text{(C.23)}$$

Likewise, the variance is found by substituting the Case 3 RTD in equation C.16:

$$\sigma^2 = \frac{1}{(n-1)!\tau_i^n}\int_0^\infty t^{n+1} e^{-t/\tau_i} dt - n^2\tau_i^2 = \frac{(n+1)!\tau_i^{n+2}}{(n-1)!\tau_i^n} - n^2\tau_i^2 = n\tau_i^2 \qquad \text{(C.24)}$$

Finally, the Case 3 RTD substituted into equation gives the skewness coefficient:

$$s^3 = \frac{1}{n^{3/2}(n-1)!\tau_i^{n+3}}\int_0^\infty t^{n+2}e^{-t/\tau_i}dt - \frac{3n\tau_i}{n^{1/2}\tau_i} - \frac{n^3\tau_i^3}{n^{3/2}\tau_i^3}$$

$$= \frac{(n+2)!\tau_i^{n+3}}{n^{3/2}(n-1)!\tau_i^{n+3}} - 3n^{1/2} - n^{3/2} = \frac{1}{n^{3/2}}(n^3+3n^2+2n-3n^2-n^3) \qquad \text{(C.25)}$$

$$= \frac{2}{\sqrt{n}}$$

C.2.3 Calculation of Local Variance

For the step-input tracer experiments discussed in chapter 3, each input and output combination had multiple trials. The time interval of the data recorder was set at 0.05 seconds, but the interval was not precise. A Savitsky-Golay filter (Press *et al.*, 1992) with an imposed interval combined the information from the multiple runs. The fitted polynomial parameters estimated at each discrete time point were used to estimate the derivative of the series.

An estimate of local variance was required for error analysis. Since data points did not fall at precise intervals, the variance for each time interval point used all points one interval step to the left and right to determine the variance of the ordinate (y-axis). Figure C.C.1 illustrates this concept. Let y_k represent the estimated y-value from the Savitsky-Golay filter at time $t_k = k\Delta t$ on the imposed time sequence. A nearby point (t_j, y_j) is corrected using the derivative at t_k:

$$\tilde{y}_j = y_j - \frac{dy_k}{dt}(t_j - t_k) \qquad \text{(C.26)}$$

If j represents the set of n points within a time step of t_k, then the local variance is defined as:

$$\sigma_k^2 = \sum_{j=1}^{n} \frac{(\tilde{y}_j - \bar{y}_k)^2}{n-1} \tag{C.27}$$

where:

$$\bar{y}_k = \sum_{j=1}^{n} \frac{\tilde{y}_j}{n} \tag{C.28}$$

The local variance is used for determining the standard errors of the estimate of the model parameters in Chapter 4.

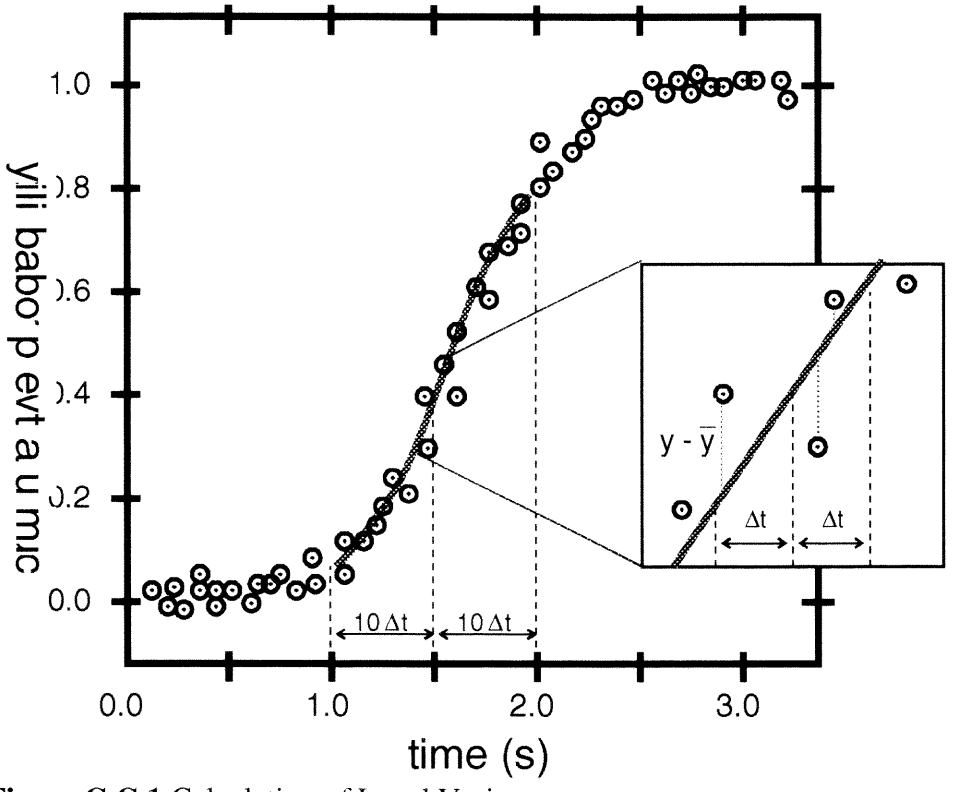


Figure C.C.1 Calculation of Local Variance

APPENDIX D

REACTOR MODEL PROGRAMS

This appendix has the FORTRAN source code for the reactor model programs. Section D.1 shows the networked ideal reactor program and D.2 shows the PaSR model program. Each program contains sufficient internal decimation for explaination. Both programs link to the *CHEMKIN-II* library.

D.1 Afterburner Version 2.4

```
C     PROGRAM AFTERBURNER
C
C
C     Integration of constant pressure kinetics problems using ideal
C     reactors
C
C     VERSION 2.4:
C    1.1.  17 Mar 94  added restart and fixed temperature options
C    1.2.  Removed element balance
C    1.3.  Added heat transfer in PFR
C    1.4.  Added tracking of CEMs and groups of products
C    1.5.  May 94 added the second PFR for a rapid cool
C    1.6.  Jun 94 added an additional Bypass between PSR1 and Choke
C    1.7.  Jul 94 added rop analysis and expressed results in mg/m*3
C    2.0   Jan 95 redesign of program to a modular format
C    2.1   Apr 96 added PSR 5, eliminated radiant heat loss
C    2.2   Jun 96 generalized file opening in PDPSR, added sensitivity
C          analysis
C    2.3   Dec 96 added heat transfer coefficient estimation based on
C          complete combustion; modified input file; modified SUBTOUTINE
C          FUN to allow isothermal reactors; added model sensitivity
C    2.4   Oct 01 modified program to match model in chapter 4; changed
C          heat transfer coefficient estimaion using PSR to estimate
C          enthalpy of combustion products
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)
      PARAMETER (LENIWK=500000, LENRWK=500000, LENCWK=50000, NK=30,
     1           NLMAX=55, LIN=5, LOUT=6, LINKCK=25, KMAX=250,
     2           LPIC=18, LRAD=19, LCEM=20,
     3           LROP=39, LOPS=9)
C
      DIMENSION IWORK(LENIWK), RWORK(LENRWK), X(KMAX), Z(KMAX),
```

```
     1          XF(KMAX), ZF(KMAX), XA(KMAX), ZA(KMAX), XK(KMAX),
     2          ZK(KMAX), XD(KMAX), ZD(KMAX), ZP(KMAX), XP(KMAX),
     3          XI(KMAX), ZI(KMAX), XIM(KMAX), ZIM(KMAX), ZSI(KMAX),
     4          ZSO4(KMAX), ZSO5(KMAX), S(KMAX), IPIC(NK), IRAD(NK),
     5          ICEM(NK), ITOP(30), VOL(LOPS), TEMP(LOPS), UA(LOPS),
     6          P(LOPS), BETA(20), ZKSI(KMAX)
      CHARACTER CWORK(LENCWK)*16, KSYM(KMAX)*16, KELE(6)*2, REACTOR*4,
     1          CNUMS*10, FILE1*12, FILE2*12, FILE3*12, FILE4*12
      LOGICAL KERR, IERR, LPHI_B, LPHI, RESTART, SERIES, UNITS ,LTAU,
     1        SENS, TGIV(LOPS), SKIP2, LUA(LOPS), MODEL
      EXTERNAL FUN
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
C
      DATA KERR/.FALSE./, ISENS/0/, X/KMAX*0.0/, KSYM/KMAX*' '/,
     1     XA/KMAX*0.0/, XK/KMAX*0.0/, XD/KMAX*0.0/,
     2     CAL2ERG /41868000/
      CNUMS = '0123456789'
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Initialize files and constants
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      OPEN (UNIT=LOUT, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='ab24.out')
      WRITE (LOUT, 15)
   15 FORMAT (
     1/'  AFTERBURNER:  Version 2.4'
     2/'                Charles Bass, October 2001',
     3/'                DOUBLE PRECISION')
C     Open the CHEMKIN LINK file
C
      OPEN (LINKCK, FORM='UNFORMATTED', file='cklink')
C
C     Initialize CHEMKIN
C
      CALL CKLEN  (LINKCK, LOUT, LENI, LENR, LENC)
      CALL CKINIT (LENIWK, LENRWK, LENCWK, LINKCK, LOUT, IWORK,
     1             RWORK, CWORK)
      CALL CKINDX (IWORK, RWORK, MM, KK, II, NFIT)
C
C     Open Output Files
C
      OPEN (UNIT=LIN, STATUS='OLD', FORM='FORMATTED', FILE='abinp.txt')
      OPEN (UNIT=LPIC, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='pics.txt')
      OPEN (UNIT=LRAD, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='radicals.txt')
      OPEN (UNIT=LCEM, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='cems.txt')
C
C
C
      NEQ   = KK + 1
```

```
      LRW   = 22 + 9*NEQ + 2*NEQ**2
      NVODE = LENR + 1
      NWT   = NVODE + LRW
      NH    = NWT  + KK
      NWDOT = NH   + KK
      NTOT  = NWDOT+ KK - 1
C
      LIW   = 30 + NEQ
      IVODE = LENI + 1
      ITOT  = IVODE + LIW - 1
C
      IF (KK .GT. KMAX) THEN
         WRITE (LOUT, *)
     1   ' Error...KMAX too small...must be at least ',KK
         KERR = .TRUE.
      ENDIF
C
      IF (LENRWK .LT. NTOT) THEN
         KERR = .TRUE.
         WRITE (LOUT, *)
     1   ' Error...LENRWK too small...must be at least', NTOT
      ENDIF
C
      IF (LENIWK .LT. ITOT) THEN
         KERR = .TRUE.
         WRITE (LOUT, *)
     1   ' Error...LENRWK too small...must be at least', NTOT
      ENDIF
C
      IF (LENIWK .LT. ITOT) THEN
         KERR = .TRUE.
         WRITE (LOUT, *)
     1   ' Error...LENIWK too small...must be at least', ITOT
      ENDIF
C
      IF (KERR) STOP
C
      CALL CKSYMS (CWORK, LOUT, KSYM, IERR)
      CALL CKSYME (CWORK, LOUT, KELE, IERR)
      IF (IERR) KERR = .TRUE.
      CALL CKWT   (IWORK, RWORK, RWORK(NWT))
      CALL CKRP   (IWORK, RWORK, RU, RUC, PATM)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Read input file
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      CALL PARSER (LIN, LOUT, 'FLAG', 4, R1, R2, R3, R4)
      RESTART = (INT(R1) .EQ. 1)
      SERIES = (INT(R1) .EQ. 2)
      LPHI_B = (INT(R2) .EQ. 1)
      LPHI = (INT(R2) .EQ. 2)
      LTAU = (INT(R3) .EQ. 1)
      SENS = (INT(R4) .EQ. 1)
      MODEL = (INT(R4) .EQ. 2)
```

```
      SKIP2 = (INT(R4) .EQ. 9)
      CALL PARSER (LIN, LOUT, 'UNITS', 1, R1, R2, R3, R4)
      UNITS = (INT(R1) .EQ. 1)
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'PICS', IPIC, IK1)
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'RADIC', IRAD, IK2)
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'CEMS', ICEM, IK3)
      CALL PARSER (LIN, LOUT, 'AMBIENT', 2, PA, TA, R3, R4)
      IF (UNITS) THEN
        PA = PA/29.9213
        TA = (TA + 459.67)*5/9
      ELSE
        PA = PA/760
        TA = TA + 273.15
      ENDIF
C
C     Default reactor pressure is equal to ambient pressure in
dynes/cm**2
C
      DO 30 I = 1, LOPS
         IF (I .EQ. 1) REACTOR = 'PSR1'
         IF (I .EQ. 2) REACTOR = 'PSR2'
         IF (I .EQ. 3) REACTOR = 'PSR3'
         IF (I .EQ. 4) REACTOR = 'PFR4'
         IF (I .EQ. 5) REACTOR = 'PSR5'
         IF (I .EQ. 6) REACTOR = 'PSR6'
         IF (I .EQ. 7) REACTOR = 'PFR7'
         IF (I .EQ. 8) REACTOR = 'PFR8'
         IF (I .EQ. 9) REACTOR = 'PFR9'
        CALL PARSER (LIN, LOUT, REACTOR, 4, VOL(I), UA(I), TEMP(I),
     1               P(I))
        TGIV(I) = (INT(UA(I)) .EQ. -1)
        LUA(I) = (INT(UA(I)) .EQ. -2)
        IF (UNITS) THEN
          IF (.NOT. LTAU) VOL(I) = VOL(I) * 28316.85
          TEMP(I) = (TEMP(I) +459.67)*5/9
          IF (.NOT. TGIV(I) .AND. .NOT. LUA(I)) UA(I) = UA(I)*0.125998
          P(I) = P(I)/29.92126
        ELSE
          TEMP(I) = TEMP(I)
          P(I) = P(I)/760
      ENDIF
C
C     BETA is the parameter vector
C
      BETA(I) = VOL(I)
      IF (TGIV(I) .OR. LUA(I)) THEN
         BETA(I+9) = TEMP(I)
      ELSE
         BETA(I+9) = UA(I)
      ENDIF
   30 CONTINUE
      CALL PARSER (LIN, LOUT, 'IMPINGER', 2, TAU_IMP, TEMP_IMP, R3, R4)
      IF (UNITS) THEN
         TEMP_IMP = (TEMP_IMP + 459.67)*5/9
      ELSE
         TEMP_IMP = TEMP_IMP + 273.15
      ENDIF
```

```
      CALL PARSER (LIN, LOUT, 'PHI', 1, PHI_D, R2, R3, R4)
      CALL PARSER (LIN, LOUT, 'BYPASS', 2, BYPASS1, BYPASS3, R3, R4)
      BETA(19) = BYPASS1
      BETA(20) = BYPASS3
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Stream input and adjustment
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     burner fuel and air
C
      CALL STREAM (LIN, LOUT, 'FUEL', KSYM, KK, IWORK, RWORK, UNITS,
     1             FLRTF, XF, ZF)
      CALL STREAM (LIN, LOUT, 'AIR', KSYM, KK, IWORK, RWORK, UNITS,
     1             FLRTA, XA, ZA)
C
C     Kiln gas and dopant
C
      CALL STREAM (LIN, LOUT, 'KILNG', KSYM, KK, IWORK, RWORK, UNITS,
     1             FLRTK, XK, ZK)
      CALL STREAM (LIN, LOUT, 'DOPANT', KSYM, KK, IWORK, RWORK, UNITS,
     1             FLRTD, XD, ZD)
      IF (FLRTD .GT. 0.0 .AND. FLRTK .GT. 0.0) THEN
         CALL MIX (KK, FLRTK, FLRTD, ZK, ZD, IWORK, RWORK)
         CALL CKYTX (ZK(2), IWORK, RWORK, XK)
      ENDIF
      IF (FLRTD .GT. 0.0 .AND. FLRTK .EQ. 0.0) THEN
         DO 10 I = 1,KK+1
            ZK(I) = ZD(I)
   10    CONTINUE
         FLRTK = FLRTD
         CALL CKYTX(ZK(2), IWORK, RWORK, XK)
      ENDIF
C
C     Air and steam injection
C
      CALL STREAM (LIN, LOUT, 'INJ1', KSYM, KK, IWORK, RWORK, UNITS,
     1             FLRTI, XI, ZI)
      CALL STREAM (LIN, LOUT, 'INJ2', KSYM, KK, IWORK, RWORK, UNITS,
     1             FLRTP, XP, ZP)
      IF (FLRTP .GT. 0.0) THEN
         CALL MIX (KK, FLRTI, FLRTP, ZI, ZP, IWORK, RWORK)
         CALL CKYTX (ZI(2), IWORK, RWORK, XI)
      ENDIF
C
C     calculate fuel/air equivalence ratios
C
      PHI_B = EQUIV_RATIO (KK, ZA(2), ZF(2), ZA(2), ZF(2), FLRTA,
     1                     FLRTF, FLRTA, FLRTF, KELE, KSYM, IWORK, RWORK)
      PHI = EQUIV_RATIO (KK, ZA(2), ZF(2), ZK(2), ZI(2), FLRTA, FLRTF,
     1                       FLRTK, FLRTI, KELE, KSYM, IWORK, RWORK)
C
C     Adjust fuel/air equivalence ratio
C
      IF (LPHI_B) THEN
```

```
         FLRTA = FLRTA*PHI_B/PHI_D
      ENDIF
      IF (LPHI) THEN
         CALL CKCOMP ('O2', KSYM, KK, IO2)
         IO2 = IO2 +1
         A = ZF(IO2)*FLRTF + ZK(IO2)*FLRTK + ZI(IO2)*FLRTI
         B = A + ZA(IO2)*FLRTA
         FLRTA = 1/ZA(IO2)*(B*PHI/PHI_D - A)
      ENDIF
C
C     Recalculate fuel/air equivalence ratios
C
      IF (LPHI .OR. LPHI_B) THEN
         PHI_B = EQUIV_RATIO (KK, ZA(2), ZF(2), ZA(2), ZF(2), FLRTA,
     1                    FLRTF, FLRTA, FLRTF, KELE, KSYM, IWORK, RWORK)
         PHI = EQUIV_RATIO (KK, ZA(2), ZF(2), ZK(2), ZI(2), FLRTA,
     1                    FLRTF, FLRTK, FLRTI, KELE, KSYM, IWORK, RWORK)
      ENDIF
C
C     Mix fuel/air stream
C
      CALL MIX (KK, FLRTF, FLRTA, ZF, ZA, IWORK, RWORK)
      CALL CKYTX (ZF(2), IWORK, RWORK, XF)
      FLRT = FLRTF
      FLRTSI = FLRTF
      FLRTKSI = FLRTK
      DO 50 N = 1, KK+1
         Z(N) = ZF(N)
         ZSI(N) = ZF(N)
         ZKSI(N) = ZK(N)
   50 CONTINUE
   70 CONTINUE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Solve series of reactors
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     IF conducting model sensivity the reconstruct orignal input
C     stream and set model parameters
C
      IF (MODEL) THEN
       DO 80 N = 1, KK+1
          Z(N) = ZSI(N)
          ZK(N) = ZKSI(N)
   80    CONTINUE
       FLRT = FLRTSI
       FLRTK = FLRTKSI
       DO 90 N = 1,9
          VOL(N) = BETA(N)
          IF (TGIV(N) .OR. LUA(N)) THEN
             TEMP(N) = BETA(N+9)
          ELSE
             UA(N) = BETA(N+9)
          ENDIF
   90    CONTINUE
```

```
       BYPASS1 = BETA(19)
       BYPASS3 = BETA(20)
      ENDIF
      CALL CKYTX (Z(2), IWORK, RWORK, X)
C
C     Write output file headings
C
      WRITE (LPIC,7510) (KSYM(IPIC(K))(:10), K=1,IK1)
      WRITE (LRAD,7510) (KSYM(IRAD(K))(:10), K=1,IK2)
      WRITE (LCEM,7510) "THC","TCLC",(KSYM(ICEM(K))(:10), K=1,IK3)
C
C     ***** PSR 1 *****
C
      IF (VOL(1) .NE. 0.0) THEN
C
C        Run PSR sub program
C
C
C     Total Heat Flux in cal/s
C
         IF (LUA(1)) TGIV(1) = .TRUE.
         QLOS = UA(1)*(TEMP(1)**4 - TEMP(2)**4)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PSR1/IN', IWORK, RWORK)


         CALL PDPSR(1, KK, II, Z(1), X, KSYM, FLRT, P(1), VOL(1),
     1              TGIV(1), QLOS, TEMP(1), RESTART, LTAU, SENS, Z)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PSR1/OUT', IWORK, RWORK)
         CALL CKYTX (Z(2), IWORK, RWORK, X)
         CALL CKRHOX (P(1)*PATM, Z(1), X, IWORK, RWORK, RHO)
         IF (LTAU) VOL(1) = VOL(1)*FLRT/RHO
         CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
         WRITE (LPIC,7511) 'PSR1', VOL(1), Z(1), (X(IPIC(K)), K=1,IK1)
         WRITE (LRAD,7511) 'PSR1', VOL(1), Z(1), (X(IRAD(K)), K=1,IK2)
         WRITE (LCEM,7511) 'PSR1', VOL(1), Z(1), THC, TCLC,
     1                     (X(ICEM(K)),K=1,IK3)
      ENDIF
C
C     Bypass 1
C
      ZF(1) = Z(1)
      DO 100 I = 1, KK
         ZF(I+1) = Z(I+1)
         XF(I) = X(I)
  100 CONTINUE
      FLRTF = BYPASS1*FLRT
      FLRT = (1-BYPASS1)*FLRT
C
C     ***** PSR 2 *****
C
      IF (VOL(2) .NE. 0.0 .AND. .NOT. SKIP2) THEN
         IF (LUA(2)) TGIV(2) = .TRUE.
         QLOS = UA(2)*(TEMP(2) - Z(1))
C
C     A short PFR is called to simulate the transition duct
C
         DT = 1.0E-4
```

```
          IPFR = 0
          TAU = 1.0
          UAV = -1
          CALL PDPFR (IPFR, LOUT, LPIC, LRAD, LCEM, ZK, IWORK, RWORK,
     1               KELE, FLRTK, P(2), TAU, DT, IPIC, IRAD, ICEM,
     2               IK1, IK2, IK3, .TRUE.)
          CALL CKYTX (ZK(2), IWORK, RWORK, XK)
          CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'PSR2/IN', IWORK, RWORK)
          CALL PDPSR(2, KK, II, ZK(1), XK, KSYM, FLRTK, P(2), VOL(2),
     1              TGIV(2), QLOS, TEMP(2), RESTART, LTAU, SENS, ZK)
          CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'PSR2/OUT', IWORK, RWORK)
          CALL CKYTX (ZK(2), IWORK, RWORK, XK)
          CALL CKRHOX (P(2)*PATM, ZK(1), XK, IWORK, RWORK, RHO)
          IF (LTAU) VOL(2) = VOL(2)*FLRTK/RHO
          CALL GROUP (KK, XK, KELE, IWORK, RWORK, THC, TCLC)
          WRITE (LPIC,7511) 'PSR2', VOL(2), ZK(1), (XK(IPIC(K)), K=1,IK1)
          WRITE (LRAD,7511) 'PSR2', VOL(2), ZK(1), (XK(IRAD(K)), K=1,IK2)
          WRITE (LCEM,7511) 'PSR2', VOL(2), ZK(1), THC, TCLC,
     1                      (XK(ICEM(K)),K=1,IK3)
C
C     cumlative save at point 3
C
        IF (ISENS .EQ. 0)
     1    CALL SAVER (KK, ZK, XK, SERIES, 'pt3pic.txt', 'pt3rad.txt',
     2                'pt3cem.txt','pt3bin', KSYM, KELE, IWORK, RWORK,
     3                IPIC, IRAD,ICEM, IK1, IK2, IK3, PHI, PHI_B)
C
C     send output through impinger
C
        IF (TAU_IMP .GT. 0.0) THEN
            CALL IMPINGER (7, RESTART, II, ZK, KSYM, TAU_IMP, TEMP_IMP,
     1                     IWORK, RWORK, ZIM)
            CALL CKYTX (ZIM(2), IWORK, RWORK, XIM)
             IF (ISENS .EQ. 0)
     1         CALL SAVER (KK, ZIM, XIM, SERIES, 'im3pic.txt',
     2                'im3rad.txt', 'im3cem.txt', 'im3bin', KSYM, KELE,
     3                IWORK, RWORK, IPIC, IRAD, ICEM, IK1, IK2, IK3,
     4                PHI, PHI_B)
          ENDIF
         ENDIF
  150 CONTINUE
C
C     ***** PSR 3 *****
C
         IF (FLRTF .GE. 0.0) THEN
            CALL MIX(KK, FLRTK, FLRTF, ZK, ZF, IWORK, RWORK)
            CALL CKYTX (ZK(2), IWORK, RWORK, XK)
         ENDIF
         IF (VOL(3) .NE. 0.0) THEN
            IF (LUA(3)) TGIV(3) = .TRUE.
            QLOS = UA(3)*(TEMP(3) - Z(1))
            CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'PSR3/IN', IWORK, RWORK)
            CALL PDPSR(3, KK, II, ZK(1), XK, KSYM, FLRTK, P(3), VOL(3),
     1              TGIV(3), QLOS, TEMP(3), RESTART, LTAU, SENS, ZK)
            CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'PSR3/OUT', IWORK, RWORK)
            CALL CKYTX (ZK(2), IWORK, RWORK, XK)
            CALL CKRHOX (P(3)*PATM, ZK(1), XK, IWORK, RWORK, RHO)
```

```
      IF (LTAU) VOL(3) = VOL(3)*FLRTK/RHO
      CALL GROUP (KK, XK, KELE, IWORK, RWORK, THC, TCLC)
      WRITE (LPIC,7511) 'PSR3', VOL(3), ZK(1), (XK(IPIC(K)), K=1,IK1)
      WRITE (LRAD,7511) 'PSR3', VOL(3), ZK(1), (XK(IRAD(K)), K=1,IK2)
      WRITE (LCEM,7511) 'PSR3', VOL(3), ZK(1), THC, TCLC,
     1                  (XK(ICEM(K)),K=1,IK3)
      ENDIF
C
C     ***** PFR 4 *****
C
      IF (VOL(4) .GT. 0.0) THEN
         TA_OLD = TA
         TA = Z(1)
         IF (LTAU) THEN
            TOUT = TEMP(4)
            TLM = (ZK(1)-TOUT)/LOG((ZK(1)-TA)/(TOUT-TA))
            CALL CKRHOY (P(4)*PATM, TLM+TA, ZK(2), IWORK, RWORK, RHO)
            VOLP = VOL(4)*FLRTK/RHO
         ELSE
            VOLP = VOL(4)
         ENDIF
      ENDIF
      IF (LUA(4) .AND. VOL(4) .GT. 0.0) THEN
           UA(4) = UA_FIND (KK, II, ZK(1), TOUT, TA, XK, KSYM,
     1                      FLRTK, PA, VOL(4), IWORK, RWORK)
           UAV = UA(4)/VOLP
      ELSEIF (TGIV(4) .AND. VOL (4) .GT. 0.0) THEN
           UAV = -1.0
           ZK(1) = TEMP(4)
      ELSEIF (VOL(4) .GT. 0.0) THEN
           UAV = UA(4)*CAL2ERG/VOLP
      ENDIF
      DT = 1.00e-4
      CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'PFR4/IN', IWORK, RWORK)
      IF (VOL(4) .GT. 0.0) THEN
         CALL PDPFR (4, LOUT, LPIC, LRAD, LCEM, ZK, IWORK, RWORK,
     1               KELE, FLRTK, P, VOL(4), DT, IPIC, IRAD, ICEM,
     2               IK1, IK2, IK3, LTAU)
      CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'PFR4/OUT', IWORK, RWORK)
      CALL CKYTX (ZK(2), IWORK, RWORK, XK)
      CALL CKRHOX (P(3)*PATM, ZK(1), XK, IWORK, RWORK, RHO)
      CALL GROUP (KK, XK, KELE, IWORK, RWORK, THC, TCLC)
      IF (.NOT. LUA(4)) VOLP = VOL(4)
      WRITE (LPIC,7511) 'PFR4', VOLP, ZK(1), (XK(IPIC(K)), K=1,IK1)
      WRITE (LRAD,7511) 'PFR4', VOLP, ZK(1), (XK(IRAD(K)), K=1,IK2)
      WRITE (LCEM,7511) 'PFR4', VOLP, ZK(1), THC, TCLC,
     1                  (XK(ICEM(K)),K=1,IK3)
      TA = TA_OLD
      ENDIF
C
C     Choke (point 4) Logging of Kiln and Flame gas streams
C
C     Flame Gas
C
      CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'CHOKE F', IWORK, RWORK)
      IF (ISENS .EQ. 0)
     1CALL SAVER (KK, Z, X, SERIES, 'pt4fpic.txt', 'pt4frad.txt',
```

```
     2            'ptf4cem.txt', 'pt4fbin', KSYM, KELE, IWORK, RWORK,
     3            IPIC, IRAD,ICEM, IK1, IK2, IK3, PHI, PHI_B)
      CALL TOP30 (KK, 'SCC Choke F', KSYM, X, ITOP)
C
C     send output through impinger
C
      IF (TAU_IMP .GT. 0.0) THEN
         CALL IMPINGER (8, RESTART, II, Z, KSYM, TAU_IMP, TEMP_IMP,
     1                 IWORK, RWORK, ZIM)
         CALL CKYTX (ZIM(2), IWORK, RWORK, XIM)
       IF (ISENS .EQ. 0)
     1   CALL SAVER (KK, ZIM, XIM, SERIES, 'im4fpic.txt', 'im4frad.txt',
     2              'im4fcem.txt', 'im3fbin', KSYM, KELE, IWORK, RWORK,
     3               IPIC, IRAD, ICEM, IK1, IK2, IK3, PHI, PHI_B)
         CALL TOP30 (KK, 'Choke F - Impinger', KSYM, X, ITOP)
      ENDIF
C
C     Kiln Gas
C
      CALL ENERGY_BAL (ZK(1), ZK(2), FLRTK, 'CHOKE K', IWORK, RWORK)
      IF (ISENS .EQ. 0)
     1CALL SAVER (KK, ZK, XK, SERIES, 'pt4kpic.txt', 'pt4krad.txt',
     2            'ptk4cem.txt', 'pt4kbin', KSYM, KELE, IWORK, RWORK,
     3            IPIC, IRAD,ICEM, IK1, IK2, IK3, PHI, PHI_B)
      CALL TOP30 (KK, 'SCC Choke K', KSYM, XK, ITOP)
C
C     send output through impinger
C
      IF (TAU_IMP .GT. 0.0) THEN
         CALL IMPINGER (9, RESTART, II, ZK, KSYM, TAU_IMP, TEMP_IMP,
     1                 IWORK, RWORK, ZIM)
         CALL CKYTX (ZIM(2), IWORK, RWORK, XIM)
       IF (ISENS .EQ. 0)
     1   CALL SAVER (KK, ZIM, XIM, SERIES, 'im4kpic.txt', 'im4krad.txt',
     2              'im4kcem.txt', 'im3kbin', KSYM, KELE, IWORK, RWORK,
     3               IPIC, IRAD, ICEM, IK1, IK2, IK3, PHI, PHI_B)
         CALL TOP30 (KK, 'Choke F - Impinger', KSYM, X, ITOP)
      ENDIF

C
C     Calculate, save normalized model 1st order sensivity coefficients
C
      IF (MODEL) THEN
         IF (ISENS .EQ. 0) THEN
            DO 160 N = 1, KK+1
               ZSO4(N) = ZK(N)
  160       CONTINUE
         ENDIF
C
C     Mean molecular weight
C
         WT = 0
         DO 165 I = 1,KK
            WT = WT + XK(I)*RWORK(NWT-1+I)
  165    CONTINUE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
C
C     Calculate normalized sensitivity coefficients and save
C     Sensitivity coefficients calculated on mole fraction basis
C     Zero or negative concentrations have sensitivity coeff of 0
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
         RMAS2MOL = 0
         IF (ISENS .GT. 0) THEN
            S(1) = (BETA(ISENS)/ZSO4(1))*(ZK(1) - ZSO4(1))/H
            DO 170 K = 1, KK
               S(K+1) = (ZK(K+1) - ZSO4(K+1))/H
               RMAS2MOL = RMAS2MOL + S(K+1)/RWORK(NWT-1+K)
  170       CONTINUE
            RMAS2MOL = BETA(ISENS)*WT*RMAS2MOL
            DO 180 K = 1, KK
                  IF (ZSO4(K+1) .GT. 0.0) THEN
                 S(K+1) = (BETA(ISENS)/ZSO4(K+1))*S(K+1) - RMAS2MOL
                  ELSE
                     S(K+1) = 0.0
                  ENDIF
  180       CONTINUE
            ITENS = INT(ISENS/10) + 1
            IONES = ISENS - (ITENS -1)*10 + 1
            FILE1 = 'sn4pic'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
     1              //'.TXT'
            FILE2 = 'sn4rad'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
     1              //'.TXT'
            FILE3 = 'sn4cem'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
     1              //'.TXT'
            FILE4 = 'sn4bin'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
            CALL SAVER (KK, S, S(2), SERIES, FILE1, FILE2, FILE3,
     1                  FILE4, KSYM, KELE, IWORK, RWORK, IPIC, IRAD,
     2                  ICEM, IK1, IK2, IK3, PHI, PHI_B)
         ENDIF
       ENDIF
C
C     Add post mixing chamber injection
C
      IF (FLRTI .GT. 0.0) THEN
         CALL MIX (KK, FLRT, FLRTI, Z, ZI, IWORK, RWORK)
         CALL CKYTX (Z(2), IWORK, RWORK, X)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'CHOKE/INJ', IWORK, RWORK)
      ENDIF
C
C     ***** PSR 5 *****
C
C     First mix in stream from bypass 3
C
      IF (BYPASS3 .GE. 0.0) THEN
C
C     Bypass 3
C
         ZF(1) = Z(1)
         DO 185 I = 1, KK
            ZF(I+1) = ZK(I+1)
  185 CONTINUE
```

```
         FLRTF = BYPASS3*FLRTK
         FLRTK = (1-BYPASS3)*FLRTK
C
C     Mix with flame stream and execute PSR
C
         CALL MIX(KK, FLRT, FLRTF, Z, ZF, IWORK, RWORK)
         CALL CKYTX (Z(2), IWORK, RWORK, X)
       ENDIF
       IF (VOL(5) .NE. 0.0) THEN
         IF (LUA(5)) TGIV(5) = .TRUE.
         QLOS = UA(5)*(TEMP(5)-TA)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PSR5/IN', IWORK, RWORK)
         CALL PDPSR(5, KK, II, Z(1), X, KSYM, FLRT, P(5), VOL(5),
     1              TGIV(5), QLOS, TEMP(5), RESTART, LTAU, SENS, Z)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PSR5/OUT', IWORK, RWORK)
         CALL CKYTX (Z(2), IWORK, RWORK, X)
         CALL CKRHOX (P(5)*PATM, Z(1), X, IWORK, RWORK, RHO)
         IF (LTAU) VOL(5) = VOL(5)*FLRT/RHO
         CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
         WRITE (LPIC,7511) 'PSR5', VOL(5), Z(1), (X(IPIC(K)), K=1,IK1)
         WRITE (LRAD,7511) 'PSR5', VOL(5), Z(1), (X(IRAD(K)), K=1,IK2)
         WRITE (LCEM,7511) 'PSR5', VOL(5), Z(1), THC, TCLC,
     1                     (X(ICEM(K)),K=1,IK3)
       ENDIF
C
C     ***** PSR 6 *****
C
C     Mix in balance of kiln gass
C
       CALL MIX(KK, FLRT, FLRTK, Z, ZK, IWORK, RWORK)
       IF (ISENS .EQ. 0)
     1 CALL SAVER (KK, Z, X, SERIES, 'pt4apic.txt', 'pt4arad.txt',
     2             'pt4acem.txt', 'pt4abin', KSYM, KELE, IWORK, RWORK,
     3              IPIC, IRAD,ICEM, IK1, IK2, IK3, PHI, PHI_B)

       CALL CKYTX (Z(2), IWORK, RWORK, X)
       IF (VOL(6) .NE. 0.0) THEN
         IF (LUA(6)) TGIV(6) = .TRUE.
         QLOS = UA(6)*(TEMP(6)-TA)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PSR6/IN', IWORK, RWORK)
         CALL PDPSR(6, KK, II, Z(1), X, KSYM, FLRT, P(6), VOL(6),
     1              TGIV(6), QLOS, TEMP(6), RESTART, LTAU, SENS, Z)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PSR6/OUT', IWORK, RWORK)
         CALL CKYTX (Z(2), IWORK, RWORK, X)
         CALL CKRHOX (P(6)*PATM, Z(1), X, IWORK, RWORK, RHO)
         IF (LTAU) VOL(6) = VOL(6)*FLRT/RHO
         CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
         WRITE (LPIC,7511) 'PSR6', VOL(6), Z(1), (X(IPIC(K)), K=1,IK1)
         WRITE (LRAD,7511) 'PSR6', VOL(6), Z(1), (X(IRAD(K)), K=1,IK2)
         WRITE (LCEM,7511) 'PSR6', VOL(6), Z(1), THC, TCLC,
     1                     (X(ICEM(K)),K=1,IK3)
       ENDIF
C
C     ***** Following PFRs *****
C
       DO 200 I = 1,3
         IPFR = I+6
```

```
C
C     if LUA is true, a heat transfer coefficient is estimated to give
C     a chosen output temperature, if only TGIV is true reactor is
C     isothermal
C
C
         IF (VOL(IPFR) .GT. 0.0) THEN
            IF (LTAU) THEN
               TOUT = TEMP(IPFR)
               TLM = (Z(1)-TOUT)/LOG((Z(1)-TA)/(TOUT-TA))
               CALL CKRHOY(P(IPFR)*PATM, TLM+TA, Z(2), IWORK, RWORK,RHO)
               VOLP = VOL(IPFR)*FLRTK/RHO
            ELSE
               VOLP = VOL(IPFR)
            ENDIF
         ENDIF
         IF (LUA(I+6) .AND. VOL(I+6) .GT. 0.0) THEN
              UA(I+6) = UA_FIND (KK, II, Z(1), TOUT, TA, X, KSYM,
     1                           FLRT, PA, VOL(I+6), IWORK, RWORK)
              UAV = UA(I+6)/VOLP
         ELSEIF (TGIV(I+6) .AND. VOL (I+6) .GT. 0.0) THEN
              UAV = -1.0
              Z(1) = TEMP(I+6)
         ELSEIF (VOL(I+6) .GT. 0.0) THEN
              UAV = UA(I+6)*CAL2ERG/VOLP
         ENDIF
         DT = 1.00e-4
         IF (VOL(I+6) .GT. 0.0) THEN
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PFR/IN', IWORK, RWORK)
            CALL PDPFR (IPFR, LOUT, LPIC, LRAD, LCEM, Z, IWORK, RWORK,
     1                  KELE, FLRT, P, VOL(I+6), DT, IPIC, IRAD, ICEM,
     2                  IK1, IK2, IK3, LTAU)
         CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'PFR/OUT', IWORK, RWORK)
         ENDIF
C
C     cumlative save at point 5
C
         IF (I .EQ. 1) THEN
            CALL CKYTX (Z(2), IWORK, RWORK, X)
           IF (ISENS .EQ. 0)
     1      CALL SAVER (KK, Z, X, SERIES, 'pt5pic.txt', 'pt5rad.txt',
     2                 'pt5cem.txt', 'pt5bin', KSYM, KELE, IWORK, RWORK,
     3                 IPIC, IRAD,ICEM, IK1, IK2, IK3, PHI, PHI_B)
            CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'SCC Exit', IWORK, RWORK)
            CALL TOP30 (KK, 'SCC Exit', KSYM, X, ITOP)
C
C     send output through impinger
C
            IF (TAU_IMP .GT. 0.0) THEN
               CALL IMPINGER (10, RESTART, II, Z, KSYM, TAU_IMP,
     1                        TEMP_IMP, IWORK, RWORK, ZIM)
               CALL CKYTX (ZIM(2), IWORK, RWORK, XIM)
               IF (ISENS .EQ. 0)
     1         CALL SAVER (KK, ZIM, XIM, SERIES, 'im5pic.txt',
     2                    'im5rad.txt', 'im5cem.txt', 'im5bin', KSYM, KELE,
     3                     IWORK, RWORK, IPIC, IRAD, ICEM, IK1, IK2, IK3,
     4                     PHI, PHI_B)
```

```
               CALL TOP30 (KK, 'Exit - Impinger', KSYM, X, ITOP)
            ENDIF
         ENDIF
C
C     cumlative save at point stack
C
         IF (I .EQ. 3) THEN
            CALL CKYTX (Z(2), IWORK, RWORK, X)
          IF (ISENS .EQ. 0)
     1      CALL SAVER (KK, Z, X, SERIES,'ptstpic.txt','ptstrad.txt',
     2                'ptstcem.txt','ptstbin', KSYM, KELE, IWORK, RWORK,
     3                 IPIC, IRAD, ICEM, IK1, IK2, IK3, PHI, PHI_B)
C
C     send output through impinger
C
            IF (TAU_IMP .GT. 0.0) THEN
               CALL IMPINGER (11, RESTART, II, Z, KSYM, TAU_IMP,
     1                        TEMP_IMP, IWORK, RWORK, ZIM)
               CALL CKYTX (ZIM(2), IWORK, RWORK, XIM)
             IF (ISENS .EQ. 0)
     1         CALL SAVER (KK, ZIM, XIM, SERIES, 'imstpic.txt',
     2                'imstrad.txt', 'imstcem.txt', 'imstbin', KSYM,
     3                 KELE, IWORK, RWORK, IPIC, IRAD, ICEM, IK1, IK2,
     4                 IK3, PHI, PHI_B)
            ENDIF
         ENDIF
  200 CONTINUE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Model Sensitivity calculations
C     Calculates the normalized first order sensitivity coefficients
C     for the ideal reactor model.  Calculates for each volume (or mean
C     residence time) parameter, the temperature if specified in an
C     isothermal or specified exit temperature problem, and bypass
C     parameters.  If a parameter is zero the sensitivity coefficient
C     is automatically zero.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Save output of unperturbed system
C
      IF (MODEL) THEN
         IF (ISENS .EQ. 0) THEN
            DO 210 N = 1, KK+1
               ZSO5(N) = Z(N)
  210       CONTINUE
         ENDIF
C
C     Mean molecular weight
C
         WT = 0
         DO 215 I = 1,KK
            WT = WT + X(I)*RWORK(NWT-1+I)
  215    CONTINUE
C
C     Calculate normalized sensitivity coefficients and save
```

```
C     Sensitivity coefficients calculated on mole fraction basis
C
         RMAS2MOL = 0
         IF (ISENS .GT. 0) THEN
            S(1) = (BETA(ISENS)/ZSO5(1))*(Z(1) - ZSO5(1))/H
            DO 220 K = 1, KK
               S(K+1) = (Z(K+1) - ZSO5(K+1))/H
               RMAS2MOL = RMAS2MOL + S(K+1)/RWORK(NWT-1+K)
  220       CONTINUE
            RMAS2MOL = BETA(ISENS)*WT*RMAS2MOL
            DO 225 K = 1, KK
                  IF (ZSO5(K+1) .GT. 0.0) THEN
                      S(K+1) = (BETA(ISENS)/ZSO5(K+1))*S(K+1)
     1                             - RMAS2MOL
                  ELSE
                     S(K+1) = 0.0
                  ENDIF
  225       CONTINUE
            BETA(ISENS) = BETA_OLD
            ITENS = INT(ISENS/10) + 1
            IONES = ISENS - (ITENS -1)*10 + 1
            FILE1 = 'sn5pic'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
     1              //'.TXT'
            FILE2 = 'sn5rad'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
     1              //'.TXT'
            FILE3 = 'sn5cem'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
     1              //'.TXT'
            FILE4 = 'sn5bin'//CNUMS(ITENS:ITENS)//CNUMS(IONES:IONES)
            CALL SAVER (KK, S, S(2), SERIES, FILE1, FILE2, FILE3,
     1                  FILE4, KSYM, KELE, IWORK, RWORK, IPIC, IRAD,
     2                  ICEM, IK1, IK2, IK3, PHI, PHI_B)
         ENDIF
C
C     Perturb next parameter
C
  230    CONTINUE
         ISENS = ISENS + 1
         IF (ISENS .GT. 20) GOTO 250
         IF (BETA(ISENS) .LE. 0.0) GOTO 230
         BETA_OLD = BETA(ISENS)
         BETA(ISENS) = BETA(ISENS) + 0.001*BETA(ISENS)
         H = BETA(ISENS) - BETA_OLD
         REWIND(LOUT)
  235   CONTINUE
        GOTO 70
  250   CONTINUE
        ENDIF
 7510 FORMAT (2X, '"Unit"', 4X, ' "V(cm3)"',3X,'"TMP(K)"',
     1        30(' "',A16,'"'))
 7511 FORMAT (2X, A6, 2X, F11.0, F11.2, 30G19.4)
      END


C
      SUBROUTINE PDPSR (IPSR, KK, II, TIN, XIN, KSYM, FLRT, PA, VOL,
     1                  TGIV, QLOS, TEMP, RESTART, LTAU, SENS, ZOUT)
C
```

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     subroutine PDPSR sets up the input file and reads
C
C     inputs:
C
C        IPSR   integer, reactor number
C        KK     integer, number of species
C        II     integer, number of reactions
C        TIN    temperature of the input stream
C        XIN    mole fractions of species
C        KSYM   character array of chemical specie symbols
C        FLRT   mass flow rate g/sec
C        PA     ambient pressure (atm)
C        VOL    reactor volume (cm**3)
C        TGIV   logical, 1 temperature fixed, 0 temperature solved
C        QLOS   heat loss (cal/sec)
C        TEMP   reactor temperature (initial guess if TGIV=0)
C        RESTART logical 1 = start from restart file
C        LTAU   logical 1 = volumes become mean residence times in sec
C        SENS   logical 1 = run sensitivity analysis
C
C     output:
C
C        ZOUT   double array of exit temperature and mass fractions
C
C     scratch:
C
C        FOSC   double array of first order sensitivity coefficients
C               wrt the pre-exponential parameter in rxn i
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
C
      PARAMETER (LENRWK=500000,LENIWK=500000,LENLWK=50000,LENCWK=50000,
     1           KMAX=250)
      DIMENSION RWORK(LENRWK), IWORK(LENIWK), XIN(*), ZOUT(*),
     1          FOSC(KMAX)
      LOGICAL RESTART, TGIV, LWORK(LENLWK), LTAU, SENS
      CHARACTER KSYM(*)*16, CWORK(LENCWK)*16, CREST*8, CRCVR*8,
     1          CSENS*8, CNUMS*10
C
      DATA LIN, LOUT, LREST, LSAVE, LRCVR, LINKCK, LSENS
     1     /10, 6,14,11,16,25,26/
      DATA ISTEPS/100/, DT/1E-5/, ATOL/1E-20/, RTOL/1E-6/
      CNUMS = '0123456789'
C
c     Convert IPSR to a string and concatenate to file name
C
      ITENS = INT(IPSR/10) + 1
      IONES = IPSR - (ITENS - 1)*10 + 1
      CRCVR = 'rcvr' // CNUMS(ITENS:ITENS) // CNUMS(IONES:IONES)
      CREST = 'rest' // CNUMS(ITENS:ITENS) // CNUMS(IONES:IONES)
      CSENS = 'sens' // CNUMS(ITENS:ITENS) // CNUMS(IONES:IONES)
C
C     Open input and output files
```

```
C
      OPEN (LIN, STATUS ='UNKNOWN',     FORM='FORMATTED',
     1      FILE='psrin')
      OPEN (LINKCK, STATUS='OLD',      FORM='UNFORMATTED',
     1      FILE='cklink')
      OPEN (LREST, STATUS='UNKNOWN', FORM='UNFORMATTED',
     1      FILE=CREST)
      OPEN (LRCVR, STATUS='UNKNOWN',FORM='UNFORMATTED',
     1      FILE=CRCVR)
      OPEN (LSENS, STATUS='UNKNOWN',FORM='UNFORMATTED',
     1      FILE=CSENS)
      OPEN (LSAVE,  STATUS='UNKNOWN',    FORM='UNFORMATTED',
     1      FILE='save')
C
C     If LTAU is .TRUE. then VOL represents the mean residence time
C     volume is set at an arbitrary 1000 cc
C
      IF (LTAU) TAU = VOL
C
C
C     Write Input File for PSR1
C
C
      IF (.NOT. TGIV) WRITE (LIN, 2001)
      IF (TGIV) WRITE (LIN, 2015)
      IF (RESTART) WRITE (LIN, 2014)
      DO 50 I = 1, KK
      IF (XIN(I) .GT. 0 ) THEN
        WRITE (LIN, 2002) KSYM(I), XIN(I)
      ENDIF
   50 CONTINUE
      WRITE (LIN, 2003) PA
      IF (.NOT. LTAU) WRITE (LIN, 2004) VOL
      IF (LTAU) THEN
         WRITE (LIN, 2004) 1000.0
         WRITE (LIN, 2021) TAU
      ENDIF
      IF (.NOT. LTAU) WRITE (LIN, 2005) FLRT
      IF (.NOT. TGIV) WRITE (LIN, 2006) TIN
      IF (.NOT. TGIV) WRITE (LIN, 2007) QLOS
      WRITE (LIN, 2008) 0
      WRITE (LIN, 2009) ISTEPS, DT
      WRITE (LIN, 2010) ISTEPS, DT
      WRITE (LIN, 2019) ATOL
      WRITE (LIN, 2020) RTOL
      WRITE (LIN, 2011) TEMP
      IF (SENS) WRITE (LIN, 2022)
      WRITE (LIN, 2012)
      REWIND (LIN)
      WRITE (LOUT, 2017) IPSR
      CALL PSR (LIN, LOUT, LINKCK,  LREST, LSAVE, LRCVR, LENLWK,
     1          LWORK, LENIWK, IWORK, LENRWK, RWORK, LENCWK, CWORK)
      CLOSE (UNIT=LIN, STATUS='DELETE')
      CLOSE (UNIT=LREST, STATUS = 'KEEP')
      CLOSE (UNIT=LRCVR, STATUS = 'KEEP')
C
C     Read PSR output to ZOUT array
```

```
C
      REWIND (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE) ISOLUT
      READ (LSAVE) NNP
      KKP = NNP - 1
      READ (LSAVE) EQUIVP, PP, TAUP, FLRTP, VP, QP
      READ (LSAVE) TP, (XIN(K), K=1,KKP)
      READ (LSAVE) (ZOUT(K), K=1,NNP)
      DO 100 I = 1,NNP
       IF (ZOUT(I) .LT. 0.0) ZOUT(I) = 0.0
  100 CONTINUE
      IF (SENS) THEN
         READ (LSAVE)
         DO 200 I = 1,II
            READ (LSAVE) N, FOSCT, (FOSC(K), K = 1,KK)
            WRITE (LSENS) N, FOSCT, (FOSC(K), K = 1,KK)
  200    CONTINUE
      ENDIF
      CLOSE (UNIT=LSENS, STATUS = 'KEEP')
      CLOSE (UNIT=LSAVE, STATUS = 'DELETE')
      WRITE (LOUT, 2018)
 2001 FORMAT ('ENRG')
 2002 FORMAT ('REAC', 1X, A16, 1X, E13.6)
 2003 FORMAT ('PRES', 1X, E13.6)
 2004 FORMAT ('VOL',  1X, E13.6)
 2005 FORMAT ('FLRT', 1X, E13.6)
 2006 FORMAT ('TINL', 1X, E13.6)
 2007 FORMAT ('QLOS', 1X, E13.6)
 2008 FORMAT ('PRNT', 1X, I2)
 2009 FORMAT ('TIME', 1X, I12, 2X, E13.6)
 2010 FORMAT ('TIM2', 1X, I12, 2X, E13.6)
 2011 FORMAT ('TEMP', 1X, E13.6)
 2012 FORMAT ('END')
 2013 FORMAT ('CNTN')
 2014 FORMAT ('RSTR')
 2015 FORMAT ('TGIV')
 2016 FORMAT ('AROP')
 2017 FORMAT (2X, 'Calling PSR for Unit: ', I2)
 2018 FORMAT (2X, 'Returning to the main program.')
 2019 FORMAT ('ATOL', 1X, E13.6)
 2020 FORMAT ('RTOL', 1X, E13.6)
 2021 FORMAT ('TAU', 2X, E13.6)
 2022 FORMAT ('SEN')
      RETURN
      END

      SUBROUTINE PDPFR (IPFR, LOUT, LPIC, LRAD, LCEM, Z, IWORK, RWORK,
     1                  KELE, FLRT, P, VOL, DT, IPIC, IRAD, ICEM, IK1,
     2                  IK2, IK3, LTAU)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
```

```
C     subroutine PDPFR sets up the input file and reads
C
C     inputs:
C
C        IPFR     integer, reactor number
C        LOUT    integer, standard output file
C        LPIC    volume step file for pics
C        LRAD    volume step file for radicals
C        LCEM    volume step file for CEMs
C        Z       array of independent variables
C        IWORK   integer working array
C        RWORK   double working array
C        KELE    character element array
C        FLRT      mass flow rate g/sec
C        P       reactor pressure (atm) then converted to (dynes/cm**3)
C        VOL     reactor volume (cm**3)
C        DT      time step (sec)
C        IPIC    array of PIC indicies
C        IRAD    array of RAD indicies
C        ICEM    array of CEM indicies
C        IK1,IK2,IK3  length of indici arrays
C        LTAU    logical = 1 VOL represents the residence time
C
C     output:
C
C        Z       double array of exit temperature and mass fractions
C
C     working:
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
C
      PARAMETER (ITOL=1, ITASK=1, IOPT=0, RTOL=1.0E-06,
     1           ATOL=1.0E-15, MF=22, KMAX = 250)
      DIMENSION IWORK(*), RWORK(*), Z(*), IPIC(*), IRAD(*), ICEM(*),
     1          X(KMAX)
      CHARACTER KELE(*)*2
      LOGICAL LTAU
      EXTERNAL  FUN
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
      PA_OLD = PA
      PA = P
      P = P*PATM
C
C     initialize values
C
      ISTATE = 1
      NEQ = KK + 1
      LRW = 22 + 9*NEQ + 2*NEQ**2
      NVODE = LENR +1
      IVODE = LENI +1
      LIW = 30 + NEQ
      TT1 = 0.0
```

```
      TT2 = TT1
      VPFR = 0.0
      IF (LTAU) TAU = VOL
C
C     Start Loop
C
C     print volume incremental solution
C
   50 CONTINUE
      IF ((MOD(TT2, 100*DT) .EQ. 0.0) .OR. ((.NOT. LTAU .AND.
     1     VPFR .GE. VOL) .OR. (LTAU .AND. TT2 .GE. TAU))) THEN
         T = Z(1)
         CALL CKYTX (Z(2), IWORK, RWORK, X)
C
         CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
         WRITE (LPIC,3002) IPFR, VPFR, T, (X(IPIC(K)), K=1,IK1)
         WRITE (LRAD,3002) IPFR, VPFR, T, (X(IRAD(K)), K=1,IK2)
         WRITE (LCEM,3002) IPFR, VPFR, T, THC, TCLC,
     1                     (X(ICEM(K)), K=1,IK3)
      ENDIF
C
C     Check for end of volume or end of residence time
C
      IF ((VPFR .GE. VOL) .AND. (.NOT. LTAU)) THEN
         WRITE (LOUT, 3001) IPFR, TT2
         GOTO 100
      ENDIF
      IF (LTAU .AND. TT2 .GE. TAU) THEN
         WRITE (LOUT, 3001) IPFR, TT2
         GOTO 100
      ENDIF
C
C     integrate to next output TT2 + DT
C
      TT2 = TT2 + DT
      IF (LTAU .AND. TT2 .GE. TAU) TT2 = TAU
C     WRITE (LOUT,317) NEQ, TT1, TT2, ITOL, RTOL, ATOL, ITASK, ISTATE,
C    1                 IOPT, NVODE, LRW, IVODE, LIW, MF
C317 FORMAT(//,'NEQ ',I4, 4X, 'TT1 ',  G9.3, 4X, 'TT2 ', G9.3, 4X, /,
C    1      'ITOL ', I4, 3X, 'RTOL ', G9.3, 3X, 'ATOL ', G9.3, 3X, /,
C    2      'ITASK ', I4, 2X, 'ISTATE ', I4, 6X, 'IOPT ', I4, 7X, /,
C    3      'NVODE ', I4, 2X, 'LRW    ', I4, 6X, 'IVODE ', I4, 6X,/ ,
C    4      'LIW   ', I4, 2X, 'MF     ', I4)
      CALL DVODE
     *            (FUN, NEQ, Z, TT1, TT2, ITOL, RTOL, ATOL, ITASK,
     1             ISTATE, IOPT, RWORK(NVODE), LRW, IWORK(IVODE),
     2             LIW, JAC, MF, RWORK, IWORK)
C
C     WRITE (*,*) 'EMERGED WITH ISTATE = ', ISTATE
      IF (ISTATE .LE. -2) THEN
         IF (ISTATE .EQ. -1) THEN
            ISTATE = 2
            GO TO 100
         ELSE
            WRITE (LOUT,*) ' ISTATE=',ISTATE
            STOP
         ENDIF
```

```
      ENDIF
C
C     Volume increment
C
      CALL CKRHOY (P, Z(1), Z(2), IWORK, RWORK, RHO)
      VPFR = VPFR + DT*FLRT/RHO
C
C
      GOTO 50
  100 CONTINUE
      P = PA
      PA = PA_OLD
 3001 FORMAT (5X, 'PFR ', I1, ' TAU (SEC) = ', F8.4)
 3002 FORMAT (2X, 'PFR',I1, 4X, F11.0, F11.4, 30E19.4)
      RETURN
      END
C
      SUBROUTINE FUN (N, TIME, Z, ZP, RPAR, IPAR)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Function Subroutine provided for DVODE.
C
C     Solves a constant pressure plug flow reactor with a constant heat
C     transfer coefficient per unit volume - UAV
C
C     Input:
C             N   Number indepent variables (not used)
C             TIME      Not used
C             Z         Array of indepent variables
C             RPAR      Working real (double) array
C             IPAR      Working integer array
C
C     Output:
C             ZP  first time derivative of Z
C
C     Common:
C             TA  Double, ambient temperature
C             UAV       Double, heat transfer coeff/vol
C                       (ergs/(sec*degK*cc))
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
C      IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
C
      DIMENSION Z(*), ZP(*), RPAR(*), IPAR(*)
C
C     Variables in Z are:  Z(1)   = T
C                          Z(K+1) = Y(K)
C
C     Call CHEMKIN subroutines
C
      P = PATM * PA
      CALL CKRHOY (P, Z(1), Z(2), IPAR, RPAR, RHO)
```

```
      CALL CKCPBS (Z(1), Z(2), IPAR, RPAR, CPB)
      CALL CKWYP  (P, Z(1), Z(2), IPAR, RPAR, RPAR(NWDOT))
      CALL CKHMS  (Z(1), IPAR, RPAR, RPAR(NH))
C
C     Form governing equation
C
      SUM = 0.0
      DO 100 K = 1, KK
         H    = RPAR(NH    + K - 1)
         WDOT = RPAR(NWDOT + K - 1)
         WT   = RPAR(NWT   + K - 1)
         ZP(K+1) = WDOT * WT / RHO
         SUM = SUM + H * WDOT * WT
 100  CONTINUE
      IF (UAV .NE. -1) THEN
         ZP(1) = -SUM / (RHO*CPB) - UAV*(Z(1)-TA)/(CPB*RHO)
      ELSE
         ZP(1) = 0
      ENDIF
C
      RETURN
      END
C
C
      SUBROUTINE GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C     Groups mole fractions into total hydrocarbons (THC), total
C     chlorinated hydrocarbons (TCLC).  THC counts each carbon atom in
C     a hydrocarbon.  TCLC counts chlorine atoms in each chlorocarbon.
C
C
C     Assumes the existence of exactly 6 elements which must include
C     C, CL, and H.  If the mechanism has more than 6 elements, modify
C     the parameter MM accordingly.
C
C     Input:
C               KK              (integer) number of species
C               X               (double) mole fraction array
C               KELE            (character) array of element strings
C               IWORK           (integer) working array
C               RWORK           (double) working array
C
C     Output:
C               THC, TCLC
C                               (double) summed mass frac
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (MM =6, KMAX = 250)
      DIMENSION X(*), IWORK(*), RWORK(*), NCE(MM,KMAX), IELE(MM)
      CHARACTER KELE(*)*2
C
C     initialize variables, and get element matrix
```

```
C
      THC = 0
      TCLC = 0
      CALL CKNCF (MM, IWORK, RWORK, NCE)
C
C     find indices of key elements
C
      CALL CKCOMP ('C',KELE,MM,IELE(1))
      CALL CKCOMP ('H',KELE,MM,IELE(2))
      CALL CKCOMP ('CL',KELE,MM,IELE(3))
C
C     count species in various categories for entire stream
C
      DO 100 K = 1,KK
         IC = NCE(IELE(1),K)
         IH = NCE(IELE(2),K)
         ICL = NCE(IELE(3),K)
         IF (IC .GE. 1 .AND. IH .GE. 1) THC = THC + X(K)*IC
         IF (IC .GE. 1 .AND. ICL .GE. 1) TCLC = TCLC + X(K)*ICL
  100 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE STREAM (LIN, LOUT, KEYWORD, KSYM, KK, ICKWRK, RCKWRK,
     1                   UNITS, FLRT, X, Z)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Subroutine to read mole fractions and temperature of an
C     input stream.  Mole fractions are normalized and converted to
C     mass fractions.  Input file is read until KEYWORD is encountered
C     or an END is reached.
C
C     Input:
C                LIN      input fortran unit
C                LOUT     output fortran unit
C                KEYWORD  character keyword of particular stream
C                KSYM     character array containing all species
C                KK       total number of species
C                UNITS    logical, .TRUE. = scfh, degF, cf
C
C     Output:
C                X        array of normalized mole fractions
C                Z        array of temperature and mass fractions
C                FLRT     double, flow rate grams/sec
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
      DIMENSION X(*), Z(*), ICKWRK(*), RCKWRK(*),
     1          VALUE(3)
      CHARACTER KSYM(*)*16, LINE*80, KEYWORD*(*), LINE2*80
      LOGICAL IERR, KERR, UNITS
      DATA KERR/.FALSE./
C
```

```
C     Check keyword and parse heading line
C
      REWIND (LIN)
  50  CONTINUE
      READ (LIN, '(A)', END=300) LINE
      IF (INDEX (LINE, 'END ') .EQ. 1) THEN
        WRITE (LOUT, 8001) KEYWORD
        STOP
      ENDIF
      IF (INDEX (LINE, 'REM ') .EQ. 1) GOTO 50
C
C     Check keyword
C
      IF (INDEX (LINE, KEYWORD) .EQ. 1) THEN
        WRITE (LOUT, '(5X,A65)') LINE
        CALL CKNPAR (LINE, 3, LOUT, LINE2, ISTART, KERR)
        IF (KERR) THEN
           WRITE (LOUT, *) 'STREAM:  ERROR IN CKNPAR'
           STOP
        ENDIF
        CALL CKXNUM (LINE2, 3, LOUT, NVAL, VALUE, KERR)
        IF (KERR) THEN
         WRITE (LOUT, *) 'STREAM:  ERROR IN CKXNUM'
         STOP
        ENDIF
        IF (NVAL .NE. 3) THEN
          WRITE (LOUT, *) 'STREAM:  INCORRECT FORMAT IN HEADING LINE'
          STOP
        ENDIF
        NOSPEC = INT (VALUE(1))
C
C     Convert Temperature from degF or degC to K and molar flow rate
C     from scfh or std cc/sec to mol/s
C
        IF (NOSPEC .NE. 0) THEN
           IF (UNITS) THEN
              Z(1) = (VALUE(2) + 459.67)*5/9
              FLRT = VALUE(3) * 1.19530/3600.0
           ELSE
              Z(1) = VALUE(2) + 273.15
              FLRT = VALUE(3) * 4.46158e-5
           ENDIF
C
C       Initialize non-zero moles
C
           DO 100 I = 1, NOSPEC
             READ (LIN, '(A)') LINE
             CALL CKSNUM (LINE, 1, LOUT, KSYM, KK, KNUM, NVAL,
     1                    VAL, KERR)
             IF (KERR) THEN
               WRITE (LOUT, *) 'ERROR READING MOLES'
               STOP
             ENDIF
             X(KNUM) = VAL
  100      CONTINUE
C
C       Normalize the mole fractions
```

```
C
           XTOT = 0.00
           DO 150 K = 1, KK
             XTOT = XTOT + X(K)
  150      CONTINUE
           DO 200 K = 1, KK
             X(K) = X(K) / XTOT
  200      CONTINUE
C
C       Initial conditions and mass fractions
C
           CALL CKXTY (X, ICKWRK, RCKWRK, Z(2))
C
C       Convert molar flow rate to mass flow rate
C
           CALL CKMMWX(X, ICKWRK, RCKWRK, WTM)
           FLRT = FLRT*WTM
C
C
         ELSE
           FLRT = 0
         ENDIF
         RETURN
       ENDIF
       GOTO 50
  300 WRITE (LOUT, 8002)
       STOP
 8001 FORMAT ('KEY WORD - ' , A7, ' - NOT FOUND.  PROGRAM STOPPED.')
 8002 FORMAT ('UNEXPECTED END TO INPUT FILE ENCOUNTERED.')
       END
C
C
      SUBROUTINE PARSER (LIN, LOUT, KEYWORD, NODAT, RVAL1, RVAL2,
     1                   RVAL3, RVAL4)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     Subroutine to read selected from the input file.  Example:
C     Data is in the form:
C
C     KEYWORD  RVAL1  RVAL2  RVAL3  RVAL4
C
C     or:
C
C     REM  comments .....
C
C     PARSER will return NODAT real values after KEYWORD.  If KEYWORD
C     is not found error is returned.  If "REM" is found the line
C     is skipped and the following line is automatically read.
C
C     Input:
C                LIN     integer, input fortran unit
C                LOUT    integer, output fortran unit
C                KEYWORD character string keyword to be checked
C                NODAT   integer, number of data items in line (1-4)
C
C     Output:
C                RVAL1   double, first real value returned
```

```
C               RVAL2   double, second real value returned
C               RVAL3   double, third real value returned
C               RVAL4   double, forth real value returned
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
      DIMENSION VALUE(4)
      CHARACTER LINE*80, KEYWORD*(*), LINE2*80
      LOGICAL KERR
      DATA KERR /.FALSE./
C
C     Read Line
C
      REWIND (LIN)
  50  CONTINUE
      READ (LIN, '(A)', END=200) LINE
      IF (INDEX (LINE, 'END ') .EQ. 1) THEN
        WRITE (LOUT, 8001) KEYWORD
        STOP
      ENDIF
      IF (INDEX (LINE, 'REM ') .EQ. 1) GOTO 50
C
C     Check keyword
C
      IF (INDEX (LINE, KEYWORD) .EQ. 1) THEN
        WRITE (LOUT, '(5X,A50)') LINE
C
C     Parse Line
C
        CALL CKNPAR (LINE, NODAT, LOUT, LINE2, ISTART, KERR)
        IF (KERR) THEN
          WRITE (LOUT, *) 'ERROR IN CKNPAR'
          STOP
        ENDIF
        CALL CKXNUM (LINE2, NODAT, LOUT, NVAL, VALUE, KERR)
        IF (KERR) THEN
          WRITE (LOUT, *) 'ERROR IN CKXNUM'
          STOP
        ENDIF
        RVAL1 = VALUE(1)
        RVAL2 = VALUE(2)
        RVAL3 = VALUE(3)
        RVAL4 = VALUE(4)
        RETURN
      ENDIF
      GOTO 50
 200  WRITE (LOUT, 8002)
      STOP
8001  FORMAT ('KEY WORD - ' , A7, ' - NOT FOUND.  PROGRAM STOPPED.')
8002  FORMAT ('UNEXPECTED END TO INPUT FILE ENCOUNTERED.')
      END

      SUBROUTINE MIX(KK, FLRT1, FLRT2, Z1, Z2, ICKWRK, RCKWRK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
```

```
C     This subroutine calculates the mixed temperature of two streams
C     along with the mixed mass flow rate and mass fractions
C
C     Input:
C               FLRT1/2 flow rate of stream 1/2
C               Z1/Z2 temperature and mass fractions of stream 1/2
C
C     Output:   Z1, FLRT1
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)
      PARAMETER (KMAX = 250)
      DIMENSION Z1(*), Z2(*), HMS1(KMAX), HMS2(KMAX), ZM(KMAX),
     1          ICKWRK(*), RCKWRK(*)
C
C     Enthalpies and mean heat capacities of each stream are found
C
      CALL CKHMS(Z1(1), ICKWRK, RCKWRK, HMS1)
      CALL CKHMS(Z2(1), ICKWRK, RCKWRK, HMS2)
      CALL CKCPBS(Z1(1), Z1(2), ICKWRK, RCKWRK, CPAV1)
      CALL CKCPBS(Z2(1), Z2(2), ICKWRK, RCKWRK, CPAV2)
C
C     Add masses and enthalpy of streams
C
      DATA LOUT/6/
C      WRITE (LOUT,'(2X,A10,2X,2F8.0)')'Init Temp',Z1(1),Z2(1)
      HT = 0
      FLRTM = FLRT1 + FLRT2
      DO 100 I = 1, KK
        HT = HT + Z1(I+1)*HMS1(I)*FLRT1 + Z2(I+1)*HMS2(I)*FLRT2
        ZM(I+1) = (Z1(I+1)*FLRT1 + Z2(I+1)*FLRT2)/FLRTM
  100 CONTINUE
C
C     Calculate first guess for mix temperature
C
      ZM(1) = (FLRT1*CPAV1*Z1(1) + FLRT2*CPAV2*Z2(1))/
     1        (FLRT1*CPAV1 + FLRT2*CPAV2)
      CALL CKCPBS(ZM(1), Z1(2), ICKWRK, RCKWRK, CPAV1I)
      CALL CKCPBS(ZM(1), Z2(2), ICKWRK, RCKWRK, CPAV2I)
      CPAV1 = (CPAV1 + CPAV1I)/2
      CPAV2 = (CPAV2 + CPAV2I)/2
      ZM(1) = (FLRT1*CPAV1*Z1(1) + FLRT2*CPAV2*Z2(1))/
     1        (FLRT1*CPAV1 + FLRT2*CPAV2)
C      WRITE (LOUT,'(2X, A10,2X,F8.3,E13.6)')'1st guess',ZM(1),HT
C
C     Interpolate to get final mixing temperature
C
      T_HIGH = 5000
      T_LOW = 300
      DO 200 J = 0,100
        CALL CKHBMS(ZM(1), ZM(2), ICKWRK, RCKWRK, HAVE)
        HTG = HAVE*FLRTM
        DIFF = (HT - HTG)/ABS(HT)
        IF (ABS(DIFF) .LE. 0.0001) GOTO 400
        ZM(1) = (1. + 0.1*DIFF)*ZM(1)
C       WRITE (LOUT,'(2X,A10,2X,F8.3,E13.6,E13.6)')'next guess',
```

```
C     1         ZM(1),HTG,DIFF
  200 CONTINUE
C
C     bisection technique if newton fails
C
      DO 300 J = 0,100
        ZM(1) = (T_HIGH + T_LOW)/2
        CALL CKHBMS(ZM(1), ZM(2), ICKWRK, RCKWRK, HAVE)
        HTG = HAVE*FLRTM
        DIFF = (HT - HTG)/ABS(HT)
        IF (DIFF .LT. 0.0) THEN
           T_HIGH = ZM(1)
        ELSE
           T_LOW = ZM(1)
        ENDIF
        IF (ABS(DIFF) .LE. 0.0001) GOTO 400
C       WRITE (LOUT,'(2X,A10,2X,F8.3,E13.6,E13.6)')'next guess',
C     1         ZM(1),HTG,DIFF
  300 CONTINUE
      WRITE(LOUT,350)
  350 FORMAT(1X,'TEMPERATURE NOT CONVERGED IN MIX')
      STOP
  400 CONTINUE
      FLRT1 = FLRTM
      DO 500 I = 1, KK+1
        Z1(I) = ZM(I)
  500 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE SYMINDICIES (LIN, LOUT, KK, KSYM, KEYWORD,
     1                        INDICIES, IDX)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     SYMINDICIES returns the indicies of the selected species in the
C     line after the KEYWORD.
C
C     Inputs:
C           LIN   integer, input file unit
C           LOUT    integer, output file unit
C           KK    integer, number of species
C           KSYM  character, array of species strings
C           KEYWORD character, prefix keyword
C
C     Output:
C           INDICIES integer, array of indicies of selected species
C           IDX    integer, number of selected species
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
      PARAMETER (KMAX=250)
      DIMENSION INDICIES(*)
      CHARACTER KSYM(*)*16, LINE*80, KEYWORD*(*)
```

```
      LOGICAL KERR
      DATA KERR /.FALSE./
C
C     Read Line
C
      REWIND (LIN)
   50 CONTINUE
      READ (LIN, '(A)', END=200) LINE
      IF (INDEX (LINE, 'END ') .EQ. 1) THEN
        WRITE (LOUT, 8001) KEYWORD
        RETURN
      ENDIF
      IF (INDEX (LINE, 'REM ') .EQ. 1) GOTO 50
C
C     Check keyword
C
      IF (INDEX (LINE, KEYWORD) .EQ. 1) THEN
        WRITE (LOUT, '(5X,A50)') LINE
      READ (LIN, '(A)', END = 200) LINE
        WRITE (LOUT, '(5X,A50)') LINE
        CALL CKCRAY (LINE, KK, KSYM, LOUT, KMAX, INDICIES, IDX, KERR)
        IF (KERR) THEN
          WRITE(LOUT, 8003) KEYWORD
          STOP
        ENDIF
        RETURN
      ENDIF
      GOTO 50
  200 WRITE (LOUT, 8002)
      STOP
 8001 FORMAT ('KEY WORD - ' , A7, ' - NOT FOUND.')
 8002 FORMAT ('UNEXPECTED END TO INPUT FILE ENCOUNTERED.')
 8003 FORMAT ('ERROR IN CKCRAY FOR LINE FOLLOWING KEYWORD - ', A10)
      END
C
C
      FUNCTION EQUIV_RATIO (KK, YF, YA, YK, YI, FLRTF, FLRTA, FLRTK,
     1                      FLRTI, KELE, KSYM, IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Returns fuel/air equivalence ratio of input streams.  Based on
C     complete combustion products of CO2, H2O, HCL, and N2.
C
C     Inputs:
C     KK          number of species
C     YF,YA,YK,YI       arrays of mass fractions of 4 streams
C     FLRT?       flow rates (g/s) of streams
C       KELE            character array of element symbols
C     KSYM        character arrays of species symbols
C     IWORK,RWORK integer and double work arrays
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (MM =6, KMAX = 250)
      DIMENSION YF(*), YA(*), YK(*), YI(*), IWORK(*), RWORK(*),
```

```
     1          NCE(MM,KMAX), Y(KMAX), X(KMAX), IELE(MM)
      CHARACTER KELE(MM)*2, KSYM(*)*16
C
C     initialize variables, and get element matrix
C
      TCL = 0
      TO = 0
      TC = 0
      TH = 0
      CALL CKNCF (MM, IWORK, RWORK, NCE)
C
C     find indices of key elements
C
      CALL CKCOMP ('C',KELE,MM,IELE(1))
      CALL CKCOMP ('H',KELE,MM,IELE(2))
      CALL CKCOMP ('CL',KELE,MM,IELE(3))
      CALL CKCOMP ('O',KELE,MM,IELE(4))
C
C     Combine Streams
C
      DO 50 K = 1,KK
         Y(K) = (YF(K)*FLRTF + YA(K)*FLRTA + YK(K)*FLRTK +YI(K)*FLRTI)/
     1          (FLRTF + FLRTA + FLRTK + FLRTI)
  50  CONTINUE
C
C     Convert to Mole Fractions
C
      CALL CKYTX (Y, IWORK, RWORK, X)
C
C     count species in various categories for entire stream on 1 mol
basis
C
      DO 100 K = 1,KK
         IC = NCE(IELE(1),K)
         IH = NCE(IELE(2),K)
         ICL = NCE(IELE(3),K)
         IO = NCE(IELE(4),K)
         TCL = TCL + X(K)*ICL
         TC = TC + X(K)*IC
         TH = TH + X(K)*IH
         TO = TO + X(K)*IO
 100  CONTINUE
      CALL CKCOMP ('O2', KSYM, KK, IO2)
      TO2 = X(IO2)
C
C     subtract free oxygen counted in previous loop
C
      TO = TO - 2*TO2
C
C     Find required O2 and PHI
C
      REQUIRED_O2 = TC + 0.25*(TH -TCL) - 0.5*TO
      EQUIV_RATIO = REQUIRED_O2/TO2
      RETURN
      END
C
C
```

```
      SUBROUTINE IMPINGER (IPSR, RESTART, NN, Z, KSYM, TAU, TEMP,
     1                     IWORK, RWORK, ZOUT)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Cools a sample stream in a water vapor saturated PSR at TEMP
C     for mean residence time TAU.  Uses a generic flow rate of 1 g/s.
C     Water vapor is first increased to reflect an equilibrium partial
C     pressure (but no greater than a mole fraction of 0.75) and HCL
C     is removed before entering the PSR.  Water vapor is removed from
C     the effluent.
C
C     Inputs:
C           IPSR    integer of recovery/restart file
C           KK    integer, number of species
C           NN    integer, number of reactions
C           Z     double array, mass fractions of stream
C           KSYM  character array, specie symbols
C           TAU   double, residence time of PSR (sec)
C           TEMP  double, temperature of PSR (K)
C           IWORK integer, working array
C           RWORK double, working array
C
C     Output:
C           ZOUT  double array, mass fractions of effulent
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
C
      PARAMETER (KMAX = 250)
      DIMENSION Z(*), IWORK(*), RWORK(*), X(KMAX), ZOUT(KMAX)
      CHARACTER KSYM(*)*16
      LOGICAL RESTART
      COMMON /RCONS/ PATM, RU, TA, PA, UAV
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
C
C     indicies of H2O and HCL
C
      CALL CKCOMP ('H2O', KSYM, KK, IH2O)
      CALL CKCOMP ('HCL', KSYM, KK, IHCL)
      CALL CKYTX (Z(2), IWORK, RWORK, X)
      FLRT = 1
C
C     Molar flow rate
C
      FLRTM = 0
      DO 50 I = 1, KK
         FLRTM = FLRTM + FLRT * Z(I+1)/RWORK(NWT+I-1)
   50 CONTINUE
      FLRTM_OLD = FLRTM
C
C     find water vapor pressure and mole frac
C
      P_H2O = EXP(18.3036 - 3816.44/(TEMP-46.13))
      P_H2O = P_H2O/760
```

```
      X_H2O = P_H2O/PA
      X_H2O_MAX =  0.75
      X_H2O = MIN(X_H2O, X_H2O_MAX)
C
C     adjust molar flow rate from mole fraction of water
C
      FLRTM = FLRTM*(1-X(IH2O)-X(IHCL))/(1-X_H2O)
C
C     adjust mass flow rate
C
      FLRT = FLRT + RWORK(NWT+IH2O-1)*(FLRTM*X_H2O -
     1       FLRTM_OLD*X(IH2O)) - RWORK(NWT+IHCL-1)*X(IHCL)
C
C     find the volume
C
      VOL = FLRTM*RU*TEMP*TAU/(PA*PATM)
C
C     readjust X array
C
      DO 100 I = 1, KK
         IF (I .EQ. IH2O) THEN
            X(I) = X_H2O
         ELSEIF (I .EQ. IHCL) THEN
            X(I) = 0.0
         ELSE
            X(I) = X(I)*FLRTM_OLD/FLRTM
         ENDIF
  100 CONTINUE
C
C     call PSR
C
      CALL PDPSR (IPSR, KK, NN, Z(1), X, KSYM, FLRT, PA, VOL, .TRUE.,
     1            0.0, TEMP, RESTART, .FALSE., .FALSE., ZOUT)
C
C     Remove water vapor
C
      CALL CKYTX (ZOUT(2), IWORK, RWORK, X)
      FLRTM = 0.0
      DO 150 I = 1, KK
         FLRTM = FLRTM + FLRT*Z(I)/RWORK(NWT+I-1)
  150 CONTINUE
      FLRTM_OLD = FLRTM
      FLRTM = FLRTM*(1-X(IH2O)-X(IHCL))
      DO 200 I = 1, KK
         IF (I .EQ. IH2O) THEN
            X(I) = 0.0
         ELSEIF (I .EQ. IHCL) THEN
            X(I) = 0.0
         ELSE
            X(I) = X(I)*FLRTM_OLD/FLRTM
         ENDIF
  200 CONTINUE
      CALL CKXTY (X, IWORK, RWORK, ZOUT(2))
      RETURN
      END
C
      SUBROUTINE SAVER (KK, Z, X, SERIES, FILE1, FILE2, FILE3, FILE4,
```

```
     1                  KSYM, KELE, IWORK, RWORK, IPIC, IRAD, ICEM,
     2                  IK1, IK2, IK3, PHI, PHI_B)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     SAVER saves mole fractions of designated species in formated text
C     files and saves both mole and mass fractions of all species in
C     an unformatted binary file.
C
C     Input:
C
C           KK    integer, number of species
C           Z     double array, temp and mass fractions
C           X     double array, mole fractions
C           SERIES      logical, .TRUE. overwrites and makes header
C           FILE? character, output files
C           KSYM  character array, specie symbols
C           KELE  character array, element symbols
C           IWORK integer array, working array
C           RWORK double array, working array
C           IPIC  integer arrray, indicies of selected species
C           IRAD,ICEM   same
C           IK?   integer, number of indicies in respective
C                 arrays
C           PHI   double, overall fuel/air equivalence ratio
C           PHI_B double, burner fuel/air equivalence ratio
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
C
      PARAMETER (LFILE1 = 31, LFILE2 = 32, LFILE3 = 33, LFILE4 =34,
     1           LOUT = 6)
      DIMENSION Z(*), X(*), IWORK(*), RWORK(*), IPIC(*), IRAD(*),
     1          ICEM(*)
      CHARACTER KSYM(*)*16, KELE(*)*2, FILE1*(*), FILE2*(*), FILE3*(*),
     1          FILE4*(*), LINE*80
      LOGICAL SERIES
C
C     Grouped CEMs
C
      CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
C
C     open output files
C
      OPEN (UNIT = LFILE1, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE1)
      OPEN (UNIT = LFILE2, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE2)
      OPEN (UNIT = LFILE3, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE3)
      OPEN (UNIT = LFILE4, FORM='UNFORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE4)
      IF (SERIES) THEN
           WRITE (LFILE1,7510) 'TEMPERATURE', (KSYM(IPIC(K)), K=1,IK1)
           WRITE (LFILE2,7510) 'TEMPERATURE', (KSYM(IRAD(K)), K=1,IK2)
           WRITE (LFILE3,7510) 'TEMPERATURE', 'PHI', 'PHI_BURNER',
```

```
     1                         'THC', 'TCLC', (KSYM(ICEM(K)), K=1,IK3)
            WRITE (LFILE4) (KSYM(K), K=1,KK)
            WRITE (LFILE4)
            WRITE (LFILE4)
      ELSE
            DO 200 I = 1,100
               READ (LFILE1,7516,END=220) LINE
               READ (LFILE2,7516,END=220) LINE
               READ (LFILE3,7516,END=220) LINE
               READ (LFILE4,END=220)
               READ (LFILE4,END=220)
               READ (LFILE4,END=220)
 200        CONTINUE
            WRITE (LOUT,*) FILE1, "NO END FOUND!"
            STOP
 220        CONTINUE
            BACKSPACE (LFILE1)
      ENDIF
      WRITE (LFILE1,7512) Z(1), (X(IPIC(K)), K=1,IK1)
      WRITE (LFILE2,7512) Z(1), (X(IRAD(K)), K=1,IK2)
      WRITE (LFILE3,7513) Z(1), PHI, PHI_B, THC, TCLC,
     1                    (X(ICEM(K)), K=1,IK3)
      WRITE (LFILE4) Z(1), PHI, PHI_B, THC, TCLC
      WRITE (LFILE4) (X(K), K=1,KK)
      WRITE (LFILE4) (Z(K), K=1,KK+1)
 230  CONTINUE
      IF (SERIES) ENDFILE (LFILE1)
      IF (SERIES) ENDFILE (LFILE2)
      IF (SERIES) ENDFILE (LFILE3)
      IF (SERIES) ENDFILE (LFILE4)
      CLOSE (LFILE1, STATUS ='KEEP')
      CLOSE (LFILE2, STATUS='KEEP')
      CLOSE (LFILE3, STATUS='KEEP')
      CLOSE (LFILE4, STATUS='KEEP')
7510  FORMAT (2X, 30(' "',A16,'"'))
7512  FORMAT (4X, F11.0, 3X, 30(5X,E14.6))
7513  FORMAT (4X, F11.0, 2X, 2(5X,F14.6), 30(5X, E14.6))
7516  FORMAT (A80)
      RETURN
      END

      SUBROUTINE TOP30 (KK, POINT, KSYM, X, ITOP)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     TOP30  Returns the indicies of the top 30 species
C     and saves symbol and mole frac in file "top30"
C
C     Input:
C           KK    integer, number of species
C           POINT character string, sample location
C           KSYM  character array of specie symbols
C           X     double array of mole fractions
C
C     Output:
C
C           ITOP  integer array of indicies
```

```
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (LTOP = 38)
      DIMENSION X(*),  ITOP(*)
      CHARACTER KSYM(*)*(*), POINT*(*), LINE*80
      LOGICAL SKIP
C
C
      DO 300 I=1,30
         RHOLD=0
         DO 200 K=1,KK
            SKIP = .FALSE.
            DO 100 J=1,I-1
               IF (ITOP(J) .EQ. K) SKIP = .TRUE.
  100       CONTINUE
            IF (.NOT. SKIP) THEN
               IF (X(K) .GT. RHOLD) THEN
                  IHOLD = K
                  RHOLD = X(K)
               ENDIF
            ENDIF
  200    CONTINUE
         ITOP(I) = IHOLD
  300 CONTINUE
      OPEN (UNIT = LTOP, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = 'top30')
      DO 400 I = 1,10000
         READ (LTOP,7516,END=420) LINE
  400 CONTINUE
      WRITE (LOUT,*)  "TOP30:  NO END FOUND!"
      STOP
  420 CONTINUE
      WRITE (LTOP,7510) POINT, (KSYM(ITOP(K)), K=1,30)
      WRITE (LTOP,7512) (X(ITOP(K)), K=1,30)
      CLOSE (LTOP, STATUS ='KEEP')
 7510 FORMAT (2X, 31(' "',A16,'"'))
 7512 FORMAT (19X, 30(5X,E14.6))
 7516 FORMAT (A80)
      RETURN
      END


      SUBROUTINE ENERGY_BAL (TEMP, Y, FLRT, POINT, IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Subroutine ENERGY_BAL logs the specific enthalpy, temperature,
C     flowrate, and total enthalpy of a stream at a designated point.
C
C     Input:
C           TEMP  double, temperature of the stream
C           Y     double array, mass fractions
C           FLRT  double, flow rate in g/s
C           POINT character, location of the stream
C           IWORK integer working array
```

```
C           RWORK double working array
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (LERG = 12)
      DIMENSION Y(*), IWORK(*), RWORK(*)
      CHARACTER POINT*(*), LINE*80
      CALL CKHBMS (TEMP, Y, IWORK, RWORK, HAVE)
      HRT = HAVE * FLRT
      OPEN (UNIT = LERG, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = 'energy_bal')
      DO 100 I = 1,10000
         READ (LERG,7502,END=120) LINE
  100 CONTINUE
      WRITE (LOUT,*)  "ENERGY_BAL:  NO END FOUND!"
      STOP
  120 CONTINUE
      WRITE (LERG, 7501) POINT, TEMP, HAVE, FLRT, HRT
      CLOSE (LERG, STATUS ='KEEP')
 7501 FORMAT (A10,F7.0,' K ',E9.3,' erg/g ',E9.3,' g/s ',E9.3,' erg/s')
 7502 FORMAT (A80)
      RETURN
      END
C
      FUNCTION UA_FIND (KK, II, TIN, TOUT, TA, X, KSYM, FLRT, PA,
     1                  TAU, IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Returns estimate of UA (ergs/(sec*K)).  Uses three fixed
C     temperature PSRs each with 1/3 of the residence time TAU
C     temperatures are 1) input 2) arithmetic mean 3) exit
C
C                Q = UA(TLM)      where Q = Hin - Hout
C
C     TLM is the co-current log mean between the input temperature, the
C     output temperature and the ambient temperature.
C
C     Inputs:
C     KK           integer, number of species
C     II           integer, number of reactions
C     TIN          double, inlet temperature (K)
C     TOUT         double, desired outlet temperature (K)
C     TA           double, ambient temperature (K)
C     X            double array(KK), mole fractions
C     KSYM         character array of specie symbols
C     FLRT         double, flow rate (g/s) of inlet stream
C     PA           double, ambient pressure (atm)
C     TAU          double, residence time of the reactor
C     IWORK,RWORK integer and double work arrays
C
C     Output:
C     UA           double, heat transfer coefficient (erg/s*K)
C
C     Scratch:
C     XSCT         double array(K), outlet mole fractions
```

```
C     ZSCT        double array(K+1), outlet temp, mass frac
C     HAVE_IN     double, average input enthalpy
C     HAVE_OUT    double, average output enthalpy
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (LOUT=6, KMAX = 250)
      DIMENSION X(*), IWORK(*), RWORK(*), XSCT(KMAX), ZSCT(KMAX)
      CHARACTER KSYM(*)*16
C
C     Find Mass Fractions and Mean Input Enthalpy
C
      DO 200 K = 1, KK
         XSCT(K) = X(K)
  200 CONTINUE
      ZSCT(1) = TIN
      CALL CKXTY(X, IWORK, RWORK, ZSCT(2))
      CALL CKHBMS(ZSCT(1), ZSCT(2), IWORK, RWORK, HAVE_IN)
C
C     A series of 3 PSRs to approximate PFR output composition
C
      TAU_I = TAU/3
      TEMP = TIN
      CALL PDPSR(97, KK, II, ZSCT(1), XSCT, KSYM, FLRT, PA, TAU_I,
     1           .TRUE., 0, TEMP, .FALSE., .TRUE., .FALSE., ZSCT)
      CALL CKYTX(ZSCT(2), IWORK, RWORK, XSCT)
      TEMP = (TIN + TOUT)/2
      CALL PDPSR(97, KK, II, ZSCT(1), XSCT, KSYM, FLRT, PA, TAU_I,
     1           .TRUE., 0, TEMP, .FALSE., .TRUE., .FALSE., ZSCT)
      CALL CKYTX(ZSCT(2), IWORK, RWORK, XSCT)
      TEMP = TOUT
      CALL PDPSR(97, KK, II, ZSCT(1), XSCT, KSYM, FLRT, PA, TAU_I,
     1           .TRUE., 0, TEMP, .FALSE., .TRUE., .FALSE., ZSCT)
C
C     Output enthalpy
C
      CALL CKHBMS(ZSCT(1), ZSCT(2), IWORK, RWORK, HAVE_OUT)
C
C     Net enthalpy loss
C
      QLOSS = FLRT*(HAVE_IN - HAVE_OUT)
C
C     Log mean temperaure and estimated UA
C
      TLM = (TIN - TOUT)/LOG((TIN - TA)/(TOUT - TA))
      UA_FIND = QLOSS/TLM
      RETURN
      END
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     END OF PROGRAM
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

D.2 PaSR Version 3.0

```
C
C     PROGRAM PASR
C
C
C     Solves an isobaric partially stirred reactor.  Reactor exchanges
C     concentrations with reactor average based on a mixing frequency
C     parameter.
C
C     VERSION 3.3.1 Last Updated June 13, 1998:
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)
      PARAMETER (LENIWK=500000, LENRWK=500000, LENCWK=50000, NK=30,
     1           NLMAX=55, LIN=5, LOUT=6, LINKCK=25, KMAX=250,
     2           LPIC=18, LRAD=19, LCEM=20, LPROF=21, LBPROF=22,
     3           NITER = 500, TTOL = 1E-3, ATOL = 1E-13)
C
      DIMENSION IWORK(LENIWK), RWORK(LENRWK), X(KMAX), Z(KMAX),
     1          XF(KMAX), ZF(KMAX), XA(KMAX), ZA(KMAX), ZOUT(KMAX),
     2          ZAVE(KMAX), IPIC(KMAX), IRAD(KMAX), FN(KMAX),
     3          ICEM(KMAX), ZPSR(KMAX), XN(KMAX), ZN(KMAX),
     4          ZD(KMAX), XD(KMAX), IPROF(KMAX)
      CHARACTER CWORK(LENCWK)*16, KSYM(KMAX)*16, KELE(6)*2, START*12,
     1          FINISH*12
      LOGICAL KERR, IERR, TGIV, SERIES, REST, LPHI, LTAU, LFREQ, LDIL,
     1        LNEWT, LSENS, MODEL, SKIP, LUA, LCONV, LPROBE
      EXTERNAL FUN
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV, PQUAV, TCOOL, BM, CM
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
      COMMON /PASRCONS/ ZAVE, FREQ, TAU, FN
C
      DATA KERR/.FALSE./, KSYM/KMAX*' '/, TEMP_RMS /0.0/,
     1     LCONV/.FALSE./, LPROBE/.TRUE./
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Initialize files and constants
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      OPEN (UNIT=LOUT, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='pasr.out')
      WRITE (LOUT, 15)
   15 FORMAT (
     1/'  PASR:   Version 3.4'
     2/'          Charles Bass, June 2, 2001'
     3/'          DOUBLE PRECISION')
C     Open the CHEMKIN LINK file
C
      OPEN (LINKCK, FORM='UNFORMATTED', file='cklink')
C
C     Initialize CHEMKIN
C
      CALL CKLEN  (LINKCK, LOUT, LENI, LENR, LENC)
      CALL CKINIT (LENIWK, LENRWK, LENCWK, LINKCK, LOUT, IWORK,
```

```
     1              RWORK, CWORK)
      CALL CKINDX (IWORK, RWORK, MM, KK, II, NFIT)
C
C     Open Output Files
C
      OPEN (UNIT=LIN, STATUS='OLD', FORM='FORMATTED',
     1      FILE='pasrinp.txt')
      OPEN (UNIT=LPIC, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='pics')
      OPEN (UNIT=LRAD, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='radicals')
      OPEN (UNIT=LCEM, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='cems')
      OPEN (UNIT=LPROF, STATUS='UNKNOWN', FORM='FORMATTED',
     1      FILE='profile')
      OPEN (UNIT=LBPROF, STATUS='UNKNOWN', FORM='UNFORMATTED',
     1      FILE='bin_prof')
C
C
C
      NEQ   = KK + 1
      LRW   = 22 + 9*(2*NEQ) + 2*(2*NEQ)**2
      NVODE = LENR + 1
      NWT   = NVODE + LRW
      NH    = NWT  + KK
      NWDOT = NH   + KK
      NTOT  = NWDOT+ KK - 1
C
      LIW   = 30 + 2*NEQ
      IVODE = LENI + 1
      ITOT  = IVODE + LIW - 1
C
      IF (KK .GT. KMAX) THEN
       WRITE (LOUT, *)
     1   ' Error...KMAX too small...must be at least ',KK
       KERR = .TRUE.
      ENDIF
C
      IF (LENRWK .LT. NTOT) THEN
         KERR = .TRUE.
         WRITE (LOUT, *)
     1  ' Error...LENRWK too small...must be at least', NTOT
      ENDIF
C
      IF (LENIWK .LT. ITOT) THEN
         KERR = .TRUE.
         WRITE (LOUT, *)
     1   ' Error...LENRWK too small...must be at least', NTOT
      ENDIF
C
      IF (LENIWK .LT. ITOT) THEN
         KERR = .TRUE.
         WRITE (LOUT, *)
     1   ' Error...LENIWK too small...must be at least', ITOT
      ENDIF
C
      IF (KERR) STOP
```

```
C
      CALL CKSYMS (CWORK, LOUT, KSYM, IERR)
      CALL CKSYME (CWORK, LOUT, KELE, IERR)
      IF (IERR) KERR = .TRUE.
      CALL CKWT   (IWORK, RWORK, RWORK(NWT))
      CALL CKRP   (IWORK, RWORK, RU, RUC, PATM)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Read input file
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      CALL PARSER (LIN, LOUT, 'FLG1', 3, R1, R2, R3, R4)
      REST = (INT(R1) .EQ. 1)
      SERIES = (INT(R1) .EQ. 2)
      LPHI = (INT(R2) .EQ. 1)
      LTAU = (INT(R3) .EQ. 1)
      CALL PARSER (LIN, LOUT, 'FLG2', 3, R1, R2, R3, R4)
      LNEWT = (INT(R1) .EQ. 1)
      LSENS = (INT(R2) .EQ. 1)
      MODEL = (INT(R2) .EQ. 2)
      SKIP = (INT(R3) .EQ. 1)
      CALL PARSER (LIN, LOUT, 'CYCL', 4, R1, CY_MIN, CY_MAX, CY_STEP)
      LFREQ = (INT(R1) .EQ. 1)
      LDIL = (INT(R1) .EQ. 2)
      IF (LDIL) THEN
       IF ((CY_MAX .GE. 1.0) .OR. (CY_MIN .LT. 0.0)) THEN
          WRITE (*,*) 'DILUTION BOUNDS MUST BE BETWEEN 0 AND 1'
       ENDIF
      ENDIF
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'PICS', IPIC, IK1)
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'RADIC', IRAD, IK2)
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'CEMS', ICEM, IK3)
      CALL SYMINDICIES (LIN, LOUT, KK, KSYM, 'PROF', IPROF, IK4)
      CALL PARSER (LIN, LOUT, 'AMBIENT', 2, PA, TA, R3, R4)
      PA = PA/760
      TA = TA + 273.15
      CALL PARSER (LIN, LOUT, 'PASR', 4, VOL, UA, TEMP, P)
      TGIV = (INT(UA) .EQ. -1)
      LUA = (INT(UA) .EQ. -2)
      TEMP = TEMP + 273.15
      P = P/760
      CALL PARSER (LIN, LOUT, 'PQUE', 2, PQTAU, PQTEMP, R3, R4)
      PQTEMP = PQTEMP + 273.15
      CALL PARSER (LIN, LOUT, 'PHI', 1, PHI, R2, R3, R4)
      CALL PARSER (LIN, LOUT, 'FREQ', 1, FREQ, R2, R3, R4)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Stream input and adjustment
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     burner fuel and air
C
      CALL STREAM (LIN, LOUT, 'FUEL', KSYM, KK, IWORK, RWORK, .FALSE.,
```

```
     1                FLRTF, XF, ZF)
      CALL STREAM (LIN, LOUT, 'AIR', KSYM, KK, IWORK, RWORK, .FALSE.,
     1                FLRTA, XA, ZA)
      CALL STREAM (LIN, LOUT, 'DIL', KSYM, KK, IWORK, RWORK, .FALSE.,
     1                FLRTN, XN, ZN)
      CALL STREAM (LIN, LOUT, 'DOPA', KSYM, KK, IWORK, RWORK, .FALSE.,
     1                FLRTD, XD, ZD)
C
C     mix the fuel and dopant streams
C
      IF (FLRTD .GT. 0.0) CALL MIX (KK, FLRTF, FLRTD, ZF, ZD, IWORK,
     1                                RWORK)
C
C     calculate fuel/air equivalence ratios
C
      PHI_B = EQUIV_RATIO (KK, ZA(2), ZF(2), ZA(2), ZF(2), FLRTA,
     1                   FLRTF, FLRTA, FLRTF, KELE, KSYM, IWORK, RWORK)
C
C     Adjust fuel/air equivalence ratio
C
      IF (LPHI) FLRTA = FLRTA*PHI_B/PHI
C
C     Recalculate fuel/air equivalence ratios
C
      PHI_B = EQUIV_RATIO (KK, ZA(2), ZF(2), ZA(2), ZF(2), FLRTA,
     1                     FLRTF, FLRTA, FLRTF, KELE, KSYM, IWORK, RWORK)
C
C     Mix fuel/air stream
C
      CALL MIX (KK, FLRTF, FLRTA, ZF, ZA, IWORK, RWORK)
C
C     Mix Diluting Nitrogen into Stream
C
      CALL MIX (KK, FLRTF, FLRTN, ZF, ZN, IWORK, RWORK)
C
C     Update mole fractions
C
      CALL CKYTX (ZF(2), IWORK, RWORK, XF)
      Z(1) = ZF(1)
      FLRT = FLRTF
      DO 50 K = 1, KK
       Z(K+1) = ZF(K+1)
       X(K) = XF(K)
   50 CONTINUE
C
C     Energy and Element balances of input stream
C
      CALL ENERGY_BAL (Z(1), Z(2), FLRT, 'Inlet', IWORK, RWORK)
      CALL ELE_BAL (KK, X, KELE, FLRT, 'Inlet', IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Solve PSR for initial Reactor Averages
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Total Heat Flux in cal/s
```

```
C
      IF (LUA) THEN
       TGIV = .TRUE.
       CALL CKHBMS (Z(1), Z(2), IWORK, RWORK, HAVE)
       HIN = FLRT*HAVE
       QLOS = 0
      ELSEIF (TGIV) THEN
       QLOS = 0
      ELSE
       QLOS = UA*(TEMP-TA)
      ENDIF
C
C     Run PSR sub program
C
      CALL PDPSR(1, KK, II, Z(1), X, KSYM, FLRT, P, VOL,
     1           TGIV, QLOS, TEMP, REST, LTAU, LSENS, ZAVE)
      CALL CKYTX (ZAVE(2), IWORK, RWORK, X)
C
C     find heat transfer, and volume parameters
C
      CALL CKRHOX (P*PATM, ZAVE(1), X, IWORK, RWORK, RHO)
      IF (LTAU) THEN
       TAU = VOL
       VOL = TAU*FLRT/RHO
      ELSE
       TAU = RHO*VOL/FLRT
      ENDIF
      IF (LUA) THEN
       CALL CKHBMS (ZAVE(1), ZAVE(2), IWORK, RWORK, HAVE)
       HOUT = FLRT*HAVE
       UAV = (HIN - HOUT)/(VOL*(ZAVE(1) - TA))
      ELSEIF (TGIV .AND. .NOT. LUA) THEN
       UAV = -1
      ELSE
       UAV = UA*41868000/VOL
      ENDIF
      DO 70 N = 1, NEQ
       ZPSR(N) = ZAVE(N)
   70 CONTINUE
      CALL ENERGY_BAL (ZPSR(1), ZPSR(2), FLRT, 'PSR', IWORK, RWORK)
      CALL ELE_BAL (KK, X, KELE, FLRT, 'PSR', IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Solve PASRs until the Solution Converges
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Frequency Flow Loop
C
      CALL TIME(START)
      IF (.NOT. LFREQ .AND. .NOT. LDIL) THEN
       CY_MAX = 1.0
       CY_MIN = 1.0
       CY_STEP = 1.0
      ENDIF
      DO 400 R = CY_MIN, CY_MAX, CY_STEP
```

```
      IF (LFREQ) FREQ = 10**R
      DO 80 N = 1, NEQ
         ZAVE(N) = ZPSR(N)
   80 CONTINUE
      IF (LDIL) THEN
         FLRTN = R*FLRTF/(1-R)
         FLRT = FLRTF
         DO 85 N = 1, NEQ
            Z(N) = ZF(N)
   85    CONTINUE
         CALL MIX (KK, FLRT, FLRTN, Z, ZN, IWORK, RWORK)
         FLRT = FLRTF
         CALL MIX (KK, FLRT, FLRTN, ZAVE, ZN, IWORK, RWORK)
      ENDIF
C
C     Write output file headings
C
      WRITE (LPIC,3001) (KSYM(IPIC(K))(:10), K=1,IK1)
      WRITE (LRAD,3001) (KSYM(IRAD(K))(:10), K=1,IK2)
      WRITE (LCEM,3001) "THC","TCLC",(KSYM(ICEM(K))(:10), K=1,IK3)
      WRITE (LPROF,3004) (KSYM(IPROF(K))(:10), K=1,IK4)
C
C     logical option SKIP skips PASR and Runs PSR instead
C
      IF (SKIP) THEN
          CALL CKYTX (Z(2), IWORK, RWORK, X)
          CALL PDPSR(1, KK, II, Z(1), X, KSYM, FLRT, P, VOL,
     1           TGIV, QLOS, TEMP, REST, LTAU, LSENS, ZAVE)
C

         DO 90 N = 1,NEQ
            Z(N) = ZAVE(N)
   90    CONTINUE
         CALL CKYTX (Z(2), IWORK, RWORK, X)
         GOTO 300
      ENDIF
C
C     Start Loop
C
      DO 200 ITER = 0, 5000
C
C     Find Mean Residence Time and Time Intervals (tf) for NBATCH
C     Reactors
C
         CALL CKRHOY (P*PATM, ZAVE(1), ZAVE(2), IWORK, RWORK, RHO)
         IF (.NOT. LTAU) TAU = RHO*VOL/FLRT
C
C     choose TF for 99.9999% mass recovery in NITER iterations
C
         TF = TAU*(-LOG(1.0E-6)/NITER)
C
C        Adjust volume for given tau
C
         IF (LTAU) VOL = FLRT*TAU/RHO
C
         CALL PDPASR (LOUT, LPROF, LBPROF, LCONV, Z, IWORK, RWORK, P,
     1                TF, NITER, IPROF, IK4, ZOUT, TEMP_RMS)
```

```
          ZAVE(1) = ZOUT(1)
          ERR = 0.0
          DO 150 N=1,KK
             FN(N+1) = ZAVE(N+1) - ZOUT(N+1)
             IF (ZOUT(N+1) .GE. ATOL)
     1          ERR = ERR+((ZAVE(N+1)-ZOUT(N+1))/ZOUT(N+1))**2
  150        CONTINUE
          ERR = SQRT(ERR)
          WRITE (*, 3005) R, FREQ, FLRTN, ZAVE(1), ERR
          CALL CKYTX (ZOUT(2), IWORK, RWORK, X)
C
          CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
          WRITE (LPIC,3002) ITER, TAU, ZOUT(1), (X(IPIC(K)), K=1,IK1)
          WRITE (LRAD,3002) ITER, TAU, ZOUT(1), (X(IRAD(K)), K=1,IK2)
          WRITE (LCEM,3002) ITER, TAU, ZOUT(1), THC, TCLC,
     1              (X(ICEM(K)), K=1,IK3)
          IF (LCONV) GOTO 220
          LCONV = (ERR .LE. TTOL)
C
C     Otherwise find new guess for ZAVE
C
          IF ((MOD(ITER,100) .EQ. 0) .AND. LNEWT) THEN
             CALL PASRNEWT (LOUT, LPROF, LBPROF, LCONV, Z, IWORK, RWORK,
     1                      P, TF, NITER, IPROF, IK4, ZOUT, X)
         ELSE
            DO 170 N = 1,KK
            ZAVE(N+1) = ZOUT(N+1)
  170         CONTINUE
         ENDIF
  200   CONTINUE
  220   CONTINUE
        LCONV = .FALSE.
        CALL ENERGY_BAL (ZOUT(1), ZOUT(2), FLRT, 'Exit', IWORK, RWORK)
        DO 250 N = 1,KK+1
           Z(N) = ZOUT(N)
  250   CONTINUE
  300   CONTINUE
        CALL SAVER (KK, Z, X, SERIES, "pic", "rad", "cem", "bin",
     1             KSYM, KELE, IWORK, RWORK, IPIC, IRAD, ICEM,
     2             IK1, IK2, IK3, PHI_B, FREQ, TAU, FLRT, P, TEMP_RMS)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Run Probe quench if PQTAU != 0 using Barat or Calculated profile
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
        IF (PQTAU .NE. 0.0) THEN
           IF ((PQTAU .GT. 0.0).AND. (LPROBE)) THEN
              TLM = (Z(1)-PQTEMP)/LOG((Z(1)-TA)/(PQTEMP-TA))
              CALL CKRHOY (P*PATM, TLM+TA, Z(2), IWORK, RWORK, RHO)
              PQVOL = FLRT*PQTAU/RHO
              PQUA = UA_FIND (KK, II, Z(1), PQTEMP, TA, X, KSYM, FLRT,
     1                        PA, PQTAU, IWORK, RWORK)
              PQUAV = PQUA/PQVOL
              DT = 1e-6
              CALL PDPFR (0, LOUT, LPIC, LRAD, LCEM, Z, IWORK, RWORK,
```

```
     1                KELE, FLRT, P, PQTAU, DT, IPIC, IRAD, ICEM, IK1,
     2                IK2, IK3, .TRUE.)
           ELSE
              IF (LPROBE) BM = PQTAU
              LPROBE = .FALSE.
              PQUAV = -1
              TCOOL = PQTEMP
              CM = (Z(1) - TCOOL)/TCOOL
              DT = 1e-6
              PQTAU = 0.003
C
C          Default Probe residence time is 3 ms
C
              CALL PDPFR (0, LOUT, LPIC, LRAD, LCEM, Z, IWORK, RWORK,
     1                KELE, FLRT, P, PQTAU, DT, IPIC, IRAD, ICEM, IK1,
     2                IK2, IK3, .TRUE.)
           ENDIF
           CALL CKYTX (Z(2), IWORK, RWORK, X)
           CALL SAVER (KK, Z, X, SERIES, "picq", "radq", "cemq",
     1             "binq", KSYM, KELE, IWORK, RWORK, IPIC, IRAD, ICEM,
     2             IK1, IK2, IK3, PHI_B, FREQ, PQTAU, FLRT, P, TEMP_RMS)
        ENDIF
        SERIES = .FALSE.
  400 CONTINUE
      CALL TIME (FINISH)
      WRITE (*,3003) START, FINISH
 3001 FORMAT ('"ITER"    ',' "tau"  ',4X,'"TMP(K)"',30(' "',A16,'"'))
 3002 FORMAT (I6, F11.4, F11.4, 30E19.4)
 3003 FORMAT (//2X, 'Start Time:  ', A12, /2X,'Finish Time: ', A12)
 3004 FORMAT ('TIME        ','T/TAU      ','CDF        ',4X,'"TMP(K)"',
     1        30(' "',A16,'"'))
 3005 FORMAT (2X,'FACTOR',F6.2,2X,'FREQ',2X,F9.1,2X,'N2',2X,
     1             F6.2,2X,'TEMP',2X,F8.2,2X,'ERR',2X,F9.5)
      END
C
      SUBROUTINE PDPSR (IPSR, KK, II, TIN, XIN, KSYM, FLRT, PA, VOL,
     1                  TGIV, QLOS, TEMP, RESTART, LTAU, SENS, ZOUT)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     subroutine PDPSR sets up the input file and reads
C
C     inputs:
C
C        IPSR     integer, reactor number
C        KK     integer, number of species
C        II     integer, number of reactions
C        TIN    temperature of the input stream
C        XIN      mole fractions of species
C        KSYM   character array of chemical specie symbols
C        FLRT     mass flow rate g/sec
C        PA     ambient pressure (atm)
C        VOL    reactor volume (cm**3)
C        TGIV   logical, 1 temperature fixed, 0 temperature solved
C        QLOS   heat loss (cal/sec)
C        TEMP   reactor temperature (initial guess if TGIV=0)
C        RESTART logical 1 = start from restart file
```

```
C        LTAU   logical 1 = volumes become mean residence times in sec
C        SENS   logical 1 = run sensitivity analysis
C
C     output:
C
C        ZOUT   double array of exit temperature and mass fractions
C
C     scratch:
C
C        FOSC   double array of first order sensitivity coefficients
C               wrt the pre-exponential parameter in rxn i
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
C
      PARAMETER (LENRWK=150000, LENIWK=150000,LENLWK=6000,LENCWK=6000,
     1           KMAX=250)
      DIMENSION RWORK(LENRWK), IWORK(LENIWK), XIN(*), ZOUT(*),
     1          FOSC(KMAX)
      LOGICAL RESTART, TGIV, LWORK(LENLWK), LTAU, SENS
      CHARACTER KSYM(*)*16, CWORK(LENCWK)*16, CREST*8, CRCVR*8,
     1          CSENS*8, CNUMS*10
C
      DATA LIN, LOUT, LREST, LSAVE, LRCVR, LINKCK, LSENS
     1     /10, 6,14,11,16,25,26/
      DATA ISTEPS/100/, DT/1E-5/, ATOL/1E-20/, RTOL/1E-6/
      CNUMS = '0123456789'
C
c     Convert IPSR to a string and concatenate to file name
C
      ITENS = INT(IPSR/10) + 1
      IONES = IPSR - (ITENS - 1)*10 + 1
      CRCVR = 'rcvr' // CNUMS(ITENS:ITENS) // CNUMS(IONES:IONES)
      CREST = 'rest' // CNUMS(ITENS:ITENS) // CNUMS(IONES:IONES)
      CSENS = 'sens' // CNUMS(ITENS:ITENS) // CNUMS(IONES:IONES)
C
C     Open input and output files
C
      OPEN (LIN, STATUS ='UNKNOWN',    FORM='FORMATTED',
     1      FILE='psrin')
      OPEN (LINKCK, STATUS='OLD',     FORM='UNFORMATTED',
     1      FILE='cklink')
      OPEN (LREST, STATUS='UNKNOWN', FORM='UNFORMATTED',
     1      FILE=CREST)
      OPEN (LRCVR, STATUS='UNKNOWN',FORM='UNFORMATTED',
     1      FILE=CRCVR)
      OPEN (LSENS, STATUS='UNKNOWN',FORM='UNFORMATTED',
     1      FILE=CSENS)
      OPEN (LSAVE,  STATUS='UNKNOWN',   FORM='UNFORMATTED',
     1      FILE='save')
C
C     If LTAU is .TRUE. then VOL represents the mean residence time
C     volume is set at an arbitrary 1000 cc
C
      IF (LTAU) TAU = VOL
C
```

```
C
C     Write Input File for PSR1
C
C
      IF (.NOT. TGIV) WRITE (LIN, 2001)
      IF (TGIV) WRITE (LIN, 2015)
      IF (RESTART) WRITE (LIN, 2014)
      DO 50 I = 1, KK
      IF (XIN(I) .NE. 0 ) THEN
        WRITE (LIN, 2002) KSYM(I), XIN(I)
      ENDIF
   50 CONTINUE
      WRITE (LIN, 2003) PA
      IF (.NOT. LTAU) WRITE (LIN, 2004) VOL
      IF (LTAU) THEN
         WRITE (LIN, 2004) 1000.0
         WRITE (LIN, 2021) TAU
      ENDIF
      IF (.NOT. LTAU) WRITE (LIN, 2005) FLRT
      IF (.NOT. TGIV) WRITE (LIN, 2006) TIN
      IF (.NOT. TGIV) WRITE (LIN, 2007) QLOS
      WRITE (LIN, 2008) 0
      WRITE (LIN, 2009) ISTEPS, DT
      WRITE (LIN, 2010) ISTEPS, DT
      WRITE (LIN, 2019) ATOL
      WRITE (LIN, 2020) RTOL
      WRITE (LIN, 2011) TEMP
      IF (SENS) WRITE (LIN, 2022)
      WRITE (LIN, 2012)
      REWIND (LIN)
      WRITE (LOUT, 2017) IPSR
      CALL PSR (LIN, LOUT, LINKCK,  LREST, LSAVE, LRCVR, LENLWK,
     1          LWORK, LENIWK, IWORK, LENRWK, RWORK, LENCWK, CWORK)
      CLOSE (UNIT=LIN, STATUS='DELETE')
      CLOSE (UNIT=LREST, STATUS = 'KEEP')
      CLOSE (UNIT=LRCVR, STATUS = 'KEEP')
C
C     Read PSR output to ZOUT array
C
      REWIND (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE)
      READ (LSAVE) ISOLUT
      READ (LSAVE) NNP
      KKP = NNP - 1
      READ (LSAVE) EQUIVP, PP, TAUP, FLRTP, VP, QP
      READ (LSAVE) TP, (XIN(K), K=1,KKP)
      READ (LSAVE) (ZOUT(K), K=1,NNP)
      DO 100 I = 1,NNP
       IF (ZOUT(I) .LT. 0.0) ZOUT(I) = 0.0
  100 CONTINUE
      IF (SENS) THEN
         READ (LSAVE)
         DO 200 I = 1,II
```

```
            READ (LSAVE) N, FOSCT, (FOSC(K), K = 1,KK)
            WRITE (LSENS) N, FOSCT, (FOSC(K), K = 1,KK)
  200     CONTINUE
      ENDIF
      CLOSE (UNIT=LSENS, STATUS = 'KEEP')
      CLOSE (UNIT=LSAVE, STATUS = 'DELETE')
      WRITE (LOUT, 2018)
 2001 FORMAT ('ENRG')
 2002 FORMAT ('REAC', 1X, A16, 1X, E13.6)
 2003 FORMAT ('PRES', 1X, E13.6)
 2004 FORMAT ('VOL',  1X, E13.6)
 2005 FORMAT ('FLRT', 1X, E13.6)
 2006 FORMAT ('TINL', 1X, E13.6)
 2007 FORMAT ('QLOS', 1X, E13.6)
 2008 FORMAT ('PRNT', 1X, I2)
 2009 FORMAT ('TIME', 1X, I12, 2X, E13.6)
 2010 FORMAT ('TIM2', 1X, I12, 2X, E13.6)
 2011 FORMAT ('TEMP', 1X, E13.6)
 2012 FORMAT ('END')
 2013 FORMAT ('CNTN')
 2014 FORMAT ('RSTR')
 2015 FORMAT ('TGIV')
 2016 FORMAT ('AROP')
 2017 FORMAT (2X, 'Calling PSR for Unit: ', I2)
 2018 FORMAT (2X, 'Returning to the main program.')
 2019 FORMAT ('ATOL', 1X, E13.6)
 2020 FORMAT ('RTOL', 1X, E13.6)
 2021 FORMAT ('TAU', 2X, E13.6)
 2022 FORMAT ('SEN')
      RETURN
      END

      SUBROUTINE PDPASR (LOUT, LPROF, LBPROF, LCONV, Z, IWORK, RWORK,
     1                  P, TF, NITER, IPROF, IK4, ZOUT, TEMP_RMS)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     subroutine PDPASR sets up the input file and reads
C
C     inputs:
C
C        LOUT   integer, standard output file
C        LPROF  integer, selected PDF profile output
C        LBPROF integer, binary data profile output
C        LCONV  logical, .true. when solution has converged
C        Z      array of independent variables
C        IWORK  integer working array
C        RWORK  double working array
C        P      double, pressure in atmosphere
C        TF     time step (sec)
C        NITER  integer, number of iterations of time TF
C        IPROF  integer array of indicies of selected species
C        IK4    integer number of elements in IPROF
C
C     output:
C
C         ZOUT    double array of exit temperature and mass fractions
```

```
C          TEMP_RMS double root mean square temperature fluctuation
C
C     working:
C          ZWORK    double array at least NEQ*2 second half is
C                   integration of the average
C          TMAX     double, stop time for DVODE integration
C          X1       double, LOG transform of TF
C          X2       double, LOG transform of TMAX
C          DX       double, LOG scale interval
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
C
      PARAMETER (ITOL=2, ITASK=1, IOPT=0, RTOL=1.0E-14,
     1           MF=22, KMAX=250, KMAX2=500)
      DIMENSION IWORK(*), RWORK(*), Z(*), ZOUT(*), ZWORK(KMAX2),
     1          ZAVE(KMAX), X(KMAX), FN(KMAX), IPROF(*), ATOL(KMAX2)
      LOGICAL LCONV
      EXTERNAL  FUN
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV, PQUAV, TCOOL, BM, CM
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
      COMMON /PASRCONS/ ZAVE, FREQ, TAU, FN
C
C     initialize values
C
      ISTATE = 1
      NEQ = KK + 1
      LRW = 22 + 9*(2*NEQ) + 2*(2*NEQ)**2
      NVODE = LENR +1
      IVODE = LENI +1
      LIW = 30 + 2*NEQ
      TT1 = 0.0
      TT2 = TT1
      P_HOLD = PA
      PA = P
      DO 20 N = 1, NEQ
       ZWORK(N) = Z(N)
       ZWORK(N+NEQ) = 0
       ATOL(N) = ABS (ZAVE(N))*RTOL + 1E-22
       ATOL(N+NEQ) = ATOL(N)
   20 CONTINUE
      CALL CKHBMS (Z(1), Z(2), IWORK, RWORK, HAVEG)
      ATOL(NEQ+1) = RTOL*ABS (HAVEG)
C
C     Rewind and skip header for PDF profile file
C
      IF (LCONV) THEN
         REWIND (LPROF)
         READ (LPROF,*)
         REWIND (LBPROF)
         WRITE (LBPROF)
         WRITE (LBPROF)
         WRITE (LBPROF)
      ENDIF
```

```
C
C     Start Loop
C
C     TF = TF/100
C     NITER = NITER + 120
      TMAX = TF*NITER
      X1 = LOG(TF)
      X2 = LOG(TMAX)
      DX = (X2 - X1)/(NITER - 1)
      TEMP_RMS = 0.0
      DO 100 ITER = 1,NITER
C
      TT3 = EXP(X1 + (ITER - 1)*DX)
C
C     F is the Probability Density  Function
C
         F1 = 1/TAU* EXP(-TT2/TAU)
         F2 = 1/TAU* EXP(-TT3/TAU)
         F = 0.5*(F2+F1)*(TT3 - TT2)
C
C     Save the solution profile on last iteration
C
         IF (LCONV) THEN
            WRITE (LPROF,3002) TT2, TT2/TAU, F, ZWORK(1),
     1            (ZWORK(IPROF(K)+1),K=1,IK4)
            DO 50 K = 1, KK+1
               ZOUT(K) = ZWORK(K)
   50       CONTINUE
            CALL CKYTX (ZOUT(2), IWORK, RWORK, X)
            WRITE (LBPROF) ZOUT(1), F, TT2, TAU, TF
            WRITE (LBPROF) (X(K), K=1,KK)
            WRITE (LBPROF) (ZOUT(K), K=1,KK+1)
         ENDIF
C
C     Integrate TEMP_RMS using trapizoidal technique
C
         F_RMS = 1/TAU*EXP(-TT2/TAU)*(ZWORK(1) - ZAVE(1))**2
         IF (ITER .GT. 1) THEN
            TEMP_RMS = TEMP_RMS + (F_RMS + F_RMS_OLD)*TF/2
         ENDIF
         F_RMS_OLD = F_RMS
C
C     integrate to next output TT3
C
         TT2 = TT3
         CALL DVODE
     *           (FUN, 2*NEQ, ZWORK, TT1, TT2, ITOL, RTOL, ATOL,
     1            ITASK, ISTATE, IOPT, RWORK(NVODE), LRW, IWORK(IVODE),
     2            LIW, JAC, MF, RWORK, IWORK)
C
C        WRITE (*,*) 'EMERGED WITH ISTATE = ', ISTATE
         IF (ISTATE .LE. -1) THEN
            IF (ISTATE .EQ. -1) THEN
               ISTATE = 2
            ELSE
               WRITE (LOUT,*) ' ISTATE=',ISTATE
               STOP
```

```
            ENDIF
         ENDIF
C        IF ((ITER .EQ. 100).OR.(ITER .EQ. 120))  TF = TF*10
  100 CONTINUE
C
C     copy average mass fractions to the output array
C
      DO 150 K = 1,KK
         ZOUT(K+1) = ZWORK(K+KK+2)
  150 CONTINUE
C
C     Normalize outputs
C
      F = 1-EXP(-1*TT2/TAU)
      ZWORK(KK+2) = ZWORK(KK+2)/F
      TEMP_RMS = TEMP_RMS/F
      DO 170 K = 1,KK
         ZOUT(K+1) = ZOUT(K+1)/F
  170 CONTINUE
      TEMP_RMS = SQRT (TEMP_RMS)
C
C     find the average temperature from the average enthalpy
C
      T_REF = 1000
      HAVE = ZWORK(NEQ+1)
      CALL CKHBMS (T_REF, ZOUT(2), IWORK, RWORK, H_REF)
      CALL CKCPBS (ZAVE(1), ZOUT(2), IWORK, RWORK, CPB)
      CALL CKCPBS (T_REF, ZOUT(2), IWORK, RWORK, CPB_REF)
      ZOUT(1) = (HAVE-H_REF)/((CPB+CPB_REF)/2) + T_REF
      DO 200 I = 1, 5000
         CALL CKHBMS (ZOUT(1), ZOUT(2), IWORK, RWORK, HAVEG)
         DIFF = (HAVE - HAVEG)/ABS(HAVE)
         IF (ABS(DIFF) .LE. 1E-6) GOTO 250
         ZOUT(1) = (1 + 0.01*DIFF)*ZOUT(1)
  200 CONTINUE
      WRITE (LOUT,*) 'Temperature did not converge in PDPASR'
  250 CONTINUE
      PA = P_HOLD
      RETURN
 3002 FORMAT (F11.8, F11.8, F11.8, F11.2, 30E19.6)
      END
C
      SUBROUTINE PASRNEWT (LOUT, LPROF, LBPROF, LCONV, Z, IWORK, RWORK,
     1                     P, TF, NITER, IPROF, IK4, ZSCRATCH, RJAC)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Subroutine to find next guess of mean mass fractions using a
C     modified Newton-Raphson technique.
C
C     This Subroutine is an adaptation of PSRJAC from PSR, a program
C     written by PETER GLARBORG of TECHNICAL UNIVERSITY OF DENMARK
C     and ROBERT J. KEE of SANDIA NATIONAL LABORATORIES.
C
C     It also uses LINPACK subroutines DCOPY, DGECO, and DGESL found
C     in DMATH.FOR
C
```

```
C     Input:
C               LOUT      integer general output file
C               LPROF     integer PDF profile output file
C               LBPROF    integer BIN PDF profileoutput file
C               LCONV     logical .TRUE. for last iteration
C               Z         double array of state variables
C               IWORK     integer working array
C               RWORK     double working array
C               P         pressure in atmospheres
C               TF        double time increment for output
C               NITER     integer number of iterations in PDPASR
C               IPROF     integer array of indicies of selected species
C               IK4       integer number of elements in IPROF
C               ZSCRATCH  double scratch array of state variables
C               RJAC      double Jacobian input of dummy arguments
C
C     Output:  common array ZAVE has next guess of average state
C     variables
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
C
      PARAMETER (KMAX = 250, ATOL = 1E-30)
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
      DIMENSION RJAC(KK,*), IPVT(KMAX), ZAVE(KMAX), Z(*),
     1          ZSCRATCH(*), IWORK(*), RWORK(*), FN(KMAX), IPROF(*)
      COMMON /PASRCONS/ ZAVE, FREQ, TAU, FN
      LOGICAL LCONV
C
C     Determine computer tolerance for relative and absolute pertibation
C
      U = 1.0
   50 CONTINUE
         U = U*0.5
         COMP = 1.0 + U
         IF (COMP .NE. 1.0) GOTO 50
      ABSOL = SQRT(2*U)
      RELAT = SQRT(2*U)
C
C     ZERO THE MATRIX STORAGE SPACE.
C
      CALL DCOPY (KK*KK, 0.0, 0, RJAC, 1)
C
C
C
C        TOP OF THE LOOPS OVER THE RESIDUE CLASSES AND
C                                            SOLUTION COMPONENTS.
C
      WRITE (*,*) 'Building Jacobian'
      DO 200 M = 1, KK
C
C            FOR A GIVEN RESIDUE CLASS AND A GIVEN SOLUTION COMPONENT,
C            PERTRB THE  VECTOR AT POINTS IN THE SAME RESIDUE CLASS.
C
         SAVE   = ZAVE(M+1)
         PERTRB = ABS(ZAVE(M)) * RELAT + ABSOL
```

```
         ZAVE(M+1)    = ZAVE(M+1) + PERTRB
C
C        Call PDPASR with the perturbed ZAVE and store in ZSCRATCH
C
      CALL PDPASR (LOUT, LPROF, LBPROF, LCONV, Z, IWORK, RWORK, P, TF,
     1             NITER, IPROF, IK4, ZSCRATCH, TEMP_RMS)
C
C              Build Jacobian
C
         DO 100 N = 1, KK
            RJAC(N, M) = (ZAVE(N+1)-ZSCRATCH(N+1)-FN(N+1))/PERTRB
  100    CONTINUE
C
C              Restore ZAVE to it original value
C
         ZAVE(M+1) = SAVE

C
C          BOTTOM OF THE LOOPS OVER THE RESIDUE CLASSES AND SOLUTION
C          COMPONENTS.
C
  200 CONTINUE
C
C     Factor the Jacobian Matrix
C
      CALL DGECO (RJAC, KK, KK, IPVT, RCOND, ZSCRATCH(2))
      IF (1.0 + RCOND .EQ. 1.0) THEN
         WRITE (LOUT, *) ' FATAL ERROR, SINGULAR JACOBIAN'
         STOP
      ENDIF
C
C     Use the Jacobian ITER times
C
      ERR1_OLD = 1E6
      ERR2_OLD = 1E6
  500 CONTINUE
C
C     Solve J*X = B  Where:
C         B = FN
C         X = Y_AVE_OLD - Y_AVE_NEW
C
C     Output of DGESL: B becomes X
C
      CALL DGESL (RJAC, KK, KK, IPVT, FN(2), 0)
      ERR1 = 0.0
      DO 250 N = 1, KK
         IF (ZAVE(N+1) .GT. ATOL)
     1       ERR1 = ERR1 + (0.5*FN(N+1)/ZAVE(N+1))**2
  250 CONTINUE
      ERR1 = SQRT(ERR1)
      IF (ERR1 .GT. ERR1_OLD) GOTO 600
      ERR1_OLD = ERR1
      DO 300 N = 1, KK
         ZAVE(N+1) = ZAVE(N+1) - 0.5*FN(N+1)
  300 CONTINUE
      CALL PDPASR (LOUT, LPROF, LBPROF, LCONV, Z, IWORK, RWORK, P,
     1             TF, NITER, IPROF, IK4, ZSCRATCH, TEMP_RMS)
```

```
      ZAVE(1) = ZSCRATCH(1)
        ERR2 = 0.0
      DO 400 N = 1, KK
         FN(N+1) = ZAVE(N+1) - ZSCRATCH(N+1)
           IF (ZAVE(N+1) .GE. ATOL)
     1        ERR2 = ERR2 + ((ZAVE(N+1)-ZSCRATCH(N+1))/ZAVE(N+1))**2
  400 CONTINUE
      ERR2 = SQRT(ERR2)
      IF (LCONV) GOTO 600
      LCONV = (ERR2 .GT. ERR2_OLD)
      ERR2_OLD = ERR2
      WRITE (*,*) 'CONVERGING     ERR1', ERR1, '  ERR2', ERR2
      GOTO 500
  600 CONTINUE
      WRITE (*,*) 'DIVERGING'
      RETURN
      END
C
      SUBROUTINE FUN (N, TIME, Z, ZP, RPAR, IPAR)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Function Subroutine provided for DVODE.
C
C     Solves a constant pressure plug flow reactor with a constant heat
C     transfer coefficient per unit volume - UAV
C
C     Input:
C             N         Number State variables (not used)
C             TIME      Double, independent time variable
C             Z         Array of indepent variables
C             RPAR      Working real (double) array
C             IPAR      Working integer array
C
C     Output:
C             ZP        first time derivative of Z
C
C     Work:
C             Y         Double Array, mass fractions
C
C     Common:
C             TA        Double, ambient temperature
C             UAV       Double, heat transfer coeff/vol
C                       (ergs/(sec*degK*cc))
C             TAU       Double, mean residence time
C             ZAVE      Double Array of reactor average states
C                       in last iteration
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
C
      PARAMETER (KMAX = 250)
      DIMENSION Z(*), ZP(*), RPAR(*), IPAR(*), ZAVE(KMAX), Y(KMAX),
     1          FN(KMAX)
      COMMON /RCONS/ PATM, RU, TA, PA, UAV, PQUAV, TCOOL, BM, CM
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
```

```
      COMMON /PASRCONS/ ZAVE, FREQ, TAU, FN
C
C     Variables in Z are:  Z(1)    = T
C                          Z(K+1)  = Y(K)
C                          Z(KK+2) = HAVE
C                          Z(K+KK+2) = YAVE(K)
C
C     Initialize Y array
C
      DO 50 K = 1,KK
         Y(K) = Z(K+1)
   50 CONTINUE
C
C     Call CHEMKIN subroutines
C
      P = PATM * PA
      CALL CKRHOY (P, Z(1), Y, IPAR, RPAR, RHO)
      CALL CKCPBS (Z(1), Y, IPAR, RPAR, CPB)
      CALL CKWYP  (P, Z(1), Y, IPAR, RPAR, RPAR(NWDOT))
      CALL CKHMS  (Z(1), IPAR, RPAR, RPAR(NH))
C
C     Find the value of the PDF (RTD)
C
      E = (1/TAU)*EXP(-1*TIME/TAU)
C
C     Form governing equation
C
      SUM1 = 0.0
      SUM2 = 0.0
      DO 100 K = 1, KK
         H    = RPAR(NH    + K - 1)
         WDOT = RPAR(NWDOT + K - 1)
         WT   = RPAR(NWT   + K - 1)
         ZP(K+1) = -1*FREQ*(Z(K+1)-ZAVE(K+1)) + WDOT * WT / RHO
         SUM1 = SUM1 + H*ZP(K+1)
         SUM2 = SUM2 + H*Z(K+1)
         ZP(K+KK+2) =  E*Z(K+1)
  100 CONTINUE
      IF (UAV .GE. 0) THEN
         ZP(1) =  -1*SUM1/CPB - UAV*(Z(1)-TA)/(CPB*RHO)
C
C     Turbulent Heat Transfer by Reynolds Analogy
C
C        ZP(1) = ZP(1) - 0.7*FREQ*(Z(1) - ZAVE(1))
      ELSE
         ZP(1) = 0.0
      ENDIF
      ZP(KK+2) = E*SUM2
C
      RETURN
      END
C
      SUBROUTINE PDPFR (IPFR, LOUT, LPIC, LRAD, LCEM, Z, IWORK, RWORK,
     1                  KELE, FLRT, P, VOL, DT, IPIC, IRAD, ICEM, IK1,
     2                  IK2, IK3, LTAU)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
C
C     subroutine PDPFR sets up the input file and reads
C
C     inputs:
C
C        IPFR     integer, reactor number
C        LOUT   integer, standard output file
C        LPIC   volume step file for pics
C        LRAD   volume step file for radicals
C        LCEM   volume step file for CEMs
C        Z      array of independent variables
C        IWORK  integer working array
C        RWORK  double working array
C        KELE   character element array
C        FLRT     mass flow rate g/sec
C        P      reactor pressure (atm) then converted to (dynes/cm**3)
C        VOL    reactor volume (cm**3)
C        DT     time step (sec)
C        IPIC   array of PIC indicies
C        IRAD   array of RAD indicies
C        ICEM   array of CEM indicies
C        IK1,IK2,IK3  length of indici arrays
C        LTAU   logical = 1 VOL represents the residence time
C
C     output:
C
C        Z      double array of exit temperature and mass fractions
C
C     working:
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
C
      PARAMETER (ITOL=1, ITASK=1, IOPT=0, RTOL=1.0E-06,
     1           ATOL=1.0E-20, MF=22, KMAX = 250, LBINQ=23)
      DIMENSION IWORK(*), RWORK(*), Z(*), IPIC(*), IRAD(*), ICEM(*),
     1          X(KMAX)
      CHARACTER KELE(*)*2
      LOGICAL LTAU
      EXTERNAL  FUN2
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV, PQUAV, TCOOL, BM, CM
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
      PA_OLD = PA
      PA = P
      P = P*PATM
      OPEN (UNIT=LBINQ, STATUS='UNKNOWN', FORM='UNFORMATTED',
     1      FILE='bin_que')
      WRITE (LBINQ)
      WRITE (LBINQ)
      WRITE (LBINQ)
C
C     initialize values
C
      ISTATE = 1
```

```
      NEQ = KK + 1
      LRW = 22 + 9*NEQ + 2*NEQ**2
      NVODE = LENR +1
      IVODE = LENI +1
      LIW = 30 + NEQ
      TT1 = 1e-12
      TT2 = TT1
      VPFR = 0.0
      IF (LTAU) TAU = VOL
C
C     Start Loop
C
C     print volume incremental solution
C
   50 CONTINUE
      T = Z(1)
      CALL CKYTX (Z(2), IWORK, RWORK, X)
      CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
      WRITE (LPIC,3002) IPFR, VPFR, T, (X(IPIC(K)), K=1,IK1)
      WRITE (LRAD,3002) IPFR, VPFR, T, (X(IRAD(K)), K=1,IK2)
      WRITE (LCEM,3002) IPFR, VPFR, T, THC, TCLC,
     1                        (X(ICEM(K)), K=1,IK3)
      WRITE (LBINQ) Z(1), 0.0, 0.0, THC, TCLC
      WRITE (LBINQ) (X(K), K=1,KK)
      WRITE (LBINQ) (Z(K), K=1,KK+1)
C
C     Check for end of volume or end of residence time
C
      IF ((VPFR .GE. VOL) .AND. (.NOT. LTAU)) THEN
         WRITE (LOUT, 3001) IPFR, TT2
         GOTO 100
      ENDIF
      IF (LTAU .AND. TT2 .GE. TAU) THEN
         WRITE (LOUT, 3001) IPFR, TT2
         GOTO 100
      ENDIF
C
C     integrate to next output TT2 + DT
C
      TT2 = TT2 + DT
      IF (LTAU .AND. TT2 .GE. TAU) TT2 = TAU
      CALL DVODE
     *            (FUN2, NEQ, Z, TT1, TT2, ITOL, RTOL, ATOL, ITASK,
     1             ISTATE, IOPT, RWORK(NVODE), LRW, IWORK(IVODE),
     2             LIW, JAC, MF, RWORK, IWORK)
C
C     WRITE (*,*) 'EMERGED WITH ISTATE = ', ISTATE
      IF (ISTATE .LE. -2) THEN
         IF (ISTATE .EQ. -1) THEN
            ISTATE = 2
            GO TO 100
         ELSE
            WRITE (LOUT,*) ' ISTATE=',ISTATE
            STOP
         ENDIF
      ENDIF
C
```

```
C     Volume increment
C
      CALL CKRHOY (P, Z(1), Z(2), IWORK, RWORK, RHO)
      VPFR = VPFR + DT*FLRT/RHO
C
C
      WRITE(LOUT,*) Z(1), TNEW
      GOTO 50
  100 CONTINUE
      P = PA
      PA = PA_OLD
 3001 FORMAT (5X, 'PFR ', I1, ' TAU (SEC) = ', F8.4)
 3002 FORMAT (2X, 'PFR',I1, F11.4, F11.4, 30E19.4)
      RETURN
      END
C
      SUBROUTINE FUN2 (N, TIME, Z, ZP, RPAR, IPAR)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Function Subroutine provided for DVODE.
C
C     Solves a constant pressure plug flow reactor with a constant heat
C     transfer coefficient per unit volume - UAV
C
C     Input:
C             N         Number of equations (not used)
C             TIME      Independent variable
C             Z         Array of dependent variables
C             RPAR      Working real (double) array
C             IPAR      Working integer array
C
C     Output:
C             ZP  first time derivative of Z
C
C     Common:
C             TA  Double, ambient temperature
C             UAV       Double, heat transfer coeff/vol
C                       (ergs/(sec*degK*cc))
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
C      IMPLICIT REAL (A-H,O-Z), INTEGER(I-N)
C
      COMMON /RCONS/ PATM, RU, TA, PA, UAV, PQUAV, TCOOL, BM, CM
      COMMON /ICONS/ KK, NWT, NH, NWDOT, LENR, LENI
C
      DIMENSION Z(*), ZP(*), RPAR(*), IPAR(*)
C
C     Variables in Z are:  Z(1)   = T
C                          Z(K+1) = Y(K)
C
C     Call CHEMKIN subroutines
C
      P = PATM * PA
      CALL CKRHOY (P, Z(1), Z(2), IPAR, RPAR, RHO)
      CALL CKCPBS (Z(1), Z(2), IPAR, RPAR, CPB)
```

```
      CALL CKWYP  (P, Z(1), Z(2), IPAR, RPAR, RPAR(NWDOT))
      CALL CKHMS  (Z(1), IPAR, RPAR, RPAR(NH))
C
C     Form governing equation
C
      SUM = 0.0
      DO 100 K = 1, KK
         H    = RPAR(NH    + K - 1)
         WDOT = RPAR(NWDOT + K - 1)
         WT   = RPAR(NWT   + K - 1)
         ZP(K+1) = WDOT * WT / RHO
         SUM = SUM + H * WDOT * WT
 100  CONTINUE
      IF (PQUAV .GE. 0.0) THEN
        ZP(1) = -SUM / (RHO*CPB) - PQUAV*(Z(1)-TA)/(CPB*RHO)
      ELSE
         IF (Z(1) .GT. 373) THEN
           ZP(1) = (0.5*BM*TIME**(-0.5))*CM*TCOOL*EXP(BM*TIME**0.5)
         ELSE
           ZP(1) = 0
         ENDIF
      ENDIF
      RETURN
      END
C
C
C
      SUBROUTINE GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C     Groups mole fractions into total hydrocarbons (THC), total
C     chlorinated hydrocarbons (TCLC).  THC counts each carbon atom in
C     a hydrocarbon.  TCLC counts chlorine atoms in each chlorocarbon.
C
C
C     Assumes the existence of exactly 6 elements which must include
C     C, CL, and H.  If the mechanism has more than 6 elements, modify
C     the parameter MM accordingly.
C
C     Input:
C               KK              (integer) number of species
C               X               (double) mole fraction array
C               KELE            (character) array of element strings
C               IWORK           (integer) working array
C               RWORK           (double) working array
C
C     Output:
C               THC, TCLC
C                               (double) summed mass frac
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (MM =6, KMAX = 250)
      DIMENSION X(*), IWORK(*), RWORK(*), NCE(MM,KMAX), IELE(MM)
```

```
      CHARACTER KELE(*)*2
C
C     initialize variables, and get element matrix
C
      THC = 0
      TCLC = 0
      CALL CKNCF (MM, IWORK, RWORK, NCE)
C
C     find indices of key elements
C
      CALL CKCOMP ('C',KELE,MM,IELE(1))
      CALL CKCOMP ('H',KELE,MM,IELE(2))
      CALL CKCOMP ('CL',KELE,MM,IELE(3))
C
C     count species in various categories for entire stream
C
      DO 100 K = 1,KK
         IC = NCE(IELE(1),K)
         IH = NCE(IELE(2),K)
         ICL = NCE(IELE(3),K)
         IF (IC .GE. 1 .AND. IH .GE. 1) THC = THC + X(K)*IC
         IF (IC .GE. 1 .AND. ICL .GE. 1) TCLC = TCLC + X(K)*ICL
  100 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE STREAM (LIN, LOUT, KEYWORD, KSYM, KK, ICKWRK, RCKWRK,
     1                   UNITS, FLRT, X, Z)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Subroutine to read mole fractions and temperature of an
C     input stream.  Mole fractions are normalized and converted to
C     mass fractions.  Input file is read until KEYWORD is encountered
C     or an END is reached.
C
C     Input:
C               LIN     input fortran unit
C               LOUT    output fortran unit
C               KEYWORD character keyword of particular stream
C               KSYM    character array containing all species
C               KK      total number of species
C               UNITS   logical, .TRUE. = scfh, degF, cf
C
C     Output:
C               X       array of normalized mole fractions
C               Z       array of temperature and mass fractions
C               FLRT    double, flow rate grams/sec
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
      DIMENSION X(*), Z(*), ICKWRK(*), RCKWRK(*),
     1          VALUE(3)
      CHARACTER KSYM(*)*16, LINE*80, KEYWORD*(*), LINE2*80
```

```
      LOGICAL IERR, KERR, UNITS
      DATA KERR/.FALSE./
C
C     Check keyword and parse heading line
C
      REWIND (LIN)
   50 CONTINUE
      READ (LIN, '(A)', END=300) LINE
      IF (INDEX (LINE, 'END ') .EQ. 1) THEN
        WRITE (LOUT, 8001) KEYWORD
        STOP
      ENDIF
      IF (INDEX (LINE, 'REM ') .EQ. 1) GOTO 50
C
C     Check keyword
C
      IF (INDEX (LINE, KEYWORD) .EQ. 1) THEN
        WRITE (LOUT, '(5X,A65)') LINE
        CALL CKNPAR (LINE, 3, LOUT, LINE2, ISTART, KERR)
        IF (KERR) THEN
           WRITE (LOUT, *) 'STREAM:  ERROR IN CKNPAR'
           STOP
        ENDIF
        CALL CKXNUM (LINE2, 3, LOUT, NVAL, VALUE, KERR)
        IF (KERR) THEN
         WRITE (LOUT, *) 'STREAM:  ERROR IN CKXNUM'
         STOP
        ENDIF
        IF (NVAL .NE. 3) THEN
          WRITE (LOUT, *) 'STREAM:  INCORRECT FORMAT IN HEADING LINE'
          STOP
        ENDIF
        NOSPEC = INT (VALUE(1))
C
C     Convert Temperature from degF or degC to K and molar flow rate
C     from scfh or std cc/sec to mol/s
C
        IF (NOSPEC .NE. 0) THEN
           IF (UNITS) THEN
              Z(1) = (VALUE(2) + 459.67)*5/9
              FLRT = VALUE(3) * 1.19530/3600.0
           ELSE
              Z(1) = VALUE(2) + 273.15
              FLRT = VALUE(3) * 4.46158e-5
           ENDIF
C
C       Initialize non-zero moles
C
           DO 100 I = 1, NOSPEC
             READ (LIN, '(A)') LINE
             CALL CKSNUM (LINE, 1, LOUT, KSYM, KK, KNUM, NVAL,
     1                    VAL, KERR)
             IF (KERR) THEN
               WRITE (LOUT, *) 'ERROR READING MOLES'
               STOP
             ENDIF
             X(KNUM) = VAL
```

```
  100       CONTINUE
C
C       Normalize the mole fractions
C
            XTOT = 0.00
            DO 150 K = 1, KK
              XTOT = XTOT + X(K)
  150       CONTINUE
            DO 200 K = 1, KK
              X(K) = X(K) / XTOT
  200       CONTINUE
C
C       Initial conditions and mass fractions
C
            CALL CKXTY (X, ICKWRK, RCKWRK, Z(2))
C
C       Convert molar flow rate to mass flow rate
C
            CALL CKMMWX(X, ICKWRK, RCKWRK, WTM)
            FLRT = FLRT*WTM
C
C
           ELSE
             FLRT = 0
           ENDIF
           RETURN
         ENDIF
         GOTO 50
  300 WRITE (LOUT, 8002)
         STOP
 8001 FORMAT ('KEY WORD - ' , A7, ' - NOT FOUND.  PROGRAM STOPPED.')
 8002 FORMAT ('UNEXPECTED END TO INPUT FILE ENCOUNTERED.')
         END
C
C
      SUBROUTINE PARSER (LIN, LOUT, KEYWORD, NODAT, RVAL1, RVAL2,
     1                   RVAL3, RVAL4)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C     Subroutine to read selected from the input file.  Example:
C     Data is in the form:
C
C     KEYWORD  RVAL1  RVAL2  RVAL3  RVAL4
C
C     or:
C
C     REM  comments .....
C
C     PARSER will return NODAT real values after KEYWORD.  If KEYWORD
C     is not found error is returned.  If "REM" is found the line
C     is skipped and the following line is automatically read.
C
C     Input:
C                LIN     integer, input fortran unit
C                LOUT    integer, output fortran unit
C                KEYWORD character string keyword to be checked
C                NODAT   integer, number of data items in line (1-4)
```

```
C
C     Output:
C               RVAL1   double, first real value returned
C               RVAL2   double, second real value returned
C               RVAL3   double, third real value returned
C               RVAL4   double, forth real value returned
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
      DIMENSION VALUE(4)
      CHARACTER LINE*80, KEYWORD*(*), LINE2*80
      LOGICAL KERR
      DATA KERR /.FALSE./
C
C     Read Line
C
      REWIND (LIN)
   50 CONTINUE
      READ (LIN, '(A)', END=200) LINE
      IF (INDEX (LINE, 'END ') .EQ. 1) THEN
        WRITE (LOUT, 8001) KEYWORD
        STOP
      ENDIF
      IF (INDEX (LINE, 'REM ') .EQ. 1) GOTO 50
C
C     Check keyword
C
      IF (INDEX (LINE, KEYWORD) .EQ. 1) THEN
        WRITE (LOUT, '(5X,A50)') LINE
C
C     Parse Line
C
        CALL CKNPAR (LINE, NODAT, LOUT, LINE2, ISTART, KERR)
        IF (KERR) THEN
          WRITE (LOUT, *) 'ERROR IN CKNPAR'
          STOP
        ENDIF
        CALL CKXNUM (LINE2, NODAT, LOUT, NVAL, VALUE, KERR)
        IF (KERR) THEN
          WRITE (LOUT, *) 'ERROR IN CKXNUM'
          STOP
        ENDIF
        RVAL1 = VALUE(1)
        RVAL2 = VALUE(2)
        RVAL3 = VALUE(3)
        RVAL4 = VALUE(4)
        RETURN
      ENDIF
      GOTO 50
  200 WRITE (LOUT, 8002)
      STOP
 8001 FORMAT ('KEY WORD - ' , A7, ' - NOT FOUND.  PROGRAM STOPPED.')
 8002 FORMAT ('UNEXPECTED END TO INPUT FILE ENCOUNTERED.')
      END

      SUBROUTINE MIX (KK, FLRT1, FLRT2, Z1, Z2, ICKWRK, RCKWRK)
```

```
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     This subroutine calculates the mixed temperature of two streams
C     along with the mixed mass flow rate and mass fractions
C
C     Input:
C               FLRT1/2 flow rate of stream 1/2
C               Z1/Z2 temperature and mass fractions of stream 1/2
C
C     Output:   Z1, FLRT1
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)
      PARAMETER (KMAX = 250)
      DIMENSION Z1(*), Z2(*), HMS1(KMAX), HMS2(KMAX), ZM(KMAX),
     1          ICKWRK(*), RCKWRK(*)
C
C     Enthalpies and mean heat capacities of each stream are found
C
      CALL CKHMS(Z1(1), ICKWRK, RCKWRK, HMS1)
      CALL CKHMS(Z2(1), ICKWRK, RCKWRK, HMS2)
      CALL CKCPBS(Z1(1), Z1(2), ICKWRK, RCKWRK, CPAV1)
      CALL CKCPBS(Z2(1), Z2(2), ICKWRK, RCKWRK, CPAV2)
C
C     Add masses and enthalpy of streams
C
      DATA LOUT/6/
C      WRITE (LOUT,'(2X,A10,2X,2F8.0)')'Init Temp',Z1(1),Z2(1)
      HT = 0
      FLRTM = FLRT1 + FLRT2
      DO 100 I = 1, KK
        HT = HT + Z1(I+1)*HMS1(I)*FLRT1 + Z2(I+1)*HMS2(I)*FLRT2
        ZM(I+1) = (Z1(I+1)*FLRT1 + Z2(I+1)*FLRT2)/FLRTM
  100 CONTINUE
C
C     Calculate first guess for mix temperature
C
      ZM(1) = (FLRT1*CPAV1*Z1(1) + FLRT2*CPAV2*Z2(1))/
     1        (FLRT1*CPAV1 + FLRT2*CPAV2)
      CALL CKCPBS(ZM(1), Z1(2), ICKWRK, RCKWRK, CPAV1I)
      CALL CKCPBS(ZM(1), Z2(2), ICKWRK, RCKWRK, CPAV2I)
      CPAV1 = (CPAV1 + CPAV1I)/2
      CPAV2 = (CPAV2 + CPAV2I)/2
      ZM(1) = (FLRT1*CPAV1*Z1(1) + FLRT2*CPAV2*Z2(1))/
     1        (FLRT1*CPAV1 + FLRT2*CPAV2)
C      WRITE (LOUT,'(2X, A10,2X,F8.3,E13.6)')'1st guess',ZM(1),HT
C
C     Interpolate to get final mixing temperature
C
      T_HIGH = 10000
      T_LOW = 0
      DO 200 J = 0,100
        CALL CKHBMS(ZM(1), ZM(2), ICKWRK, RCKWRK, HAVE)
        HTG = HAVE*FLRTM
        DIFF = (HT - HTG)/ABS(HT)
```

```
        IF (ABS(DIFF) .LE. 0.0001) GOTO 400
        ZM(1) = (1. + 0.1*DIFF)*ZM(1)
C        WRITE (LOUT,'(2X,A10,2X,F8.3,E13.6,E13.6)')'next guess',
C     1          ZM(1),HTG,DIFF
  200 CONTINUE
C
C     bisection technique if newton fails
C
      DO 300 J = 0,100
        ZM(1) = (T_HIGH + T_LOW)/2
        CALL CKHBMS(ZM(1), ZM(2), ICKWRK, RCKWRK, HAVE)
        HTG = HAVE*FLRTM
        DIFF = (HT - HTG)/ABS(HT)
        IF (DIFF .LT. 0.0) THEN
           T_HIGH = ZM(1)
        ELSE
           T_LOW = ZM(1)
        ENDIF
        IF (ABS(DIFF) .LE. 0.0001) GOTO 400
C        WRITE (LOUT,'(2X,A10,2X,F8.3,E13.6,E13.6)')'next guess',
C     1          ZM(1),HTG,DIFF
  300 CONTINUE
      WRITE(LOUT,350)
  350 FORMAT(1X,'TEMPERATURE NOT CONVERGED IN MIX')
      STOP
  400 CONTINUE
      FLRT1 = FLRTM
      DO 500 I = 1, KK+1
        Z1(I) = ZM(I)
  500 CONTINUE
      RETURN
      END
C
C
      SUBROUTINE SYMINDICIES (LIN, LOUT, KK, KSYM, KEYWORD,
     1                         INDICIES, IDX)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     SYMINDICIES returns the indicies of the selected species in the
C     line after the KEYWORD.
C
C     Inputs:
C           LIN   integer, input file unit
C           LOUT    integer, output file unit
C           KK    integer, number of species
C           KSYM  character, array of species strings
C           KEYWORD character, prefix keyword
C
C     Output:
C           INDICIES integer, array of indicies of selected species
C           IDX    integer, number of selected species
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z), INTEGER (I-N)
```

```
      PARAMETER (KMAX=250)
      DIMENSION INDICIES(*)
      CHARACTER KSYM(*)*16, LINE*80, KEYWORD*(*)
      LOGICAL KERR
      DATA KERR /.FALSE./
C
C     Read Line
C
      REWIND (LIN)
   50 CONTINUE
      READ (LIN, '(A)', END=200) LINE
      IF (INDEX (LINE, 'END ') .EQ. 1) THEN
        WRITE (LOUT, 8001) KEYWORD
        RETURN
      ENDIF
      IF (INDEX (LINE, 'REM ') .EQ. 1) GOTO 50
C
C     Check keyword
C
      IF (INDEX (LINE, KEYWORD) .EQ. 1) THEN
        WRITE (LOUT, '(5X,A50)') LINE
      READ (LIN, '(A)', END = 200) LINE
        WRITE (LOUT, '(5X,A50)') LINE
        CALL CKCRAY (LINE, KK, KSYM, LOUT, KMAX, INDICIES, IDX, KERR)
        IF (KERR) THEN
          WRITE(LOUT, 8003) KEYWORD
          STOP
        ENDIF
        RETURN
      ENDIF
      GOTO 50
  200 WRITE (LOUT, 8002)
      STOP
 8001 FORMAT ('KEY WORD - ' , A7, ' - NOT FOUND.')
 8002 FORMAT ('UNEXPECTED END TO INPUT FILE ENCOUNTERED.')
 8003 FORMAT ('ERROR IN CKCRAY FOR LINE FOLLOWING KEYWORD - ', A10)
      END
C
C
      FUNCTION EQUIV_RATIO (KK, YF, YA, YK, YI, FLRTF, FLRTA, FLRTK,
     1                      FLRTI, KELE, KSYM, IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Returns fuel/air equivalence ratio of input streams.  Based on
C     complete combustion products of CO2, H2O, HCL, and N2.
C
C     Inputs:
C     KK          number of species
C     YF,YA,YK,YI       arrays of mass fractions of 4 streams
C     FLRT?       flow rates (g/s) of streams
C       KELE            character array of element symbols
C     KSYM        character arrays of species symbols
C     IWORK,RWORK integer and double work arrays
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
```

```
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (MM =6, KMAX = 250)
      DIMENSION YF(*), YA(*), YK(*), YI(*), IWORK(*), RWORK(*),
     1          NCE(MM,KMAX), Y(KMAX), X(KMAX), IELE(MM)
      CHARACTER KELE(MM)*2, KSYM(*)*16
C
C     initialize variables, and get element matrix
C
      TCL = 0
      TO = 0
      TC = 0
      TH = 0
      CALL CKNCF (MM, IWORK, RWORK, NCE)
C
C     find indices of key elements
C
      CALL CKCOMP ('C',KELE,MM,IELE(1))
      CALL CKCOMP ('H',KELE,MM,IELE(2))
      CALL CKCOMP ('CL',KELE,MM,IELE(3))
      CALL CKCOMP ('O',KELE,MM,IELE(4))
C
C     Combine Streams
C
      DO 50 K = 1,KK
         Y(K) = (YF(K)*FLRTF + YA(K)*FLRTA + YK(K)*FLRTK +YI(K)*FLRTI)/
     1          (FLRTF + FLRTA + FLRTK + FLRTI)
  50  CONTINUE
C
C     Convert to Mole Fractions
C
      CALL CKYTX (Y, IWORK, RWORK, X)
C
C     count species in various categories for entire stream on 1 mol
basis
C
      DO 100 K = 1,KK
         IC = NCE(IELE(1),K)
         IH = NCE(IELE(2),K)
         ICL = NCE(IELE(3),K)
         IO = NCE(IELE(4),K)
         TCL = TCL + X(K)*ICL
         TC = TC + X(K)*IC
         TH = TH + X(K)*IH
         TO = TO + X(K)*IO
 100  CONTINUE
      CALL CKCOMP ('O2', KSYM, KK, IO2)
      TO2 = X(IO2)
C
C     subtract free oxygen counted in previous loop
C
      TO = TO - 2*TO2
C
C     Find required O2 and PHI
C
      REQUIRED_O2 = TC + 0.25*(TH -TCL) - 0.5*TO
      EQUIV_RATIO = REQUIRED_O2/TO2
      RETURN
```

```
      END
C
C
      SUBROUTINE SAVER (KK, Z, X, SERIES, FILE1, FILE2, FILE3, FILE4,
     1                  KSYM, KELE, IWORK, RWORK, IPIC, IRAD, ICEM,
     2                  IK1, IK2, IK3, PHI, FREQ, TAU, FLRT, P, TEMP_RMS)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     SAVER saves mole fractions of designated species in formated text
C     files and saves both mole and mass fractions of all species in
C     an unformatted binary file.
C
C     Input:
C
C           KK    integer, number of species
C           Z     double array, temp and mass fractions
C           X     double array, mole fractions
C           SERIES      logical, .TRUE. overwrites and makes header
C           FILE? character, output files
C           KSYM  character array, specie symbols
C           KELE  character array, element symbols
C           IWORK integer array, working array
C           RWORK double array, working array
C           IPIC  integer arrray, indicies of selected species
C           IRAD,ICEM   same
C           IK?   integer, number of indicies in respective
C                 arrays
C           PHI   double, overall fuel/air equivalence ratio
C           FREQ  double, turbulent frequency (Hz)
C           TAU   double, mean residence time (sec)
C           FLRT  double, mass flow rate (g/sec)
C           P     double, reactor pressure (atm)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER(I-N)
C
      PARAMETER (LFILE1 = 31, LFILE2 = 32, LFILE3 = 33, LFILE4 =34,
     1           LOUT = 6)
      DIMENSION Z(*), X(*), IWORK(*), RWORK(*), IPIC(*), IRAD(*),
     1          ICEM(*)
      CHARACTER KSYM(*)*16, KELE(*)*2, FILE1*(*), FILE2*(*), FILE3*(*),
     1          FILE4*(*), LINE*80
      LOGICAL SERIES
      COMMON /RCONS/ PATM, RU, TA, PA, UAV, PQUAV, TCOOL, BM, CM
C
C     Grouped CEMs
C
      CALL GROUP (KK, X, KELE, IWORK, RWORK, THC, TCLC)
C
C     open output files
C
      OPEN (UNIT = LFILE1, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE1)
      OPEN (UNIT = LFILE2, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE2)
```

```
      OPEN (UNIT = LFILE3, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE3)
      OPEN (UNIT = LFILE4, FORM='UNFORMATTED', STATUS = 'UNKNOWN',
     1      FILE = FILE4)
      IF (SERIES) THEN
           WRITE (LFILE1,7510) 'TEMPERATURE', (KSYM(IPIC(K)), K=1,IK1)
           WRITE (LFILE2,7510) 'TEMPERATURE', (KSYM(IRAD(K)), K=1,IK2)
           WRITE (LFILE3,7510) 'TEMPERATURE', 'RMS TEMP', 'PHI',
     1          'FREQUENCY', 'TAU', 'Volume', 'Heat Trans', 'Heat Trans',
     2          'Mass Flow','THC', 'TCLC', (KSYM(ICEM(K)), K=1,IK3)
           WRITE (LFILE4) (KSYM(K), K=1,KK)
           WRITE (LFILE4)
           WRITE (LFILE4)
      ELSE
           DO 200 I = 1,100
              READ (LFILE1,7516,END=220) LINE
              READ (LFILE2,7516,END=220) LINE
              READ (LFILE3,7516,END=220) LINE
              READ (LFILE4,END=220)
              READ (LFILE4,END=220)
              READ (LFILE4,END=220)
  200      CONTINUE
           WRITE (LOUT,*) FILE1, "NO END FOUND!"
           STOP
  220      CONTINUE
           BACKSPACE (LFILE1)
      ENDIF
      CALL CKRHOX (P*PATM, Z(1), X, IWORK, RWORK, RHO)
      VOL = TAU*FLRT/RHO
      UA = UAV*VOL/41868000
      PQUA = PQUAV*VOL/41868000
      WRITE (LFILE1,7512) Z(1), (X(IPIC(K)), K=1,IK1)
      WRITE (LFILE2,7512) Z(1), (X(IRAD(K)), K=1,IK2)
      WRITE (LFILE3,7513) Z(1), TEMP_RMS, PHI, FREQ, TAU, VOL, UA,
     1                PQUA, FLRT, THC, TCLC, (X(ICEM(K)), K=1,IK3)
      WRITE (LFILE4) Z(1), PHI, FREQ, THC, TCLC
      WRITE (LFILE4) (X(K), K=1,KK)
      WRITE (LFILE4) (Z(K), K=1,KK+1)
  230 CONTINUE
      IF (SERIES) ENDFILE (LFILE1)
      IF (SERIES) ENDFILE (LFILE2)
      IF (SERIES) ENDFILE (LFILE3)
      IF (SERIES) ENDFILE (LFILE4)
      CLOSE (LFILE1, STATUS ='KEEP')
      CLOSE (LFILE2, STATUS='KEEP')
      CLOSE (LFILE3, STATUS='KEEP')
      CLOSE (LFILE4, STATUS='KEEP')
 7510 FORMAT (2X, 31(' "',A16,'"'))
 7512 FORMAT (4X, F11.2, 3X, 30(5X,E14.6))
 7513 FORMAT (4X, F11.2, 2X, 8(5X,F14.4), 30(5X, E14.6))
 7516 FORMAT (A80)
      RETURN
      END

      SUBROUTINE ENERGY_BAL (TEMP, Y, FLRT, POINT, IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
C
C     Subroutine ENERGY_BAL logs the specific enthalpy, temperature,
C     flowrate, and total enthalpy of a stream at a designated point.
C
C     Input:
C           TEMP  double, temperature of the stream
C           Y     double array, mass fractions
C           FLRT  double, flow rate in g/s
C           POINT character, location of the stream
C           IWORK integer working array
C           RWORK double working array
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (LERG = 12)
      DIMENSION Y(*), IWORK(*), RWORK(*)
      CHARACTER POINT*(*), LINE*80
      CALL CKHBMS (TEMP, Y, IWORK, RWORK, HAVE)
      HRT = HAVE * FLRT
      OPEN (UNIT = LERG, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = 'energy_bal')
      DO 100 I = 1,1000
         READ (LERG,7502,END=120) LINE
  100 CONTINUE
      WRITE (LOUT,*) FILE1, "NO END FOUND!"
      STOP
  120 CONTINUE
      WRITE (LERG, 7501) POINT, TEMP, HAVE, FLRT, HRT
      CLOSE (LERG, STATUS ='KEEP')
 7501 FORMAT (A10,F7.0,' K ',E9.3,' erg/g ',E9.3,' g/s ',E9.3,' erg/s')
 7502 FORMAT (A80)
      RETURN
      END
C
C
      SUBROUTINE ELE_BAL (KK, X, KELE, FLRT, POINT, IWORK, RWORK)
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C     Gives the mass flow rate of six elements at point POINT.  Appends
C     results in file 'element_bal'.
C
C     Input:
C                KK              (integer) number of species
C                X               (double) mole fraction array
C                KELE            (character) array of element strings
C                FLRT            (double) total mass flowrate in g/s
C                POINT           (character) the sample point
C                IWORK           (integer) working array
C                RWORK           (double) working array
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
```

```
      PARAMETER (LELE = 12, MM = 6, KMAX = 250)
      DIMENSION X(*), IWORK(*), RWORK(*), NCE(MM,KMAX), IELE(MM)
      CHARACTER KELE(*)*2, POINT*(*), LINE*120
      OPEN (UNIT = LELE, FORM='FORMATTED', STATUS = 'UNKNOWN',
     1      FILE = 'element_bal')
      DO 100 I = 1,1000
         READ (LELE,8502,END=120) LINE
  100 CONTINUE
      WRITE (LOUT,*) FILE1, "NO END FOUND!"
      STOP
  120 CONTINUE
C
C     initialize variables, and get element matrix
C
      TC = 0
      TH = 0
      TO = 0
      TCL = 0
      TAR = 0
      TN = 0
      CALL CKNCF (MM, IWORK, RWORK, NCE)
C
C     find indices of key elements
C
      CALL CKCOMP ('C',KELE,MM,IELE(1))
      CALL CKCOMP ('H',KELE,MM,IELE(2))
      CALL CKCOMP ('O',KELE,MM,IELE(3))
      CALL CKCOMP ('CL',KELE,MM,IELE(4))
      CALL CKCOMP ('AR',KELE,MM,IELE(5))
      CALL CKCOMP ('N',KELE,MM,IELE(6))
C
C     count species in various categories for entire stream
C
      DO 200 K = 1,KK
         TC = TC + NCE(IELE(1),K)*X(K)
         TH = TH + NCE(IELE(2),K)*X(K)
         TO = TO + NCE(IELE(3),K)*X(K)
         TCL = TCL + NCE(IELE(4),K)*X(K)
         TAR = TAR + NCE(IELE(5),K)*X(K)
         TN = TN + NCE(IELE(6),K)*X(K)
  200 CONTINUE
      CALL CKMMWX (X, IWORK, RWORK, WTM)
      TC = TC * FLRT/WTM * 12.0112
      TH = TH * FLRT/WTM * 1.00797
      TO = TO * FLRT/WTM * 15.9994
      TCL = TCL * FLRT/WTM * 35.4530
      TAR = TAR * FLRT/WTM * 39.9480
      TN = TN * FLRT/WTM * 14.0067
      WRITE (LELE, 8501) POINT, TC, TH, TO, TCL, TAR, TN
      RETURN
 8501 FORMAT (A10, 2X, '(in g/s)', 2X, 'C ', E9.3, '  H ', E9.3, '  O ',
     1        E9.3, '  CL ', E9.3, '  AR ', E9.3, '  N ', E9.3)
 8502 FORMAT (A120)
      END
C
      FUNCTION UA_FIND (KK, II, TIN, TOUT, TA, X, KSYM, FLRT, PA,
     1                  TAU, IWORK, RWORK)
```

```
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     Returns estimate of UA (ergs/(sec*K)).  Uses three fixed
C     temperature PSRs each with 1/3 of the residence time TAU
C     temperatures are 1) input 2) arithmetic mean 3) exit
C
C                 Q = UA(TLM)      where Q = Hin - Hout
C
C     TLM is the co-current log mean between the input temperature, the
C     output temperature and the ambient temperature.
C
C     Inputs:
C     KK            integer, number of species
C     II            integer, number of reactions
C     TIN           double, inlet temperature (K)
C     TOUT          double, desired outlet temperature (K)
C     TA            double, ambient temperature (K)
C     X             double array(KK), mole fractions
C     KSYM          character array of specie symbols
C     FLRT          double, flow rate (g/s) of inlet stream
C     PA            double, ambient pressure (atm)
C     TAU           double, residence time of the reactor
C     IWORK,RWORK integer and double work arrays
C
C     Output:
C     UA            double, heat transfer coefficient (erg/s*K)
C
C     Scratch:
C     XSCT          double array(K), outlet mole fractions
C     ZSCT          double array(K+1), outlet temp, mass frac
C     HAVE_IN       double, average input enthalpy
C     HAVE_OUT      double, average output enthalpy
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
      PARAMETER (LOUT=6, KMAX = 250)
      DIMENSION X(*), IWORK(*), RWORK(*), XSCT(KMAX), ZSCT(KMAX)
      CHARACTER KSYM(*)*16
C
C     Find Mass Fractions and Mean Input Enthalpy
C
      DO 200 K = 1, KK
         XSCT(K) = X(K)
  200 CONTINUE
      ZSCT(1) = TIN
      CALL CKXTY(X, IWORK, RWORK, ZSCT(2))
      CALL CKHBMS(ZSCT(1), ZSCT(2), IWORK, RWORK, HAVE_IN)
C
C     A series of 3 PSRs to approximate PFR output composition
C
      TAU_I = TAU/3
      TEMP = TIN
      CALL PDPSR(97, KK, II, ZSCT(1), XSCT, KSYM, FLRT, PA, TAU_I,
     1           .TRUE., 0, TEMP, .FALSE., .TRUE., .FALSE., ZSCT)
```

```
      CALL CKYTX(ZSCT(2), IWORK, RWORK, XSCT)
      TEMP = (TIN + TOUT)/2
      CALL PDPSR(97, KK, II, ZSCT(1), XSCT, KSYM, FLRT, PA, TAU_I,
     1            .TRUE., 0, TEMP, .FALSE., .TRUE., .FALSE., ZSCT)
      CALL CKYTX(ZSCT(2), IWORK, RWORK, XSCT)
      TEMP = TOUT
      CALL PDPSR(97, KK, II, ZSCT(1), XSCT, KSYM, FLRT, PA, TAU_I,
     1            .TRUE., 0, TEMP, .FALSE., .TRUE., .FALSE., ZSCT)
C
C     Output enthalpy
C
      CALL CKHBMS(ZSCT(1), ZSCT(2), IWORK, RWORK, HAVE_OUT)
C
C     Net enthalpy loss
C
      QLOSS = FLRT*(HAVE_IN - HAVE_OUT)
C
C     Log mean temperaure and estimated UA
C
      TLM = (TIN - TOUT)/LOG((TIN - TA)/(TOUT - TA))
      UA_FIND = QLOSS/TLM
      RETURN
      END
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C     END OF PROGRAM
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

APPENDIX E

RTD RECONSTRUCTION AND PARAMETER FITTING TECHNIQUES AND PROGRAMS

This appendix contains several MATLAB script programs used to identify the RTDs and fit model parameters. MATLAB uses a C-like scripting language that can deal with entire vectors and matrices in a single statement. The documentation describes the program direction, but does not explain individual MATLAB calls.

E.1 Fitpoly: a MATLAB M-File for a 5 Parameter Model Fit

```
function [g1,g2,g3] = fitpoly6(model,t,U1,Y1,t1,U2,Y2,t2,U3,Y3,t3)
%
%  [g1 g2 g3] = FITPOLY6(model,t,U1,Y1,t1,U2,Y2,t2,U3,Y3,t3)
%
%  FITPOLY fits a 5 parameter model the RTD data from three input - output
%  combinations in the mixing chamber of the EPA-AEERL Combustion Divisions
%  RKIS.  The input-output combinations are:
%
%                        1.      B1 to 4
%                        2.      C  to 4
%                        3.      B1 to 3
%
%  FITPOLY uses a nonlinear curve fit routine that uses the Nelder-Meade simplex
%  search algorithm (FMIN) to minimize the a merit function designed to
%  find parameters based on a least squares criterion.  The merit function is the
%  sum of the squares of the errors between the model and the data.  The response
%  of three separate input and output combinations are fitted simultaneously.
%
%  FITPOLY fits the integrated data.  The model outputs are simulated by the
%  control toolbox function LSIM.  The integrated analyzer response is the input
%  function.  The output is subtracted from averaged normalized step data and
%  squared to give the merit function.  Required input is:
%
%       U1, U2, U3      averaged analyzer data corresponding to t
%
```

```
%
%       Y1, Y2, Y3      ordered normalized step-down system responses
%
%       t1, t2, t3      time vectors correcponding to the above
%
%       t               the time vector
%
%     model             string of the model transfer function as such:
%                       [Y1_pred,Y2_pred,Y3_pred,g1,g2,g3] = model(t,U1,U2,U3,c)
%                       where c is the vector of parameters
%
% Additionally a function err = modelf(c0) must be created that returns the squared
% error of the model.
%
echo on
clc
dt = 0.05;
%
% Set up evaluation string to analyze model results
%
evalstr = [model,'(t,U1,U2,U3,c)']
%
% model string for FMIN minimization.  This model produces a merit function based
% on least square criterion that FMIN minimizes
%
model = [model,'f'];
n = length(t);
%
% Trim analyzer response based on the lengh of t, assumes t begins at 0
%
U1 = U1(1:n);
U2 = U2(1:n);
U3 = U3(1:n);
%
% Trim system response data within range of t
%
indicies = find((t1>=min(t))&(t1<=max(t)+0.05));
Y1 = Y1(indicies);
t1 = t1(indicies);
indicies = find((t2>=min(t))&(t2<=max(t)+0.05));
Y2 = Y2(indicies);
t2 = t2(indicies);
indicies = find((t3>=min(t))&(t3<=max(t)+0.05));
Y3 = Y3(indicies);
t3 = t3(indicies);
%
% Pass the Data to the Model
```

```
%
global t U1 Y1 t1 U2 Y2 t2 U3 Y3 t3
%
%  Set up graphs and plot handles
%
  global Axeshandle1 Plothandle1 Axeshandle2 Plothandle2 Axeshandle3 Plothandle3
  figure('Name','Fit to Normalized System Step Response');
  hold on;
  Axeshandle1 = subplot(2,2,1);
  plot(t1,Y1,'c-','EraseMode','none');
  xlabel('B1 to 4: Time (sec)'),
  ylabel('Cumulative Probability');
  Axeshandle2 = subplot(2,2,2);
  plot(t2,Y2,'c-','EraseMode','none');
  xlabel('C to 4: Time (sec)'),
  ylabel('Cumulative Probability');
  Axeshandle3 = subplot(2,2,3);
  plot(t3,Y3,'c-','EraseMode','none');
  xlabel('B1 to 3: Time (sec)'),
  ylabel('Cumulative Probability');
  axes(Axeshandle1)
  set(Axeshandle1,'NextPlot','add')
  Plothandle1 = plot(t1,Y1,'y-','EraseMode','xor');
  axes(Axeshandle2)
  set(Axeshandle2,'NextPlot','add')
  Plothandle2 = plot(t2,Y2,'y-','EraseMode','xor');
  axes(Axeshandle3)
  set(Axeshandle3,'NextPlot','add')
  Plothandle3 = plot(t3,Y3,'y-','EraseMode','xor');
%
%  Optimizeparameters
%
  c00 = [2   4  4   .8   0]';
  op = foptions;
  op(18) = 1;
  op(1) = 1;
  op(2) = 1e-6;
  op(14) = 5000;
  c = fmins(model,c00,op,[]);
%   c = c00;
  a1 = c(1);
  a2 = c(2);
  a3 = c(3);
  a4 = c(4);
  a0 = c(5);

  [Y1_pred,Y2_pred,Y3_pred,g1_pred,g2_pred,g3_pred] = eval(evalstr);
```

```
%
%  Replot step responses for the record
%
  figure('Name','Fit to Normalized System Step Response Data');
  plot(t1,Y1,'.',t,Y1_pred,'-');
  xlabel('B1 to 4: Time (sec)'),
  ylabel('Cumulative Probability');
  figure('Name','Fit to Normalized System Step Response Data');
  plot(t2,Y2,'.',t,Y2_pred,'-');
  xlabel('C to 4: Time (sec)'),
  ylabel('Cumulative Probability');
  figure('Name','Fit to Normalized System Step Response Data');
  plot(t3,Y3,'.',t,Y3_pred,'-');
  xlabel('B1 to 3: Time (sec)'),
  ylabel('Cumulative Probability');
%
%  Analysis of residuals
%
  residual1 = Y1 - interp1(t,Y1_pred,t1,'spline');
  residual2 = Y2 - interp1(t,Y2_pred,t2,'spline');
  residual3 = Y3 - interp1(t,Y3_pred,t3,'spline');
  figure('Name','Model Residuals');
  subplot(2,2,1),
  plot(t1,residual1,'r-');
  xlabel('B1 to 4: Time (sec)'),
  ylabel('Cumulative Probability');
  subplot(2,2,2),
  plot(t2,residual2,'r-');
  xlabel('C to 4: Time (sec)'),
  ylabel('Cumulative Probability');
  subplot(2,2,3),
  plot(t3,residual3,'r-');
  xlabel('B1 to 3: Time (sec)'),
  ylabel('Cumulative Probability');
%
%  Correlation of the residuals
%
  rxx1 = correla2(residual1,residual1);
  rxx2 = correla2(residual2,residual2);
  rxx3 = correla2(residual3,residual3);
  figure('Name','Auto Correlation of the Residuals');
  subplot(2,2,1),
  plot(t1,rxx1,'r-');
  xlabel('B1 to 4: Time (sec)'),
  subplot(2,2,2),
  plot(t2,rxx2,'r-');
  xlabel('C to 4: Time (sec)'),
```

```
  subplot(2,2,3),
  plot(t3,rxx3,'r-');
  xlabel('B1 to 3: Time (sec)');
%
%  Compute the covariance matrix and estimate the confidence bounds
%
%
%  Compute numeric derivatives wrt a
%
  d1 = 1/a1;
  d2 = 1/a2;
  d3 = 1/a3;
  ah4 = a4 + 0.0001;
  ah0 = a0 + 0.0001;
  dh1 = d1 + 0.0001;
  dh2 = d2 + 0.0001;
  dh3 = d3 + 0.0001;
  ah1 = 1/dh1;
  ah2 = 1/dh2;
  ah3 = 1/dh3;
  h = ah0 - a0;

  c = [a1 a2 a3 ah4 a0];
  [Y1_pred_h,Y2_pred_h,Y3_pred_h,g1_pred,g2_pred,g3_pred] = eval(evalstr);
  dyda4 = [(Y1_pred_h - Y1_pred)/h;(Y2_pred_h - Y2_pred)/h;(Y3_pred_h - Y3_pred)/h];

  c = [ah1 a2 a3 a4 a0];
  [Y1_pred_h,Y2_pred_h,Y3_pred_h,g1_pred,g2_pred,g3_pred] = eval(evalstr);
  dyda1 = [(Y1_pred_h - Y1_pred)/h;(Y2_pred_h - Y2_pred)/h;(Y3_pred_h - Y3_pred)/h];

  c = [a1 ah2 a3 a4 a0];
  [Y1_pred_h,Y2_pred_h,Y3_pred_h,g1_pred,g2_pred,g3_pred] = eval(evalstr);
  dyda2 = [(Y1_pred_h - Y1_pred)/h;(Y2_pred_h - Y2_pred)/h;(Y3_pred_h - Y3_pred)/h];

  c = [a1 a2 ah3 a4 a0];
  [Y1_pred_h,Y2_pred_h,Y3_pred_h,g1_pred,g2_pred,g3_pred] = eval(evalstr);
  dyda3 = [(Y1_pred_h - Y1_pred)/h;(Y2_pred_h - Y2_pred)/h;(Y3_pred_h - Y3_pred)/h];

  c = [a1 a2 a3 a0 ah0];
  [Y1_pred_h,Y2_pred_h,Y3_pred_h,g1_pred,g2_pred,g3_pred] = eval(evalstr);
  dyda0 = [(Y1_pred_h - Y1_pred)/h;(Y2_pred_h - Y2_pred)/h;(Y3_pred_h - Y3_pred)/h];

%
%  Model Variance
%
  df = length([residual1;residual2;residual3])-5
  sigma_sq = sum([residual1;residual2;residual3].^2);
```

```
  sigma_sq = sigma_sq/df
%
%  Obtain variance vector and differentiate system response data
%  and create variance matrix
%
  [Y1s,y1,sigma1] = sgfilt2(t1,Y1,t,10,2);
  [Y2s,y2,sigma2] = sgfilt2(t2,Y2,t,10,2);
  [Y3s,y3,sigma3] = sgfilt2(t3,Y3,t,10,2);
  variance = [sigma1;sigma2;sigma3].^2;
  variance = [ variance variance variance variance variance]';
%
%  Create Covariance Matrix
%
  dY = [dyda0 dyda1 dyda2 dyda3 dyda4];
  alpha = dY'./variance*dY
  size(alpha)
  C = inv(alpha)
  Formal_Variances = diag(C)
%
%  Standard Errors
%
  theta1_se = sqrt(C(1,1));
  tau1_se = sqrt(C(2,2));
  tau2_se = sqrt(C(3,3));
  tau3_se = sqrt(C(4,4));
  tau4_se = sqrt(C(5,5));
  tau1 = d1;
  tau2 = d2;
  tau3 = d3;
  theta1 = a0;
  tau4 = a4;
  fprintf(1,'theta1=\t%6.4f [%6.4f]\n',theta1, theta1_se)
  fprintf(1,'tau1 =\t%6.4f [%6.4f]\n',tau1, tau1_se)
  fprintf(1,'tau2 =\t%6.4f [%6.4f]\n',tau2, tau2_se)
  fprintf(1,'tau3 =\t%6.4f [%6.4f]\n',tau3, tau3_se)
  fprintf(1,'tau4 =\t%6.4f [%6.4f]\n',tau4, tau4_se)
%
break
%
%  Comparison to differentiated data
%
%
%  differentiate and normalize analyzer response using a 10-2 SG filter
%
  [Us,u1] = sgfilter(t,U1,10,2);
  [Us,u2] = sgfilter(t,U2,10,2);
  [Us,u3] = sgfilter(t,U3,10,2);
```

```
  U1 = u1/trapz(t,u1);
  U2 = u2/trapz(t,u2);
  U3 = u3/trapz(t,u3);
%
%  normalize differentiated system responses
%
  y1 = y1/trapz(t,y1);
  y2 = y2/trapz(t,y2);
  y3 = y3/trapz(t,y3);
%
%  find system impulse response
%
  c = [a1 a2 a3 a4 a0];
  [Y1_pred,Y2_pred,Y3_pred,g1_pred,g2_pred,g3_pred] = eval(evalstr);
%
%  Add a 4 sec tail to each function
%
  n = length(t);
  t_add = [t(n)+dt:dt:t(n)+4]';
  t = [t;t_add];
  pad = zeros(length(t_add),1);
  y1 = [y1;pad];
  y2 = [y2;pad];
  y3 = [y3;pad];
  Y1_pred = [Y1_pred;pad];
  Y2_pred = [Y2_pred;pad];
  Y3_pred = [Y3_pred;pad];
  g1_pred = [g1_pred;pad];
  g2_pred = [g2_pred;pad];
  g3_pred = [g3_pred;pad];
  u1 = [u1;pad];
  u2 = [u2;pad];
  u3 = [u3;pad];
  n_old = n;
  n = length(t);
  n_freq = (n-1)/2 +1;
  nqf = 1/(2*dt);
  freq = [0:nqf/(n_freq-1):nqf]';
  figure('Name','System Response');
  plot(t,Y1_pred,'y-',t,y1,'c+');
  xlabel('B1 to 4: Time (sec)'),
  ylabel('Probability Density'),
  axis([0 t(n_old) min(y1) 4/3*max(y1)]);
  figure('Name','System Response');
  plot(t,Y2_pred,'y-',t,y2,'c+');
  xlabel('C to 4: Time (sec)'),
  ylabel('Probability Density'),
```

```
  axis([0 t(n_old) min(y2) 4/3*max(y2)]);
  figure('Name','System Response');
  plot(t,Y3_pred,'y-',t,y3,'c+');
  xlabel('B1 to 3: Time (sec)');
  ylabel('Probability Density'),
  axis([0 t(n_old) min(y3) 4/3*max(y3)]);
%
%  reconstruct transfer function B1 to 4
%
  RangeStr = 'B1 to 4';
  [g1_pred g1_rcst g1 t_shift] = RECONST(RangeStr,u1,Y1_pred,y1,g1_pred,t,dt,n);
  Spec_A(RangeStr,u1,y1,g1,g1_rcst,g1_pred,freq,n_freq,dt)
%
%  Compute Moments B1 to 4
%
  mean1 = trapz(t_shift,t_shift.*g1);
  vari1 = trapz(t_shift,(t_shift-mean1).^2.*g1);
  skew1 = 1/(vari1^(3/2))*trapz(t_shift,(t_shift-mean1).^3.*g1);
%
%  C to 4
%
  RangeStr = 'C to 4';
  [g2_pred g2_rcst g2 t_shift] = RECONST(RangeStr,u2,Y2_pred,y2,g2_pred,t,dt,n);
  Spec_A(RangeStr,u2,y2,g2,g2_rcst,g2_pred,freq,n_freq,dt)
%
%  Compute Moments C to 4
%
  mean2 = trapz(t_shift,t_shift.*g2);
  vari2 = trapz(t_shift,(t_shift-mean2).^2.*g2);
  skew2 = 1/(vari2^(3/2))*trapz(t_shift,(t_shift-mean2).^3.*g2);
%
%  B1 to 3
%
  RangeStr = 'B1 to 3';
  [g3_pred g3_rcst g3 t_shift] = RECONST(RangeStr,u3,Y3_pred,y3,g3_pred,t,dt,n);
  Spec_A(RangeStr,u3,y3,g3,g3_rcst,g3_pred,freq,n_freq,dt)
%
%  Compute Moments B1 to 3
%
  mean3 = trapz(t_shift,t_shift.*g3);
  vari3 = trapz(t_shift,(t_shift-mean3).^2.*g3);
  skew3 = 1/(vari3^(3/2))*trapz(t_shift,(t_shift-mean3).^3.*g3);
%
%   Input
%
  meana1 = trapz(t,t.*u1);
  varia1 = trapz(t,(t-meana1).^2.*u1);
```

```
  skewa1 = 1/(varia1^(3/2))*trapz(t,(t-meana1).^3.*u1);

  meana2 = trapz(t,t.*u2);
  varia2 = trapz(t,(t-meana2).^2.*u2);
  skewa2 = 1/(varia2^(3/2))*trapz(t,(t-meana2).^3.*u2);

  meana3 = trapz(t,t.*u3);
  varia3 = trapz(t,(t-meana3).^2.*u3);
  skewa3 = 1/(varia3^(3/2))*trapz(t,(t-meana3).^3.*u3);
%
%  Response
%
  meanr1 = trapz(t,t.*y1);
  varir1 = trapz(t,(t-meanr1).^2.*y1);
  skewr1 = 1/(varir1^(3/2))*trapz(t,(t-meanr1).^3.*y1);

  meanr2 = trapz(t,t.*y2);
  varir2 = trapz(t,(t-meanr2).^2.*y2);
  skewr2 = 1/(varir2^(3/2))*trapz(t,(t-meanr2).^3.*y2);

  meanr3 = trapz(t,t.*y3);
  varir3 = trapz(t,(t-meanr3).^2.*y3);
  skewr3 = 1/(varir3^(3/2))*trapz(t,(t-meanr3).^3.*y3);
%
  fprintf(1,'\nRTDs\n')
  fprintf(1,'mean1 =  %6.4f \tvariance1 = %6.4f \tskewness1 = %6.4f\n',mean1,vari1,skew1)
  fprintf(1,'mean2 =  %6.4f \tvariance2 = %6.4f \tskewness2 = %6.4f\n',mean2,vari2,skew2)
  fprintf(1,'mean3 =  %6.4f \tvariance3 = %6.4f \tskewness3 = %6.4f\n',mean3,vari3,skew3)
  fprintf(1,'\nAnalyzer Response\n')
  fprintf(1,'mean1 =  %6.4f \tvariance1 = %6.4f \tskewness1 = %6.4f\n',meana1,varia1,skewa1)
  fprintf(1,'mean2 =  %6.4f \tvariance2 = %6.4f \tskewness2 = %6.4f\n',meana2,varia2,skewa2)
  fprintf(1,'mean3 =  %6.4f \tvariance3 = %6.4f \tskewness3 = %6.4f\n',meana3,varia3,skewa3)
  fprintf(1,'\nSystem Response\n')
  fprintf(1,'mean1 =  %6.4f \tvariance1 = %6.4f \tskewness1 = %6.4f\n',meanr1,varir1,skewr1)
  fprintf(1,'mean2 =  %6.4f \tvariance2 = %6.4f \tskewness2 = %6.4f\n',meanr2,varir2,skewr2)
  fprintf(1,'mean3 =  %6.4f \tvariance3 = %6.4f \tskewness3 = %6.4f\n',meanr3,varir3,skewr3)
```

E.2 Example Model Function

```
function [Y1,Y2,Y3,g1,g2,g3] = model1(t,U1,U2,U3,c)
%
%  Returns simulation and step response of model1 with parameter vector c to
%  a set of data given inthe global variable Data.
%
```

```
% model 1 assumes the form of the transfer function of two
% PSRs in series and a delay expressed in the parameter a00
% which is imposed after the LSIM function:
%
%                          a2*a3
% G(s) =        ---------------------- * exp(-a00*s)
%               (s + a2) * (s + a3)
%
% model 2 assumes the form of the transfer function of two
% PSRs in series added to one PSR:
%
%                          a1*a3                              a1
% G(s) =        a0* ------------------------  +  (1 - a0) * --------
%                   (s + a1) * (s + a3)                    (s + a1)
%
% which simplifies to:
%
%               a1*[(1 - a0)*s + a3]
% G(s) =        ----------------------
%               (s + a1) * (s + a3)
%
% model 3 assumes the form of the transfer function of a single PSR
%
%                 a2
% G(s) =        --------
%               (s + a2)
%

a1 = c(1);
a2 = c(2);
a3 = c(3);
a0 = c(4);
a00 = c(5);
num1 = [a3*a2];
den1 = conv([1 a3],[1 a2]);
num2 = [a1*(1-a0),a1*a3];
den2 = conv([1 a1],[1 a3]);
num3 = [a2];
den3 = [1 a2];
Y1 = lsim(num1,den1,U1,t);
Y1 = interp1([[-5:0.05:-0.05]';t]+a00,[zeros(100,1);Y1],t,'linear');
Y2 = lsim(num2,den2,U2,t);
Y3 = lsim(num3,den3,U3,t);
g1 = impulse(num1,den1,t);
g1 = interp1([[-5:0.05:-0.05]';t]+a00,[zeros(100,1);g1],t,'linear');
g2 = impulse(num2,den2,t);
g3 = impulse(num3,den3,t);
```

E.3 Satvisky-Golay Filter

```
function [ys,dydx2,sigma] = sgfilt2(x,y,x2,win,order)
%
%  Satvisky-Golay Windowed Filter (windowed on x2)
%  [ys,dydx2,sigma] = sgfilt2(x,y,x2,win,order)
%
%  Returns the smoothed dependent variable, ys, and first derivative, dydx2
%  wrt x2.  win (integer) specifies the window to the left and right of the
%  point to be smoothed.  order specified the order of the fitted polynomial.
%
%  The SG filter fits a polynomial of specified order to the points in the
%  "window" about a point in x2 using least square criteria.  All pairs of x,y
%  which fall within the bounds of this window are used.  The first derivative
%  is found from the coefficients of the polynomial.  The process is repeated
%  for each individual point.
%
%  sigma is the array of sample standard deviations of y corresponding to the
%  x values within the sequence interval, dx2, about x(i)
%
%  Least square fit is found using QR decomposition
%
if (order > win)
  fprintf('\n\Polynomial order is larger than window size.\n');
  break;
end
[m1,n1] = size(x);
[m2,n2] = size(y);
[m3,n3] = size(x2);
if ((n1 ~= n2) | (m1 ~= m2))
  fprintf('\n\Input vectors must be the same size\n');
  err = 1;
  break;
end
if (n1 > m1)
  x = x';
end
if (n2 > m2)
  y = y';
end
if (n3 > m3)
  x2 = x2';
end
if (min(x2) < min(x)) | (max(x2) > max(x))
```

```
  fprintf('\n\Second time independent vector must lie within the bounds of the first\n');
  err = 1;
  break;
end
n = length(x);
n2 = length(x2);
dydx2 = zeros(n2,1);
ys = zeros(n2,1);
sigma = zeros(n2,1);
order = round(order);
win = round(win);
dx2 = 1.5*(max(x2)-min(x2))/(n2-1);
%
% Create linearized polynomial regression matrix
% also a pseudo regression matrix for x2 to use the
% resulting parameters in a polynomial of order 'order'
%
X = [ones(n,1),x];
X2 = [ones(n2,1),x2];
for i = 2:order
  X = [X x.^i];
  X2 = [X2 x2.^i];
end
%
% solve regression for each point
%
for i = 1:n2
%
% x2 window indicies
%
  win_lower2 = i - win;
  win_upper2 = i + win;
  if (win_lower2 < 1)
    win_lower2 = 1;
  end
  if (win_upper2 > n2)
    win_upper2 = n2;
  end

%
% Indicies of x within the bounds of the x2 window
%
  win_indicies = find((x>=x2(win_lower2))&(x<=x2(win_upper2)));
%
% least Squares using QR factorization solves an over specified
% equation Ax=b where A=X, x=a (the parameters) and b=y
% the estimator (best fit) of y is c
```

```
%
  b = y(win_indicies);
  A = X(win_indicies,:);
  [Q,R] = qr(A);
  c = Q'*b;
  a = R\c;
  ys(i) = a'*X2(i,:)';
  dydx2(i) = [[1:order]'.*a(2:order+1)]'*X2(i,1:order)';
%
%  Find the standard deviations within the intervals of x2
%
  interval = find((x>=x2(i)-dx2)&(x<=x2(i)+dx2));
  sigma(i) = std(y(interval)-(x(interval)-x2(i))*dydx2(i));
end
```

REFERENCES

Angus, R. M., and Leon Lapidus, "Characterization of Multiple Variable Linear Systems from Random Inputs," *AIChE Journal,* **9**, 810 (1963).

Barat, Robert Benedict, *Characterization of the Mixing/Chemistry Interaction in the Toroidal Jet Stirred Combustor*, Ph.D. Thesis, Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge MA (1990).

Barat, Robert B., "Jet-Stirred Combustor Behavior Near Blowout: Observations and Implications," *Combustion Science and Technology*, **84**, 187 (1992).

Bédat, B., F. N. Egolfopoulos, T. Poinsot, "Direct Numerical Simulation of Heat Release and NO_x Formation in Turbulent Nonpremixed Flames," *Combustion ad Flame*, 119, 69 (1999).

Bass, C., R. B. Barat, G. Sacchi, T. Lee, J. P. Longwell, A. F. Sarofim, P. M. Lemieux, "Fundamental Experimental and Modeling Studies of Chlorocarbon Incineration in Afterburners," Work-in-Progress Poster Session, *25th International Symposium on Combustion*, The Combustion Institute, Pittsburgh, PA (1994).

Bédat, B., R. K Cheng, "Experimental Study of Premixed Flames in Intense Isotropic Turbulence," *Combustion and Flame*, **100**, 485 (1995).

Beér, J. M, and K. B. Lee, "The Effect of the Residence Time Distribution on the Performance and Efficiency of Combustors," *Tenth Symposium (International) on Combustion,* The Combustion Institute, Pittsburgh, PA, 1187 (1965).

Beér, J. M., and N. A. Chigier, *Combustion Aerodynamics*, Krieger, Malabar, FL (1972).

Bikas, G., and N. Peters, "Kinetic Modeling of *n*-Decane Combustion and Autoignition," *Combustion and Flame*, **126**, 1456 (2001).

Bozzelli, Joseph W., personal communication, 1996.

Brukh, Roman, Tara Salem, Thana Slanvetpan, Robert Barat, Somenath Mitra, "Process Modeling and On-Line Monitoring of Benzene and Other Species during the Two Stage Combustion of ethylene in Air," Submitted for Publication to *Advances in Environmental Research* (2001).

Cannon, S. M., B. S. Brewster, L. D. Smoot, "Stochastic Modeling of CO and NO in Premixed Methane Combustion," *Combustion and Flame*, **113**, 135 (1998).

Chiang, H., B. Park, J. W. Bozzelli, *Twenty Fifth Symposium (International) on Combustion,* The Combustion Institute, Pittsburgh, PA, (1994).

Chiang, H., Ph.D. Thesis, Department of Chemical Engineering, Chemistry, and Environmental Science, New Jersey Institute of Technology, Newark, NJ (1995).

Chen, J. E., T. Echekki, W. Kollmann, "The Mechanism of Two-Dimensional Pock Formation in Lean Premixed Methane-Air Flames with Implications to Turbulent Combustion," *Combustion and Flame*, **116**, 15 (1999).

Cohen, N., and S. W. Benson, "Estimation of Heats of Formation of Organic Compounds by Additivity Methods," *Chemical Reviews*, **93**, 2419 (1993).

Correa, S.M., "Turbulence-Chemistry Interactions in the Intermediate Regime of Premixed Combustion," *Combustion and Flame*, **93**, 41 (1993).

Correa, S.M., M. E. Braaten, "Parallel Simulations of Partially Stirred Methane Combustion," *Combustion and Flame*, **94**, 469 (1993).

Correa, S.M., "A Direct Comparison of Pair-Exchange and IEM Models in Premixed Combustion," *Combustion and Flame*, **103**, 194 (1995).

Cundy, V. A., T. W. Lester, C. Leger, G. Miller, A. N. Montestruc, S. Acharya, A. M. Sterling, D. W. Pershing, J. S. Lighty, G. D. Silcox, W. D. Owens, "Rotary Kiln Incineration – Combustion Chamber Dynamics," *Journal of Hazardous Materials*, **22**, 195 (1989).

Curl, R. L., "Dispersion and Phase Mixing: 1. Theory and Effect of Simple Reactors," *AIChE Journal*, **9**, 175 (1963).

Danckwerts, P. V., "The Effect of Incomplete Mixing in Homogeneous Reactions," *Chemical Engineering Science*, **8**, 93 (1958a).

Danckwerts, P. V., "Local Residence-Times in Continuous-Flow Systems," *Chemical Engineering Science*, **9**, 78 (1958b).

David, R., "Turbulent Reactive Flow of Liquids in Isothermal Stirred Tanks," *Turbulent Reacting Flows*, P. A. Libby and F. A. Williams, editors, Academic Press, New York, NY (1994).

Dean, Anthony M., and Joseph W. Bozzelli, "Combustion Chemistry of Nitrogen," in *Gas-Phase Combustion Chemistry*, W. C. Gardiner editor, Springer-Verlag, New York, NY (2000).

Dopazo, C. and E. E. O'Brien, "An Approach to the Autoignition of a Turbulent Mixture," *Acta Astronautica*, **1**, 1239 (1974).

Dougherty, Eugene P., Jenn-Tai Hwang, and Herschel Rabitz, "Further Developments and Applications of the Green's Function Method of Sensitivity Analysis in Chemical Kinetics," *Journal of Chemical Physics*, **71**, 1794 (1970).

Draper, N. R., and H. Smith, *Applied Regression Analysis, 3rd Ed.*, John Wiley & Sons, Inc., New York, NY (1998).

Echekki, T., A. R. Kerstein, T. D. Dreeben, "'One-Dimensional Turbulence' Simulation of Turbulent Jet Diffusion Flames: Model Formulation and Illustrative Applications," *Combustion and Flame*, **125**, 1083 (2001).

Ewan, B. C. R., F. Boysan, J. Swithenbank, "Closing the Gap Between Finite Difference and Stirred Reactor Combustor Modeling Procedures," *Twentieth Symposium (Int.) On Combustion*, Combustion Institute, Pittsburgh PA, 541 (1984).

Fogler, Scott H., *Elements of Chemical Reaction Engineering 2d Ed.,* Prentice-Hall, Englewood Cliffs, NJ (1992).

Frank, Paul M., *Introduction to System Sensitivity Theory*, Academic Press, New York, NY (1978).

Glarborg, Peter., Robert J. Kee, Joseph F. Grcar, James A. Miller. 1986. *PSR: A FORTRAN Program for Modeling Well-Stirred Reactors*, Sandia National Laboratories Report SAND86-8209, Livermore, CA (1986).

Hallett, W. L. H., "Swirl Generator for Independent Variation of Swirl and Velocity Profile," *AIAA Journal*, **24**, 1212 (1986).

Heyberger, B., F. Battin-Leclerc, V. Warth, R. Fournet, G. M. Côme, and G.Scacchi, "Comprehensive Mechanism for the Gas-Phase Oxidation of Propene," *Combustion and Flame*, **126**, 1780 (2001).

Himmelblau, D. M., and K. B. Bischoff, *Process Analysis and Simulation, Deterministic Systems*, John Wiley & Sons, New York, NY (1968).

Hsia, T. C., *System Identification: Least-Squares Methods*, Lexington Books, Lexington, MA (1977).

Ho, Wen Pin, Robert B. Barat, and Joseph W. Bozzelli, "Thermal Reactions of CH_2Cl_2 in H_2/O_2 Mixtures: Implications for Chlorine Inhibition of CO Conversion to CO_2," *Combustion and Flame*, **88**, 265 (1992a).

Ho, Wenpin, and Joseph W. Bozzelli, "Validation of a Mechanism for Use in Modeling CH2Cl2 and/or CH3Cl Combustion and Pyrolysis," *Twenty-Forth Symposium (International) on Combustion,* The Combustion Institute, Pittsburgh, PA, 743 (1992b).

Hottel, Hoyt C., and Adel F. Sarofim, *Radiative Transfer*, McGraw-Hill, New York, NY (1967).

Huang, Jiawei and Selim M. Senkan, "Polycyclic Aromatic Hydrocarbon and Soot Formation in Premixed Flames," *Twenty Sixth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, PA (1996).

Kee, R. J., J. F. Grcar, M. D. Smooke, and J. A. Miller, *A FORTRAN Program for Modeling Steady Laminar One-Dimensional Premixed Flames*, Sandia National Laboratories Report SAND85-8240, Livermore, CA (1993).

Kee, R. J., F. M. Rupley, and J. A. Miller, *CHEMKIN-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas Phase Chemical Kinetics*, Sandia National Laboratories Report SAND89-8009B, Livermore, CA (1994).

Kramer, Mark A., Herschel Rabitz, Joseph M. Calo, and Robert J. Kee, "Sensitivity Analysis in Chemical Kinetics: Recent Developments in Computational Comparisons," *International Journal of Chemical Kinetics*, **16**, 559 (1984).

Kridiotis, A. C., J. P. Longwell, A. F. Sarofim, and E. Bar-Ziv, "Application of a Stochastic Model of Imperfect Mixing to the Combustion of Fuel-Lean CO-H_2 Mixtures in Air," *Chemical Engineering Science*, **44**, 1039 (1989).

Lemieux, Paul M., William A. Linak, Joseph A. McSorley, and Jost O. L. Wendt, "Transient Suppression Packaging for Reduced Emissions from Rotary Kiln Incinerators," *Combustion Science and Technology*, **85**, 203 (1992).

Lemieux, P., J. Ryan, C. Bass, and R. Barat, "Emissions of Trace Products of Incomplete Combustion from a Pilot-Scale Incinerator Secondary Combustion Chamber," *Journal of the Air and Waste Management Association*, **46**, 309-316 (1996).

Levenspiel, Octave, *Chemical Reaction Engineering 2d Ed.*, John Wiley & Sons, New York, NY (1972).

Lou, J. C., and Y. S. Chang, "Thermal Oxidation of Chloroform," *Combustion and Flame*, **109**, 188 (1997).

Marinov, N. M., W. J Pitz, C. K Westbrook, A. M. Vincitore, M. J. Castaldi, S. M. Senkan, "Aromatic and Polycyclic Aromatic Hydrocarbon Formation in a Laminar Premixed *n*-Butane Flame," *Combustion and Flame*, **114**, 192 (1998).

Mitra, S., *Development and Evaluation of a Automatic GC Equipped with a Multi-Absorbent Pre-Concentration Device*, Contract # 68-02-4127 Report to US EPA, Research Triangle Park, NC, (1990).

Nasserzadeh, V., J. Swithenbank, C. Schofield, D. W. Scott, A. Loader, "Effects of High Speed Jets and Internal Baffles on the Gas Residence Times in Large Municipal Incinerator," *Environmental Progress*, **13**, 124 (1994).

Nauman, Bruce E., "Residence Time Distributions and Micromixing," *Chemical Engineering Communications*, **8**, 53 (1981).

Nauman, E. B., and B. A. Buffham, *Mixing in Continuous Flow Systems*, John Wiley and Sons, New York, NY (1987).

Nelder, J.A., and R. Mead, "A Simplex Method for Function Minimization," *The Computer Journal*, **7**, 308 (1965).

Nenniger, J. E., A. C. Kridiotis, J. Chomiak, J. P. Longwell, A. F. Sarofim, "Characterization of a Toroidal Well Stirred Reactor," *Twentieth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, PA, 473 (1984).

Norris, A. T., S. B. Pope, "Modeling of Extinction in Turbulent Diffusion Flames by the Velocity-Dissipation-Composition PDF Method," *Combustion and Flame*, **100**, 211 (1995).

O'Neil, P. V., *Advanced Engineering Mathematics, 3rd Ed.*, Wadsworth Publishing Company, Belmont, CA (1991).

Oran, Elaine. S., Jay P. Boris, *Numerical Simulation of Reactive Flow, 2nd Ed.*, Cambridge University Press, Cambridge, UK (2001).

Perry, Robert H., Don W. Green, editors, *Perry's Chemical Engineers' Handbook, sixth Edition*, McGraw-Hill, New York, NY (1984).

Pope, S. B., "Computations of Turbulent Combustion: Progress and Challenges," *Twenty-Third Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, PA, 591 (1990).

Pope, Stephen. B., *Turbulent Flows*, Cambridge University Press, Cambridge, UK (2000).

Press, W. H., S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical Recipes in C, the Art of Scientific Computing, 2d Ed.,* Cambridge University Press, Cambridge, UK (1992).

Ravichandran, M., and F. C. Gouldin, "Residence Time Calculations Using Numerical Simulation of Incineration Flows," *Combustion Science and Technology*, **91**, 257 (1993).

Ritter, Edward R., Joseph W. Bozzelli, and Anthony M. Dean, "Kinetic Study on Thermal Decomposition of Chlorobenzene Diluted in H_2," *Journal of Physical Chemistry*, **94**, 2493 (1990a).

Ritter, Edward, and Joseph W. Bozzelli, "Reactions of Chlorinated Benzenes in H_2 and H_2/O_2 Mixtures: Thermodynamic Implications on Pathways to Dioxin," *Combustion Science and Technology*, **74**, 117 (1990b).

Rooney, W. C., L. T. Biegler, "Design for Model Parameter Uncertainty Using Nonlinear Confidence Regions," *AIChE Journal*, **47**, 1794 (2001).

Saxena, V., S. B. Pope, "PDF Simulations of Turbulent Combustion Incorporating Detailed Chemistry," *Combustion and Flame*, **117**, 340 (1990).

Senkan, S. M., "Survey of Rate Coefficients in the C-H-Cl-O System," *Gas-Phase Combustion Chemistry*, W. C. Gardiner, editor, Springer-Verlag, New York, NY (2000).

Schafer, R. W., R. M. Mersereau, M. A. Richards, "Constrained Iterative Restoration Algorithms," *Proceedings of the IEEE*, **69**, 432 (1981).

Sgro, L. A., C. P. Koshland, D. Lucas, R. F. Sawyer, "Post Flame Reaction Chemistry of Dichloromethane: Variations in Equivalence Ratio and Temperature," *Combustion and Flame*, **120**, 492 (2000).

Smith, G. P., D. M. Golden, M. Frenklach, N. W. Moriarty, B. Eiteneer, M. Goldenberg, C. T Bowman, R. K. Hanson, S. Song, W. C. Gardiner, V. V. Lissianski, Z. Qin, "GRI Mechanism 3.0," http://www.me.berkeley.edu/gri_mech/, (1999).

Spencer, Jordan L., Richard R. Lunt, Stanley A. Leshaw, "Identification of Micromixing Mechanisms in Flow Reactors: Transient Inputs of Reactive Tracers," *Industrial and Engineering Chemistry*, **19**, 135 (1980).

Subramaniam, S., S. B. Pope, "A Mixing Model for Turbulent Reactive Flows based on Euclidean Minimum Spanning Trees," *Combustion and Flame*, **115**, 487 (1998).

Swithenbank, J., I. Poll, M. W. Vincent, D. D. Wright, "Combustion Design Fundamentals," *Fourteenth Symposium (Int.) On Combustion*, Combustion Institute, Pittsburgh PA, 627 (1972).

Syred, N., and J. M. Beér, "Combustion in Swirling Flows: A Review," *Combustion and Flame*, **23**, 143 (1974).

Thomas, Murray J., Brian S. Higgins, Donald Lucas, Catherine R. Koshland, and Robert F. Sawyer, "Phosgene Formation from 1,1,1,-Trichloroethane Oxidation," *Combustion and Flame*, **98**, 350 (1994).

Topps, J. E. C., "An Optical Technique for the Investigation of Flow in Gas Turbine Combustors," *Seventeenth Symposium (International) on Combustion,* The Combustion Institute, Pittsburgh, PA, 347 (1978).

Townsend, D. I., J. D. Wilson, C. N. Park, "Mechanisms for Formation and Options for Control of Emissions of PCDDs and PCDFs from Incineration," *Proceedings of the 1995 International Incineration Conference*, University of California, Irvine, CA, 331 (1995).

Trenholm, A., P. Gorman, G. Jungclaus, *Performance Evaluation of Full-Scale Hazardous Waste Incinerators, Volume I Executive Summary*, U.S. EPA Report EP-600/2-84-181a, Industrial Environmental Research Laboratory, Cincinnati, OH (1984).

Tuma, J. J., *Engineering Mathematics Handbook 3rd Ed.*, McGraw-Hill, New York, NY (1987).

Turns, Stephen R., *An Introduction to Combustion: Concepts and Applications*, McGraw-Hill, New York, NY (1996).

Vaughn, C. B., Ph.D. Thesis, Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA (1988).

Warnatz, J., U. Maas, R. W. Dibble, *Combustion: Physical and Chemical Fundamentals, Modeling and Simulation, Experiments*, Springer-Verlag, Berlin (1999).

Wendt, Jost O. L., and William P. Linak, "Mechanisms Governing Transients from the Batch Incineration of Liquid Wastes in Rotary Kilns," *Combustion Science and Technology*, **61**, 169 (1988).

Westmoreland, P. R., J. B. Howard, J. P. Longwell, and A. M. Dean, "Prediction of Rate Constants for Combustion and Pyrolysis Reactions by Bimolecular QRRK," *AIChE Journal*, **23**, 1971 (1986).

Xu, Jun, Stephen B. Pope, " PDF Calculations of Turbulent Nonpremixed Flames with Local Extinction," *Combustion and Flame*, **123**, 281 (2000).

Zwietering, T. N., "The Degree if Mixing in Continuous Flow Systems," *Chemical Engineering Science*, **11**, 1 (1959).