Microkinetic Modeling of Complex Reaction Networks using Automated Network Generation

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Udit Gupta

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Advised by Prodromos Daoutidis and Aditya Bhan

April 2018

Acknowledgments

I would like to thank my advisors, Prof. Prodromos Daoutidis and Prof. Aditya Bhan, for providing me the opportunity to work on this project and for being patient with my deadlines on the research. They have supported my ideas even when I was unsure of them myself, and were always there for guidance. They gave me enough freedom to pursue my ideas, encouraged collaborations, offered interesting problems to work on, and above all were always available for advice to sort out my problems, technical and otherwise. I would also like to thank Prof. Wei-Shou Hu for his guidance and help with the biochemical reaction network generation work. I also thank Dr. Srinivas Rangarajan for taking out the time to answer my unending doubts. Technical discussions with him really helped me solve issues as well as helped me think of solutions with a totally new perspective. He has been a great resource for learning and will continue to be.

I would also like to thank Tung Le for helping me understand the terminology and concepts in biochemical engineering.

I thank the members of the Daoutidis group, with whom I shared my office, and the Bhan group, with whom I had valuable interaction time-to-time. I would like to specifically thank my colleagues in the Daoutidis group, Dr. Nitish Mittal, Andrew Allman, Manjiri Moharir, Conor O'Brien, Matt Palys, Wentao Tang, Shaaz Khatib, Dr. Davood Pourkargar, and former members Prof. Ana Torres, Dr. Alex Marvin, Dr. Dimitrios Georgis, Dr. Seongmin Heo, Dr. Adam Kelloway, Dr. Michael Zachar with whom I have had innumerable engaging technical discussions on common research problems. I would also thank members of the Bhan group, Anurag Kumar, Andrew

To My Family

Abstract

Complex reaction networks are found in a variety of engineered and natural chemical systems ranging from petroleum processing to atmospheric chemistry and including biomass conversion, materials synthesis, metabolism, and biological degradation of chemicals. These systems comprise of several thousands of reactions and species interrelated through a highly interconnected network. These complex reaction networks can be constructed automatically from a small set of initial reactants and chemical transformation rules. Detailed kinetic modeling of these complex reaction systems is becoming increasingly important in the development, analysis, design, and control of chemical reaction processes. The key challenges faced in the development of a kinetic models for complex reaction systems include (1) multi-time scale behaviour due to presence of fast and slow reactions which introduces stiffness in the system, (2) lack of lumping schemes that scale well with the large size of the network, and (3) unavailability of accurate reaction rate constants (activation energies and pre-exponential factors). Model simplification and order reduction methods involving lumping, sensitivity analysis and time-scale analysis address the challenges of size and stiffness of the system. Although there exist numerical methods for simulation of large scale, stiff models, the use of such models in optimization-based tasks (e.g. parameter estimation, control) results in ill-conditioning of the corresponding optimization task.

This research presents methods, computational tools, and applications to address the two challenges that emerge in the development of microkinetic models of complex reaction networks in the context of chemical and biochemical conversion – (a) identifying

the different time scales within the reaction system irrespective of the chemistry, and (b) identifying lumping and parameterization schemes to address the computational challenge of parameter estimation. The first question arises due to presence of both fast and slow reactions simultaeneously within the system. The second challenge is directly related to the estimation of the reaction rate constants that are unknown for these chemical reaction networks. Addressing these questions is a key step towards modeling, design, operation, and control of reactors involving complex systems.

In this context, this thesis presents methods to address the computational challenges in developing microkinetic models for complex reaction networks. Rule Input Network Generator (RING) [1, 2], a network generation computational tool, is used for the network generation and analysis. First, the stiffness is addressed with the implementation of a graph-theoretic framework. Second, lumping and parameterization schemes are studied to address the size challenge of these reaction networks. A particular lumping and parameterization scheme is used to develop the microkinetic model for an olefin interconversion reaction system. Further, RING is extended for application of biochemical reaction network generation and analysis.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Complex reaction systems are prevalent in many areas of chemical and biochemical transformations [8, 9]. Examples of individual chemical systems include petrochemical processes, biomass conversion, combustion of fuels, nanoparticles synthesis, atmospheric chemistry of volatile organic compounds, degradation of xenobiotics in the environment, and biological systems [10, 11, 12, 13, 14, 15, 16, 17, 18, 19]. These reaction networks are of particular recent interest because of the emergence of new feedstocks and chemistries, e.g. for biomass and methane processing. Biomass, for example, can be converted into a plethora of valuable compounds such as platform chemicals, cosmetics, solvents, pharma- and neutra-ceuticals, etc. using a wide spectrum of chemistries spanning homogeneous and heterogeneous, catalytic and noncatalytic, and thermochemical and biochemical routes [20].

These complex networks show two common characteristics. First, their size is large; for example, combustion of hexadecane – a model diesel compound – can involve up to 6000 species (compounds) and 20,000 reactions [14]. Tropospheric degradation network of volatile organic chemicals can have up to 4000 species and 12,000 reactions [21] while biological systems such as the metabolic network of *Escherichia Coli* is reported to have up to 1000 species and 2000 reactions. Further, in many petrochemical processes such as hydrocracking and fluid catalytic cracking , the reactors convert crude oil feedstock containing several hundred compounds into a variety of products through a complex

reaction network containing several thousand intermediate species and reactions [22, 23]. Second, the species and reactions within the reaction network are highly interconnected because each experimentally observed product can be potentially formed from initial reactants by tens to hundreds of reaction pathways and mechanisms.

Detailed kinetic modeling of complex reaction systems is becoming increasingly important in the development, analysis, design, and control of chemical reaction processes. A detailed kinetic model of such systems can help identify important pathways and therefore can be used to optimize process conditions for achieving the desired product composition and properties. Microkinetic modeling is an essential step towards rigorous design, optimization, and control of these reaction systems; however, the development of microkinetic models, with the underlying parameter estimation problem, is computationally challenging, with two key challenges being model stiffness and size. Stiffness arises from the difference in the order of magnitude of reaction rate constants, while the large model size is due to the large number of species and reactions typically present in such networks. The challenges present in generating the model include (1) multi-time scale behaviour due to presence of fast and slow reactions, (2) lack of lumping schemes that scale well with the large size of the network, and (3) unavailability of accurate reaction rate constants (activation energies and pre-exponential factors) [24]. Model simplification and order reduction methods involving lumping, sensitivity analysis and time-scale analysis address the challenges of size and stiffness of the system [25]. Although there exist numerical methods for simulation of large scale, stiff models, the use of such models in optimization-based tasks (e.g. parameter estimation, control) results in ill-conditioning of the corresponding optimization task.

This research aims to address these two challenges that emerge in the development of microkinetic models of complex reaction networks in the context of chemical and biochemical conversion – (a) identifying the different time scales within the reaction system irrespective of the chemistry, and (b) identifying lumping and parameterization schemes to address the computational challenge of parameter estimation. The first question arises due to presence of both fast and slow reactions simultaeneously within the system. The second challenge is directly related to the estimation of the reaction rate constants that are unknown for these chemical reaction networks. Addressing these questions is a key step towards modeling, design, operation, and control of reactors involving complex systems.

In this context, this thesis presents methods to address the computational challenges

in developing microkinetic models for complex reaction networks. Rule Input Network Generator (RING) [1, 2], a network generation computational tool, is used for the network generation and analysis. First, the stiffness is addressed with the implementation of a graph-theoretic framework. Second, lumping and parameterization schemes are studied to address the size challenge of these reaction networks. A particular lumping and parameterization scheme is used to develop the microkinetic model for an olefin interconversion reaction system. Further, RING is extended for application of biochemical reaction network generation and analysis.

1.1   Time scale analysis of complex reaction networks

In many complex reaction networks the reactions occur on vastly different time scales. Some reactions dominate the initial dynamics and may reach a pseudo-steady state quickly, whereas others occur slowly and may dominate the dynamics on a long time scale. The dynamics of such systems are described by a large number of variables and differential equations with kinetic parameters of widely-differing orders of magnitude. As a result, accurate computations that resolve the fast and slow time scale dynamics for very large networks of the kind that arise in the above systems are computationally challenging. Moreover, the slow dynamics are often of primary interest, and to analyze them one has to construct the governing equations for slowly-varying quantities. In Chapter 3, a graph-theoretic framework is developed for time scale decomposition of complex reaction networks to separate the slow and fast time scales, and to identify pseudo-species that evolve only in the slow time scale. The reaction network is represented using a directed bi-partite graph and cycles that correspond to closed walks are used to identify interactions between species participating in fast/equilibrated reactions. Subsequently, an algorithm which connects the cycles to form the pseudo-species is utilized to eliminate the fast rate terms. These pseudo-species are used to formulate reduced, non- stiff kinetic models of the reaction system. Two reaction systems are considered to show the efficacy of this framework in the context of thermochemical and biochemical processing.

1.2 Network generation and analysis of biochemical reaction networks

Biochemical reaction systems, encompassing enzymes, present tremendous synthetic potential. They form various metabolic pathways and generate thousands of different chemical species in microorganisms, plants and animals. Many of them, especially those involved in catabolism and anabolism, are well characterized with the enzymes, reaction intermediates and products well known (reviewed in [26]). Many others, notably the secondary metabolites and complex glycans, have only scantly been surveyed, with their full range of synthetic potential yet to be explored. With recent advances in genomic science, analytical technology and synthetic biology, we possess the capability of designing, reconstituting and synthesizing new pathways [27, 28, 29]. We also have an unprecedented ability to discover new compounds. With the large repertoire of enzymes and not-yet-fully-characterized biosynthetic genes, the potential number of combinations of pathways that can be formed by those enzyme is enormous. Computational tools are therefore necessary to construct, model, and elucidate the transformations occurring in both natural and synthetic biochemical reaction networks [30, 31, 32]. Automated network generators that identify the reactive motifs in molecules as well as the reaction that chemically modifies these motifs, and further connect series of reactions and molecules into networks, will have wide applications for exploring, identifying new compounds in not-yet fully explored pathways and constructing new synthetic pathways. Chapter 4 describes how Rule Input Network Generator (RING), a network generation computational tool, can be adopted to generate a variety of complex biochemical reaction networks. The reaction language incorporated in RING allows representation of chemical compounds in biological systems with various structural complexity. Complex molecules such as oligosaccharides in glycosylation pathways can be described using a simplified representation of their monosaccharide building blocks and glycosidic bonds. The automated generation and topological network analysis features in RING also allow for: (1) constructing biochemical reaction networks in a rule-based manner, (2) generating graphical representations of the networks, (3) querying molecules containing a particular structural pattern, (4) finding the shortest synthetic pathways to a user-specified species, and (5) performing enzyme knockout to study their effect on the reaction network. Case studies involving two biochemical reaction systems: (1) Synthesis of 2-ketoglutarate from xylose in bacterial cells and (2) N-glycosylation in

mammalian cells are presented to demonstrate the capabilities of RING for robust and exhaustive network generation and the advantages of its post-processing features.

1.3   Microkinetic modeling of Olefin Interconversion on self-pillared pentasil MFI

Chapter 5 presents a microkinetic model of an olefin interconversion reaction system using RING. Specifically, the chapter demonstrates (a) specification of kinetic parameters of each reaction rule in a rule-based manner and (b) specification of chemical functionality-based lumping, to construct and solve a thermodynamically consistent kinetic model subsequent to network generation in RING. Parameters involving rate constants are estimated using sequential optimization by fitting the mathematical model and the experimental data. Kinetic modeling results – the concentration of various products at different points along a plug flow reactor for experimentally-specified reaction conditions are shown. The model captures the trend in concentration profile for each chemical species (ethylene, propylene, C_5 - C_9 aliphatic hydrocarbons, and aromatics). The kinetic parameters estimated in this work lie within the acceptable error limits (within $\sim$ 2-3 orders of magnitude) of that reported in the literature.

Background

In Chapter 1, several examples of complex reaction systems were given. Despite the large size, it was argued that the reaction networks can be constructed from a relatively smaller set of chemical transformation rules. Automated network generators have therefore been developed to construct such networks from initial reactants and pre-specified reaction rules. In this chapter, we provide a detailed discussion of relevant background developments in cheminformatics and state-of-the art in network generation and kinetic modeling.

2.1 Network generation and analysis: a review

Rule-based automated reaction network generators are computational tools that take in a set of molecules as reactants, and iteratively apply the set of input reaction rules, to construct a comprehensive list of possible reactions. Network generators have been developed and applied in different fields such as pyrolysis & oxidation, catalysis, and biological systems. Table 2.1 lists and describes several of them. All automated network generators have five common and essential features [33]. First, an unambiguous representation is required for molecules and reactions. This is usually represented as character strings for input and output [34, 35, 36]. Second, an internal representation of molecules is required, such as molecular trees, adjacency matrices, or chemical graphs,

thereby enabling quick structure manipulation. Adjacency matrix is the most common representation format owing to its simplicity. An adjacency matrix "M" of a molecule is a square matrix containing connectivity and bond order information between every two atoms. Thus, M(i,j) = 0 implies that the ith and j^{th} atoms are not connected while a positive nonzero value would indicate the strength of the bond (1 is a single bond, 2 is a double bond, etc). The diagonal values indicate the number of unpaired electrons in the atoms. The Bond-electron matrix, therefore, is an adjacency matrix. Third, an internal representation of reaction rules that can be applied iteratively on the molecules is required. A common representation scheme is to employ a matrix for reaction rules "R" proposed by Dugundji and Ugi [37] and later used in other tools such as NETGEN [34], BNICE [38, 39, 40, 41]. Baltanas & Froment [42] used a Boolean matrix to represent molecules for generation of networks for modeling paraffin cracking and isomerization on bifunctional catalysts. The Boolean matrix is similar to the adjacency matrix; however, bonds of a higher order (e.g. double bonds) and information on charges (such as +1 for carbenium ions) are stored separately. This method, therefore, is similar to that of Dugundji and Ugi [37]. Transformations in RDL [35] and RDL++ [43], on the other hand, are input by the user as English-language-like statements describing changes in the charge/ bonding of atoms participating in the reaction rule which get directly applied on the internal graph description of molecules. Fourth, all network generators have a generation scheme that iteratively applies the reaction rules to all input and generated molecules so that the resultant network is exhaustive. The scheme should ensure that all possible reactions of a given set of reactants are generated corresponding to that reaction rule. Faulon and Sault [44] describe such a generation scheme as deterministic network generation.

Combinatorial explosion is an important practical problem that can significantly increase execution time and lead to a large proportion of unimportant or improbable reactions. The fifth essential feature of most network generation tools, therefore, is to employ a systematic procedure to curtail this effect. When kinetic parameters are available apriori, quantitative estimates of the magnitude of the reaction rates allows for the identification of "important"/"unimportant" reactions and species that should be included in, or excluded from, the network. For example, the tool NETGEN adopts rate-based [45] network pruning criteria. This requires generation and kinetic modeling in concurrence because the rates calculated on-the-fly are used to determine if a particular species will react further. In the absence of such kinetic information, either

topological or experts-based constraints can be provided. For example, species rank-based criteria [46] network pruning criteria prevents reactions that involve species of ranks greater than a specified value, while the tools RDL [35] and RDL++ [43] allow for the specification of constraints that molecules should satisfy to undergo a particular transformation. Faulon and Sault [44] propose stochastic (or sampling) network generation algorithms, in contrast to the deterministic scheme, for concurrent generation and reduction of networks. These algorithms scale in polynomial time but require on-the-fly estimation of rate constants which is achieved, in their case, through quantitative structure property relationships for free-radical chemistries.

Kinetic modeling is a common application of automated network generation, wherein the appropriate differential algebraic system of equations that captures the dynamics of the system is formulated. The model is then solved with kinetic parameters estimated, predicted, or specified, to obtain product yield information. Network generation in conjunction with kinetic modeling has been extensively applied for hydrocarbon [34, 47], and biochemical systems [48]. Complex reaction networks, however, have also been analyzed for: (a) deriving topological properties such as average path length of the network [49], (b) identifying synthetic/ degradation pathways [50], and (c) deriving and testing plausible mechanisms and overall rate expressions [51, 52, 53]. The use of additional thermodynamic data in conjunction with the reaction network has further enabled quantitative analysis of networks in terms of: (a) generating thermodynamically meaningful flux distributions in biochemical systems [39], (b) extracting functional information such as regulatory sites in biological systems [54], and (c) identifying thermodynamically feasible synthesis routes [41] to form chemicals, or biological degradation pathways [38] to decompose molecules.

Table 2.1: A list of Reaction network generators, and a description of their essential features, and their areas of applications

Name	Description	Remarks	References
NETGEN	Network generator and model builder based on 'BE' & 'R' matrices. Uses adjacency matrix representation of molecules and reaction rules.	Rate based and rank based pruning. (ii) Application in gas phase pyrolysis, nanoparticle synthesis, and biochemical reactions. (iii) Linked to MOPAC[55] for thermochemistry.	[34, 41, 56, 57, 46, 45, 15]
EXGAS	Kinetic model builder using a tree datastructure for internal molecule representation.	(i) Applied in gas phase combustion and oxidation. (ii) A library of free radical chemistry rules used in reaction network generation. (iii) Tree representation system based on Chinnick. et al[58].	[59, 60]
COMGEN	Network generator based on chemical graph theory. String representation of molecules, reactant pattern based on Blurock et al.[61], and topological indices for molecule identification.	(i) Hydrocarbon gas phase chemistry. (ii) Thermochemistry was calculated from a database.	[36]
RMG	Kinetic models of free radical chemistry of hydrocarbons. Kinetics estimated from semi-empirical relations obtained from theoretical calculations.	(i) Applications in hydrocarbon pyrolysis. (ii) Accurate calculations of kinetics and thermodynamics, and formulation of kinetic models to predict product yields and conversion.	[47, 62, 63]

<div align="center">

Table 2.1 – continued from previous page

</div>

Name	Description	Remarks	References
RDL	English-like language based description of reaction rules. Object-oriented framework using elements of graph theory.	(i) Reaction network generated from scratch depending upon reaction rules input, thus offering flexibility in describing the system. (ii) Constraints on rules to prevent combinatorial explosion.	[35, 64, 65]
RDL++	Extends RDL with additional features to enable description of solid-acid catalyzed reactions of hydrocarbons.	(i) Applied in microkinetic modeling of heterogeneous catalytic systems, and data analysis and knowledge extraction in high-throughput experimentation.	[43, 66, 67, 68]
KING	An automated mechanism generator. Uses 'BE' and 'R' matrix for molecule and reaction representation.	(i) Applied in combustion chemistry. (ii) Reactions are determined combinatorially, as a linear combination of elementary steps.	[69]
BioNETGEN	Rule-based generation of biological reaction network. Graph based representation of molecules with each node being a building block of the macromolecule of the biological system.	(i) Application in reaction network generation in biological systems and subsequent dynamic modeling.	[70, 71, 72]
BNICE	Computational framework for generating and analyzing biological reaction pathways.	(i) Reaction rules are obtained from the enzyme function information in KEGG database. (ii) Incorporation of group-contribution based thermochemistry estimation for flux analysis and pathways prediction.	[73, 41, 39, 38]

Table 2.1 – continued from previous page

Name	Description	Remarks	References
SynBioSS	Modeling and simulation tool for synthetic biological systems. Complete enumeration of sets of biomolecular reactions based on user input of molecular parts involved in gene expression and regulation.	(i) Multiscale simulation of the generated reaction network using stochastic algorithms.	[74]
BioTrans	Computational tool for predicting metabolism of chemicals in a mixture. Generated paths of different compounds are interconnected through common metabolites. ODE models solved to predict the time profiles of the each of the compounds.	(i) Application in modeling of biotransformations of VOCs that commonly pollute water.	[75]
DESHARKY	Monte Carlo algorithm finds metabolic pathways to a target species by exploring the KEGG database of enzymatic reactions.	(i) Application in generating pathways in metabolic networks.	[76]
ReBiT	Accepts a molecular structure as input and returns a list of three-digit EC code for the reactions that either generate or react with the input molecule using a database of over 600 conserved structure generalized enzyme-catalyzed reactions.	(i) Application in modeling of metabolic networks.	[77]

2.2   The structure of RING: an overview

RING consists of three modules: a reaction language compiler, a reaction network generator, and a post-processing module [5], These modules were developed for network construction and pathways enumeration for chemical reaction networks. We have constructed an additional new module for reaction network display and applied it for biochemical reaction network generation. Figure 2.1 shows these modules as implemented for biochemical reaction networks.
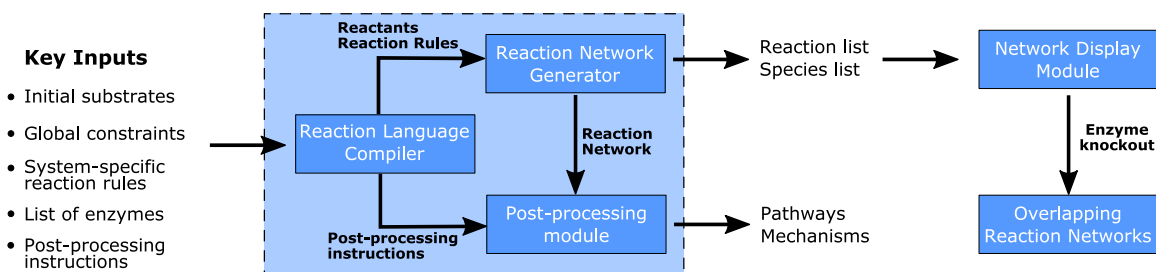
Figure 2.1: The modular structure of RING [Adapted from [5]]

2.2.1   Reaction Language Compiler

The compiler translates the inputs from the user into relevant instructions for the network generator. These inputs, written in an English-like reaction language, includes information on the initial substrates, global constraints, reaction rules, and a set of post-processing instructions for network analysis [5]. The initial substrates, provided as SMILES-like strings, are the chemical species that initiate the reaction network. The global constraints are molecular restrictions imposed upon all molecules in the reaction network. The reaction rules define the structural requirements for reactants, and products as well as chemical transformations of a reactant due to the enzymes present in the system. The chemical transformations that were previously considered in RING include elementary, non-elementary, unimolecular and biomolecular transformations [5]. Recently, RING has been upgraded to address the termolecular reactions where a cofactor participates as a co-substrate in the reaction. The post-processing instructions involve directives for RING after the network generation such as pathway identification, overall mechanism elucidation, molecule and reaction queries, thermochemistry estimation or lumping of chemical species.

2.2.2   Reaction network generator

The reaction network generator module takes the output from the Reaction Language Compiler to construct the reaction network by iterative application of the rules upon the initial substrates and the products generated thereof. The network propagation terminates when the user-specified constraints prevents the intermediate to undergo further reactions. The final output from the network generator include a list of species and reactions consistent with the reaction rules.

2.2.3   Post-processing module

The post-processing module identifies the pathway among the predicted network based on the post-processing instructions specified by the user. For example, post-processing module may be instructed to identify the pathways that include user-specified molecules, or that have the minimal number of total reaction steps. It uses a reverse depth-first network traversal algorithm to identify all possible pathways exhaustively (Rangarajan et al. 2014).

2.2.4   Network display module

The network display module shows a graphic of the generated reaction network, which includes initial substrates, intermediates and products as nodes and reactions as edges. If the user specifies the target products, it presents the reaction paths leading to such products. In the case of knockout study, it superimposes the reaction network of knockout studies over the initial reaction network to depict reactions that are eliminated by knockout. This feature of RING is demonstrated in Chapter 4. RING generates an input file containing the species and network connectivity information which is passed as input to Graphviz (Gansner and North 2000), an open source graph visualization software, to generate a visual representation of the reaction network.

2.3   Time-scale analysis

Model reduction methods based on time-scale analysis include numerical approaches like the computational singular perturbation method [78, 79, 80] where the eigenvalues of the Jacobian of the kinetic system of differential equations are used to identify the slow invariant manifold [81]; the intrinsic low-dimensional manifold method [82]

where an eigenvalue-eigenvector decomposition of the Jacobian matrix is performed with the assumption that the fast subspace vanishes quickly [83, 84]; geometric-based analysis [85, 86] where a comprehensive investigation of the features of trajectories in the concentration phase space starting from many different initial conditions is used; and analytical, projection-based methods [87, 6, 88, 89, 90, 91]. All of these methods, however, require considerable computational effort in practical applications to complex, large scale systems [92] and have been mostly applied to homogeneous reaction systems. Alternatively, mechanism reduction methods based on reaction rate evaluation [93, 94, 95, 45] allow elimination of unimportant species and reactions from the reaction network, thereby, reducing the computational complexity of the system. The reaction rate evaluation is possible when the system has well-defined kinetics like in the case of gas-phase chemistry, due to the existence of a vast kinetic database for gas-phase chemical reactions (NIST Chemical Kinetics Database). For systems where there is significant uncertainty in the kinetic parameters, eliminating unimportant species and reactions based on an approximate set of kinetic constants may lead to erroneous results.

2.4 Biochemical Reaction Network Generation and Analysis

Several automated network generators have been developed to generate and enumerate pathways in biochemical reaction systems. In BNICE, a computational framework that uses a graph-theoretic matrix representation of biochemical compounds and enzyme reaction rules [38, 39, 40, 41], molecules are represented using a bond-electron matrix (BEM) where the diagonal elements denote non-bonded valence electrons while the non-diagonal elements give the connectivity between different atoms of the molecule. The reaction rules are represented using a similar matrix and are obtained from the enzyme function information in the Kyoto Encyclopedia of Genes and Genomes (KEGG) database [96]. The reactions are then generated through matrix operations. These operations can become computationally intensive when examining networks comprising complex molecules, e.g., oligosaccharides (glycans) in glycosylation network that can have over a hundred atoms. DESHARKY, a Monte Carlo algorithm finds metabolic pathways to a target species by exploring the KEGG database of enzymatic reactions [76]. ReBiT, accepts a molecular structure as input and returns a list of three-digit

EC code for the reactions that either generate or react with the input molecule using a database of over 600 conserved structure generalized enzyme-catalyzed reactions [77]. The University of Minnesota Biocatalysis/Biodegradation Database (UM-BBD), a database of microbial biodegradation reactions for xenobiotics, uses a Pathway Prediction System (PPS) with a series of generalized reaction rules to propose step by step pathways [97, 98].

GlycoVis is a network generation and visualization program that utilizes matrix manipulation of vector-represented species to generate the reaction network [4]. The algorithm uses a 7-digit number to denote a species and a set of reaction rules manipulates the digits of the number to generate other species. Similar implementation for network generation using a 9-digit sequence is shown in [99]. Glycosylation Network Analysis Toolbox (GNAT), an open-source MATLAB based toolbox generates reaction network by defining enzyme class with detailed specificity information involving enzymatic functional group, linkage and substrate specificity [100]. Formal grammar involving pattern-matching algorithm for generation of glycosylation networks is shown in [101].

2.5   Parameter Estimation

Rule Input Network Generator (RING), a computational tool for generation and analysis of chemical reaction networks [5, 102], is used for the reaction network generation. The kinetic modeling module in RING solves the catalyst packed bed as a steady state plug flow reactor (PFR). The conservation of mass on each species generates a set of differential-algebraic equations (DAEs). The set of differential equations (Eq. 2.1) is written for gas-phase species with $r_i(C_i)$ denoting the net rate of formation of species i. For catalytic systems, the quasi-steady-state approximation (QSSA) is used for surface intermediates (Eq. 3.3) and the resulting DAEs are solved along with the site balance equation (Eq. 2.3).

$$\text{Mass Balance: } \frac{dF_i}{dV} = r_i(C_i) \ \forall i \in S_{bulk} \tag{2.1}$$

$$\text{QSSA: } r_j(C_j) = 0 \ \forall j \in S_{surface} \tag{2.2}$$

$$\text{Site Balance: } \sum_j C_j + C_{site} = C_{site}^o \ \forall j \in S_{surface} \tag{2.3}$$

$$\text{Vol. flow rate: } v = \frac{RT \times \sum_i F_i}{P} \tag{2.4}$$

$$\text{Initial Conditions: } F_i(0) = F_{i0}; \; C_i v = F_i \; \forall i \in S_{bulk} \tag{2.5}$$

$$C_j(0) = C_{j0} \; \forall j \in S_{surface} \tag{2.6}$$

where F is the molar flow rate, C is the concentration, v is the volumetric flow rate; S_{bulk} and $S_{surface}$ are sets of species in the bulk phase, and species on the surface, respectively. F_0 and C_0 are initial flow rates and concentrations. P is the pressure of the system, R is the universal gas constant, and T is the system temperature.

The kinetic module in RING, for simulating the above kinetic model, incorporates the implicit differential-algebraic solver (IDAS) with sensitivity analysis [103]. The solver calculates the molar flow rates of gas-phase species and concentrations of surface species along the reactor length. The sensitivities estimated give the effect on the yield of species due to change in the kinetic rate constants (defined as parameters). Although computational and experimental studies reported in the literature provide an estimate of kinetic parameters, a slight error of $\sim$ 1-5 kcal/mol in the activation energy value at 623 K causes an amplification of $\sim$ 2-50 for the rate constant. The rate constants are therefore estimated using experimental datasets. The parameter estimation problem is a non-linear programming (NLP) problem where a scalar objective function f(p) is minimized (Eq. 2.7). Constraints may be added to this formulation, which could be bounds on the parameter values, or specific thermodynamic constraints on activation energies. The resulting problem has the form:

$$minimize f(p) = \sum_{i=1}^{n_{expts}} \sum_{j=1}^{m_{species}} w_{ij} \left(F_{ij}^{expt} - F_{ij}^{pred}(p) \right)^2 \tag{2.7}$$

where F_{ij}^{expt} is the observed molar flow rate of species j of ith experimental dataset, $F_{ij}^{pred}(p)$ is the predicted molar flow rate of species j for i^{th} experimental operating conditions using the p parameter values, and w_{ij} (Eq. 2.8) correspond to different weights used in order to normalize the contributions of each term:

$$w_{ij} = \left(\frac{1}{F_{ij}^{expt}} \right)^2 \tag{2.8}$$

We use the sequential optimization method [104] to estimate the kinetic rate constants, shown in Figure 1. The values of the state variables (chemical species), as well as the sensitivities are estimated using IDAS. These are passed into the NLP solver for solving

the minimization problem. The NLP solver, Interior Point OPTimizer (IPOPT), calculates a set of parameter values in the direction of steepest descent depending on the gradients estimated by the DAE solver [105]. The solution of the DAEs and the solution of the minimization problem are done sequentially. As already stated, the parameter estimation problem is a non-linear problem that may converge to local minima which may be significantly poorer than the global minimum [106]. The solution of any local optimization depends on the initial guess provided for the parameter values. A multi-start approach is used to provide the initial points for optimization to obtain greater reliability and improved solutions to the optimization problem. The set of initial guesses is generated by considering specific bounds on the parameters and discretizing the range to generate an ensemble of initial guesses that span the parameter space in an unbiased manner using Latin Hypercube sampling [107].

Sequential Optimization
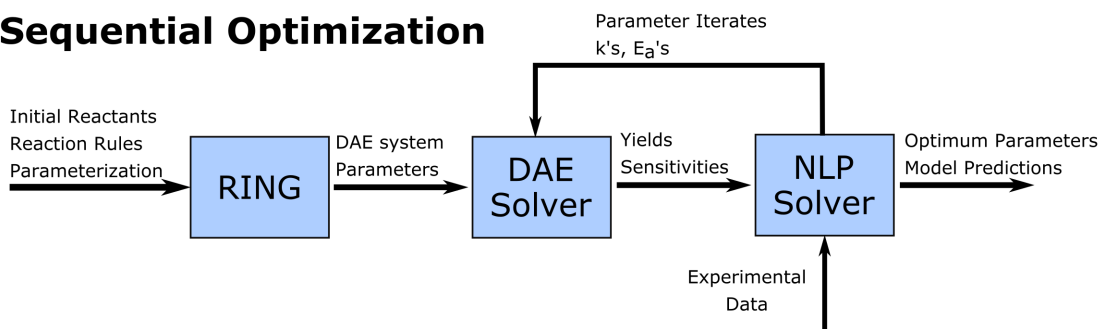
Figure 2.2: The sequential optimization setup for the parameter estimation problem.

2.6   Estimability Analysis

Parameter correlation is a major factor in the uncertainty and non-uniqueness of parameter estimates. Estimability criteria [108, 109, 110, 111] identify parameters that have a major influence on the model predictions by using local sensitivities of these parameters evaluated at desired operating conditions. The essence of the estimability criteria is to identify initially the most sensitive parameter and successively identify other estimable parameters by removing the correlation of the already identified parameters. An iterative process has been proposed for identifying estimable parameters [111]. The sum of squares for each of the columns of the sensitivity matrix (Eq. 2.9) is calculated to identify the most estimable parameter.

$$\text{Sensitivity Coefficient Matrix, } Z = \begin{bmatrix} \frac{\partial lnC_1}{\partial ln\theta_1}\big|_{t=t_1} & \cdots & \frac{\partial lnC_1}{\partial ln\theta_P}\big|_{t=t_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial lnC_R}{\partial ln\theta_1}\big|_{t=t_1} & \cdots & \frac{\partial lnC_R}{\partial ln\theta_P}\big|_{t=t_1} \\ \frac{\partial lnC_1}{\partial ln\theta_1}\big|_{t=t_2} & \cdots & \frac{\partial lnC_1}{\partial ln\theta_P}\big|_{t=t_2} \\ \vdots & \ddots & \vdots \\ \frac{\partial lnC_R}{\partial ln\theta_1}\big|_{t=t_N} & \cdots & \frac{\partial lnC_R}{\partial ln\theta_P}\big|_{t=t_N} \end{bmatrix} \qquad (2.9)$$

where C is the concentration of species, θ is the reaction rate constant.

The orthogonalization step (as shown in Eq. 2.10, 2.11) allows for removal of the correlation effect of the selected parameter from the other parameters. A residual matrix, R_L, is generated at each iteration.

$$\hat{Z}_L = X_L \left(X_L^T X_L\right)^{-1} X_L^T Z \qquad (2.10)$$

$$\text{Residual Matrix, } R_L = Z - \hat{Z}_L \qquad (2.11)$$

where X_L is the corresponding column of the parameter selected in the iteration step. The residual matrix generated, after the orthogonalization step, is used as the new sensitivity matrix and the next most sensitive parameter is selected. Because of correlation present in the parameters, the magnitude of the sum of squares of sensitivities approaches zero and the iteration scheme stops as the sensitivity matrix becomes singular.

Time Scale Decomposition in Complex Reaction Systems: A Graph Theoretic Analysis

In this chapter, a graph-theoretic framework is proposed for generation of non-stiff reduced models of isothermal reaction systems with fast and slow reactions. A directed bi-partite graph is used to represent the reaction network and the reactions are characterized as fast or slow using a kinetic threshold and an equilibrium tolerance. Cycles that correspond to closed walks are then used to identify interactions between species participating in fast/equilibrated reactions. Subsequently, an algorithm which connects these cycles to generate pseudo-species that evolve in the slow time scale alone is presented. The result is an automated, generic procedure for generating non-stiff reduced models in terms of these pseudo-species, while enforcing typical quasi-equilibrium or complete conversion constraints for fast reactions. The efficacy of the developed framework is illustrated through its application on two chemical reaction systems: 1-butene cracking and carbon metabolism in erythrocytes.

3.1   Methodology

Consider a reaction network of a homogeneous, isothermal system with n chemical species (S) and m reactions (R), with α_{ij} and β_{ij} the stoichiometric coefficients of the

reactants and products, respectively, and k_i the kinetic constant for reaction R_i:

$$\sum_{j=1}^{n} \alpha_{ij} S_j \xrightarrow{k_i} \sum_{j=1}^{n} \beta_{ij} S_j, \qquad i = 1, ..., m \tag{3.1}$$
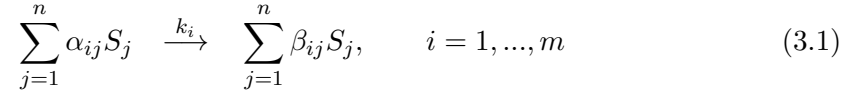
Let C_j be the concentration of species S_j and $C = (C_1, C_2, ..., C_n)^T$ be the vector of concentrations. The reaction rate r_i is generally expressed as a product of a reaction rate constant, k_i, and a nonlinear function of concentrations, $f_i(C)$:

$$r_i(C) = k_i f_i(C) \tag{3.2}$$

In the case of reversible reactions, the forward and reverse reactions are represented separately in Eq. 3.1. A kinetic model of a batch (fixed volume) system, derived from the mass balances for the species, results in a set of ordinary differential equations (ODEs) that gives the time evolution of the concentrations, C_j, $j = 1, ..., n$:

$$\frac{dC_j}{dt} = \sum_i (\beta_{ij} - \alpha_{ij}) \times r_i(C) \tag{3.3}$$

In a plug flow reactor, these ODEs are reformulated with respect to reactor volume (V). The spatial evolution of the molar flow rates, F_j, $j = 1, ..., n$, at steady state is then given by:

$$\frac{dF_j}{dV} = \sum_i (\beta_{ij} - \alpha_{ij}) \times r_i(C) \tag{3.4}$$

In a heterogeneous (gas-solid) system, the quasi-steady-state assumption (QSSA) for surface intermediates is typically employed [112] involving adsorption/desorption reactions. Let Q be the subset of $S = \{S_1, S_2, ..., S_n\}$ containing the surface intermediates. Then the QSSA assumption applied on these species results in the following algebraic equations

$$\sum_i (\beta_{ij} - \alpha_{ij}) \times r_i(C) = 0 \quad \forall j \in Q \tag{3.5}$$

Together with the differential equations for the gas-phase species, the kinetic model in this case is a differential-algebraic equation (DAE) system.

The framework developed in the present work is applicable to all types of reaction systems discussed above. Model stiffness can result from large reaction rate constants in the case of irreversible reactions or high forward/reverse reaction rates in the case of

reversible reactions. A systematic framework is developed for identifying such fast/e-quilibrated reactions and generating pseudo-species evolving in a slow time scale, while enforcing quasi-equilibrium or complete conversion constraints. The steps involved are: (1) graph representation for the reaction network, (2) identification of fast/equilibrated reactions, (3) identification of fast sub-graphs, (4) identification of cycles, and (5) generation of pseudo-species. Each of these steps is discussed in detail below.

3.1.1 Graph representation for the reaction network

A directed bi-partite graph $\mathcal{G}_B = (S, R, E)$ with two disjoint sets of vertices, one including species (S) and the other including reactions (R) [113, 114], and the set of directed links (E) - ordered pairs of one node in S and one node in R - is used to represent the reaction network. A species is identified as a reactant/product based on the direction of the edge. An edge directed from the species set to the reaction set implies that the species is a reactant, while an edge directed from the reaction set to the species set implies that the species is a product. An example reaction scheme along with its graph representation is shown in Figure 3.1. The bi-partite graph allows representation of both bimolecular and unimolecular reactions as opposed to other graphical representations (e.g., a digraph with nodes denoting species and edges denoting reactions which can only capture unimolecular reactions [115]).
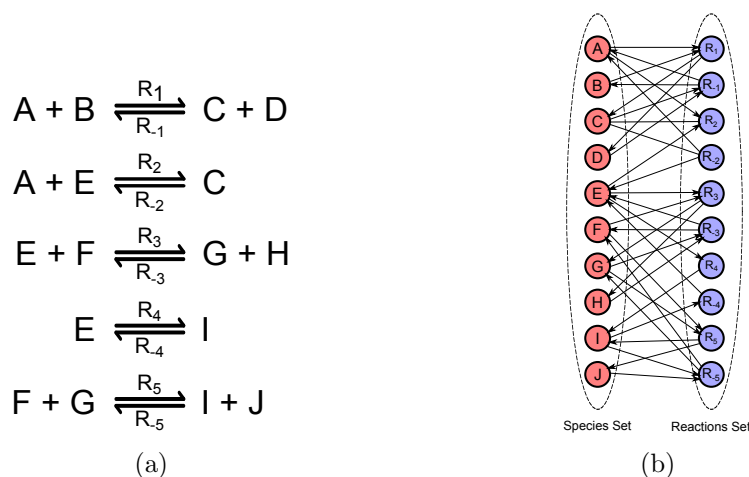
Figure 3.1: (a) Reaction scheme with species represented as letters and R_1/R_{-1}, R_2/R_{-2}, R_3/R_{-3}, R_4/R_{-4}, and R_5/R_{-5} representing reactions and (b) its directed bi-partite representation.

3.1.2   Identification of fast/equilibrated reactions

In the next step, the reactions are classified as fast and slow using a kinetic threshold (k^{min}) and an equilibrium tolerance (δ). For irreversible reactions, a kinetic threshold value, k^{min} (where k^{min} has dimensions of inverse time) is assumed and any reaction with a pseudo-first-order rate constant above this threshold is considered to be fast. Note that for a bimolecular reaction, the reaction rate constant can be normalized to get the units of inverse time by using the concentration of one of the reactants present in excess or in case of biochemical systems, using enzyme activity, enzyme concentration, or the average concentration of some of the species [116].

Reversible reactions that equilibrate over a short initial time (or space time) have fast forward and reverse rates. These reactions can be similarly identified based on the kinetic threshold, however, a reaction may not satisfy the equilibrium condition if the concentrations of species in the reaction vary over several orders of magnitude. Hence, fast reversible reactions that equilibrate are identified by defining a term, the equilibrium index, which captures the ratio of the forward reaction rate and the reverse reaction rate:

$$\text{Equilibrium Index} = \frac{\text{Forward reaction rate}}{\text{Reverse reaction rate}} = \frac{k_{\text{forward}} \times f_{\text{forward}}(\text{C})}{k_{\text{reverse}} \times f_{\text{reverse}}(\text{C})} \qquad (3.6)$$

An equilibrated reaction will have an equilibrium index value of unity. An equilibrium tolerance δ, will be used to characterize reactions that can be assumed equilibrated. To this end, the forward and reverse reaction rates of a reversible reaction are calculated through a forward simulation of the ODEs (using best available estimates of kinetic parameters if these are not known exactly). The equilibrium index is calculated for each reversible reaction over the whole spatial/temporal region of interest; reactions with equilibrium index values always lying within the tolerance δ, are considered equilibrated:

$$\left| 1 - \frac{k_{\text{forward}} \times f_{\text{forward}}(\text{C})}{k_{\text{reverse}} \times f_{\text{reverse}}(\text{C})} \right| \leq \delta \qquad (3.7)$$

Note that in the case of reversible reaction, we use the same number for both the forward and reverse reactions, but with a negative sign for the latter as shown in figure 3.1a. **Algorithm 1** below describes the steps for identifying the fast/equilibrated reactions.

Two datastructures, *FastReactionList* (E_{fast}) and *SlowReactionList* (E_{slow}) are generated for the fast and slow reaction edges respectively, and two datastructures, *ReactantMap* (RM_j) and *ProductMap* (PM_j). *ReactantMap* contains reactions in which species S_j participates as a reactant whereas *ProductMap* contains reactions in which species S_j participates as a product.

Algorithm 1 Reaction Identification($\mathcal{G}_\mathcal{B}(S, R, E)$)

1: Forward simulation (C_0, k) ▷ Perform a forward simulation of the ODEs to calculate the equilibrium index of reversible reactions.

2: **for** i = 1 : size(E) **do** ▷ Check the type of the corresponding reaction (reversible or irreversible) and also whether the reaction satisfies the identification criterion for fast reactions. The routines checkThresholdCriterion(E_i) and checkEquilibriumIndexCriterion(E_i) examine irreversible and reversible reactions respectively.

3:     **if** $E_i \in \mathcal{IRS}$ **then**

4:         checkThresholdCriterion(E_i)

5:     **else if** $E_i \in \mathcal{RS}$ **then**

6:         checkEquilibriumIndexCriterion(E_i)

7:     **end if**

8:     **if** fastReaction(E_i) **then**

9:         Put E_i in E_{fast} ▷ The routine fastReaction(E_i) checks if edge E_i is fast and two datastructures, *FastReactionList* (E_{fast}) and *SlowReactionList* (E_{slow}) are generated for the fast and slow reaction edges respectively. *ReactantMap* (RM_j) contains reactions in which species S_j participates as a reactant and *ProductMap* (PM_j) contains reactions in which species S_j participates as a product.

10:         **for** j = 1 : size($\mathcal{N}_R$) **do**

11:             Put E_i in RM_j for S_j

12:         **end for**

13:         **for** j = 1 : size($\mathcal{N}_P$) **do**

14:             Put E_i in PM_j for S_j

15:         **end for**

16:     **else**

17:         Put E_i in E_{slow}

18:     **end if**

19: **end for**

3.1.3 Identification of fast sub-graphs

The sub-graphs in $\mathcal{G}_{\mathcal{B}}(S, R, E)$ which contain only fast reaction edges and are connected with the remaining reaction network through slow reactions only are identified in this next step. Figure 3.2a shows the same reaction scheme as in Figure 3.1a, where some of reactions (R_1/R_{-1}, R_3/R_{-3}, R_4/R_{-4}) are considered fast and the corresponding sub-graphs are explicitly identified. The fast reactions are shown with red arrows while the slow reactions are shown using black arrows. It can be seen that only slow reactions (black arrows) pass through an enclosed dashed boundary, illustrating that sub-graphs comprising only fast reactions interact with the remaining reaction network through slow reactions only.
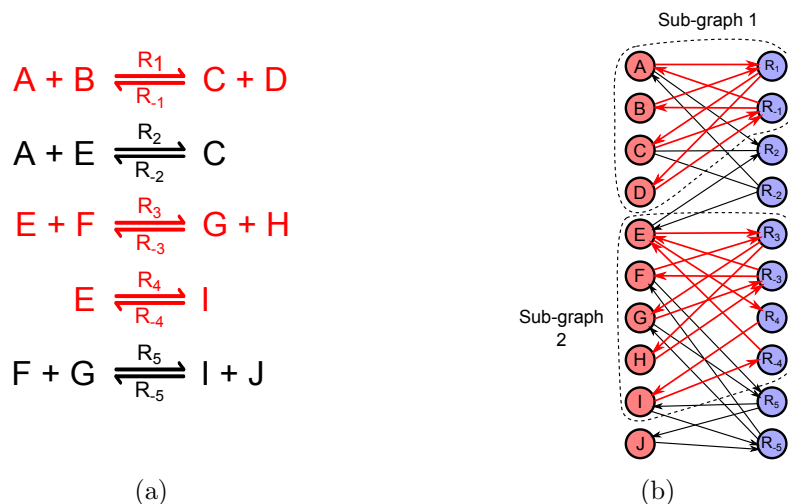
Figure 3.2: (a) Reaction scheme with the R_1/R_{-1}, R_3/R_{-3}, R_4/R_{-4} reactions considered fast and (b) the directed bi-partite representation illustrating the identified fast sub-graphs.

In a general reaction network, these sub-graphs can be identified using a breadth-first search (BFS) graph traversal algorithm [117]. **Algorithm 2** below describes the steps followed in identifying the fast sub-graphs. In the general case, the procedure runs over all species S and checks if species S_j participates in a fast reaction. If a species participates in a fast reaction, it is added to a *SpeciesQueue* and in a sub-graph, $\mathcal{SG}$. Using species S_j, the procedure runs over the reactions in the ReactantMap, RM and the reactions in the ProductMap, PM, finding the product and reactant species, respectively, of the reactions that S_j participates in. These species are then added to the

SpeciesQueue and to the same sub-graph as species S_j. If a reaction is bimolecular, the co-reactant is also found and added to the *SpeciesQueue* and the same sub-graph. Further, the species S_j is removed from the *SpeciesQueue* and the next species in the queue is selected, and the procedure is repeated. The procedure terminates when the *SpeciesQueue* is empty resulting in a list of all fast sub-graphs in the reaction network. If a species does not participate in any fast reaction, the species is added to the true slow species datastructure, $\mathcal{TS}$.

Algorithm 2 Finding fast sub-graphs

1: **for** j = 1 : size(S) **do**   ▷ Go over all species S
2:   found = FALSE
3:   **if** (size(RM_j) > 0 || size(PM_j) > 0) **then** ▷ Check if species S_j participates in
                                                                                a fast reaction through corresponding
                                                                                RM and PM sizes. A non-zero size
                                                                                implies that species S_j participates in
4:     SQ.push(S_j)   a fast reaction.
                                                                                ▷ If a species participates in a fast
5:     **for** l = 1 : $\mathcal{SG}$ **do**   reaction, it is added to a SpeciesQueue,
6:       **if** ($\mathcal{SG}_l$.count(S_j) > 0) **then** SQ.
7:         index = l   ▷ Check if the species has already
8:         found = TRUE   been added in a sub-graph. If yes, the
9:       **end if**   index of the corresponding sub-graph
10:       **if** (! found) **then**   is stored, if not, a sub-graph with a
11:         index = size($\mathcal{SG}$) + 1   new index (line 11) is assigned to the
12:       **end if**   species.
13:     **end for**
14:     **while** (! SQ.empty()) **do**
15:       $S_j = SQ$.front()
16:       $\mathcal{SGS}$[index].insert(S_j)
17:       **for** i = 1 : size(RM_j) **do**   ▷ Using species S_j, the procedure runs
18:         $\mathcal{SGR}$[index].insert(R_i)   over the reactions in the ReactantMap,
19:         **if** (size($\mathcal{N}_R$) == 2) **then** RM and the ProductMap, PM, find-
20:           $\mathcal{SGS}$[index].insert(co-react(S_j,   ing the product and reactant species,
       R_i))   respectively, of the reactions that S_j
21:           SQ.push(co-react(S_j, R_i))   participates in. These species are then
22:         **end if**   added to the SQ and to the same
23:         **for** j = 1 : size($\mathcal{N}_P$) **do**   sub-graph as species S_j. If a reaction
24:           $\mathcal{SGS}$[index].insert(S_j)   is bimolecular, the co-reactant/co-
25:           SQ.push(S_j)   product is also found and added to the
26:         **end for** SQ and the same sub-graph as shown
27:       **end for**   in lines 19-22 and 30-33.

28:         **for** i = 1 : size(PM_j) **do**   ▷ Further, the species S_j is removed
29:             $\mathcal{SGR}$[index].insert(R_i)   from the SQ and the next species in the
30:             **if** (size($\mathcal{N}_P$) == 2) **then**   queue is selected, and the procedure
31:                 $\mathcal{SGS}$[index].insert(co-prod(S_j,   is repeated. The procedure terminates
        R_i))   when the SQ is empty resulting in a list
32:                 SQ.push(co-prod(S_j, R_i))   of all fast sub-graphs in the reaction
33:             **end if**   network.
34:             **for** j = 1 : size($\mathcal{N}_R$) **do**
35:                 $\mathcal{SGS}$[index].insert(S_j)
36:                 SQ.push(S_j)
37:             **end for**
38:         **end for**
39:         SQ.pop()
40:     **end while**   ▷ If a species does not participate in
41:   **else**   any fast reaction, the species is added
42:     $\mathcal{TS}$.insert(S_j)   to the true slow species datastructure,
43:   **end if** $\mathcal{TS}$.
44: **end for**

3.1.4 Identification of cycles

The interactions between species participating in fast reactions within each sub-graph
are identified in this step. Specifically, cycles are identified that correspond to closed
walks over the fast edges in the graph. Figure 3.3 shows sub-graph 1 identified in
Figure 3.2 with reaction R_1/R_{-1} considered as fast. The procedure for identifying
cycles involves starting at a species node, e.g. node A or node B, traversing in the
direction of the arrow to the reaction node, R_1 (corresponding to the fast reaction),
traversing to one of the products of the reaction, e.g. node C or node D, traversing
to the node for the corresponding reverse reaction, R_{-1}, and finally, traversing back
to the starting species node, to complete the cycle. The cycles identified for this fast
bimolecular reaction are shown in Figure 3.3.
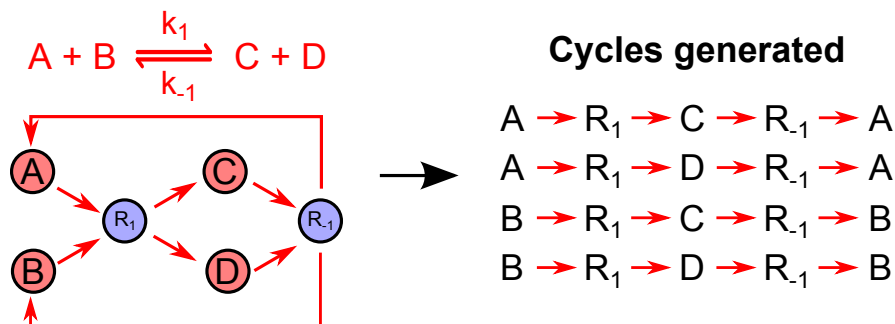
Figure 3.3: Cycles generated for a fast bi-molecular reaction.

For a general sub-graph involving more than one reaction, the backtrack algorithm [118] is used to generate the cycles for all the species and fast reactions. **Algorithm 3** below describes the steps followed in generating cycles for reactions in a fast sub-graph. The procedure runs over all sub-graphs, $\mathcal{SG}$. Within each sub-graph, the algorithm runs over each species, present in the sub-graph, stored in datastructure $\mathcal{SGS}$. The species S_j is used as a starting node to generate the cycle for the reactions that it participates in. The algorithm then goes over all the reactions that use species S_j as a reactant, finding the second node in the cycle. w_i stores the stoichiometric coefficient of the reactant R_i using the routine stoichiometry(S_j, R_i) which requires the species and reaction as inputs. Next, the algorithm goes over the products of the reaction R_i, finding the third node of the cycle. The fourth node corresponding to the reverse reaction is found using the definition for the routine reverseRxn(R_i). w_{-i} stores the stoichiometric coefficient of the reactant R_{-i} using the routine stoichiometry(S_t, R_{-i}). The information about stoichiometric coefficients and species pair in a cycle for a reaction is stored in datastructures $\mathcal{NP}$ and $\mathcal{NPI}$. The procedure ends when all species within a sub-graph and all sub-graphs are processed.

Algorithm 3 Cycles generation

1: **for** l = 1 : size($\mathcal{SG}$) **do**	▷ Go over all sub-graphs
2: **for** j = 1 : size($\mathcal{SGS}$) **do**	▷ Within each sub-graph, a species in the sub-graph is selected
3: **if** (size(ReactionsProcessed[S_j]) <size(RM_j)) **then**	▷ Check on whether all the reactions of the corresponding species have been accounted
4: NQ.push(S_j)	▷ If not, the species is inserted in a *NodeQueue*, *NQ*
5: **for** i = 1 : size(RM_j) **do**	▷ Go over the reactant map of the
6: NQ.push(R_i)	species and inserts a reaction into NQ
7: ReactionsProcessed[S_j].insert(R_i)	▷ Add reaction R_i to ReactionsProcessed datastructure for species S_j
8: w_i = stoichiometry(S_j, R_i)	▷ Store the stoichiometric coefficient for reaction R_i
9: **for** t = 1 : size($\mathcal{N}_P$) **do**	▷ Go over the products of reaction,
10: NQ.push(S_t)	insert the species into NQ and
11: R_{-i} = reverseRxn(R_i)	store their stoichiometric coeffi-
12: NQ.push(R_{-i})	cients
13: ReactionsProcessed[S_t].insert(R_{-i})	▷ Add reaction R_{-i} into NQ
14: w_{-i} = stoichiometry(S_t, R_{-i})	▷ Add reaction R_{-i} to Reac-
15: $\mathcal{NP}$.insert(S_j, S_t)	tionsProcessed datastructure for
16: $\mathcal{NPI}$.insert(pair(S_j, S_t), R_i, pair(w_{-i}, w_i))	species S_t
17: NQ.push(S_j)	▷ The datastructures, $\mathcal{NP}$ and
18: **while** NQ.front() != R_i **do**	$\mathcal{NPI}$ store information regarding
19: NQ.pop()	the species pair, corresponding
20: **end while**	reaction and the coefficients.
21: **end for**	▷ On finishing the required cycle,
22: **while** NQ.front() != S_j **do**	some nodes are removed to account
23: NQ.pop()	for all the remaining species and
24: **end while**	reactions.
25: **end for**	
26: **end if**	
27: **end for**	
28: **end for**	

3.1.5   Identification of pseudo-species using the cycles

From each cycle, a pseudo-species (the sum of the two species involved in the cycle) can be readily identified such that the contributions of the fast reaction rates cancel out. For example, for the first cycle, $A \rightarrow R_1 \rightarrow C \rightarrow R_{-1} \rightarrow A$, as shown in Figure 3.3, the corresponding pseudo-species is $(A + C)$. Eq. 3.8 and Eq. 3.9 show the mass balances for species A and C respectively. Both equations contain the fast reaction rate terms corresponding to the fast bimolecular reaction ($k_{-1}f_{-1}(\text{C})$ and $k_1 f_1(\text{C})$) and slow reaction rate terms corresponding to the slow reactions, $\sum S.T_1$ and $\sum S.T_3$ for species A and C respectively. Eq. 3.10 shows the mass balance for the pseudo-species $(A + C)$, where only the slow reaction rate terms are present with the fast reaction rate terms being eliminated. Similarly, the other pseudo-species corresponding to the other identified cycles are $A + D$, $B + C$, and $B + D$. Note that one of these species is linearly dependent on the remaining three species. Therefore, the fourth species $(B + D)$ should be removed from the set of pseudo-species to avoid such a redundancy. The choice of this species is arbitrary implying that the set of pseudo-species generated is not unique.

Original Model

$$\frac{dC_A}{dt} = k_{-1}f_{-1}(\text{C}) - k_1 f_1(\text{C}) + \sum S.T_1 \tag{3.8}$$

$$\frac{dC_C}{dt} = -k_{-1}f_{-1}(\text{C}) + k_1 f_1(\text{C}) + \sum S.T_3 \tag{3.9}$$

Model in terms of pseudo-species

$$\frac{d[C_A + C_C]}{dt} = \sum S.T_1 + \sum S.T_3 \tag{3.10}$$

Since each cycle can be considered as a species pair, the occurrences of a species pair in all fast reactions are stored, to be used later for generating pseudo-species.

Remark: The generation of pseudo-species following the above procedure corresponds to a particular choice of coordinate change such that the corresponding coefficient matrix belongs to the left null space of the stoichiometric matrix of the fast reactions. For the

above sub-graph the fast reaction stoichiometric matrix is $\begin{bmatrix} -1 \\ -1 \\ 1 \\ 1 \end{bmatrix}$ and the choice of

pseudo-species corresponds to a coordinate matrix $\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$ with $\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \times$

$\begin{bmatrix} -1 \\ -1 \\ 1 \\ 1 \end{bmatrix} = 0$. In this sense, the proposed procedure implements in a graph-theoretic

setting the projection-based approach in [87], [6].

Species participating in multiple reactions within a sub-graph

In a general sub-graph, a species can participate in multiple fast reactions. Figure 3.4a shows the third and the fourth reactions from Figure 3.1 considered as fast along with the directed bi-partite representation of the corresponding sub-graph from Figure 3.2b.

$$E + F \underset{R_{-3}}{\overset{R_3}{\rightleftharpoons}} G + H$$

$$E \underset{R_{-4}}{\overset{R_4}{\rightleftharpoons}} I$$

(a)   (b)
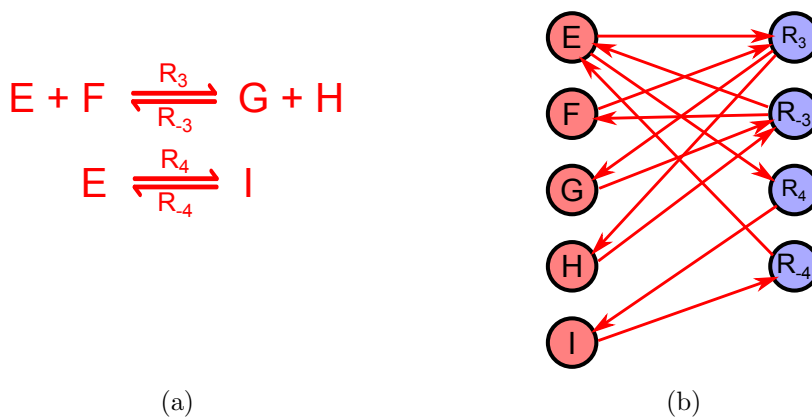
Figure 3.4: (a) Reaction scheme with R_3/R_{-3} and R_4/R_{-4} reactions considered fast and (b) the directed bi-partite representation of the corresponding sub-graph.

The species E participates in both reactions which contribute fast rate terms $(k_{-3}f_{-3}(C), k_3f_3(C), k_{-4}f_{-4}(C),$ and $k_4f_4(C))$ shown in Eq. 3.11. Therefore, to eliminate all fast reaction rate terms from the corresponding mass balance, the cycles identified for each reaction need to be combined. Considering the cycles from each reaction,

E $\rightarrow$ R$_3$ $\rightarrow$ G $\rightarrow$ R$_{-3}$ $\rightarrow$ E for the first reaction and E $\rightarrow$ R$_4$ $\rightarrow$ I $\rightarrow$ R$_{-4}$ $\rightarrow$ E for the second reaction, the pseudo-species that will be invariant in the fast time scale is $E + G + I$. The mass balance for this pseudo-species is shown in Eq. 3.14 and indeed involves only slow reaction terms. Figure 3.5 illustrates the combination of the two cycles which generates the above pseudo-species. Similarly, this procedure can be applied to the second cycle for species E in reaction R_3 to generate the pseudo-species $E + H + I$, as well as for the other species in the sub-network.

E + G + I
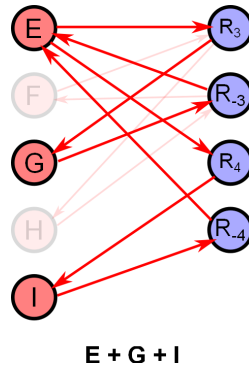
Figure 3.5: Cycles correponding to fast reactions of species E used to generate the pseudo-species

Original Model

$$\frac{dC_E}{dt} = k_{-3}f_{-3}(\mathrm{C}) - k_3 f_3(\mathrm{C}) + k_{-4}f_{-4}(\mathrm{C}) - k_4 f_4(\mathrm{C}) + \sum S.T_5 \qquad (3.11)$$

$$\frac{dC_G}{dt} = -k_{-3}f_{-3}(\mathrm{C}) + k_3 f_3(\mathrm{C}) + \sum S.T_7 \qquad (3.12)$$

$$\frac{dC_I}{dt} = -k_{-4}f_{-4}(\mathrm{C}) + k_4 f_4(\mathrm{C}) + \sum S.T_9 \qquad (3.13)$$

Model in terms of pseudo-species

$$\frac{d[C_E + C_G + C_I]}{dt} = \sum S.T_5 + \sum S.T_7 + \sum S.T_9 \qquad (3.14)$$

Algorithm 4 shown below describes the steps followed in the pseudo-species generation procedure by combining the cycles identified in section 3.1.4. The species-pairs are sorted based on their occurrences in datastructure $\mathcal{NP}$. The procedure starts from the

most frequently occurring species pair $\mathcal{NP}_t$, identified through the cycle generation procedure. If the individual species in a pair participate in the same reactions, they form a pseudo-species since all the fast reaction rate terms cancel out. If the species individually participate in other fast reactions, they are then inserted into a *SpeciesQueue*, *SQ* for identifying other cycles of these species. A species S_j is selected from the *SQ* and its presence in other cycles is checked. If a reaction edge that has not been processed is found, then the other species in the corresponding pair is identified and inserted in the *SQ*. If the reaction edge has already been processed, a check regarding participation of the products of the reaction edge in a different unimolecular reaction is performed. If such a reaction exists, the coefficient of species S_j is updated within the pseudo-species generated, to account for the reaction rate terms from both the product species. A species is removed from the *SQ* if all the reactions that the species participates in have been accounted for. The procedure ends when n_f - m_f number of pseudo-species are generated, where n_f denotes the number of species that participate in the m_f fast/e-quilibrated reactions. Note that for a sub-graph with only unimolecular reactions, only one pseudo-species is generated following the above described procedure, which is the summation of all the species within the sub-graph.

Algorithm 4 Pseudo-species generation

1: **for** $t = 1 : \text{size}(\mathcal{NP})$ **do**   ▷ Go over all the species pairs

2:     PairProcessed.insert($\mathcal{NP}_t$)   ▷ Store processed pairs

3:     $\mathcal{R}_t = \mathcal{NPI}[\mathcal{NP}_t].\text{getReactions}()$ ▷ Identify the corresponding set of reactions for the pair using the routine getReactions()

4:     **for** $j = 1 : \text{size}(\mathcal{NP}_t)$ **do**   ▷ Go over each species in the pair and select a species to initiate the procedure

5:       $S_j = \mathcal{NP}_t[j]$

6:       ReactionsProcessed[S_j].insert($\mathcal{R}_t$)

7:       $SQ.\text{push}(S_j)$ ▷ Store reaction for current species and add the species to *SpeciesQueue*, *SQ*

8:       Coefficient.insert(S_j, $\mathcal{NPI}[\mathcal{NP}_t].\text{getCoefficient}(S_j)$)

9:     **end for**   ▷ Store species coefficient that will appear in the pseudo-species

10:     **while** (**!**$SQ.\text{empty}()$) **do**

11:       $S_j = SQ.\text{front}()$

12:       **for** $k = 1 : \text{size}(\mathcal{NP})$ **do**   ▷ Iterate over the species entered into the *SQ*
▷ Go over all the pairs and identify pairs that contain the present species

13:         **if** ($\mathcal{NP}_k.\text{count}(S_j) > 0$ && PairProcessed.count($\mathcal{NP}_k$) == 0)

14:         $\mathcal{R}_l = \mathcal{NPI}[\mathcal{NP}_k].\text{getReactions}()$

15:         **if** (ReactionsProcessed.count($\mathcal{R}_l$) == 0) **then**   ▷ If pair has already been processed, then skip

16:           ReactionsProcessed[S_j].insert($\mathcal{R}_l$)

17:           $S = \text{OtherPairSpecies}(\mathcal{NP}_k, S_j)$ ▷ If not, the reaction R_l corresponding to the pair is checked if already processed

18:           $SQ.\text{push}(S)$

19:           $\mathcal{W}_j = \text{Coefficient.find}(S_j)$

20:           Coefficient.insert(S, $\mathcal{W}_j * \dfrac{\mathcal{NPI}[\mathcal{NP}_k].\text{getCoefficient}(S)}{\mathcal{NPI}[\mathcal{NP}_k].\text{getCoefficient}(S_j)}$)   ▷ If not, the other species in pair is added to the *SQ*

```
21:              ReactionsProcessed[S].insert(R_l)                    ▷ The coefficient of S is
                                                                        calculated based on the
22:         else if (ReactionsProcessed.count(R_l) > 0)               coefficient of S_j in the
      then                                                             pseudo-species and their
23:              S = OtherSpecies(NP_k, S_j)                          respective stoichiometric
24:              NP_uni = copairSpecies(R_l, S_j)                     coefficients
25:              if   checkUnimolecularReaction(NP_uni)               ▷ If the reaction has
      then                                                             already been processed,
26:                  if checkCoeff.find(S_j) !=                       check if there exists a
                       sumofCoefficient(NP_uni)                       uni-molecular   reaction
27:                      Coefficient.insert(S,                        between the conjugate
                         sumofCoefficient(NP_uni))                    species in the two pairs.
28:                      updateOtherCoefficients()
29:                  end if                                           ▷ Further, check and
30:                  SQ.push(S)                                       update the coefficient of
31:                  Coefficient.insert(S,                            species S_j as sum of the
                     NPI[NP_t].getCoefficient(S))                     coefficients of the conju-
32:                  end if                                           gate species if necessary.
33:             end if
34:         end if
35:     end for
36:     if (size(ReactionsProcessed[S_j]) == NR_j)
37:         SQ.pop()                                                  ▷ If all the reactions for
38:     end if                                                        a species have been pro-
39:   end while                                                      cessed, the fast reaction
40:   PS.insert(Coefficient)                                          terms have been ac-
41: end for                                                          counted for and cancelled
                                                                     out using the conjugate
                                                                     species.   The species is
                                                                     removed from the SQ.
```

The generation of pseudo-species via the cycle identification procedure is automated in an algorithm for each sub-graph. The algorithms presented in this work are implemented as a computational tool written in C++ to automate this graph theoretic framework;

this tool is used in all the examples presented below.

3.1.6   Equation formulation for the reduced model

The fast/equilibrated reactions, identified in section 1, are used to formulate the algebraic constraints in the reduced model. Complete conversion constraints are enforced for all the fast irreversible reactions:

$$f_i(\text{C}) = 0 \tag{3.15}$$

The quasi-equilibrium assumption is enforced for all the reversible reactions:

$$f_i(\text{C}) - \frac{k_{-i}}{k_i} \times f_{-i}(\text{C}) = 0 \tag{3.16}$$

In a system involving n_f species participating in m_f fast reactions, Eq. 3.15 and Eq. 3.16 constitute m_f algebraic constraints, $\text{F}(\text{C}) = 0$. The m_f algebraic constraints are assumed independent such that the Jacobian $(\partial \text{F}(\text{C})/\partial C)$ has full row rank. If not, a subset of independent constraints are selected [119]. A reduced order description of the system (Eq. 3.3 or Eq. 3.4) in terms of the slow pseudo-species ζ of order equal to the degrees of freedom $(n_f - m_f)$, can be obtained by considering a coordinate change of the form:

$$\begin{bmatrix} \zeta \\ \eta \end{bmatrix} = T(\text{C}) = \begin{bmatrix} \phi \times \text{C} \\ F(\text{C}) \end{bmatrix} \tag{3.17}$$

where ϕ is the coefficient matrix obtained from the generation of the pseudo-species. This essentially projects the description of the system on the equilibrium state space where $\eta = 0$. The differential equations for ζ will only depend on the slow reaction rates and will take the form (referring to Eq. 3.3):

$$\frac{d\zeta}{dt} = \phi \times \frac{dC}{dt} \bigg|_{C = T^{-1}(\zeta, 0)} \tag{3.18}$$

with initial conditions consistent with the algebraic constraints $\text{F}(\text{C}) = 0$.

Alternatively, the slow dynamics of the system can be simulated in terms of a reduced set of the original species. For this, the set of original species is partitioned into a reduced $(n_f - m_f)$ species set C_r and the remaining ones, $\widehat{C_r}$. Given the full row rank

of $\partial F(C)/\partial C$, the algebraic constraints $F(C) = 0$ can be solved for $\widehat{C_r}$:

$$\widehat{C_r} = F'(C_r) \tag{3.19}$$

Using Eq. 3.18 and Eq. 3.19, the slow dynamics of the system is given by

$$\frac{d\zeta}{dt} = \phi \times \frac{d}{dt}\begin{bmatrix} C_r \\ F'(C_r) \end{bmatrix} = \phi \times \begin{bmatrix} I_{n_f - m_f} \\ \frac{\partial F(C_r)}{\partial C_r} \end{bmatrix}\frac{dC_r}{dt} = P\frac{dC_r}{dt} \tag{3.20}$$

Based on the independence assumption, it can be shown that P is non-singular [90] which gives

$$\frac{dC_r}{dt} = P^{-1}\frac{d\zeta}{dt} \tag{3.21}$$

The explicit representation of the reduced model is given by Eq. 3.21 with the initial condition for species C_r selected so that $F(C_r(0), F'(C_r(0)) = 0$.

3.2 Results and Discussion

The developed graph-theoretic framework is applied to two case studies. The first case study is a reaction system involving 1-butene cracking which comprises of reversible unimolecular reactions only. In the second case study, a biochemical reaction system involving carbon metabolism in erythrocytes is considered with both reversible and irreversible bimolecular reactions.

3.2.1 Cracking and isomerization of 1-butene

1-butene cracks to form ethene and isomerizes to form 2-butene and isobutene [120]. The reaction scheme containing 10 species and 15 reactions on a Bronsted-acid catalyst is shown in Figure 3.6.
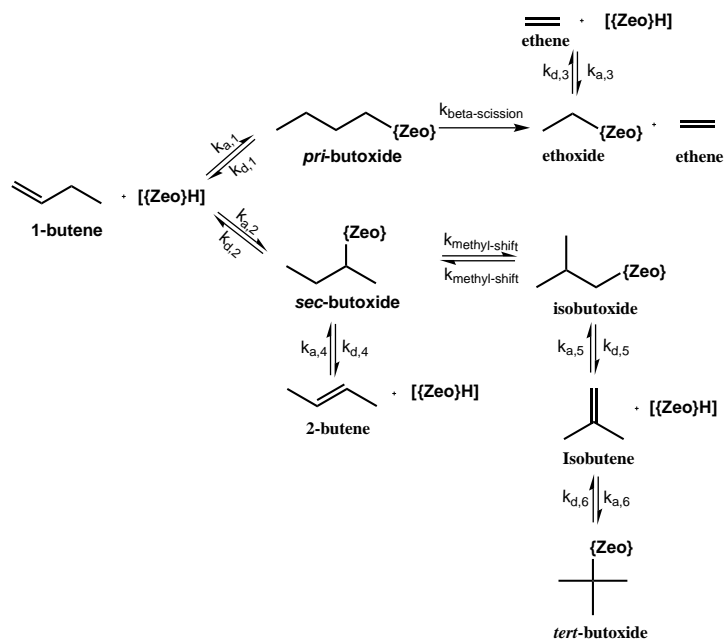
Figure 3.6: Reaction scheme for 1-butene cracking and isomerization; rate constants k_{a} correspond to the olefin adsorption reactions, k_{d} correspond to the desorption reactions, $k_{\mathrm{methyl\text{-}shift}}$ correspond to the methyl shift reactions, and $k_{\mathrm{beta\text{-}scission}}$ corresponds to the β-scission reaction. [{Zeo}H] denotes the Bronsted-acid sites in a zeolite catalyst and the species containing [{Zeo}] denote a surface alkoxide intermediate.

The kinetic parameters involving pre-exponential factors and activation energies are taken from literature reports [120, 121, 122]. The original model (Eq. 3.4) is simulated with the initial flow rate of 1-butene = 2.14 mmol/h and initial free site concentration = 0.219 mmol/g_{cat} for a temperature T = 623 K. An equilibrium tolerance of $\delta = 0.05$ was used to identify fast equilibrated reversible reactions. Using this tolerance, five unimolecular reactions are found to be equilibrated and the two corresponding sub-networks are shown in Figure 3.7.

Kinetic parameters for 1-butene cracking

The kinetic constants for each reaction at absolute temperature T are calculated using the arrhenius equation.

$$k = Ae^{-E_a/RT} \tag{3.22}$$

where A is the pre-exponential factor, E_a is the activation energy, and R is the universal gas constant.

Table 3.1: Kinetic parameter values for reactions in cracking and isomerization of 1-butene

Reaction Type	Parameter	Pre-exponential factor, A	E_a (kJ/-mol)	Ref.
Olefin Adsorption	$k_{a,1}$	1.32×10^3 (1/atm/s)	34	[121, 122]
Olefin Adsorption	$k_{a,2}$	4.939×10^3 (1/atm/s)	-22	[121, 122]
Olefin Adsorption	$k_{a,3}$	2.67×10^4 (1/atm/s)	54	[121, 122]
Olefin Adsorption	$k_{a,4}$	4.939×10^3 (1/atm/s)	-22	[121, 122]
Olefin Adsorption	$k_{a,5}$	1.32×10^3 (1/atm/s)	34	[121, 122]
Olefin Adsorption	$k_{a,6}$	6.8×10^4 (1/atm/s)	-59	[121, 122]
Beta-scission	$k_{beta-scission}$	1.7×10^{19} (1/s)	257	[120]
Methyl shift	$k_{methylshift}$*	1.246×10^{13} (1/s)	10	-
Methyl-shift	$k_{methylshift}$*	1.246×10^{13} (1/s)	10	-

* $k_{methylshift}$ is calculated assuming A is $k_B T/h$ (where k_B = Boltzmann constant, T = Temperature, and h = Planck constant) and E_a is assigned to be 10 kJ/mol

The kinetics for the desorption reactions are calculated from the thermochemistry of olefin adsorption. The thermodynamic values for the surface alkoxide intermediates are calculated using group contribution values.
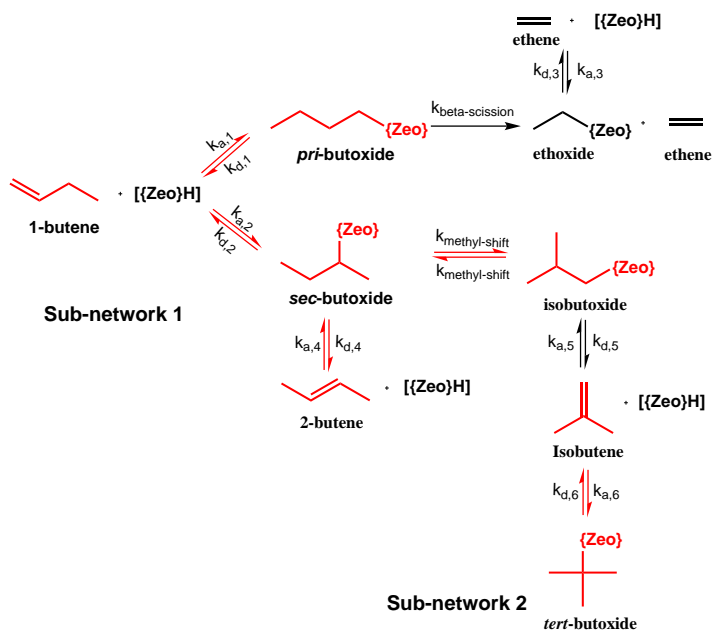
Figure 3.7: Fast reaction sub-networks identified for 1-butene cracking and isomerization reactions

The corresponding pseudo-species generated for each sub-network are shown below:

1. Pseudo-species 1:

2. Pseudo-species 2:

The reduced model is formulated using the above pseudo-species along with the initial conditions derived from the quasi-equilibrium constraints. A comparison between the original and the reduced model evolution profiles is shown in Figure 3.8.
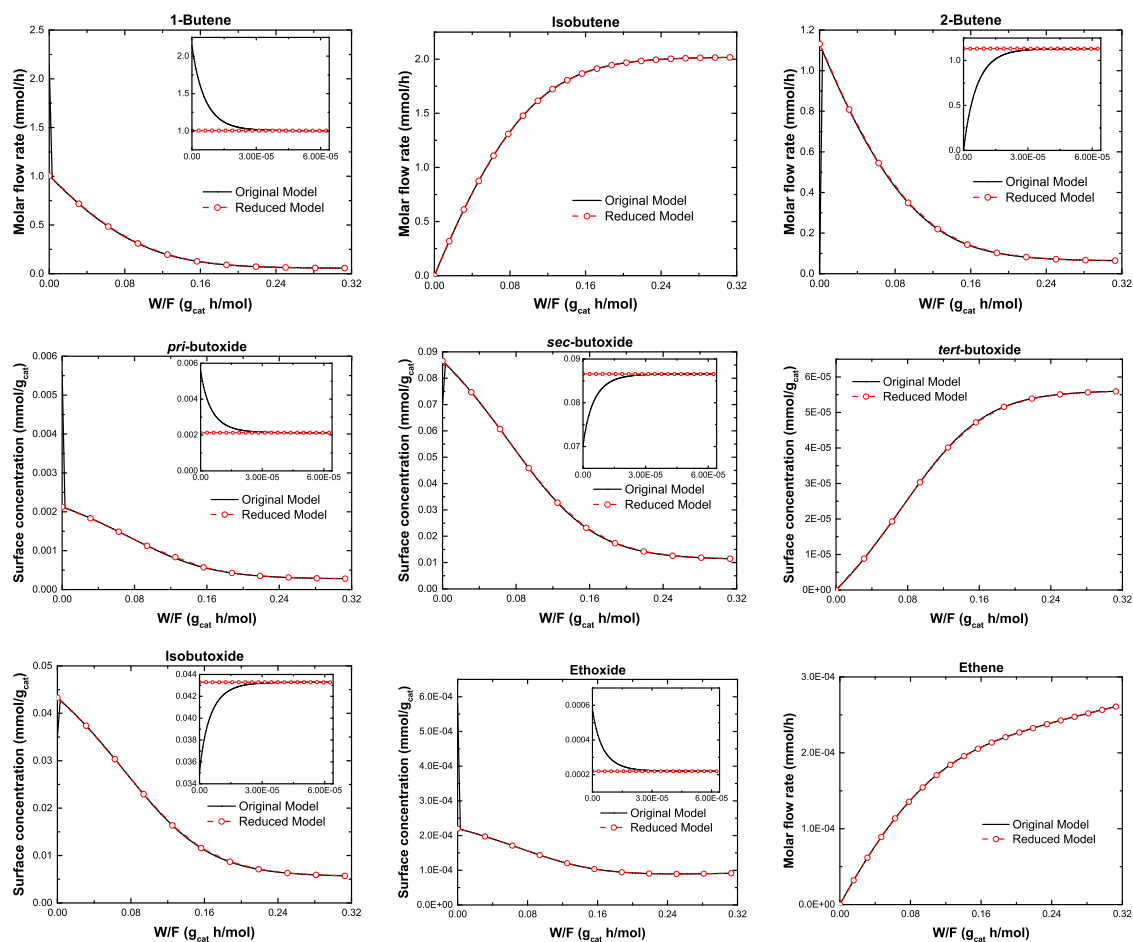
Figure 3.8: A comparison between the original and the reduced model evolution profiles as a function of reactor volume of various species in 1-butene cracking and isomerization reaction scheme. The solid line (–) denotes the original model and the dashed line (-◯-) denotes the reduced model.

The reduced model eliminates the initial fast transient due to incorporation of the quasi-equilibrium constraints as shown in the insets of Fig. 3.8. As shown in Table 3.2, a significant reduction in the number of integration steps (by an order of magnitude) is observed and five (out of eight) model parameters are eliminated by employing the quasi-equilibrium approximation.

Table 3.2: A comparison of integration steps and model parameters between the original and reduced models proposed for 1-butene cracking. The IDAS package [3] was used to simulate the models using a relative tolerance of 10^{-4} and an absolute tolerance of 10^{-6}.

	Original Model	**Reduced Model**
No. of steps	228	38
No. of residual evaluations	4875	240
No. of Jacobian evaluations	342	14
No. of non-linear iterations	426	55
No. of model parameters	8	3

3.2.2   Carbon metabolism in Erythrocytes

Erythrocytes are simple systems, due to the lack of compartmentalization, allowing for the study of glycolysis with minimum interference from other pathways (see [116]). The reaction scheme for carbon metabolism in erythrocytes containing 20 species and 25 reactions (15 reversible and 10 irreversible) is shown in Fig. 3.13 [6].
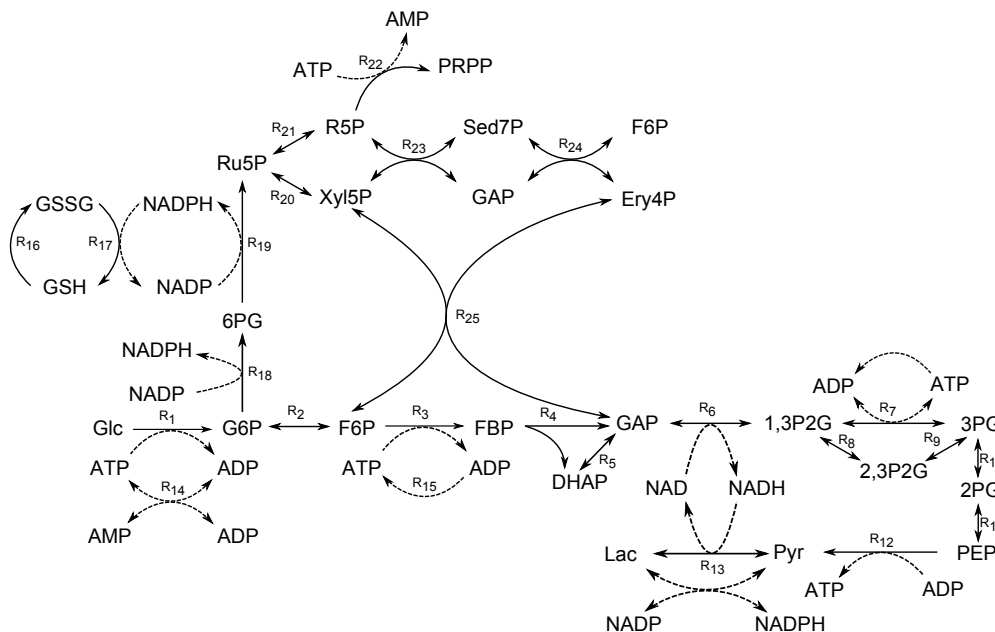
Figure 3.9: Reaction scheme for carbon metabolism in erythrocytes. Adapted from [6].

The rate expressions and the pseudo-first order rate constants for the system are taken from literature reports [116, 123]. The effect of the equilibrium tolerance and kinetic threshold criteria on the accuracy and computational effort of the resulting reduced models was examined. Four different cases shown in Table 3.3 were considered. In case 1, only fast reversible reactions were identified by using an equilibrium index of $\delta = 0.01$. The forward and reverse reaction rates were calculated through a forward simulation of the ODEs listed in the supporting information. In case 2, the equilibrium index criterion was relaxed ($\delta = 0.2$) to include more fast reversible reactions. In case 3, fast irreversible reactions were also introduced with a kinetic threshold $k^{min} = 10^6$ h^{-1}. In case 4, the kinetic threshold was further relaxed ($k^{min} = 10^5$ h^{-1}). Table 3.3 shows the reactions selected for each case. The initial conditions for the original model are listed in Table 3.4. The corresponding sub-networks identified are shown for each respectively.

Table 3.3: Different cases considered for identifying fast/equilibrated reactions

	Identification Criteria	Reactions assumed fast
Case 1	$\delta = 0.01$	R_5, R_{10}, R_{11}, R_{20}
Case 2	$\delta = 0.2$	R_2, R_5, R_{10}, R_{11}, R_{20}, R_{21}
Case 3	$\delta = 0.2$ and $k^{min} = 10^6$ h^{-1}	R_2, R_3, R_5, R_{10}, R_{11}, R_{20}, R_{21}
Case 4	$\delta = 0.2$ and $k^{min} = 10^5$ h^{-1}	R_2, R_3, R_5, R_{10}, R_{11}, R_{20}, R_{21}, R_{22}

Table 3.4: Initial conditions for erythrocytes model

$C_{G6P} = 0.0385$ mM	$C_{6PG} = 0.0049$ mM
$C_{F6P} = 0.0157$ mM	$C_{Ru5P} = 0.016$ mM
$C_{FBP} = 0.007$ mM	$C_{Xyl5P} = 0.016$ mM
$C_{GAP} = 0.0057$ mM	$C_{R5P} = 0.018$ mM
$C_{DHAP} = 0.14$ mM	$C_{SH7P} = 0.0199$ mM
$C_{1,3P2G} = 0.0005$ mM	$C_{E4P} = 0.0076$ mM
$C_{3PG} = 0.0685$ mM	$C_{NADP} = 0.0014$ mM
$C_{2,3P2G} = 5.7$ mM	$C_{ATP} = 1.83$ mM
$C_{2PG} = 0.01$ mM	$C_{AMP} = 0.037$ mM
$C_{PEP} = 0.017$ mM	$C_{GSH} = 3.15$ mM

Case 1: $\delta = 0.01$

Reactions R_5, R_{10}, R_{11}, and R_{20} are treated as fast, with 7 species participating in these fast reactions. The reduced model is initialized so that the corresponding quasi-equilibrium constraints, listed in Table 3.7, are satisfied at initial time t = 0. The reduced model eliminates the initial fast transient due to the incorporation of the quasi-equilibrium constraints as shown in the insets of Figure 3.14.
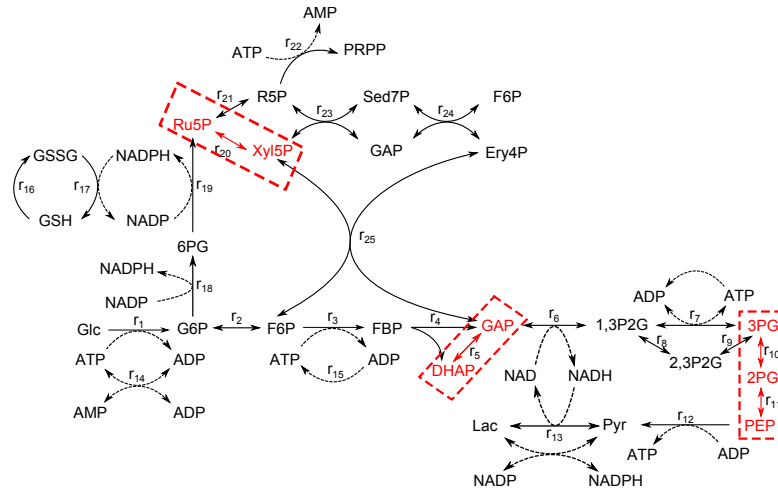
Figure 3.10: Reaction scheme for carbon metabolism in erythrocytes.

4 reactions satisfy the criteria with 7 species participating in these fast reactions.

1. Pseudo-species 1:   3PG + 2PG + PEP

2. Pseudo-species 2:   GAP + DHAP

3. Pseudo-species 3:   Ru5P + Xyl5P

Case 2: $\delta = 0.2$

Reactions R_2 and R_{21} along with reactions in case 1 are added to the set of fast reactions. The same effect in the initial concentrations and the evolution profile of the species (shown in Figure 3.14) in the reduced model is seen as in case 1.
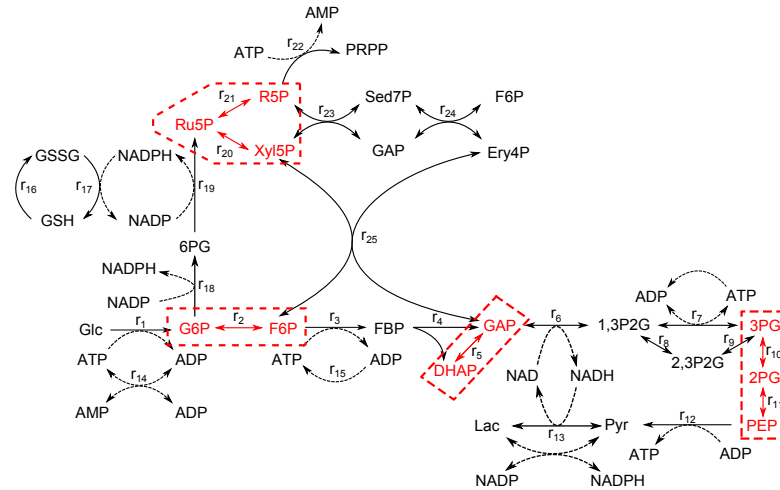
Figure 3.11: Reaction scheme for carbon metabolism in erythrocytes. Adapted from [6].

6 reactions satisfy the criteria with 10 species participating in these fast reactions.

1. Pseudo-species 1:   3PG + 2PG + PEP

2. Pseudo-species 2:   GAP + DHAP

3. Pseudo-species 3:   R5P + Ru5P + Xyl5P

4. Pseudo-species 4:   G6P + F6P

Case 3: $\delta = 0.2$ and $k^{min} = 10^6 \ h^{-1}$

Reaction R_3 is added to the list of fast reactions and the corresponding constraint $(C_{\text{F6P}}C_{\text{MgATP}} = 0)$ results in complete conversion of the reactant species F6P as seen in Figure 3.14. The reduced model is initialized with zero concentration of species F6P. This results in an increase in the concentration of FBP, GAP, and DHAP with time. The evolution profile of the species subsequently aligns with that of the original model.
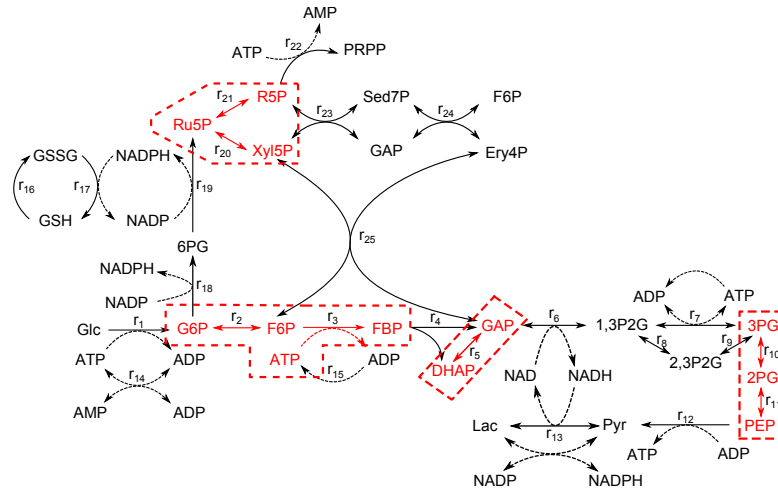
Figure 3.12: Reaction scheme for carbon metabolism in erythrocytes. Adapted from [6].

7 reactions satisfy the criteria with 12 species participating in these fast reactions.

1. Pseudo-species 1:   3PG + 2PG + PEP

2. Pseudo-species 2:   GAP + DHAP

3. Pseudo-species 3:   R5P + Ru5P + Xyl5P

4. Pseudo-species 4:   G6P + F6P + FBP

5. Pseudo-species 5:   ATP + FBP

Case 4: $\delta = 0.2$ **and** $k^{min} = 10^5 \ h^{-1}$

Reaction R_{22} is added to the list of fast reactions and the corresponding constraint ($C_{\text{R5P}}C_{\text{MgATP}} = 0$) results in complete conversion of the reactant species R5P as seen in Figure 3.14. The reduced model is initialized with zero concentration of species R5P. The equilibrium constraint imposed on reactions R_{20} and R_{21} result in zero initial concentration of species Xyl5P and Ru5P. Because of this, accumulation of species E4P takes place with time as can be seen in Figure 3.14, with the evolution profile deviating from that of the original model.
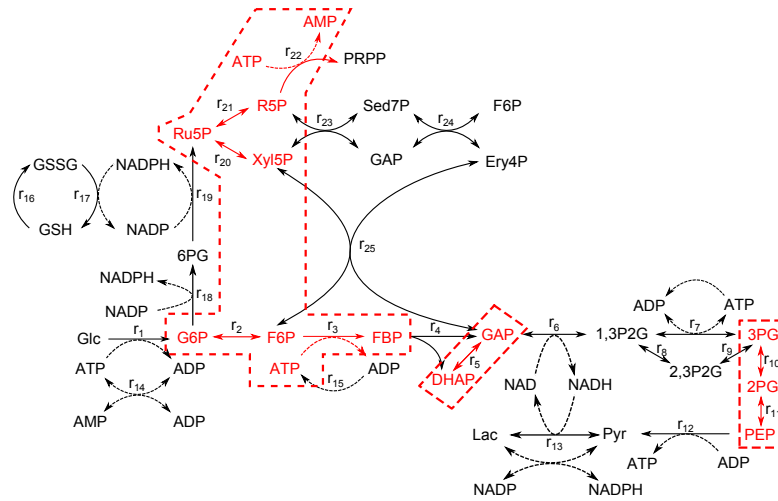
Figure 3.13: Reaction scheme for carbon metabolism in erythrocytes. Adapted from [6].

8 reactions satisfy the criteria with 13 species participating in these fast reactions.

1. Pseudo-species 1: 3PG + 2PG + PEP

2. Pseudo-species 2: GAP + DHAP

3. Pseudo-species 3: G6P + F6P + FBP

4. Pseudo-species 4: ATP + FBP + AMP

5. Pseudo-species 5: R5P + Ru5P + Xyl5P + AMP

Mass balances and rate expressions for the species in the erythrocytes model

The mass balances for the species in the erythrocytes model are listed below:

$$\frac{dC_{\text{G6P}}}{dt} = r_1 - r_{18} - r_2 \tag{3.23}$$

$$\frac{dC_{\text{F6P}}}{dt} = r_2 + r_{24} + r_{25} - r_3 \tag{3.24}$$

$$\frac{dC_{\text{FBP}}}{dt} = r_3 - r_4 \tag{3.25}$$

$$\frac{dC_{\text{GAP}}}{dt} = r_4 + r_5 + r_{23} + r_{25} - r_{24} - r_6 \tag{3.26}$$

$$\frac{dC_{\text{DHAP}}}{dt} = r_4 - r_5 \tag{3.27}$$

$$\frac{dC_{\text{1,3P2G}}}{dt} = r_6 - r_8 - r_7 \tag{3.28}$$

$$\frac{dC_{\text{3PG}}}{dt} = r_9 + r_7 - r_{10} \tag{3.29}$$

$$\frac{dC_{\text{2,3P2G}}}{dt} = r_8 - r_9 \tag{3.30}$$

$$\frac{dC_{\text{2PG}}}{dt} = r_{10} - r_{11} \tag{3.31}$$

$$\frac{dC_{\text{PEP}}}{dt} = r_{11} - r_{12} \tag{3.32}$$

$$\frac{dC_{\text{6PG}}}{dt} = r_{18} - r_{19} \tag{3.33}$$

$$\frac{dC_{\text{Ru5P}}}{dt} = r_{19} - r_{20} - r_{21} \tag{3.34}$$

$$\frac{dC_{\text{Xyl5P}}}{dt} = r_{20} - r_{23} - r_{25} \tag{3.35}$$

$$\frac{dC_{\text{R5P}}}{dt} = r_{21} - r_{23} - r_{22} \tag{3.36}$$

$$\frac{dC_{\text{SH7P}}}{dt} = r_{23} - r_{24} \tag{3.37}$$

$$\frac{dC_{\text{E4P}}}{dt} = r_{24} - r_{25} \tag{3.38}$$

$$\frac{dC_{\text{NADP}}}{dt} = r_{13} + r_{17} - r_{18} - r_{19} \tag{3.39}$$

$$\frac{dC_{\text{ATP}}}{dt} = r_7 + r_{12} - r_1 - r_3 - r_{14} - r_{15} - r_{22} \tag{3.40}$$

$$\frac{dC_{\text{AMP}}}{dt} = r_{22} - r_{14} \tag{3.41}$$

$$\frac{dC_{\mathrm{GSH}}}{dt} = r_{16} - r_{17} \tag{3.42}$$

Table 3.5: Reaction rate constants for glycolysis in erythrocytes

$r_1^{max} = 9.96 \times 10^1 \mathrm{h}^{-1}$	$r_{14}^{max} = 7.50 \times 10^3 \mathrm{h}^{-1}$
$r_2^{max} = 9.56 \times 10^3 \mathrm{h}^{-1}$	$r_{15}^{max} = 3.56 \times 10^{-1} \mathrm{h}^{-1}$
$r_3^{max} = 5.81 \times 10^5 \mathrm{h}^{-1}$	$r_{16}^{max} = 3.00 \times 10^{-2} \mathrm{h}^{-1}$
$r_4^{max} = 1.03 \times 10^6 \mathrm{h}^{-1}$	$r_{17}^{max} = 7.53 \times 10^3 \mathrm{h}^{-1}$
$r_5^{max} = 7.30 \times 10^4 \mathrm{h}^{-1}$	$r_{18}^{max} = 3.88 \times 10^3 \mathrm{h}^{-1}$
$r_6^{max} = 9.06 \times 10^3 \mathrm{h}^{-1}$	$r_{19}^{max} = 2.86 \times 10^3 \mathrm{h}^{-1}$
$r_7^{max} = 6.68 \times 10^3 \mathrm{h}^{-1}$	$r_{20}^{max} = 3.77 \times 10^4 \mathrm{h}^{-1}$
$r_8^{max} = 3.44 \times 10^3 \mathrm{h}^{-1}$	$r_{21}^{max} = 2.51 \times 10^3 \mathrm{h}^{-1}$
$r_9^{max} = 2.6 \times 10^0 \mathrm{h}^{-1}$	$r_{22}^{max} = 8.71 \times 10^4 \mathrm{h}^{-1}$
$r_{10}^{max} = 3.90 \times 10^5 \mathrm{h}^{-1}$	$r_{23}^{max} = 1.68 \times 10^0 \mathrm{h}^{-1}$
$r_{11}^{max} = 7.95 \times 10^4 \mathrm{h}^{-1}$	$r_{24}^{max} = 1.2 \times 10^1 \mathrm{h}^{-1}$
$r_{12}^{max} = 2.69 \times 10^2 \mathrm{h}^{-1}$	$r_{25}^{max} = 6.01 \times 10^0 \mathrm{h}^{-1}$
$r_{13}^{max} = 1.02 \times 10^1 \mathrm{h}^{-1}$	

Table 3.6: Concentrations for cofactors and restrictions imposed in the model

$$C_{\text{Mg}} = 0.7$$

$$C_{\text{CADN}} = 2.0$$

$$C_{\text{ADP}} = C_{\text{CADN}} - C_{\text{AMP}} - C_{\text{ATP}}$$

$$K_{eq,\text{MgATP}} = 0.081$$

$$K_{eq,\text{MgADP}} = 0.81$$

$$C_{\text{MgATP}} = C_{\text{Mg}} * C_{\text{ATP}}/K_{eq,\text{MgATP}}$$

$$C_{\text{MgADP}} = C_{\text{Mg}} * C_{\text{ADP}}/K_{eq,\text{MgADP}}$$

$$P = 0.94$$

$$C_{\text{CO}_2} = 0.2$$

$$C_{\text{Ado}} = 0.0012$$

$$C_{\text{NAD}} = 0.056$$

$$C_{\text{NADH}} = 0.0023$$

$$C_{\text{NDP}} = 0.0644$$

$$C_{\text{NADPH}} = C_{\text{NDP}} - C_{\text{NADP}}$$

$$C_{\text{Pyr}} = 0.077$$

$$C_{\text{GSN}} = 3.15$$

$$C_{\text{GSSG}} = C_{\text{GSN}} - C_{\text{GSH}}$$

$$N_{\text{HK}} = (1 + \frac{C_{\text{MgATP}}}{1.44})(1 + \frac{C_{\text{Mg}}}{1.00}) + (1.55 + \frac{C_{\text{G6P}}}{6.9 \times 10^{-2}})(1 + \frac{C_{\text{Mg}}}{1.00} + \frac{C_{2,3\text{P2G}}}{2.70} + \frac{C_{\text{Mg}}C_{2,3\text{P2G}}}{3.44})$$

$$(3.43)$$

$$N_{\text{PFK}} = 1 + (1.07 \times 10^{-3}) \times \frac{(1 + \frac{C_{\text{ATP}}}{0.01})^4 \times (1 + \frac{C_{\text{Mg}}}{0.44})^4}{(1 + \frac{C_{\text{F6P}}}{0.10})^4 \times (1 + \frac{C_{\text{AMP}}}{0.033})^4} \qquad (3.44)$$

$$N_{\text{ALD}} = (1.94 \times 10^{12} + (2.73 \times 10^{14}C_{\text{FBP}}) + (3.38 \times 10^{13}C_{\text{GAP}}) + (1.77 \times 10^{14}C_{\text{DHAP}})$$
$$+ (1.55 \times 10^{15}C_{\text{FBP}}C_{\text{GAP}}) + (9.31 \times 10^{14}C_{\text{GAP}}C_{\text{DHAP}})) \qquad (3.45)$$

$$N_{\text{PK}} = 1 + (19) \times \frac{(1 + \frac{C_{\text{ATP}}}{3.39})^4}{(1 + \frac{C_{\text{PEP}}}{0.225})^4 \times (1 + \frac{C_{\text{FBP}}}{0.005})} \tag{3.46}$$

$$
\begin{aligned}
Det_{\text{GSSGR}} = {} & 2.86 \times 10^{22} + 4.77 \times 10^{27} C_{\text{NADPH}} + 5.67 \times 10^{26} C_{\text{GSSG}} + 1.29 \times 10^{21} C_{\text{GSH}} \\
& + 4.09 \times 10^{26} C_{\text{NADP}} + 6.65 \times 10^{31} C_{\text{GSSG}} C_{\text{NADPH}} + 2.14 \times 10^{26} C_{\text{NADPH}} C_{\text{GSH}} \\
& + 8.1 \times 10^{30} C_{\text{NADP}} C_{\text{GSSG}} + 9.18 \times 10^{24} C_{\text{GSH}}^2 + 1.84 \times 10^{25} C_{\text{NADP}} C_{\text{GSH}} \\
& + 2.05 \times 10^{28} C_{\text{NADP}} C_{\text{GSH}} + 5.38 \times 10^{30} C_{\text{NADPH}} C_{\text{GSSG}} C_{\text{GSH}} \\
& + 3.44 \times 10^{32} C_{\text{NADPH}} C_{\text{GSSG}} C_{\text{GSH}} + 1.53 \times 10^{30} C_{\text{NADPH}} C_{\text{GSH}}^2 \\
& + 4.05 \times 10^{32} C_{\text{GSSG}} C_{\text{NADP}} C_{\text{GSH}} + 2.95 \times 10^{30} C_{\text{GSH}}^2 C_{\text{NADP}} \\
& + 3.85 \times 10^{34} C_{\text{NADPH}} C_{\text{GSSG}} C_{\text{GSH}}^2 + 4.53 \times 10^{34} C_{\text{GSSG}} C_{\text{GSH}}^2 C_{\text{NADP}}
\end{aligned}
\tag{3.47}
$$

$$
\begin{aligned}
Det_{\text{G6PD}} = {} & 1.45 \times 10^{15} + 1.83 \times 10^{20} C_{\text{NADP}} + 4.29 \times 10^{19} C_{\text{G6P}} + 5.742 \times 10^{17} C_{\text{6PG}} \\
& + 2.04 \times 10^{20} C_{\text{NADPH}} + 6.84 \times 10^{24} C_{\text{NADP}} C_{\text{G6P}} + 7.26 \times 10^{22} C_{\text{6PG}} C_{\text{NADP}} \\
& + 6.01 \times 10^{24} C_{\text{G6P}} C_{\text{NADPH}} + 5.01 \times 10^{24} C_{\text{6PG}} C_{\text{NADPH}} \\
& + 8.65 \times 10^{27} C_{\text{6PG}} C_{\text{NADP}} C_{\text{G6P}} + 1.1 \times 10^{29} C_{\text{G6P}} C_{\text{6PG}} C_{\text{NADPH}} \tag{3.48}
\end{aligned}
$$

$$
\begin{aligned}
Det_{\text{6PGD}} = {} & 1.69 \times 10^{15} + 4.95 \times 10^{18} C_{\text{NADP}} + 7.26 \times 10^{18} C_{\text{6PG}} + 3.45 \times 10^{15} C_{\text{CO}_2} \\
& + 5.58 \times 10^{19} C_{\text{NADPH}} + 2.44 \times 10^{23} C_{\text{6PG}} C_{\text{NADP}} + 1.01 \times 10^{19} C_{\text{NADP}} C_{\text{CO}_2} \\
& + 2.4 \times 10^{23} C_{\text{6PG}} C_{\text{NADPH}} + 5.18 \times 10^{18} C_{\text{CO}_2} C_{\text{R5P}} + 1.14 \times 10^{20} C_{\text{CO}_2} C_{\text{NADPH}} \\
& + 3.14 \times 10^{22} C_{\text{R5P}} C_{\text{NADPH}} + 1.01 \times 10^{24} C_{\text{6PG}} C_{\text{NADP}} C_{\text{CO}_2} \\
& + 1.63 \times 10^{25} C_{\text{6PG}} C_{\text{NADP}} C_{\text{R5P}} + 1.52 \times 10^{22} C_{\text{CO}_2} C_{\text{NADP}} C_{\text{R5P}} \\
& + 1.35 \times 10^{26} C_{\text{6PG}} C_{\text{R5P}} C_{\text{NADPH}} + 1.84 \times 10^{24} C_{\text{CO}_2} C_{\text{NADPH}} C_{\text{R5P}} \\
& + 1.52 \times 10^{27} C_{\text{6PG}} C_{\text{NADP}} C_{\text{CO}_2} C_{\text{R5P}} + 1.25 \times 10^{28} C_{\text{CO}_2} C_{\text{NADPH}} C_{\text{6PG}} C_{\text{R5P}}
\end{aligned}
\tag{3.49}
$$

$$
\begin{aligned}
Det_{\text{TK1}} = {} & 2.63 \times 10^{16} C_{\text{SH7P}} + C_{\text{R5P}} \times (4.4 \times 10^{16} + 4.92 \times 10^{16} C_{\text{SH7P}}) + 5.96 \times 10^{16} C_{\text{GAP}} \\
& + 6.94 \times 10^{16} C_{\text{SH7P}} C_{\text{GAP}} + C_{\text{Xyl5P}} \times (7.35 \times 10^{16} + 2.44 \times 10^{17} C_{\text{R5P}} \\
& + 3.38 \times 10^{17} C_{\text{GAP}}) \tag{3.50}
\end{aligned}
$$

$$
\begin{aligned}
Det_{\text{TA}} = {} & 1.64 \times 10^{17} C_{\text{E4P}} + 2.39 \times 10^{16} C_{\text{GAP}} + C_{\text{F6P}}(1.36 \times 10^{16} + 3.92 \times 10^{17} C_{\text{E4P}} \\
& + 1.11 \times 10^{17} C_{\text{GAP}}) + C_{\text{SH7P}}(3.4 \times 10^{16} + 2.11 \times 10^{18} C_{\text{E4P}} + 4.41 \times 10^{17} C_{\text{GAP}})
\end{aligned}
\tag{3.51}
$$

$$
\begin{aligned}
Det_{\text{TK2}} = {} & 3.01 \times 10^{17} C_{\text{E4P}} + 5.96 \times 10^{16} C_{\text{GAP}} + C_{\text{F6P}}(1.25 \times 10^{16} + 1.6 \times 10^{17} C_{\text{E4P}} \\
& + 3.31 \times 10^{16} C_{\text{GAP}}) + C_{\text{Xyl5P}}(7.34 \times 10^{16} + 1.67 \times 10^{18} C_{\text{E4P}}
\end{aligned}
$$

$$+ 3.38 \times 10^{17} C_{\text{GAP}}) \tag{3.52}$$

$$r_1 = \frac{6.30 \times \frac{C_{\text{MgATP}}}{1.44} \times (1 + \frac{13.35 \times C_{\text{Mg}}}{6.30 \times 1.14})}{N_{\text{HK}}} \tag{3.53}$$

$$r_2 = \frac{\frac{1116 \times C_{\text{G6P}}}{0.182} - \frac{928 \times C_{\text{F6P}}}{0.0714}}{1 + \frac{C_{\text{G6P}}}{0.182} + \frac{C_{\text{F6P}}}{0.0714}} \tag{3.54}$$

$$r_3 = \frac{250 \times \frac{C_{\text{F6P}}}{0.1 + C_{\text{F6P}}} \times \frac{C_{\text{MgATP}}}{0.068 + C_{\text{MgATP}}}}{N_{\text{PFK}}} \tag{3.55}$$

$$r_4 = \frac{3.7 \times 10^{-4} \times ((6.64 \times 10^{19} C_{\text{FBP}}) - (7.81 \times 10^{20} C_{\text{GAP}} C_{\text{DHAP}}))}{N_{\text{ALD}}} \tag{3.56}$$

$$r_5 = \frac{\frac{5415 \times C_{\text{DHAP}}}{0.838} - \frac{59964 \times C_{\text{GAP}}}{0.43}}{1 + \frac{C_{\text{DHAP}}}{0.838} + \frac{C_{\text{GAP}}}{0.43}} \tag{3.57}$$

$$r_6 = (1 \times 10^5 \times P \times C_{\text{GAP}} \times C_{\text{NAD}}) - (5.59 \times 10^6 \times C_{\text{1,3P2G}} \times C_{\text{NADH}}) \tag{3.58}$$

$$r_7 = (1 \times 10^5 \times C_{\text{1,3P2G}} \times C_{\text{ADP}}) - (55.6 \times C_{\text{3PG}} \times C_{\text{ATP}}) \tag{3.59}$$

$$r_8 = \frac{2.75 \times 10^5 \times C_{\text{1,3P2G}}}{(1 + \frac{C_{\text{2,3P2G}}}{0.04})} \tag{3.60}$$

$$r_9 = \frac{0.52 \times C_{\text{2,3P2G}}}{0.2 + C_{\text{2,3P2G}}} \tag{3.61}$$

$$r_{10} = (1 \times 10^5 \times C_{\text{3PG}}) - (6.8 \times 10^5 \times C_{\text{2PG}}) \tag{3.62}$$

$$r_{11} = (1 \times 10^5 \times C_{\text{2PG}}) - (5.9 \times 10^4 \times C_{\text{PEP}}) \tag{3.63}$$

$$r_{12} = \frac{250 \times \frac{(C_{\text{PEP}}/0.225)}{(1 + (C_{\text{PEP}}/0.225))} \times \frac{((C_{\text{MgADP}}/0.474)}{(1 + (C_{\text{MgADP}}/0.474)))}}{N_{\text{PK}}} \tag{3.64}$$

$$r_{13} = \frac{162 \times C_{\text{NADPH}} \times C_{\text{Pyr}}}{0.414 + C_{\text{Pyr}}} \tag{3.65}$$

$$r_{14} = \frac{2.4 \times C_{\text{ATP}} \times C_{\text{Ado}}}{(0.8 + C_{\text{ATP}}) \times (4 \times 10^{-4} + C_{\text{Ado}})} \tag{3.66}$$

$$r_{15} = 0.356 \times C_{\text{ATP}} \tag{3.67}$$

$$r_{16} = 0.03 \times C_{\text{GSH}} \tag{3.68}$$

$$r_{17} = \frac{1.25 \times 10^{-7} \times ((4.82 \times 10^{34} \times C_{\text{NADPH}} \times C_{\text{GSSG}}) - (9.18 \times 10^{32} \times C_{\text{GSH}}^2 \times C_{\text{NADP}}))}{Det_{\text{GSSGR}}} \tag{3.69}$$

$$r_{18} = \frac{9.3 \times 10^{-8} \times ((4.72 \times 10^{27} \times C_{\text{NADP}} \times C_{\text{G6P}}) - (8.04 \times 10^{26} \times C_{\text{6PG}} \times C_{\text{NADPH}}))}{Det_{\text{G6PD}}}$$

$$(3.70)$$

$$r_{19} = \frac{2.1 \times 10^{-6} \times ((8.71 \times 10^{24} \times C_{\text{NADP}} \times C_{\text{6PG}}) - (5.13 \times 10^{25} \times C_{\text{R5P}} \times C_{\text{NADPH}} \times C_{\text{CO}_2}))}{Det_{\text{6PGD}}}$$

$$(3.71)$$

$$r_{20} = \frac{\frac{4.642 \times 10^3 \times C_{\text{Ru5P}}}{1.9 \times 10^{-1}} - \frac{6.67 \times 10^3 \times C_{\text{Xyl5P}}}{0.5}}{1 + \frac{C_{\text{Ru5P}}}{1.9 \times 10^{-1}} + \frac{C_{\text{Xyl5P}}}{0.5}}$$

$$(3.72)$$

$$r_{21} = \frac{\frac{1.7 \times 10^3 \times C_{\text{Ru5P}}}{7.8 \times 10^{-1}} - \frac{7.26 \times 10^2 \times C_{\text{R5P}}}{2.2}}{1 + \frac{C_{\text{Ru5P}}}{7.8 \times 10^{-1}} + \frac{C_{\text{R5P}}}{2.2}}$$

$$(3.73)$$

$$r_{22} = \frac{1.1 \times C_{\text{MgATP}} \times C_{\text{R5P}}}{(0.01 + C_{\text{MgATP}}) \times (0.57 + C_{\text{R5P}})}$$

$$(3.74)$$

$$r_{23} = \frac{3.3 \times 10^{-4} \times ((1.61 \times 10^{22} \times C_{\text{Xyl5P}} \times C_{\text{R5P}}) - (7.81 \times 10^{21} \times C_{\text{SH7P}} \times C_{\text{GAP}}))}{Det_{\text{TK1}}}$$

$$(3.75)$$

$$r_{24} = \frac{6.9 \times 10^{-4} \times ((1.32 \times 10^{22} \times C_{\text{SH7P}} \times C_{\text{GAP}}) - (3.64 \times 10^{22} \times C_{\text{E4P}} \times C_{\text{F6P}}))}{Det_{\text{TA}}}$$

$$(3.76)$$

$$r_{25} = \frac{3.3 \times 10^{-4} \times ((1.1 \times 10^{23} \times C_{\text{Xyl5P}} \times C_{\text{E4P}}) - (3.72 \times 10^{21} \times C_{\text{F6P}} \times C_{\text{GAP}}))}{Det_{\text{TK2}}}$$

$$(3.77)$$

Table 3.7: List of the pseudo-species generated and the algebraic constraints enforced for the fast/equilibrated reactions in the different cases considered for identifying fast/e-quilibrated reactions

Case	Pseudo-species generated	Algebraic constraints
1	3PG + 2PG + PEP GAP + DHAP Ru5P + Xyl5P	$K_5 \times C_{\text{DHAP}} - C_{\text{GAP}} = 0$ $K_{10} \times C_{\text{3PG}} - C_{\text{2PG}} = 0$ $K_{11} \times C_{\text{2PG}} - C_{\text{PEP}} = 0$ $K_{20} \times C_{\text{Ru5P}} - C_{\text{Xyl5P}} = 0$
2	3PG + 2PG + PEP GAP + DHAP R5P + Ru5P + Xyl5P G6P + F6P	$K_2 \times C_{\text{G6P}} - C_{\text{F6P}} = 0$ $K_5 \times C_{\text{DHAP}} - C_{\text{GAP}} = 0$ $K_{10} \times C_{\text{3PG}} - C_{\text{2PG}} = 0$ $K_{11} \times C_{\text{2PG}} - C_{\text{PEP}} = 0$ $K_{20} \times C_{\text{Ru5P}} - C_{\text{Xyl5P}} = 0$ $K_{21} \times C_{\text{Ru5P}} - C_{\text{R5P}} = 0$
3	3PG + 2PG + PEP GAP + DHAP R5P + Ru5P + Xyl5P G6P + F6P + FBP ATP + FBP	$K_2 \times C_{\text{G6P}} - C_{\text{F6P}} = 0$ $C_{\text{F6P}}C_{\text{MgATP}} = 0$ $K_5 \times C_{\text{DHAP}} - C_{\text{GAP}} = 0$ $K_{10} \times C_{\text{3PG}} - C_{\text{2PG}} = 0$ $K_{11} \times C_{\text{2PG}} - C_{\text{PEP}} = 0$ $K_{20} \times C_{\text{Ru5P}} - C_{\text{Xyl5P}} = 0$ $K_{21} \times C_{\text{Ru5P}} - C_{\text{R5P}} = 0$
4	3PG + 2PG + PEP GAP + DHAP R5P + Ru5P + Xyl5P + AMP G6P + F6P + FBP ATP + FBP + AMP	$K_2 \times C_{\text{G6P}} - C_{\text{F6P}} = 0$ $C_{\text{F6P}}C_{\text{MgATP}} = 0$ $K_5 \times C_{\text{DHAP}} - C_{\text{GAP}} = 0$ $K_{10} \times C_{\text{3PG}} - C_{\text{2PG}} = 0$ $K_{11} \times C_{\text{2PG}} - C_{\text{PEP}} = 0$ $K_{20} \times C_{\text{Ru5P}} - C_{\text{Xyl5P}} = 0$ $K_{21} \times C_{\text{Ru5P}} - C_{\text{R5P}} = 0$ $C_{\text{R5P}}C_{\text{MgATP}} = 0$
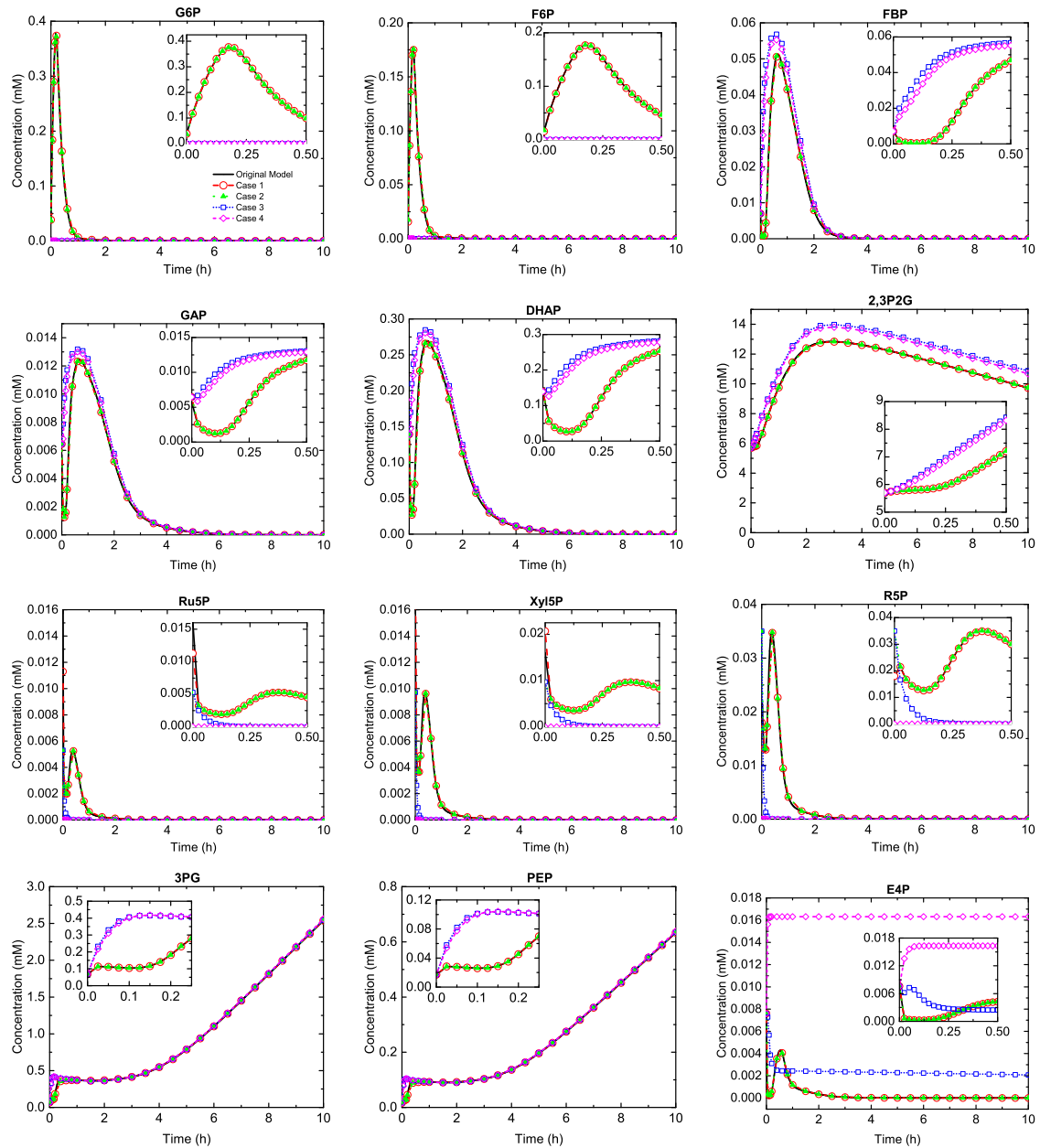
Figure 3.14: A comparison between original and reduced model evolution profiles of various species for the different cases of the identification criteria for fast/equilibrated reactions. The solid line (−) denotes the original model evolution profiles, the symbol (-◯-) denotes the reduced model evolution profiles of various species for case 1, the symbol (-△-) denotes the reduced model evolution profiles of various species for case 2, the symbol (-□-) denotes the reduced model evolution profiles of various species for case 3 and the symbol (-◇-) denotes the reduced model evolution profiles of various species for case 4.

The evolution profiles of the species in Figure 3.14 illustrate that as more and more reactions are considered fast, the dynamics of the reaction system is constrained to a lower dimension. As shown in Table 3.8, a reduction in the integration steps is observed as additional reactions are treated as equilibrated/instantaneous. A similar decrease in the number of model parameters is observed with an increase in the number of algebraic constraints added to the reduced model.

Table 3.8: A comparison between the original and reduced model for carbon metabolism in erythrocytes for the different cases considered for identifying fast/equilibrated reactions listed in Table 3.3

	Original Model	Case 1	Case 2	Case 3	Case 4
No. of integration steps	461	404	386	268	260
No. of residual evaluations	1401	1354	1388	1091	1077
No. of Jacobian evaluations	35	35	37	33	34
No. of non-linear iterations	701	651	645	428	394
No. of model parameters	134	124	116	110	108

The results show that the proposed reduction framework allows the user to systematically address the trade-offs between accuracy and computational complexity in the resulting reduced models.

3.3   Conclusion

A graph-theoretic framework is developed to generate non-stiff non-linear reduced models. Within this framework, a set of pseudo-species that evolve only in the slow time scale are generated as a linear combination of original species via a cycle identification procedure. A reduced model is formulated using these pseudo-species and algebraic constraints arising from fast/equilibrated reactions. The incorporation of complete conversion or quasi-equilibrium constraints allows a reduction in the number of model parameters. The efficacy of the developed framework is illustrated through application on two chemical systems. The cracking reaction scheme of 1-butene over zeolite acids was studied and an order of magnitude reduction in the number of integration steps was observed by incorporating quasi-equilibrium constraints. Further, the trade-off between

the accuracy and the computational complexity of the resulting reduced models for carbon metabolism in erythrocytes system was studied by gradually relaxing the criteria for identifying fast/equilibrated reactions. The developed graph-theoretic framework is an automatic, generic procedure that generates non-stiff reduced models of isothermal reaction systems.

NOMENCLATURE

The following symbols are used in the algorithms presented.

- $\mathcal{G}_\mathcal{B}(S, R, E)$: The bi-partite graph
- C_0 : The set of initial concentration values of species
- k : The set of kinetic parameters
- $\mathcal{IRS}$: The set of irreversible reactions
- $\mathcal{RS}$: The set of reversible reactions
- $\mathcal{N}_R$: The set of reactants of a reaction
- $\mathcal{N}_P$: The set of products of a reaction
- RM_i : Reactant map for species S_i where species is a reactant
- PM_i : Product map for species S_i where species is a product
- E_{fast} : The set of fast edges in the bi-partite graph
- E_{slow} : The set of slow edges in the bi-partite graph
- SQ : Queue to process the species
- NQ : Queue to process the nodes while generating cycles
- $\mathcal{SG}$: Sub-graph consisting of fast reactions only
- $\mathcal{SGS}$: The set of species in the respective sub-graph
- $\mathcal{SGR}$: The set of reactions in the respective sub-graph
- $\mathcal{TS}$: The set of true slow species
- $\mathcal{NR}_i = \mathcal{RM}_i + \mathcal{PM}_i$: The set of reactions for species S_i
- $\mathcal{NP}$: The set of all species pairs
- $\mathcal{NPI}$: The set of species pairs with the corresponding reaction and weights

- w_j : Stoichiometric coefficient of corresponding species in reaction R_j

- $\mathcal{W}_i$: Coefficient of corresponding species S_i in the pseudo-species

- $\mathcal{NP}_i$: The set of species for a species pair

- $\mathcal{NP}_{uni}$: The set of species being checked for a uni-molecular reaction

- $\mathcal{PS}$: The set of pseudo-species generated for a sub-graph

3.3.1 Flowsheet for the steps involved in the graph-theoretic framework

The steps followed in the framework for generating the pseudo-species are shown in Figure 3.15.
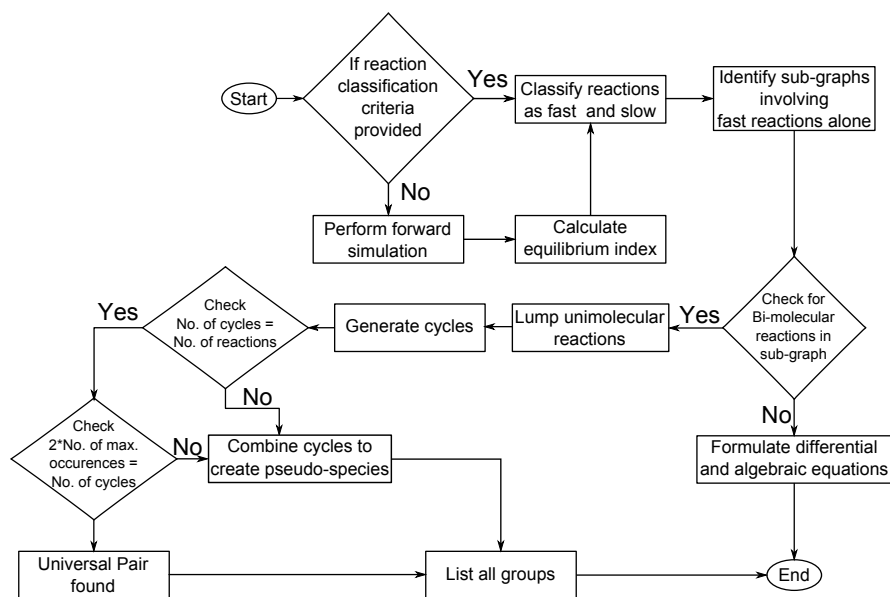
Figure 3.15: Flowsheet of the steps involved in the graph-theoretic framework

3.3.2 Computational requirements of the algorithms

The computational requirements for the algorithms presented in section 2 are discussed below.

Algorithm 1: The algorithm runs over all the edges of the bi-partite graph. For a bimolecular reaction, shown in Figure 3, a total of 8 edges are present. So, for a reaction network with m bimolecular reactions, the maximum number of edges is $8m$. The routines checkThresholdCriterion(E_i) and checkEquilibriumIndexCriterion(E_i) check if the corresponding reaction satisfies the identification criterion for fast reaction. If the reaction does satisfy the identification criterion, the edge is added to ReactantMap or ProductMap of the species participating in the reaction. The computations in this algorithm are therefore, of $O(m)$.

Algorithm 2: The algorithm runs over all the species in the reaction network. The breadth first search algorithm used requires $O(m + n)$ operations [117].

Algorithm 3: The algorithm runs over all the identified sub-graphs in the bi-partite graph. The nested loop runs over all the species within each sub-graph. The total operations for these two loops are of $O(n)$.

Algorithm 4: The algorithm runs over all the identified species pair. As shown in section 2.5, three species pair are found in a bi-molecular reaction. The outer loop runs over $3m_f$ species pair, followed by a nested while loop that runs over n_f fast species and finally another nested for loop that runs over $3m_f$ species pairs. For m_f fast reactions, the number of computations are of $O(m_f^2 n_f)$.

3.3.3 Comparison with the null space-based analysis

The null space-based analysis in the case of n species and m reactions in [87] utilizes $O(4n^2m + 13m^3)$ operations [124] for a (n x m) matrix in the singular value decomposition algorithm.

The most intensive algorithm in the manuscript is the pseudo-species generation algorithm (Algorithm 4). As shown in section 2.5, three species pair are found in a bi-molecular reaction. The outer loop runs over $3m_f$ species pairs, followed by a nested

while loop that runs over n_f fast species and finally another nested for loop that runs over $3m_f$ species pairs. For m_f fast reactions, the number of computations for Algorithm 4 can be considered to be $O(m_f^2 n_f)$. For a large reaction network, $n > n_f$ and $m \gg m_f$ indicating that the present algorithm requires less operations. The upper value of fast species, n_f can be bound to $4m_f$ considering four distinct species in all m_f reactions.

For the breadth first search algorithm, the computational complexity involves $O(m+n)$ operations [117] which is fewer than the operations involved in Algorithm 4 and can be neglected.

Automated Network Generation and Analysis of Biochemical Reaction Pathways Using RING

In this chapter, we show that RING can be adopted naturally for the generation and analysis of biochemical reaction networks. Specifically, we employed RING to elucidate the reaction network for two complex biochemical systems, the transformation of five-carbon sugar xylose to 2-ketoglutarate, and the generation of N-glycosylation networks in mammalian cells. We also demonstrated that RING can be used to predict specific biochemical pathways to species of interest. A new reaction network display module was also added onto RING to allow depicting the effects of enzyme knockout on the reaction network.

4.1 Synthesis of 2-Ketoglutarate from Xylose

2-Ketoglutarate (2KG, also known as α-KG) is an important intermediate in the TCA cycle and amino acid metabolism. It has a broad scope of applications such as a dietary supplement, or an agent that possesses wound healing, anti-oxidative stress, immunomodulatory, and bone anabolic activities [125, 126]. 2KG has also been used as a precursor for the synthesis of heterocyclic compounds [127] or a biodegradable polymer with potential use in tissue scaffolding or drug delivery [128]. In this study, we used RING to identify all possible synthetic routes of 2KG from xylose, a lignocellulosic sugar

61

derived from renewable feedstock.

The reaction rules were written based on the enzymatic reactions in the KEGG database. For generating the relevant reaction network, a subset of reactions were chosen from the large number of enzyme reactions in the KEGG database. Only reaction classes involved in the molecular transformation from a ketose to a 2-keto-carboxylic acid were considered. Those include oxidoreduction, transferase, hydrolase, lyase, isomerase and ligase reactions. Within each reaction class, we excluded reaction sub-classes that involve substrates containing ether, ester, amino, sulfate, halide and or peroxide groups. All the reaction sub-classes considered in this study are listed in Table 1 along with the names of rules used for each sub-class. An example of reaction rule for the enzyme sub-class Transferase221a is shown in Figure 2. The reaction rule specifies the biochemical transformation of C, or H or O atoms (reaction center) within a reactant similar to those seen in chemical reaction networks [1]. The reaction rules coded into RING for the network generation are provided in the Supporting Information.

Table 4.1: List of reaction rule names and enzyme class involved in 2KG pathways

Reaction Rule Name	Enzyme Class	Enzyme Class Selection Basis
1) Oxidoreductase reactions		
Oxido111a	EC 1.1.1	The reaction rules were written for reaction classes 1.1, 1.2, and 1.3. The other reaction sub-classes were not considered because their substrates, which contain sulfate, diphenols, peroxide, amino groups, are not of interest.
ReverseOxido111a	EC 1.1.1	
Oxido111b	EC 1.1.1	
ReverseOxido111b	EC 1.1.1	
Oxido111c	EC 1.1.1	
ReverseOxido111c	EC 1.1.1	
Oxido111d	EC 1.1.1	
Dehydrogenase111d	EC 1.1.1	
Oxido121a	EC 1.2.1	
Oxido121b	EC 1.2.1	
2) Transferase reactions		
Transferase221a	EC 2.2.1	The reaction rules were written for reaction classes 2.2, 2.3, and 2.7. The other reaction sub-classes were not considered because their substrates, which contain amino, selenium, sulfate groups, are not of interest.
Transferase221b	EC 2.2.1	
Transferase233a	EC 2.3.3	
Transferase271a	EC 2.7.1	
ReverseTransferase271a	EC 2.7.1	
Transferase272b	EC 2.7.2	
ReverseTransferase272b	EC 2.7.2	
3) Hydrolase reactions		
Hydrolaseester311a	EC 3.1.1	The reaction rules were written for reaction classes 3.1 and 3.7. The other reaction sub-classes were not considered because their substrates, which contain ether, ester, amino, sulfate groups, are not of interest.
Hydrolaseester311b	EC 3.1.1	
Hydrolaseester312c	EC 3.1.2	
Hydrolaseester312d	EC 3.1.2	
ReverseHydrolaseester312d	EC 3.1.2	

Hydrolaseeester371a	EC 3.7.1	
4) Lyase reactions		
Lyase411a	EC 4.1.1	The reaction rules were written for reaction classes 4.1 and 4.2. The other reaction sub-classes were not considered because their substrates, which contain amino, sulfate, halide groups, are not of interest.
Lyase412a	EC 4.1.2	
Lyase421a	EC 4.2.1	
Lyase421b	EC 4.2.1	
Lyase421c	EC 4.2.1	
5) Isomerase reactions		
Isomerase532a	EC 5.3.2	The reaction rules were written for reaction classes 5.3 and 5.4. The other reaction sub-classes were not considered because their substrates, which contain amino or cyclic groups, are not of interest.
Isomerase532b	EC 5.3.2	
Isomerase542a	EC 5.4.2	
ReverseIsomerase542a	EC 5.4.2	
6) Ligase reactions		
Ligase621a	EC 6.2.1	The reaction rules were written for reaction classes 6.2 and 6.4. The other reaction sub-classes were not considered because their substrates, which contain sulfate, amino groups, are not of interest.
Ligase641a	EC 6.4.1	
Ligase641b	EC 6.4.1	
Ligase641c	EC 6.4.1	

A   **RULE DESCRIPTION**

```
rule Transferase221a{
linear reactant r1{
    C labeled c1
    O labeled o1 double bond to c1
    C labeled c2 single bond to c1 {connected to 2 $ with single
bond}
    O labeled o2 single bond to c2
    C labeled c3 single bond to c1
    H labeled h1 single bond to c3
    O labeled o3 single bond to c3
    H labeled h2 single bond to o3
  }
linear reactant r2{
    C labeled c4 {connected to 2 $ with any bond}
    O labeled o4 double bond to c4
    H labeled h3 single bond to c4
  }
constraints{! r1 is olefinicMol && ! r2 is olefinicMol && r1 is
phosphorylated && r2 is phosphorylated && r1.size ≥ 12 && r2.size
≥ 12 && r1.size ≤ 16 && r2.size ≤ 16}
break bond (c1, c3)
break bond (o3, h2)
increase bond order (c3, o3)
decrease bond order (c4, o4)
form bond (c1, c4)
form bond (o4, h2)
product constraints on mol{mol is phosphorylated}
}
```

B   **RULE ILLUSTRATION**

C   **CONSTRAINTS**

Reactant r_1
- Must not contain C=C
- Must contain phosphorylated group (-OP(=O)(OH)$_2$)
- Must have molecule size between 12 and 16

Reactant r_2
- Must not contain C=C
- Must contain phosphorylated group (-OP(=O)(OH)$_2$)
- Must have molecule size between 12 and 16

Products
- Must contain phosphorylated group (-OP(=O)(OH)$_2$)

D   **REACTION EXAMPLE**

Figure 4.1: (A) An example of rule implementation in the synthesis of 2KG from xylose. The rule describes the structural requirements for the two reactants of the reaction catalyzed by Transferase221a enzyme. Reactant 1 must contain a terminal -CH(OH)C(=O)CH$_2$OH substructure. Reactant 2 must contain a terminal -CH2OH substructure. A local constraint requires both reactants 1 and 2 to not contain a C=C group, contain a terminal -OP(=O)(OH)$_2$ group, and have molecular sizes between 12 and 16 atoms (excluding hydrogen atoms). A constraint also requires products to contain a terminal -OP(=O)(OH)$_2$ group. (B) and (C) Rule illustration and constraints for the reaction rule defined in Fig. 2A. (D) Representation of an example reaction.

RING generated a reaction network consisting of 4574 species and 12703 reactions. A large number of different reaction paths in the network can convert xylose to 2KG with varying number of steps. The post-processing module of RING was used to further screen biochemical pathways for generation of 2KG. A subset of these reaction paths with a length of at most 10 steps is shown in Figure 4.2. Table 4.2 shows the number of distinct pathways generated for each pathway length. Among the pathways generated is the ubiquitous route of xylose phosphorylation, molecular transformation through pentose phosphate pathway to enter glycolysis, followed by TCA cycle, involving 13 reaction steps (Figure 4.3). The reaction network only focusses on the primary carbon skeleton. The generation of co-substrates, such as ATP/ADP/Pi and NAD/NADH is not considered. A number of reactions require some intermediates as co-substrate, including ribose-5-phosphate (R5P) for a transketolase reaction, and oxaloacetate (OAA) for the formation of citrate from acetyl-CoA. Furthermore, co-products are also formed. In this case, the molecular transformation through transaldolase and transketolase (and an isomerase) converts 3 xylose-5-phosphate to 2 fructose-6-phosphate (F6P) and 1 glyceraldehyde-3-phosphate (G3P) as co-products. The 2 F6P is readily converted to 4 G3P. The generation of these co-substrates takes place though separate pathways. In this case study, we will not further demonstrate the pathways for the formation of glycolysis or TCA cycle intermediates. Instead, we show these pathways in Figures 4.4 and 4.5.
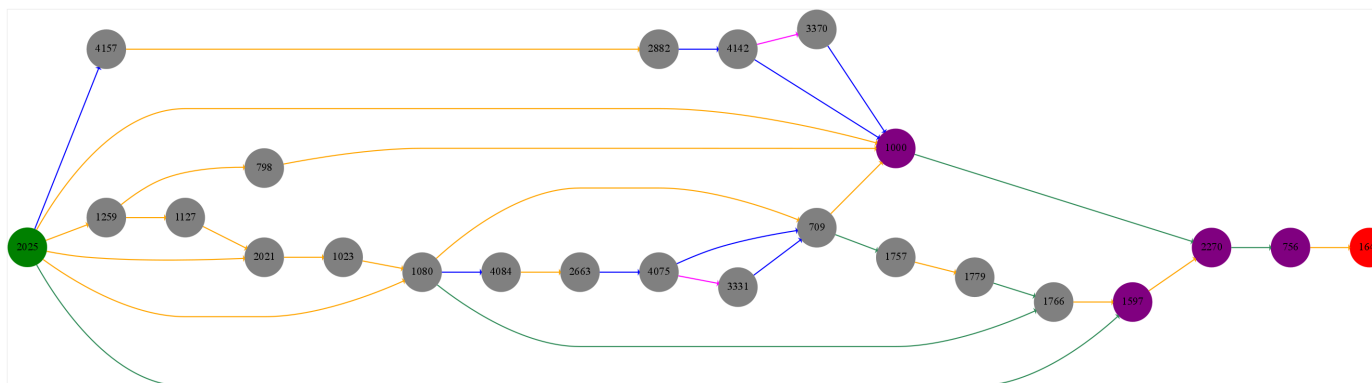
Figure 4.2: The network display for a set of pathways from xylose to 2KG with at most 10 steps. The nodes in the graph represent species along the pathways. The starting node xylose is shown in green, the end node 2KG is shown in red, and the intermediate species are shown in grey. The nodes in purple correspond to species that are present in majority of pathways. Nodes #1000 and #1597 (in purple) are present on two similar pathways whereas nodes #2270 and #756 are present in every pathway. The edges in the graph represent reactions along the pathways. The edges are colored based on the enzyme sub-class. The chemical structures of all the species present in this graph is provided in Table S3.1 of the Supporting Information.

Table 4.2: Number of distinct pathways generated from Xylose to 2KG for each pathway length.

Pathway Length	No. of Distinct Pathways
4	1
6	1
7	2
8	3
9	2
10	6
11	9
12	21
13	39
14	60

Overall Stoichiometry:

$$10 \, NAD^+ + 3 \, C(C(C(C(C=O)O)O)(CO)O + 2.5 \, C(=O)(O)O \rightarrow 2.5 \, C(=O)(CCC(=O)C(=O)O)O + 5 \, O=C=O + 10 \, H^+ + 10 \, NADH$$
$$\text{Xylose} \qquad\qquad\qquad\qquad\qquad \text{2-ketoglutarate}$$
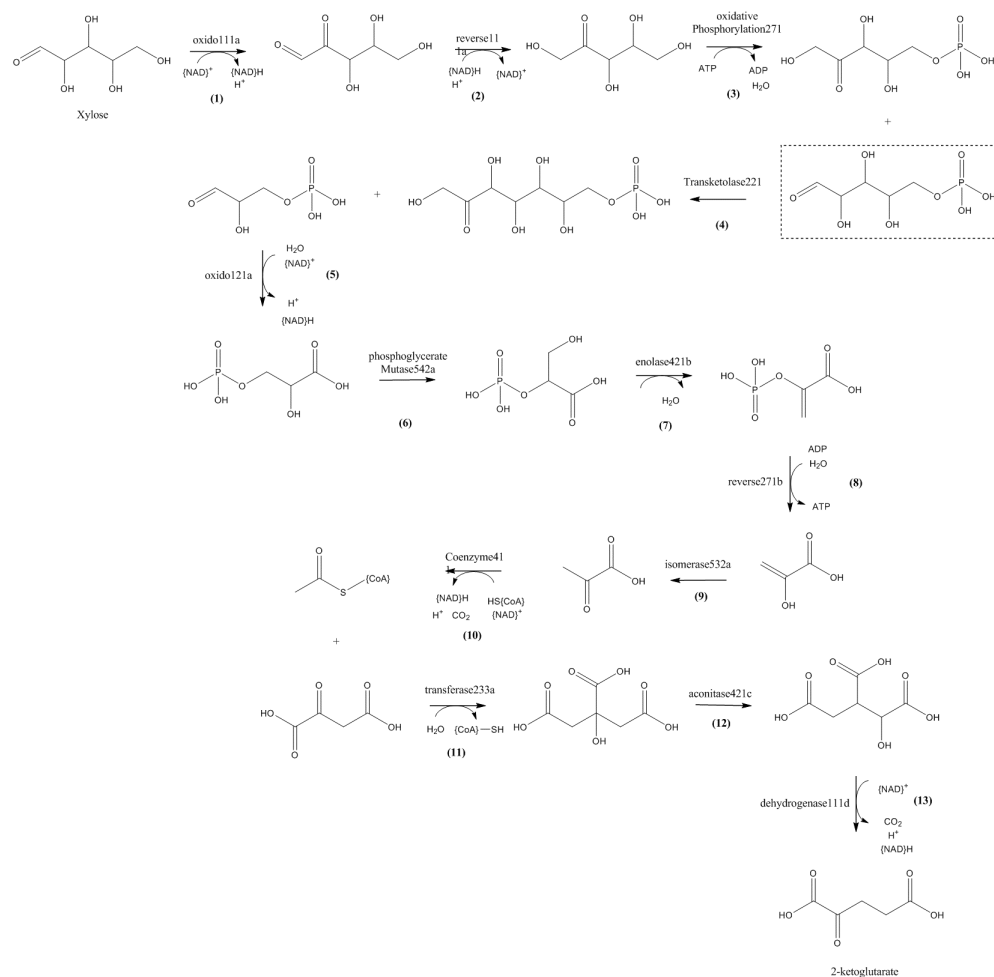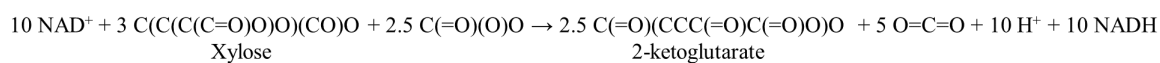
Figure 4.3: Phosphorylative route from xylose to 2KG predicted by RING. The pathway contains 13 reaction steps labeled with the representative reaction rules used to generate the respective reaction. The overall stoichiometry involving the reactants and the products is shown at the bottom.

Overall Stoichiometry

3 ATP + 3 C(C(C(C(C=O)O)O)(CO)O --> 3 O + P(=O)(OCC(C=O)O)(O)O + 2 P(=O)(OCC(C(C(C(=O)CO)O)O)O)(O)O + 3 ADP

Xylose   G3P   F6P
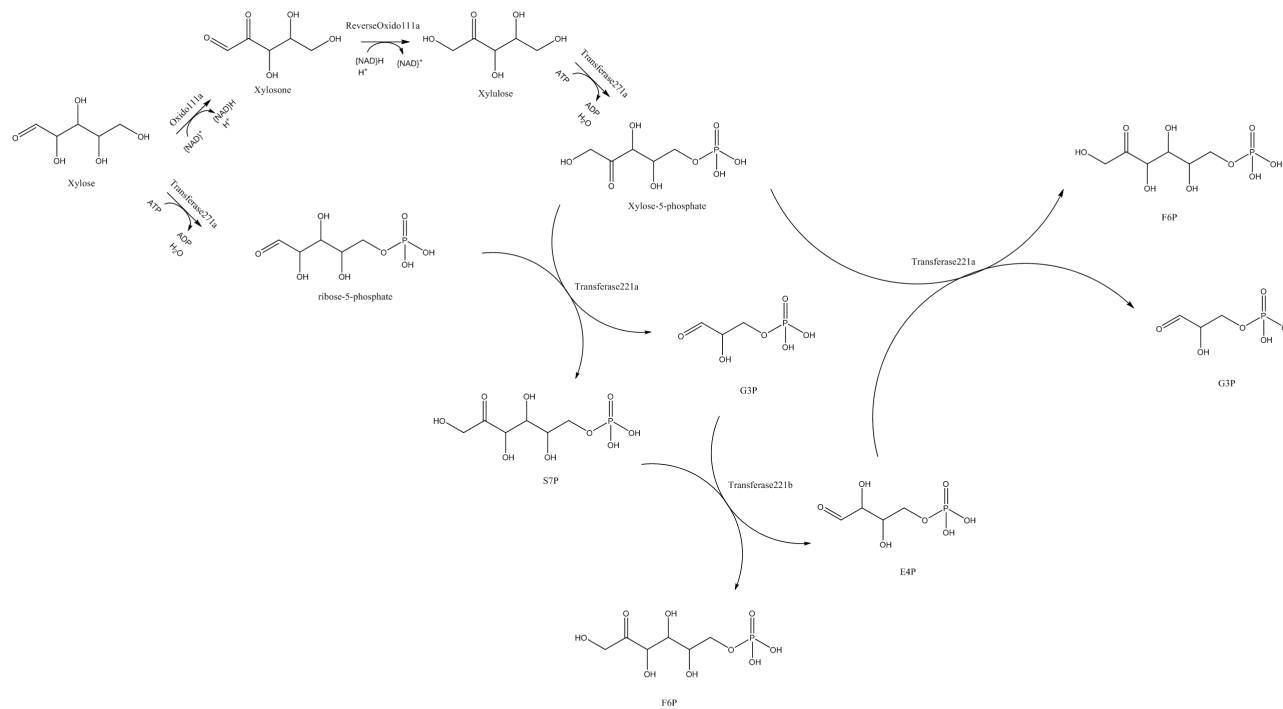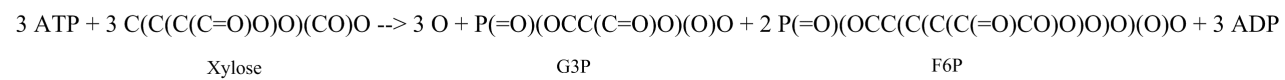
Figure 4.4: The reaction network generated for Pentose Phosphate Pathway (PPP). The overall stoichiometry involving the reactants and the products is shown at the bottom.

Overall Stoichiometry:

10 NAD+ + 3 C(C(C(C(=O)O)O)(CO)O + 2.5 C(=O)(O)O --> 2.5 C(=O)(CCC(=O)C(=O)O)O + 5 O=C=O + 10 H+ + 10 NADH

Xylose   2-ketoglutarate
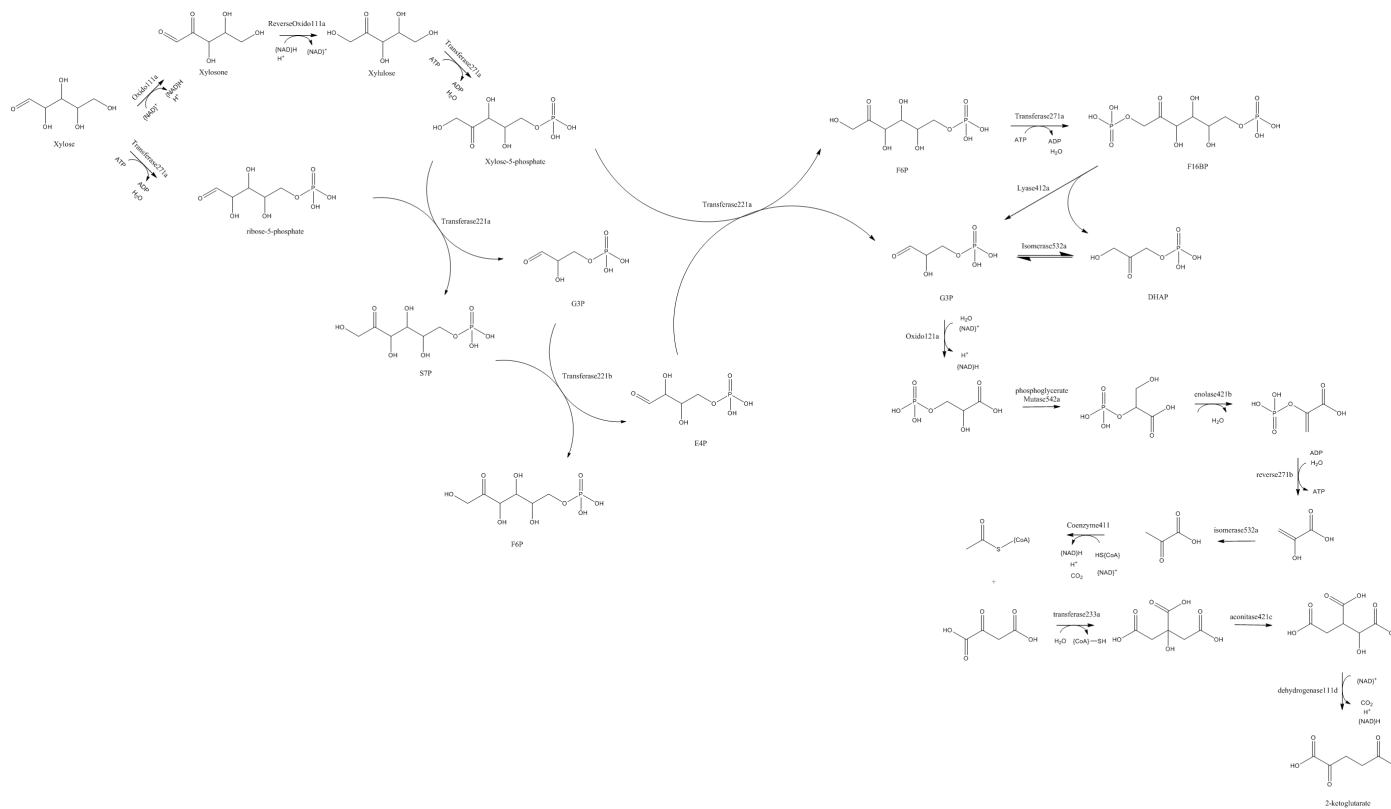
Figure 4.5: The reaction network generated for Pentose Phosphate Pathway (PPP) with glycolysis and TCA cycle. The overall stoichiometry involving the reactants and the products is shown at the bottom.

Among all the pathways generated, the shortest one has four reactions, with two each of dehydratase and oxidoreductase reactions (Figure 4.6). Interestingly, the pathway does not involve phosphorylation using ATP, as seen in glycolysis and the Pentose Phosphate Pathway (PPP). Since this pathway does not involve any phosphorylation or carbon-carbon bond cleavage, it has a higher carbon yield than the conventional route through glycolysis and TCA cycle as illustrated by the respective stoichiometric equations shown in Figure 4.6 and Figure 4.3. This pathway was indeed recently constructed through metabolic engineering in E. coli [129].

Several pathways differ from one another only because the order of occurrence of each elementary step is different. Pathways that have the same number of each elementary step are identified as similar. Figure 4.6 shows two similar pathways containing four reaction steps to form 2KG. The two species (#1000 and #1597) are formed in the two similar pathways containing reaction steps of rules Dehydratase421a and Oxido121a. Such similar pathways were considered as one distinct pathway.

Overall Stoichiometry:

$$2NAD^+ + C(C(C(C(C=O)O)O)(CO)O \rightarrow 2H^+ + 2NADH + C(=O)(CCC(=O)C(=O)O)O$$

Xylose   2-Ketoglutarate
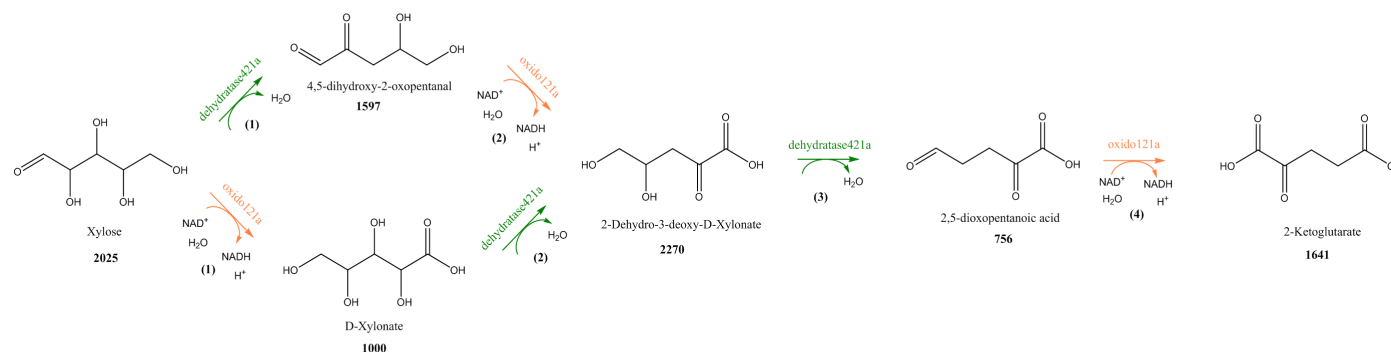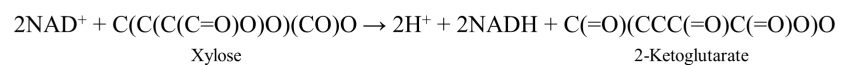
Figure 4.6: Non-phosphorylative route from xylose to 2KG predicted by RING. The pathway contains 4 reaction steps labeled with the representative reaction rules used to generate the respective reaction. The overall stoichiometry involving the reactants and the products is shown at the bottom.

We also examined other distinct pathways generated by RING that had a small number (less than 10) of reaction steps. The pathway that involves six reaction steps is shown in Figure 4.7. The three intermediates shown in the dashed boxes were present in the shortest pathway of length four discussed above (Nodes #1597, #2270 and #756 in purple in Figure 4.2). We note that all pathways with at most 10 reaction steps contain these intermediate species shown in Figure 4.2.

Overall Stoichiometry:

$$2NAD+ + C(C(C(C(C=O)O)O)(CO)O \rightarrow 2H+ + 2NADH + C(=O)(CCC(=O)C(=O)O)O$$
$$\text{Xylose} \qquad\qquad\qquad\qquad\qquad\qquad \text{2-ketoglutarate}$$
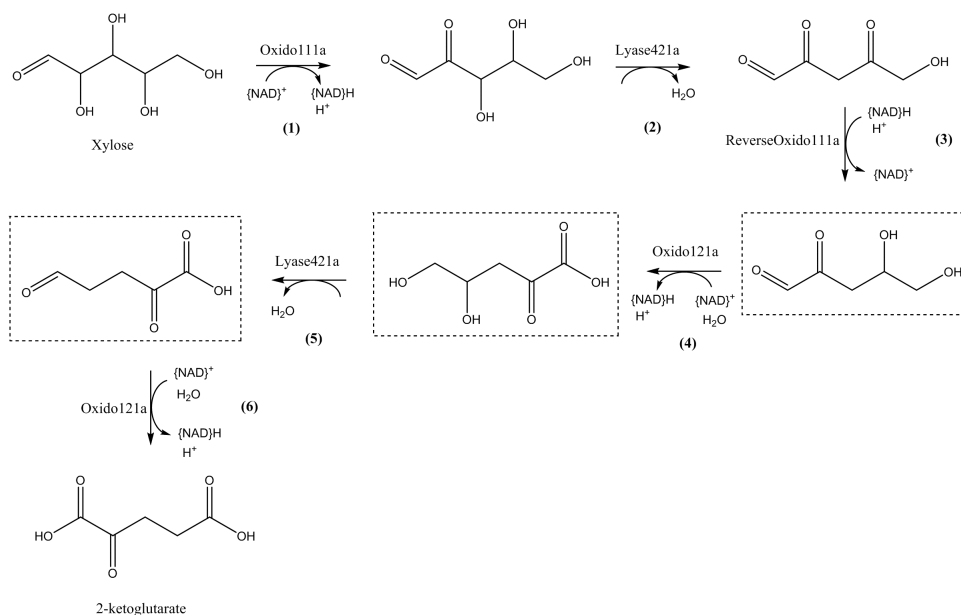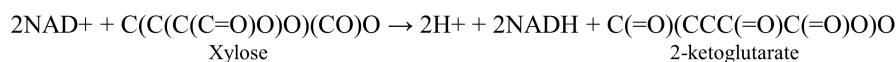
Figure 4.7: A non-phosphorylative route from xylose to 2KG predicted by RING. The pathway contains 6 reaction steps labeled with the representative reaction rules used to generate the respective reaction. The three boxed species are common with the pathway shown in Figure 4 of the main text. The overall stoichiometry involving the reactants and the products is shown at the bottom.

4.2 N-Glycosylation in mammalian cells

Protein N-Glycosylation is a highly branched reaction network with both convergent and divergent branches, whereby monosaccharides are sequentially added or removed from the glycan. N-glycosylation reactions start in the Endoplasmic Reticulum (ER) by forming a core glycan on the N-glycosylation site and continue to grow and diversify in the Golgi compartments. Since each monosaccharide (in the form of a nucleotide sugar) is added or removed as a unit, we treated it as a distinct entity rather than describing all of its chemical elements. The addition or removal of a monosaccharide unit is accompanied by the formation or breakage of glycosidic bonds. In each glycosidic bond formation reaction, only a limited number of carbon atoms on the acceptor and donor monosaccharides are involved as defined by the enzyme-substrate specificity.

In writing RING rules, each monosaccharide unit of a glycan was represented by a symbol using the modified IUPAC condensed nomenclature established by the Consortium for Functional Glycomics, such as Man for mannose and GlcNAc for N-acetylglucosamine (Figure 4.8). The α- and β-glycosidic bonds between two monosaccharides were represented as "a" and "b" accompanied by the carbon positions involved in the bond formation. For example, "A16" represents the $\alpha 1,6$ bond between two mannose molecules as shown in Figure 4.8C. The ten enzymes that constitute a large portion of the mammalian N-glycan biosynthetic pathway are listed in Table 4.3 along with their substrate specificities. An illustration of the reaction rule in RING for the β-1,4-mannosyl-glycoprotein 4-β-N-acetylglucosaminyltransferase (GnTIII) enzyme is shown in Figure 4.9. RING generated a reaction network of N-glycosylation using the enzyme substrate specificity listed in Table 4.3. Man9 glycan, which is generated from ER, was the initial reactant. The resulting reaction network consisted of 350 species and 796 reactions, terminating with 15 fully processed (terminal) glycans.
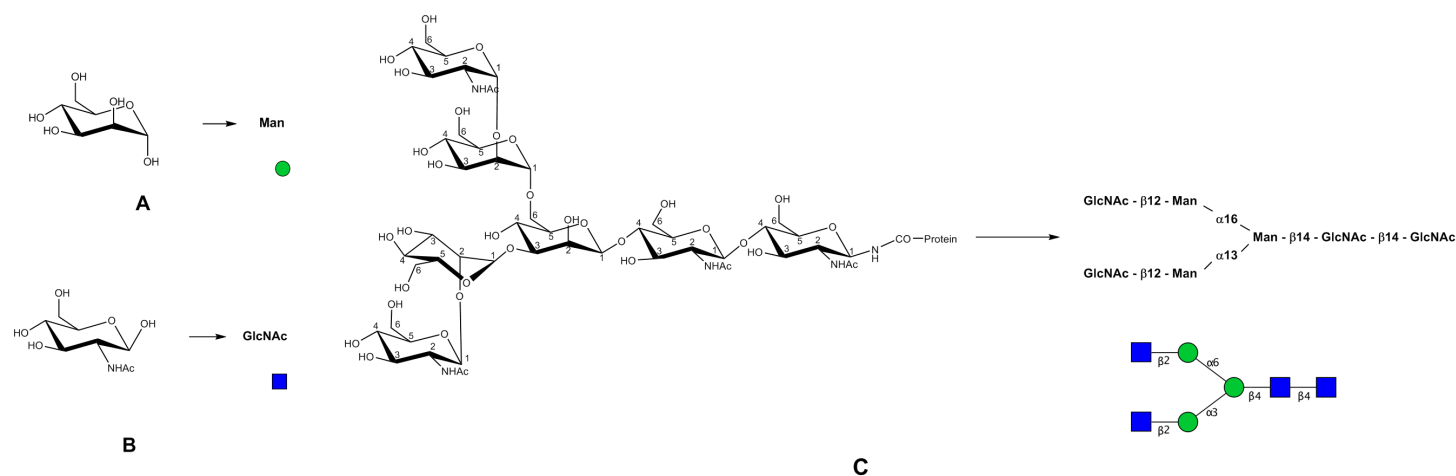
Figure 4.8: (A) Representation of Mannose in RING, (B) Representation of GlcNAc in RING and (C) Representation of the GlcNAcβ1,2Manα1,6Man (α1,3Manβ1,2GlcNAc)β1,4GlcNAcβ1,4GlcNAc- structure in RING. The structure contains four GlcNac and three Mannose molecules linked with each other via glycosidic bonds. The graphical representations of the nucleotide sugars are shown below the pseudo-chemical representations in RING.
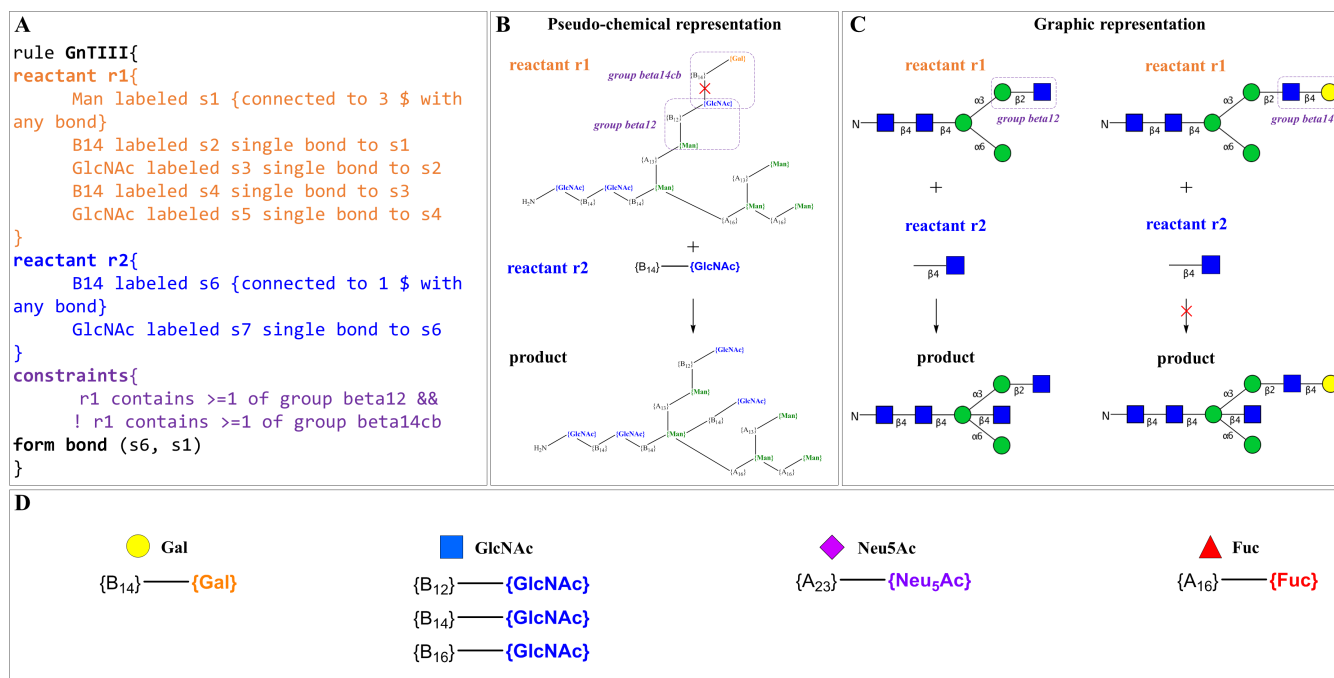
Figure 4.9: (A) An example of rule implementation in N-glycosylation model. The rule describes substrate specificity for the two reactants of the reaction catalyzed by -1,4-mannosyl-glycoprotein 4-β-N-acetylglucosaminyltransferase (GnTIII) enzyme. Reactant 1 must contain Man-β1,4-GlcNAc-β1,4-GlcNAc-Asn substructure. Reactant 2 must be UDP-GlcNAc, shortened as GlcNAc with an overhanging β-glycosidic "bond". A local constraint requires that reactant 1 must contain a pre-added β1,2GlcNAc on the α1,3Man branch. If all the requirements are satisfied, a β-glycosidic bond will be formed between s1 (Man) of reactant 1 and s7 (GlcNAc) of reactant 2 as stated in the "form bond" line. The product glycan will contain the GlcNAc-β1,4-Man-β1,4-GlcNAc-β1,4-GlcNAc-Asn substructure. (B) and (C) Pseudo-chemical illustration and constraints for the reaction rule defined in Fig. 4.9A. The symbolic representation was generated using output from RING. (D) Graphical and pseudo-chemical (by RING) representations of nucleotide sugars.
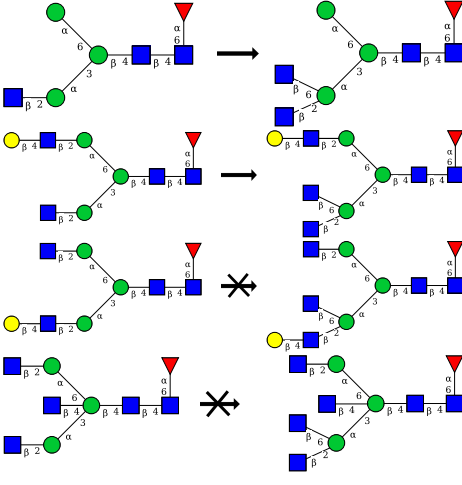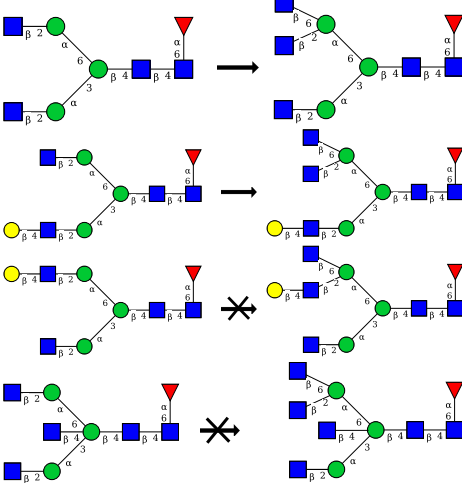
We compared the N-glycosylation network generated by RING with that previously obtained by [4]. The two are similar with respect to network topology and glycans (Table 4.4). Noticeably, RING identified three additional terminal glycans as listed in Table 4.5. The reaction pathways to these terminal glycans are reported in the Figure 4.10 - 4.12. The small differences mainly arose from the implementation of substrate specificity in the reaction rules for enzymes GnTIII and GnTIV.

Table 4.3: List of enzymatic requirements and restrictions considered in the reaction rules [Adapted from [4]]

Enzyme Name	Symbol	Glycan Substrate → Glycan Product	Enzyme-Substrate Specificity
Mannosyl-oligosaccharide 1,2-α-mannosidase	Man I		Requirement: Free $\alpha(1,2)$ Man; ordered removal
Mannosyl-oligosaccharide 1,3-1,6-α-mannosidase	Man II		Requirement: Free $\alpha(1,3)$ or $\alpha(1,6)$ Man, with opposing $\beta(1,2)$ GlcNAc
α-1,3-mannosyl-glycoprotein 2-β-N-acetylglucosaminyltransferase	GnT I		Requirement: All $\alpha(1,2)$ Man removed, only one substrate
α-1,6-mannosyl-glycoprotein 2-β-N-acetylglucosaminyltransferase	GnT II		Requirement: $\beta(1,2)$ GlcNAc must add to $\alpha(1,3)$ Man branch first Restriction: Inhibited by bisecting $\beta(1,4)$ GlcNAc Restriction: Once $\beta(1,4)$ Gal adds to opposing branch, activity is inhibited
β-1,4-mannosyl-glycoprotein 4-β-N-acetylglucosaminyltransferase	GnT III		Requirement: $\beta(1,2)$ GlcNAc must add to $\alpha(1,3)$ Man branch first Restriction: Any prior Gal addition precludes activity

α-1,3-mannosyl-glycoprotein 4-β-N-acetylglucosaminyltransferase	GnT IV		Requirement: Prior addition of $\beta(1,2)$ GlcNAc to $\alpha(1,3)$Man branch required
			Requirement: Prior addition of $\alpha(1,6)$ Fuc is required
			Restriction: Prior addition of $\beta(1,4)$ Gal to $\alpha(1,3)$ Man branch precludes activity
			Restriction: Inhibited by bisecting $\beta(1,4)$ GlcNAc
α-1,6-mannosyl-glycoprotein 6-β-N-acetylglucosaminyltransferase	GnT V		Requirement: Prior addition of $\beta(1,2)$ GlcNAc to $\alpha(1,6)$Man branch required
			Requirement: Prior addition of a(1,6) Fuc is required
			Restriction: Prior addition of $\beta(1,4)$ Gal to $\alpha(1,6)$ Manbranch precludes activity
			Restriction: Inhibited by bisecting $\beta(1,4)$ GlcNAc

Glycoprotein 6-α-L-fucosyltransferase	FucT		Requirement: Prior addition of at least one GlcNAc Restriction: Inhibited by bisecting $\beta(1,4)$ GlcNAc Restriction: Fully capped glycans with $\beta(1,4)$ Gal are not a substrate
β-N-acetylglucosaminylglycopeptide β-1,4-galactosyltransferase	GalT		Requirement: Free GlcNAc on any branch
β-Galactoside α-2,3/6-sialyltransferase	SiaT		Requirement: Free $\beta(1,4)$ Gal on any branch

Table 4.4: A comparison of the number of species, number of reactions, and the number of terminal glycans generated using GlycoVis and RING.

	GlycoVis	RING
Species	344	350
Reactions	768	793
Terminal Glycans	12	15

Table 4.5: Three additional terminal glycans generated by RING.

Terminal Glycans	Glycan Structure
T1	
T2	
T3	
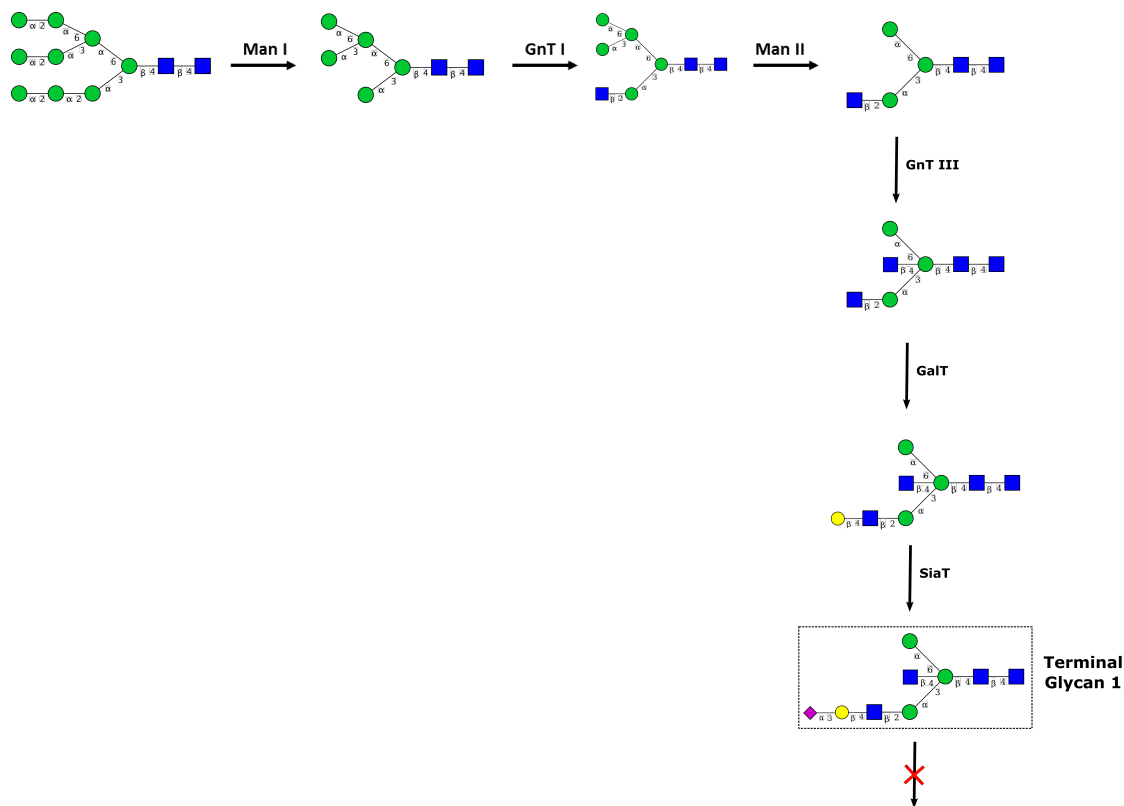
Figure 4.10: Pathway to the additional terminal glycan T1 generated in RING. The terminal glycan T1 was not seen in [4] because the substrates of enzyme GnTIII were limited to fucosylated glycans only.
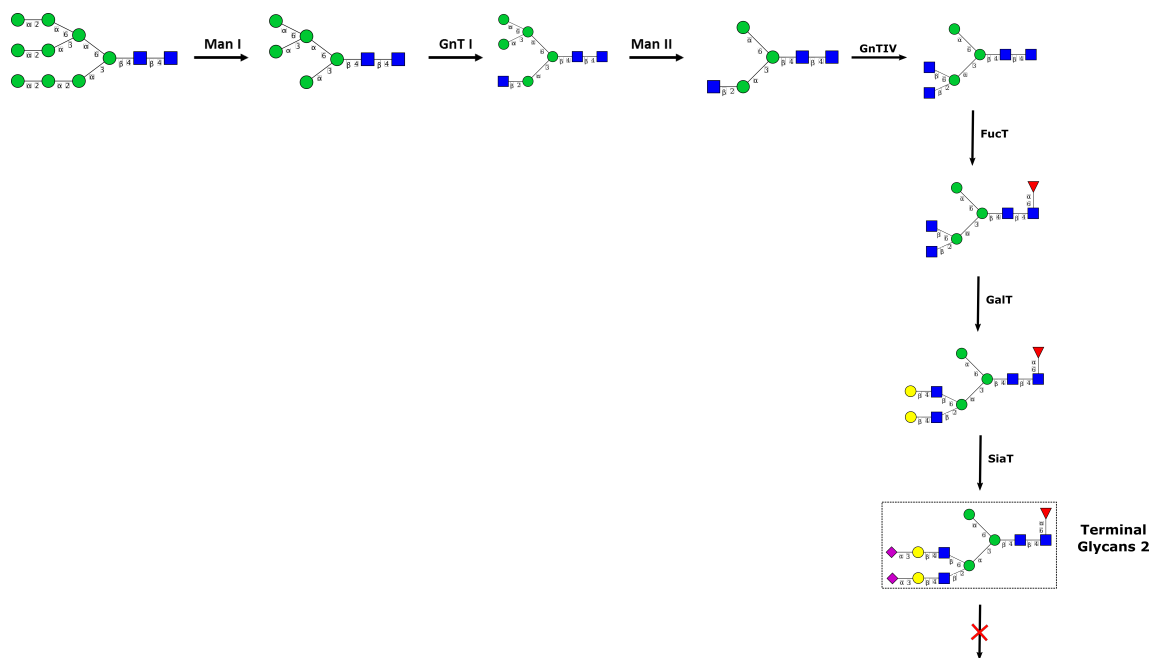
Figure 4.11: Pathway to the additional terminal glycan T2 generated in RING. The terminal glycan T2 was not seen in [4] because the substrates of enzyme GnTIV were restricted to be bi-antennary glycans only. In this study, the enzyme GnTIV can also act on hybrid glycan substrates.
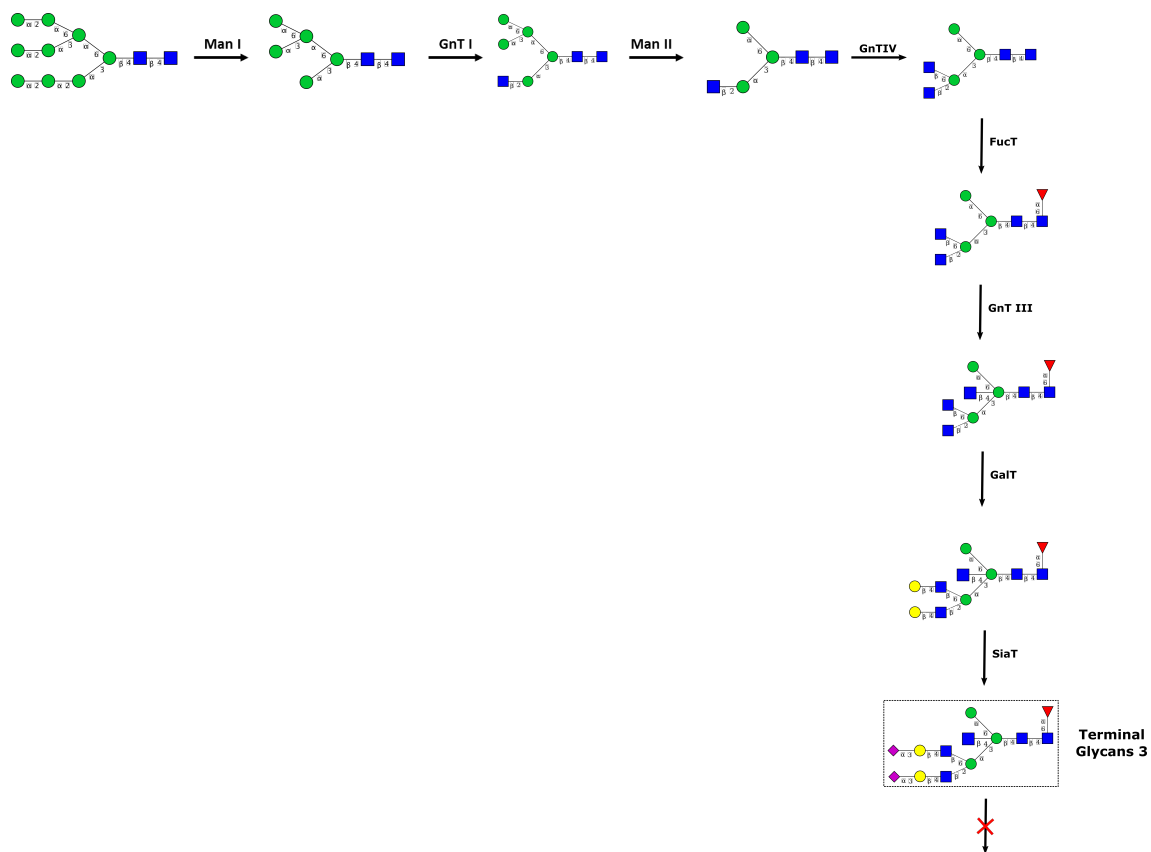
Figure 4.12: Pathway to the additional terminal glycan T3 generated in RING. The terminal glycan T3 was not seen in [4] because the substrates of enzyme GnTIV were restricted to be bi-antennary glycans only. In this study, the enzyme GnTIV can also act on hybrid glycan substrates.

4.3 Demonstration of the network display module in RING

To generate the knockout network, users can list the targeted enzymes in the reaction rule file. The output is a DOT file that can be used to create network display using Graphviz. Figure 4.13 shows the wild type N-glycosylation network generated using RING. Figure 4.14 shows the network resulted from knockout of enzymes GnTIII and GnTV.

N-glycosylation	RING
Reaction rules	10
Species	350
Reactions	793
Terminal glycans	15
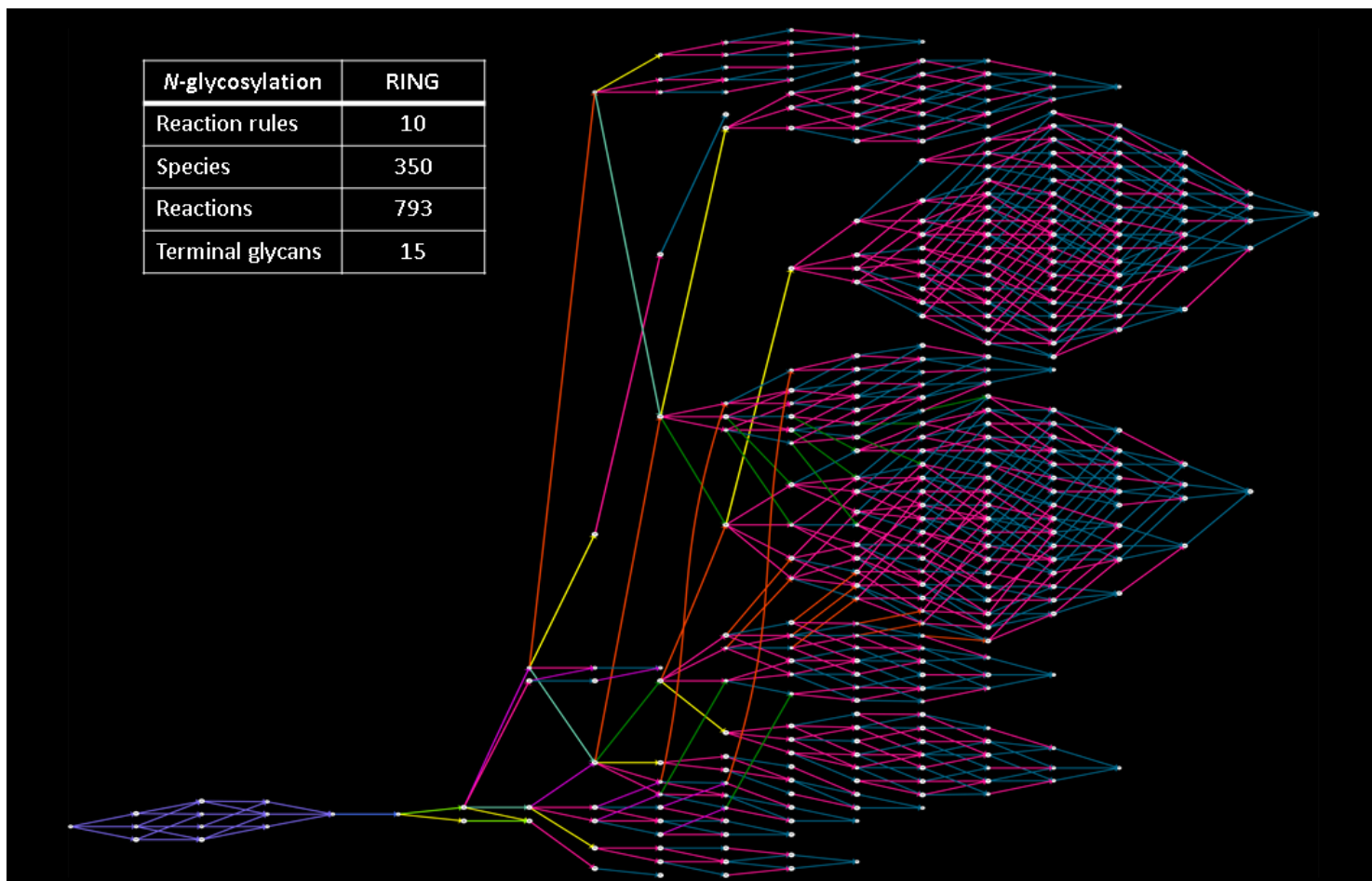
Figure 4.13: Visual representation of the wild type N-glycosylation network generated using RING. Nodes represent glycans and edges being reactions. Edges are colored by the respective reaction rule.
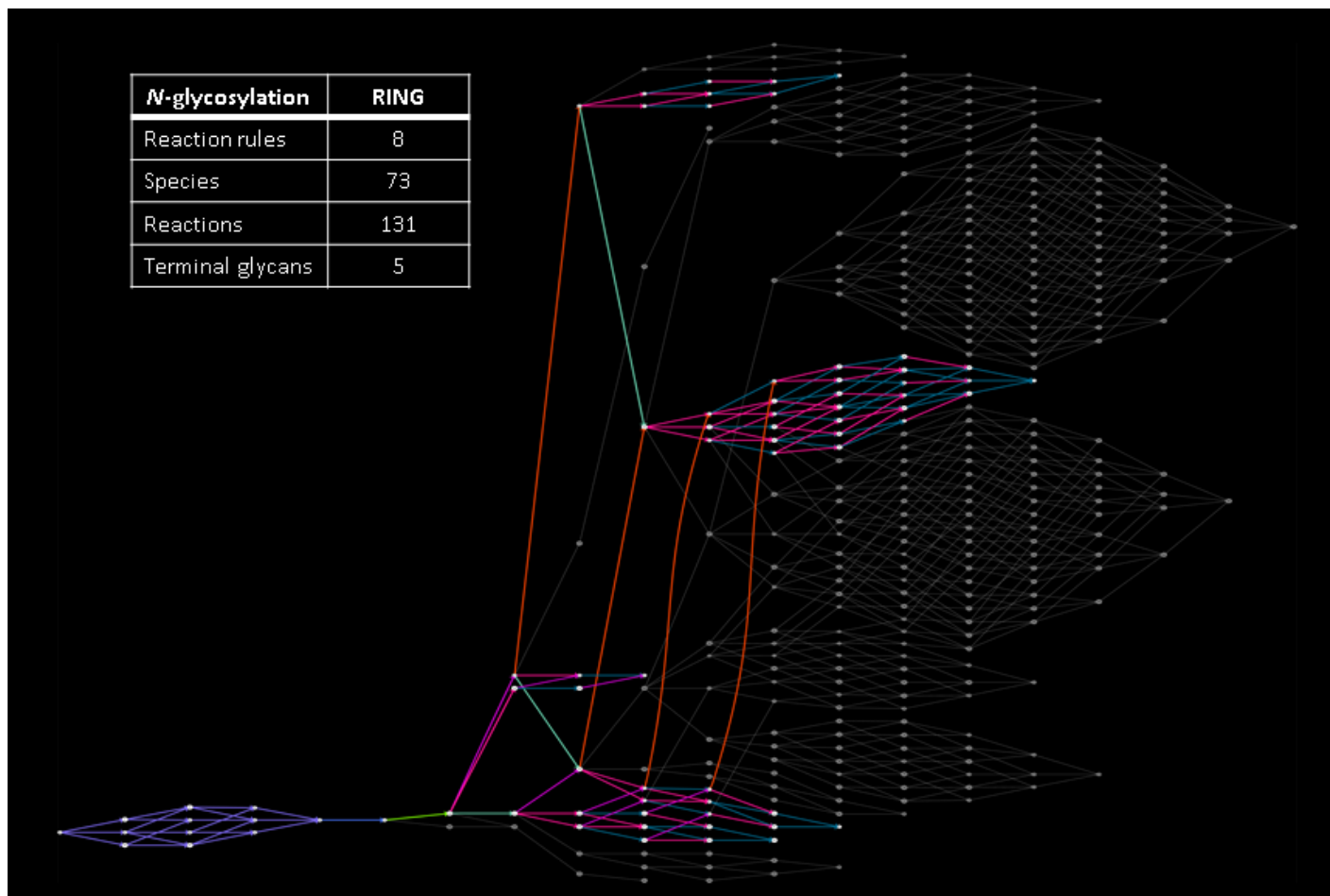
Figure 4.14: The resulting network from the knockout of enzymes GnTIII and GnTV. Nodes represent glycans and edges being reactions. Edges are colored by the respective reaction rule. The eliminated species and reactions are colored grey.

4.4 Discussion

RING adopts a string representation for reactant patterns based on SMARTS (SMiles ARbitrary Target Specification, [130]), which contains well-defined rules and symbols to represent patterns in a molecule. The biochemical reaction network generators mentioned in the introduction have been used to construct and enumerate metabolic pathways with given input substrates and enzymatic reaction rules. RING can generate these metabolic pathways with similar facility. The matrix operations in the Dugundji-Ugi algebraic model of BE and R matrices [37] can become computationally intensive when examining networks comprising complex molecules, e.g., oligosaccharides (glycans) in glycosylation network that can have over a hundred atoms. The simplified representation of molecule sub-fragments into pseudo-atoms in the reaction language along with the topological network analysis features incorporated in RING enable its generic application to different biochemical systems.

A number of network generators have been developed for generating glycosylation networks. GlycoVis is a visualization program that displays the glycan distribution in the N-glycosylation network [4]. Its network generator utilizes matrix manipulation of vector-represented glycan species to generate the reaction network. The algorithm uses a 6-digit number to denote species and a set of reaction rules to manipulate the digits. Similar implementation for N-glycosylation network generation using a 9-digit sequence is shown in [99]. Glycosylation Network Analysis Toolbox (GNAT), an open-source MATLAB based toolbox, generates glycosylation networks by defining enzyme class with detailed specificity information involving enzymatic functional group, linkage and substrate specificity [100]. A formal grammar involving a pattern-matching algorithm for generation of O-glycosylation networks was shown in [101]. These network generators provided different methods for glycosylation network generation and analysis. However, they were developed specifically for glycosylation networks and cannot be directly used for other biochemical systems. RING offers a generic user-friendly platform for reaction rule specification as demonstrated in the case studies.

The application of RING is not limited to the three biochemical systems demonstrated above. Another potential application of RING is to predict novel natural product compounds. After biosynthetic gene clusters are identified in genome sequences [131], a set of enzymatic rules can be derived. Based on that, RING generates a network with all possible species. The molecule query and enzyme knockout features in RING can classify the generated species by molecular groups and associate them with reaction rules

and potentially gene cluster families. Finally, RING can identify unknown pathways leading to products of interest, which might be used as a guide for retrosynthesis or pathway engineering.

4.5 Conclusion

In this study, we described the application of RING to generate a variety of complex biochemical reaction networks through three case studies. In the first case study, we generated reaction pathways from xylose to 2KG in Escherichia coli using reaction rules derived from the KEGG database, and reproduced a novel pathway recently reported [129]. In the second case study, RING was applied to a highly branched convergent and divergent reaction network of N-glycosylation and regenerated the network that was similarly generated using a MATLAB ®based tool [4].

The versatility of RING in generating networks was also demonstrated through enzyme knockout simulations in N-glycosylation reaction system. The network display module allowed visualizing the effects of enzyme knockout on the reaction network. A superimposition of the knockout network on the wild type can assist users to quickly identify species or pathways that are not present in the knockout network.

Microkinetic Modeling of Olefin Interconversion on Self-pillared Pentasil MFI

Olefin interconversion for upgrading light olefins to produce heavier hydrocarbon fuels over acid-type catalysts has been widely investigated for many years ([132], [133]). Olefin interconversion involves acid-catalyzed carbenium ion chemistry involving adsorption and desorption, oligomerization, β-scission, cyclization, isomerization, and hydride transfer reactions. This chapter presents a detailed microkinetic model for olefin interconversion. The reaction network is generated using reaction rules based on reported mechanisms from the literature using Rule Input Network Generator (RING). The reaction network is lumped using chemical functionality-based lumping to reduce its size. The lumped reaction network is used for parameter estimation with experimental datasets at 723 K and varying space times (W/F = 0-2 gcat-h/mol). The following sections describe the steps involved in developing the microkinetic model. The experimental procedure is provided in the appendix.

5.1 Network generation

The reaction rules used along with their constraints are summarized in Table 5.1. Figure 5.1 illustrates a subset of reactions showing the interconnectivity of these reaction rules. The reaction rule file is given in section S1 of the Supporting Information.
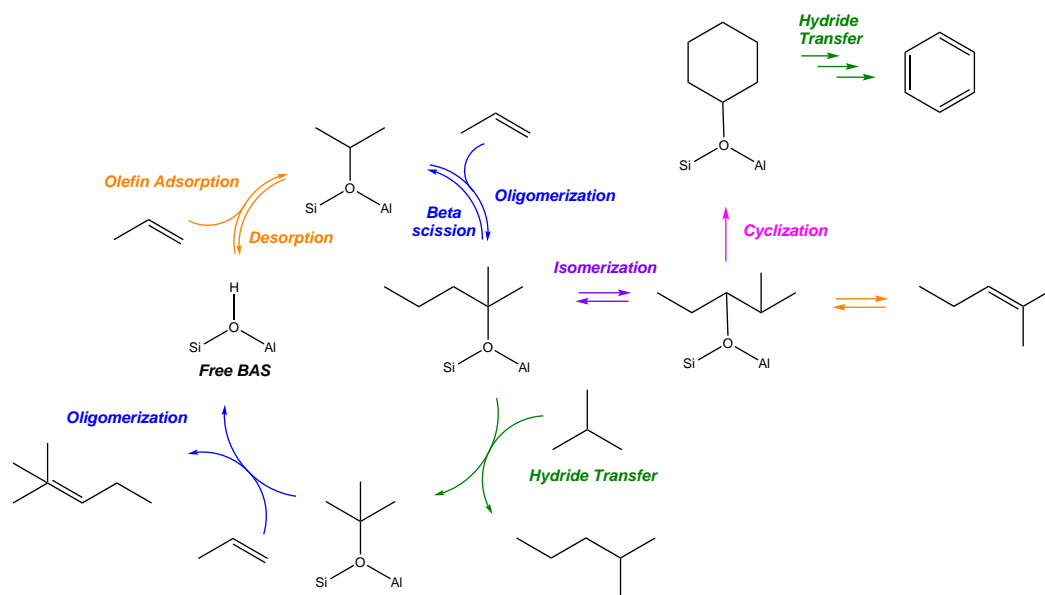
Figure 5.1: An illustration of a subset of reactions showing the interconnectivity of the reactions rules.

5.1.1   Reaction Rules

Olefin Adsorption/Desorption

The olefin adsorption involves a physisorption step wherein the olefin double bond inter-acts with the Brønsted Acid Site (BAS), followed by a chemisorption step wherein the proton transfers from BAS to one of the carbon atoms in the double bond and simultae-neous C-O bond formation at the adjacent lattice oxygen [134, 135]. The reaction rule defined in the network generation scheme involves two reactants - an olefin and a free BAS with the product being an alkoxide. The olefin adsorption reactions are considered to be fast and equilibrated independent of the carbon number. The olefin desorption is defined as the reverse reaction step of adsorption for thermochemical consistency. Chemisorption enthalpies and entropies of linear and branched alkenes/alkoxides are estimated using group additivity [136, 120].

Theoretical calculations reported by [137] show different stable intermediate species on the catalyst surface at relevant cracking temperatures, 773-873K. For linear alkenes, the alkoxide species is more stable than the carbenium ion, whereas, for branched alkenes, the carbenium ion is more stable than the alkoxide species. In our representation of the

reaction rule, we consider the alkoxides as the only intermediate species on the catalyst surface.

Aromatics Adsorption/Desorption

The aromatic adsorption involves protonation of the aromatic cycle in where the aromatic double bond interacts with the BAS. The chemisorption step involves proton transfer from a BAS to a carbon atom of an aromatic double bond and simultaneously a C-O bond formation at the lattice oxygen [138]. The reaction rule defined in the network generation involves two reactants - an aromatic and a free BAS with the product being a cyclic alkoxide. Aromatic adsorption reactions are considered to be fast and equilibrated independently of the carbon number. The aromatic desorption is defined as the reverse reaction step of adsorption for thermochemical consistency. 5-membered rings are not allowed to desorb since they are not observed in the effluent.

Olefin Oligomerization and Beta scission

Olefin oligomerization involves addition of a gas-phase species on an alkoxide. The reaction step is the reverse of olefin cracking where the mechanism requires protonation of an olefin to form an alkoxide intermediate, followed by β-scission of the alkoxide to form a smaller olefin and a smaller alkoxide. The smaller alkoxide subsequently desorbs to form another olefin and leaves behind a proton to regenerate the acid site. Olefin cracking occurs through different modes: A (3°→ 3°), B (2°→ 3°, 3°→ 2°), C (2°→ 2°), D (1°→ 2°, 2°→ 1°), E (1°→ 3°, 3°→ 1°), and F (1°→ 1°) [139, 140, 120]. Methane was used as an internal standard in the experiments and therefore, a constraint on methoxide formation as a product for olefin cracking is imposed since no change in methane flow rates is observed in the effluent across all experiments.

Isomerization

Skeletal isomerization reactions involve hydride shift and methyl shift for acyclic alkoxides, ring methyl shift and ring allyl shift for cyclic alkoxides. The mechanism involves the shift (moving of the electrons) of a hydride or methyl to an adjacent carbon resulting in a new carbocation where the substituent moved from [141]. The reactions are considered fast and equilibrated independently of the carbon number [140, 142].

Ring Closure
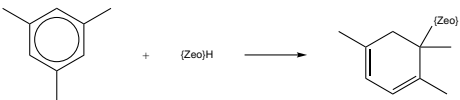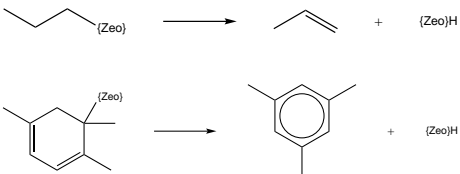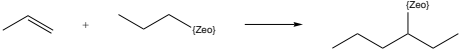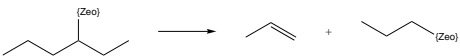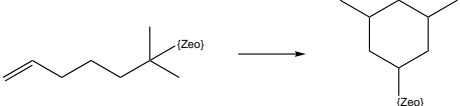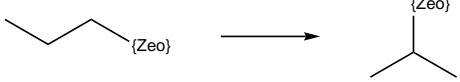
Alkoxide cyclization is a step preceeding the formation of aromatics. The reaction involves a carbenium ion ($\geq C_6$ hydrocarbons) undergoing ring closure to form cycloalkane or cycloalkene intermediates. 5-membered ring species were not observed in any significant concentration in the effluent. Hence, only 6-membered ring formation is considered in the reaction rule based on experimental observations. Further, only polymethylbenzenes formation is considered for C_8 and higher hydrocarbons since no ethyl or higher exocyclic alkyl fragments were observed in the effluents.

Hydride Transfer

The mechanism for hydride transfer starts with a hydrogen-rich hydrocarbon attacking the C-O alkoxy bond of the adsorbed intermediate, resulting in a hydride-sharing cationic species that has to undergo a rotation, so that the positive charge within the complex stays stabilized by the negative charge left on the deprotonated acid site [143, 144]. The reaction rule defined involves shuttling of the hydride among the alkoxide and the gas-phase species. A similar reaction rule is written for cyclic intermediates and cyclic gas-phase species as well. For thermodynamic consistency, the reactions are divided in two groups to account for the reversibility.

Table 5.1: List of reaction rules and constraints considered in the reaction rules

Reaction rule	Illustrative example	Rule constraints
Olefin adsorption		Species contains one or more C=C bond and is not an aromatic
Aromatic adsorption		Species contains aromatic C=C bond part of a 6-membered ring
Desorption		6-membered rings can desorb alone
Oligomerization		Acyclic species participate in reaction, sum of the size of reactants ≤ 11
Beta-scission		No methyl/methane formation
Cyclization		Restrict 5-membered ring formation
Hydride Shift		-

Methyl Shift		-
Ring Methyl Shift		Requires two C=C ring bonds
Hydrogen Transfer		sum of the size of reactants ≤ 19, reactants are not cyclic
Hydrogen Transfer with Cyclics		sum of the size of reactants ≤ 19, cyclic species has at maximum two C=C ring bonds

The original network generated involves 4246 species and 19716 reactions. A kinetic model for such a large system is computationally intensive. Therefore, lumping of the reaction network is required to reduce its size.

5.2 Lumping

The species are lumped using chemical functionality-based lumping as described in [102]. Additional constraints are added to select the lump representative. This is important considering that the lumps have a wide range of thermodynamic values of species within each lump. These values for each generated species are estimated using group contribution values as described in [5]. Straight or monomethyl-branched species are observed in the effluent as well as observations from [145], therefore, lumped representatives are identified by finding straight or mono-methyl branched species in the lump. An example of a lump with its corresponding representative is shown in Fig. 5.2.
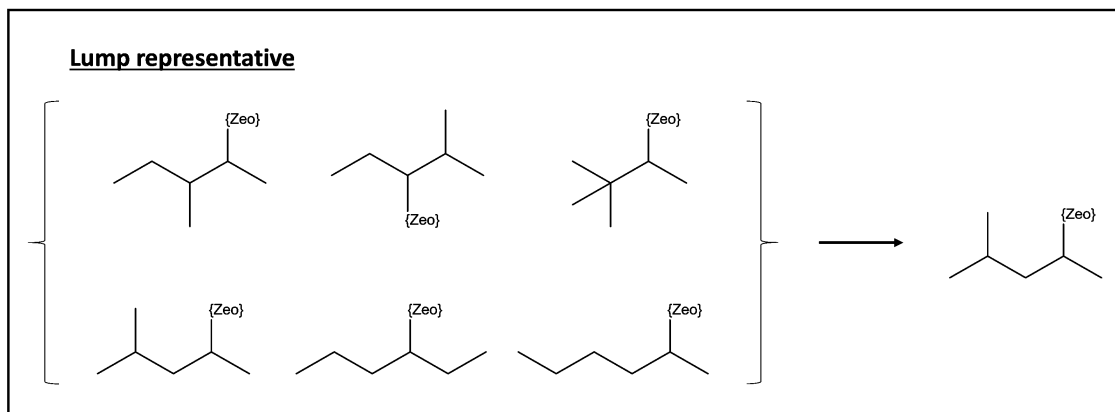
Figure 5.2: Lump representative for a C_6 secondary carbenium ion species. The lumped representative is constrained to mono-methyl branched carbenium ions.

The molecular lumping is done for all olefins, aromatics, paraffins and primary, secondary, and tertiary alkoxides considering the carbon number of each species. The resulting reaction network consists of 127 species and 7802 reactions. The isomer lumps along with their representative molecules are provided in the Supporting Information.

5.3   Parameterization

The lumped reaction network still contains a large number of reactions and it is computationally impractical to estimate the rate constants of each reaction given the experimental datasets. Hence, the rate constant for every elementary reaction is defined either based on the type of alkoxide involved before and after reaction or the size of the alkoxide participating in the reaction as a reactant/product for the various reaction rules. The initial guess for these kinetic parameters for this study have been taken from the literature. Table 5.2 contains all the kinetic parameters used in this study and the corresponding literature source.

Table 5.2: The initial guess for the kinetic parameters used in modeling Olefin interconversion chemistry. k refers to rate constant at 723K

Kinetic information

(i) Olefin adsorption

The olefin adsorption reactions are considered to be fast and equilibrated independently of the carbon number and operating conditions. Sarazen et al., report an experimental study for light alkene conversion to involve equilibration of skeletal and regioisomers under all conditions of pressure (2-400 kPa), temperature (473-533 K) and conversions on MFI catalyst [142].

(ii) Aromatics adsorption

The aromatic adsorption reactions are considered to be fast and equilibrated independent of the carbon number and operating conditions.

(iii) Beta-scission (reverse of Oligomerization)

The beta-scission kinetic parameters are differentiated on the basis of type of alkoxide before and after the reaction as well as the resulting alkoxide size.

Modes A ($3° \rightarrow 3°$), B ($2° \rightarrow 3°$, $3° \rightarrow 2°$), C ($2° \rightarrow 2°$), D ($1° \rightarrow 2°$, $2° \rightarrow 1°$), E ($1° \rightarrow 3°$, $3° \rightarrow 1°$), and F ($1° \rightarrow 1°$)

k_2: $C_9 \rightarrow C_2 + C_7$

k_2: $C_8 \rightarrow C_2 + C_6$

k_2: $C_7 \rightarrow C_2 + C_5$

k_2: $C_6 \rightarrow C_2 + C_4$

k_2: $C_5 \rightarrow C_2 + C_3$

k_2: $C_4 \rightarrow C_2 + C_2$
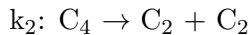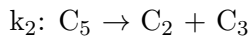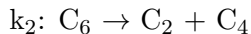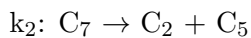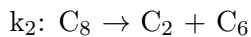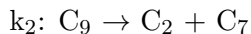
The above rate constant is used for every reaction whether the C_2 species formed is an alkoxide or a gas-phase species. Parameter k_2 for each mode is estimated independently.

k_3: $C_9 \rightarrow C_3 + C_6$

k_3: $C_8 \rightarrow C_3 + C_5$

k_3: $C_7 \rightarrow C_3 + C_4$

k_3: $C_6 \rightarrow C_3 + C_3$
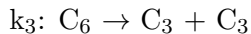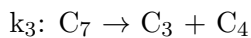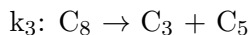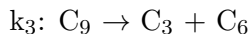
The above rate constant is used for every reaction whether the C_3 species formed is an alkoxide or a gas-phase species. Parameter k_3 for each mode is estimated independently. We note that there is an overlap for reactions involving C_5 beta-scission being considered under k_2 parameter.

k_4: $C_9 \rightarrow C_4 + C_5$

k_4: $C_8 \rightarrow C_4 + C_4$
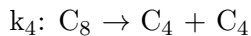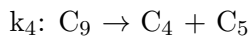
The above rate constant is used for every reaction whether the C_4 species formed is an alkoxide or a gas-phase species. Parameter k_4 for each mode is estimated independently. We note that there is an overlap for some reactions with the above two parameter definitions involving generation of C_4 species.

The initial rate constants and activation energies are taken from [120, 146, 67, 147].

Different parameterizations have been used for beta-scission reactions in the literature. Hinrichsen and coworkers [148, 147, 149] parameterized the rate constants on the basis of the mode of beta-scission. With the assumption that quaternary carbon atoms cannot react and primary carbenium ions cannot undergo oligomerization reactions due to instability of the carbenium ion, only four reaction rate constants for modes B, C, D, and E were estimated. [150] considered the influence of the chain length of the gas-phase olefin in the rate constant for the oligomerization reactions and cracking reactions using an empirical correlation. In this work, the rate constants are parameterized considering both the chain length as well as the mode of beta-scission reaction.

(iv) Cyclization

$k_{cyclization}$: 6.81E+08 1/s

C_6: k $k_{cyclization}$ 1/s Ea 67.1 kJ/mol
C_7: k $k_{cyclization}$*78.2 1/s Ea 38.1 kJ/mol
C_8: k $k_{cyclization}$*78.2 1/s Ea 38.1 kJ/mol
C_9: k $k_{cyclization}$*78.2 1/s Ea 38.1 kJ/mol

Kinetics for cyclization of C_6 species is taken from [151]; Values for higher alkenes are taken from [152]. Only one parameter is used to estimated the rate constants for cyclization reactions. For C_7 and higher alkoxides, a constant factor is used to calculate their cyclization rate constants from the C_6 alkoxide species cyclization rate constant. This is done to reduce the number of parameters to be estimated. Further, C_6 - C_9 primary and secondary alkoxides participate in the cyclization. Differentiation of the type of alkoxide undergoing cyclization is not considered to reduce the number of parameters to be estimated.

(v) Hydride shift, methyl shift, and ring allyl shift

The isomerization reactions are considered to be fast and equilibrated independent of the carbon number and operating conditions. Sarazen et al., report the experimental evidence for skeletal equilibration consistent with rapid hydride and methyl shifts of alkoxide intermediates under different pressure (2-400 kPa), temperatures (473 - 553 K) and conversions [142].

(vi) Hydride transfer

Acyclic species

$3° \rightarrow 3°$: k 3.87e-03 Ea 91 kJ/mol n 0.0

$3° \rightarrow 2°$: k 3.87e-03 Ea 105 kJ/mol n 0.0

$3° \rightarrow 1°$: k 3.87e-03 Ea 90 kJ/mol n 0.0

$2° \rightarrow 3°$: k 3.87e-03 Ea 125 kJ/mol n 0.0

$2° \rightarrow 2°$: k 3.87e-03 Ea 103 kJ/mol n 0.0

$2° \rightarrow 1°$: k 3.87e-03 Ea 116 kJ/mol n 0.0

$1° \rightarrow 3°$: k 3.87e-03 Ea 108 kJ/mol n 0.0

$1° \rightarrow 2°$: k 3.87e-03 Ea 114 kJ/mol n 0.0

$1° \rightarrow 1°$: k 3.87e-03 Ea 133 kJ/mol n 0.0

Cyclic species

k 3.87e-03 Ea 103 kJ/mol n 0.0

The initial rate constants and activation energies are taken from [67, 144]. The parameterization incorporates the characteristics of the alkoxides formed before and after hydride transfer. It has been reported that alkoxides with different backbone structures differ in reactivity because of the effects of substitution in the stability of the protonated species formed at the hydride transfer transition state [153]. An increase in hydride transfer rate constant on BEA is reported with increase in the chain length of donor alkane species [153], implying that chain length should be considered in the parameterization. In this work, the parameterization does not consider the chain length as a parameter in order to have a smaller set of parameters for estimation.

5.4   Results and Discussion

5.4.1   Parameter estimation

The parameter estimation involves estimating 29 parameters involving 18 rate constants for beta-scission reactions, one parameter for cyclization and 10 rate constants for hydrogen transfer reactions as listed in the previous section. The experimental datasets used for the parameter estimation include 16 datasets with propene feed at pressure 27 kPa, space velocity 0-2 g-h/mol and temperature 723K on SPP zeolite (Si/Al=75-88), with seven datasets involving propene feed with a mixture of hydrocarbon co-feeds. The best solution found has an objective function value of 27.21. Figure 5.3 shows the model comparison with the 16 experimental datasets of propene feed alone. Figure 5.4 presents parity plots of the various species for all the experimental data used in the parameter estimation. The model comparisons for the experiments involving propene with mixture of hydrocarbons co-feed are added in the Supporting Information.
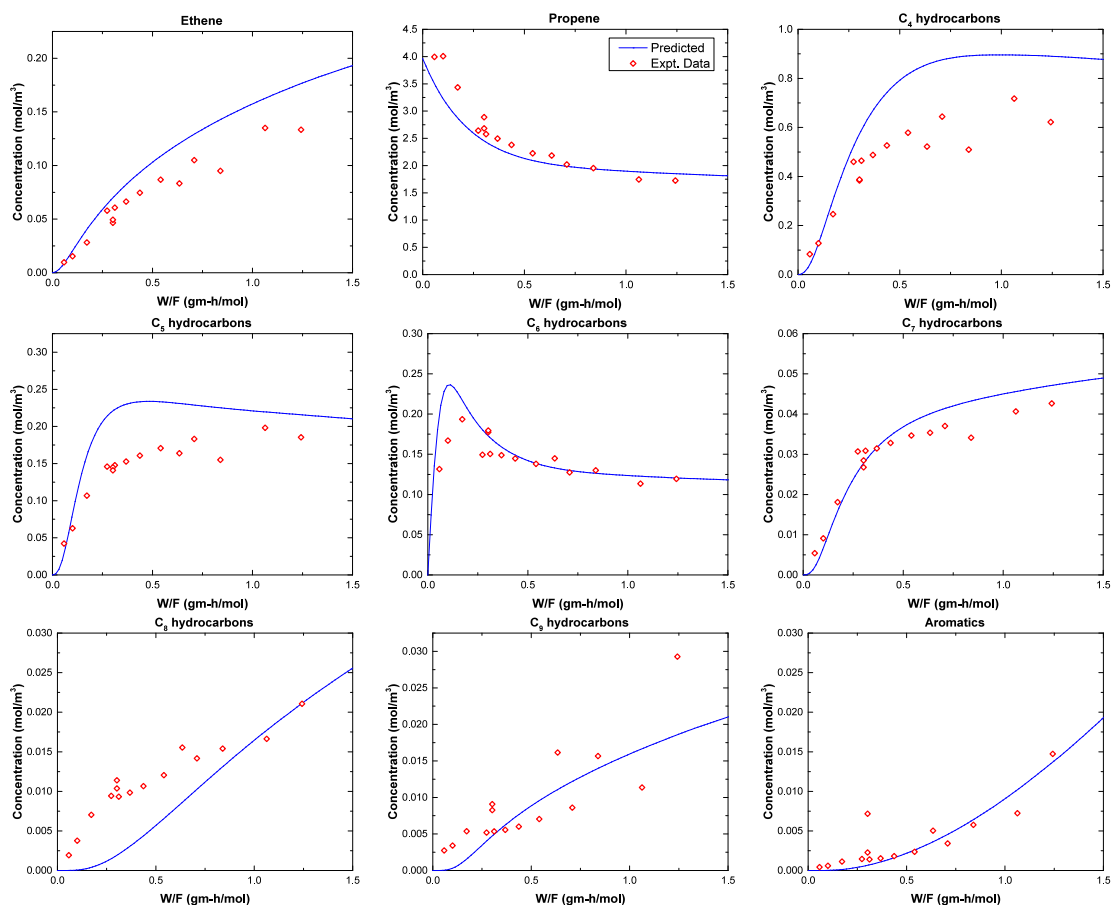
Figure 5.3: Model comparison among various species with experimental data of propene feeds at pressure 27 kPa, space velocity 0-2 g-h/mol and temperature 723K on SPP zeolite (Si/Al=75-88). C_4-C_9 hydrocarbons are represented as a sum of all hydrocarbons of specific carbon number
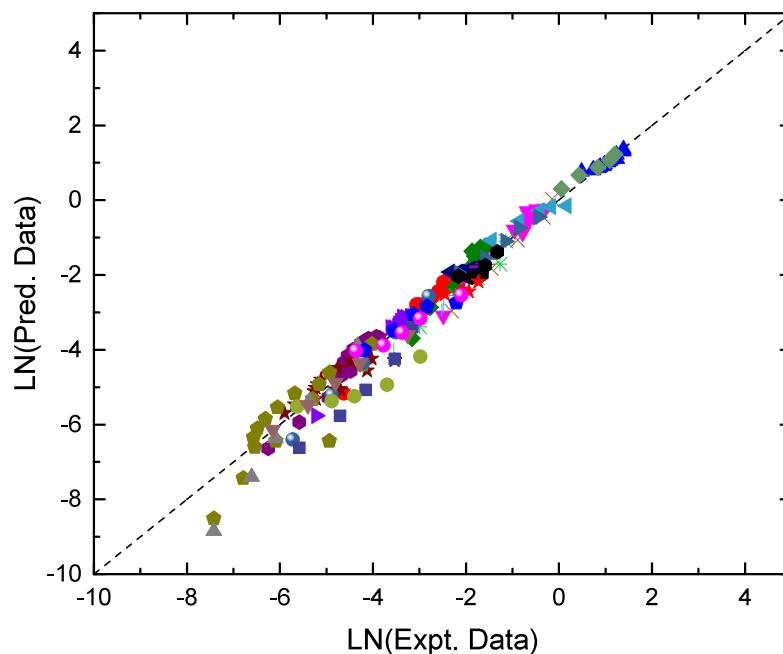
Figure 5.4: Parity plot of the various species for all the experimental datasets used in the parameter estimation.

The estimated parameters are listed in Table 5.3. A comparison of the rate constants with the reported numbers in the literature is shown in Table 5.4. A single-event kinetic model consisting of oligomerization, cracking, isomerization, and adsorption/desoprtion reactions was proposed by [147] when studying 1-Pentene cracking over ZSM-5 catalyst within a temperature range from 633 - 733 K. The estimated rate constants are shown in Table 5.4. Cyclization and hydride transfer reactions were not added in the reaction network based on the product distribution. Further, quaternary carbon atoms were assumed to not react. These assumptions resulted in simplification of the reaction network as well as the number of parameters to be estimated. Their values on comparison with experimental activation energies for olefin cracking in literature [120, 146] showed a difference ($\sim$ 15-50 kJ/mol) which at 723K can result in a difference of $\sim$ 3-4 orders of magnitude. The rate constants estimated in this work lie within this error range.

[154] proposed a lumped kinetic model where the components were grouped into C_2; C_3; C_4; C_5; C_6; C_7^+ and Rest (aromatics and paraffins). The reaction network was lumped down to nine chemical reactions. This molecular lumping simplified the parameter estimation problem, however, the model was shown to be applicable only for abundant

C_3 - C_7 olefins only. [155] also proposed a lumped kinetic model accounting for 10 key oligomerization reactions and estimated a rate constant for each of the reactions. In the above studies, the nature of the surface intermediate and its effect on the reaction rates was not considered. The model comparisons in this work predict profiles for species ranging from C_2 - C_9 along with aromatic species. The characteristics of the surface intermediates are retained in order to estimate rate constants that are dependent on the type of surface intermediates.

Table 5.3: Optimal kinetic parameter values at 723K temperature[1]

Reacton type	Parameters	Predicted Values
Beta-Scission		
Mode A (3°→ 3°)	k_2	6.09E+07
	k_3	5.70E+07
	k_4	2.11E+09
Mode B (2°→ 3°, 3°→ 2°)	k_2	3.42E+06
	k_3	1.2756E+06
	k_4	1.26E+08
Mode C (2°→ 2°)	k_2	695.795
	k_3	163.459
	k_4	1750.24
Mode D (1°→ 2°, 2°→ 1°)	k_2	0.0043
	k_3	7.7378
	k_4	221.932
Mode E (1°→ 3°, 3°→ 1°)	k_2	3662.36
	k_3	449.051
	k_4	2.17E+06
Mode F (1°→ 1°)	k_2	213.859
	k_3	1.80E+05
	k_4	2.78E+07
Cyclization	$k_{cyclization}$	1.38E+08
Hydride Transfer	$k_{3\rightarrow3}$	3.2435
	$k_{3\rightarrow2}$	5.7866
	$k_{3\rightarrow1}$	6.5763
	$k_{2\rightarrow3}$	0.022
	$k_{2\rightarrow2}$	1.4849
	$k_{2\rightarrow1}$	0.5298
	$k_{1\rightarrow3}$	2.802
	$k_{1\rightarrow2}$	0.6638
	$k_{1\rightarrow1}$	21.9279
	k_{Cyclic}	3.2834

[1] Beta-scission are unimolecular reactions, Cyclization and Hydride Transfer are bimolecular reactions. Unimolecular rate constants are in (1/s), bimolecular rate constants are in (1/(atm s)).

Table 5.4: Comparison between estimated kinetic parameters with literature

Reacton type	Predicted Values	Reported Values
Beta-Scission rate constants at 723K (1/s)		
Mode B ($3° \rightarrow 2°$)	4.36E+07	8.86E+05 [1]
Mode C ($2° \rightarrow 2°$)	956.85	74.83 [1]
Mode D ($2° \rightarrow 1°$)	76.55	0.54 [1]
Mode E ($3° \rightarrow 1°$)	2055.71	10.68 [1]
Hydride Transfer rate constants at 723K (mol/H^+ s kPa)		
Hexoxide - Isobutane ($3° \rightarrow 3°$)	0.032	0.013 [2]
Propoxide - Isobutane ($2° \rightarrow 3°$)	0.00022	0.0071 [2]

[1] [147].
[2] [153].

5.4.2   Propene with hydrocarbon mixture co-feeds

Figures 5.5 - 5.9 show the model comparison of experiments involving mixture of hydrocarbons cofeed with propene. It is observed that the C_8 and C_9 aliphatics undergo complete conversion into smaller alkoxides and olefins in the initial 5 % reactor bed length. This is due to the high k_4 rate constants value estimated. The k_4 rate constant values are $\sim$ O(1-3) higher than k_2, k_3 rate constants in their respective beta-scission modes. The higher rate constant value results in high reaction fluxes observed in beta-scission steps for C_8 and C_9 alkoxide species, shown in Figure 5.10 (marked in red). Figure 5.11 shows the surface coverages for C_8 and C_9 alkoxide species for experiment involving hydrocarbon mixture cofeed with propene. The surface is not dominated by the C_8 and C_9 alkoxide species in the initial 5% of the reactor bed length further corroborating the fact that the high coversion in the initial 5% of the reactor bed length is due to the high estimated k_4 rate constant. The surface concentration of both C_8 and C_9 alkoxide species is seen to vary $\sim$ 1-2 orders of magnitude only. High space velocity data is required to predict the profile of C_8 and C_9 aliphatic species and estimate k_4 rate constants of different modes with high certainty.
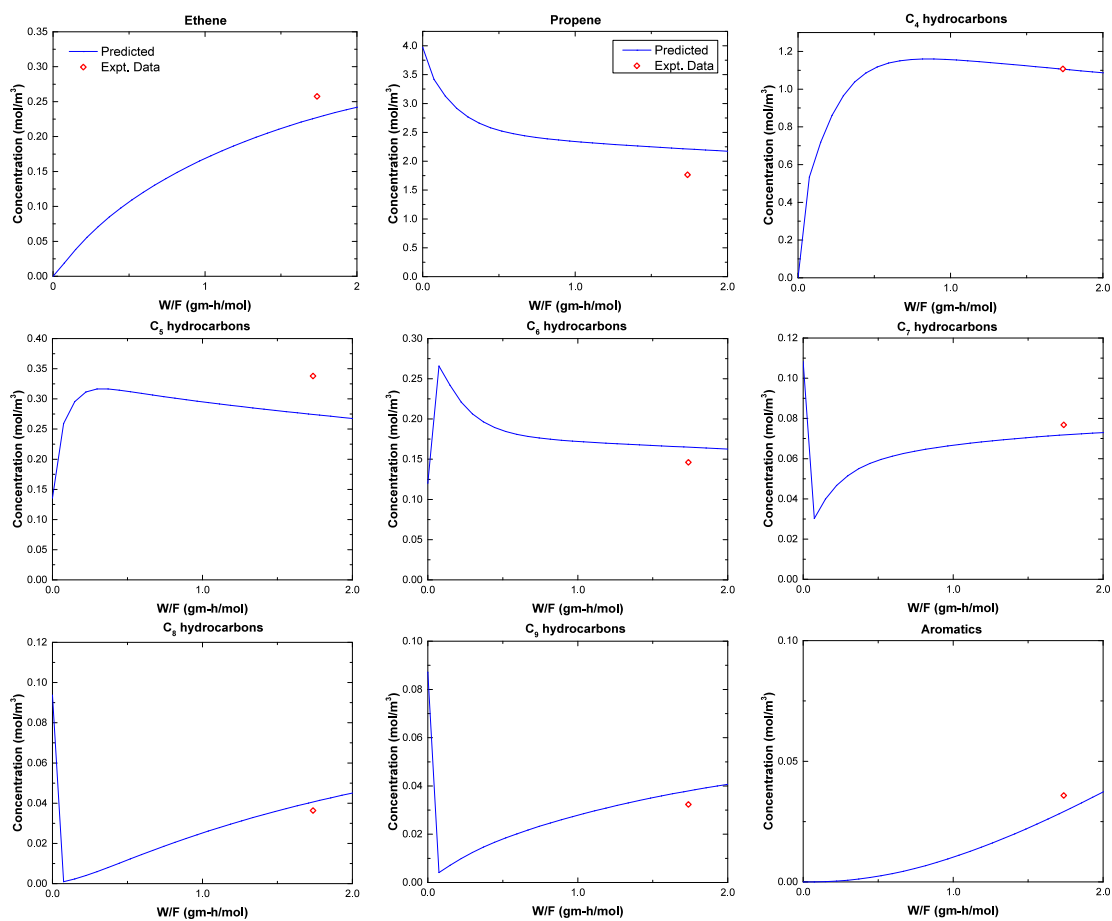
Figure 5.5: Model comparison with experiemental data at 723K among various species with propene and mixture of hydrocarbons cofeed. The mixture of hydrocarbons involve mole fraction of olefinic species C_2:C_3:C_4:C_5:C_6:C_7:C_8:C_9 = (0:0.879:0:0.03:0.027:0.024:0.021:0.019). C_4-C_9 hydrocarbons are represented as a sum of all hydrocarbons of specific carbon number.
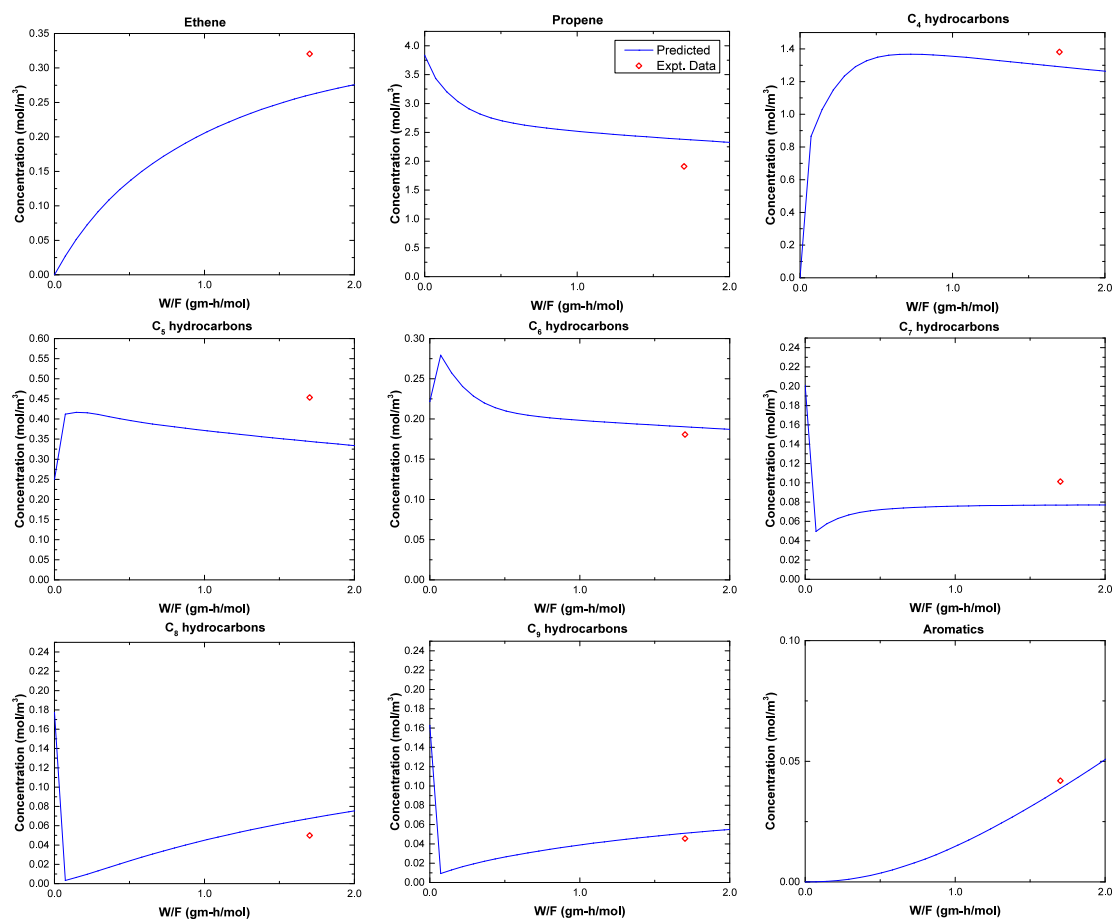
Figure 5.6: Model comparison with experiemental data at 723K among various species with propene and mixture of hydrocarbons cofeed. The mixture of hydrocarbons involve mole fraction of olefinic species C_2:C_3:C_4:C_5:C_6:C_7:C_8:C_9 = (0:0.791:0:0.051:0.046:0.041:0.037:0.034). C_4-C_9 hydrocarbons are represented as a sum of all hydrocarbons of specific carbon number
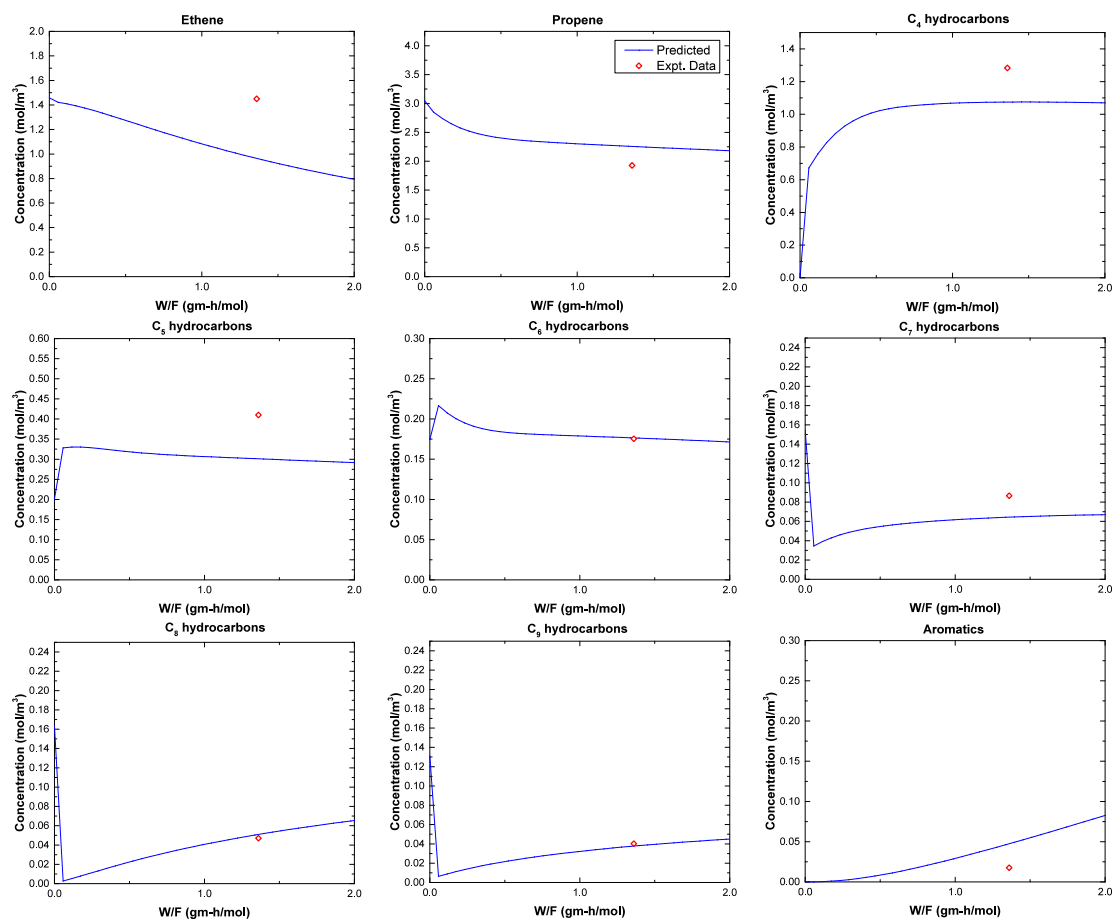
Figure 5.7: Model comparison with experiemental data at 723K among various species with propene and mixture of hydrocarbons cofeed. The mixture of hydrocarbons involve mole fraction of olefinic species C_2:C_3:C_4:C_5:C_6:C_7:C_8:C_9 = (0.274:0.572:0:0.038:0.033:0.029:0.031:0.025). C_4-C_9 hydrocarbons are represented as a sum of all hydrocarbons of specific carbon number
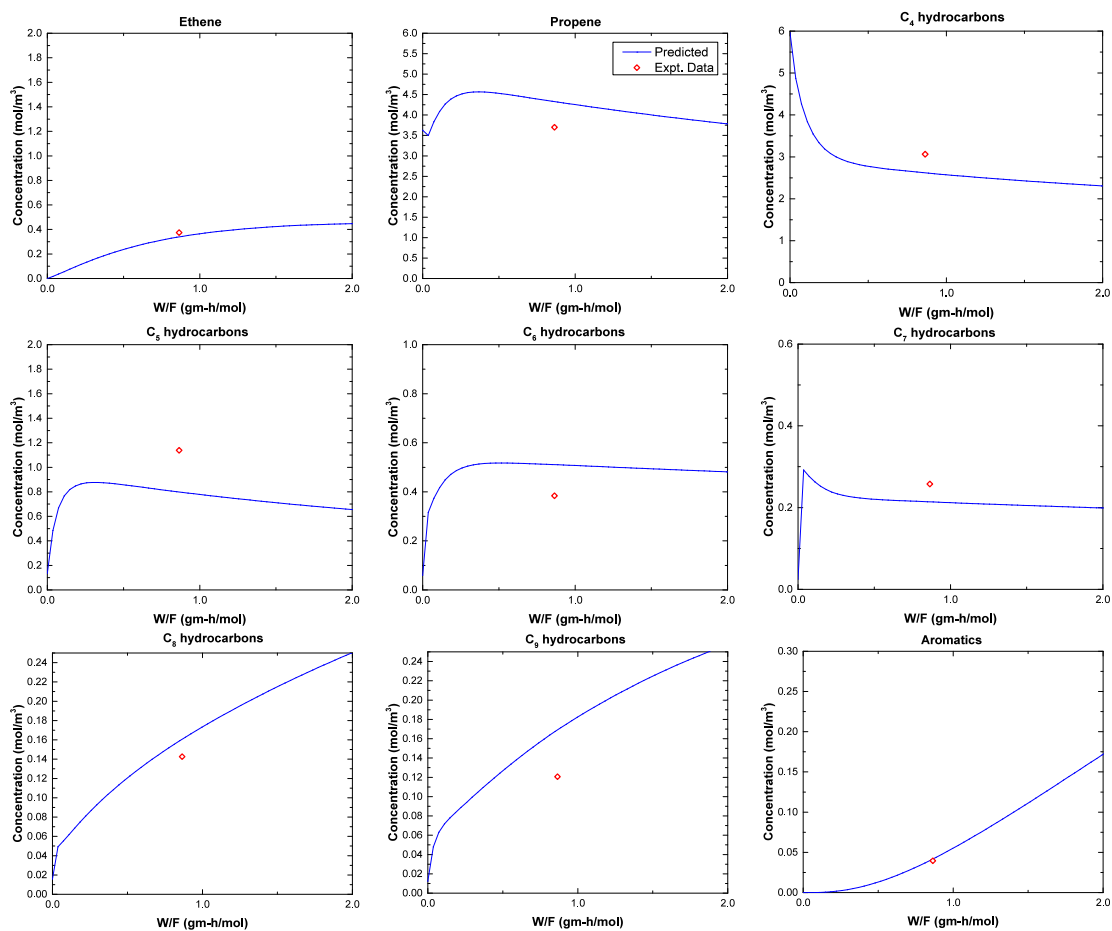
Figure 5.8: Model comparison with experiemental data at 723K among various species with propene and mixture of hydrocarbons cofeed. The mixture of hydrocarbons involve mole fraction of olefinic species C_2:C_3:C_4:C_5:C_6:C_7:C_8:C_9 = (0:0.369:0.606:0.014:0.006:0.003:0.002:0.001). C_4-C_9 hydrocarbons are represented as a sum of all hydrocarbons of specific carbon number
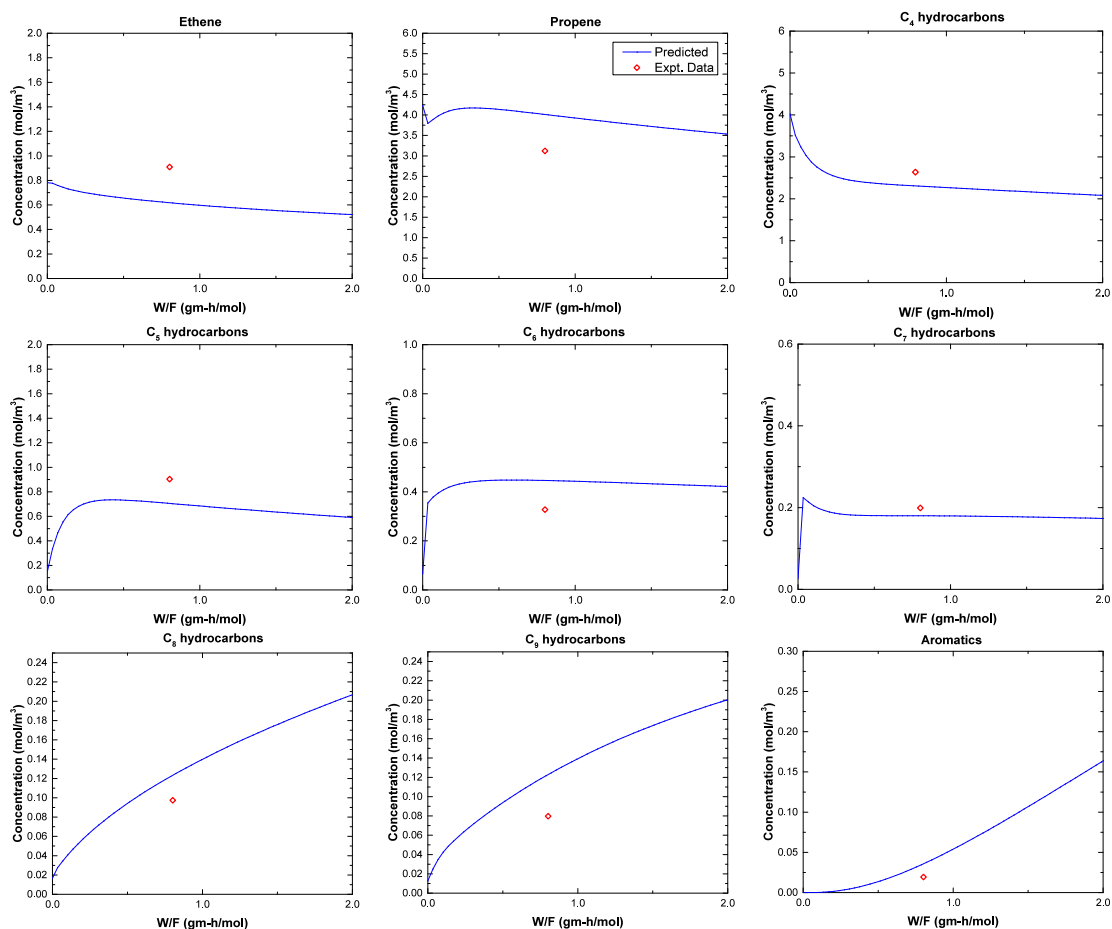
Figure 5.9: Model comparison with experiemental data at 723K among various species with propene and mixture of hydrocarbons cofeed. The mixture of hydrocarbons involve mole fraction of olefinic species C_2:C_3:C_4:C_5:C_6:C_7:C_8:C_9 = (0.084:0.455:0.432:0.016:0.007:0.003:0.002:0.001). C_4-C_9 hydrocarbons are represented as a sum of all hydrocarbons of specific carbon number
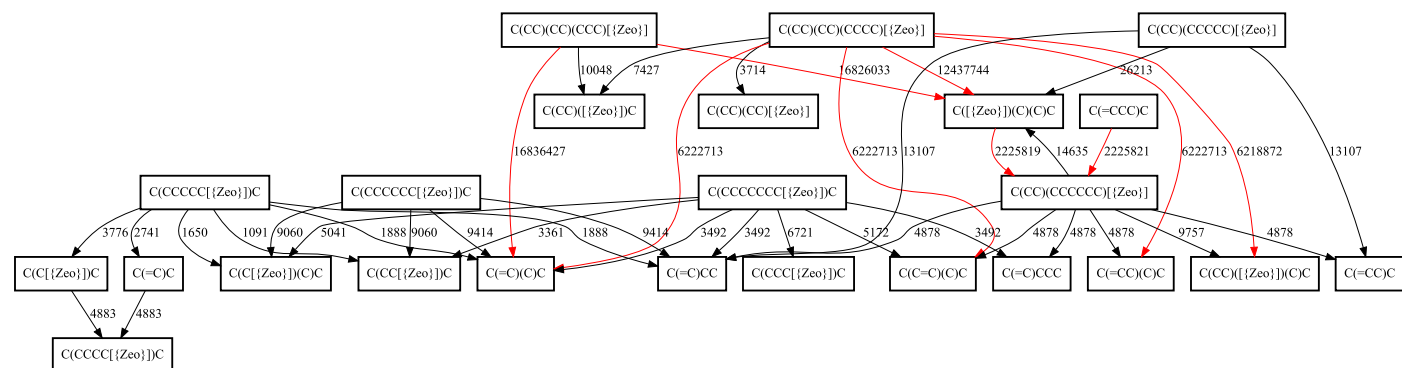
Figure 5.10: Reaction fluxes (mmol/g s) of species (represented using SMILES strings) for experiment involving hydrocarbon mixture cofeed with propene at 1% reactor bed length. Red arrows represent the high reaction fluxes in the beta-scission steps for C_8 and C_9 alkoxide species.

Surface Alkoxides Mole Fraction
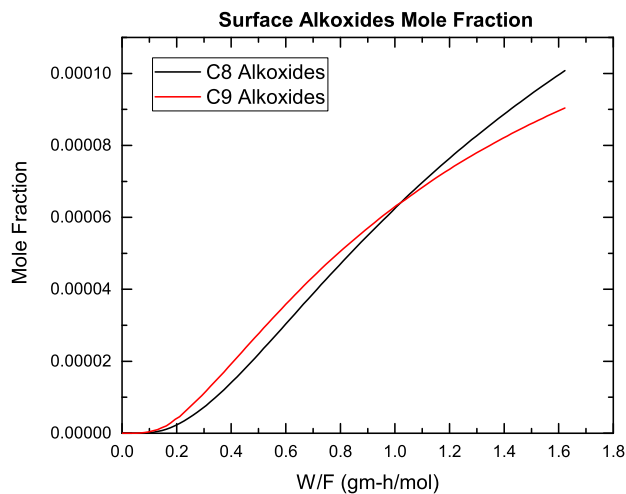
Figure 5.11: Surface coverages for C_8 and C_9 alkoxide species for experiment involving hydrocarbon mixture cofeed with propene.

5.5   Conclusion

This chapter presents a microkinetic model for an olefin interconversion reaction system. Olefin interconversion reaction chemistry is defined through a set of reaction rules that are coded into RING. The reaction network generated is then lumped based on chemical functionality-based lumping while incorporating constraints on the lump representative. The mathematical model generated is parameterized into 29 parameters. Sequential optimization is used to find the optimum set of parameters. The best solution shows good agreement with the experimental datasets with an objective function value of 27.21. Complete conversion of C_8 and C_9 aliphatic species is associated with high values of k_4 estimated within each mode respectively. High space velocity experiments are required to estimate the initial concentration profiles of C_8 and C_9 aliphatic species resulting in estimation of parameter k_4 with high certainty.

Summary and Future

6.1   Summary and Discussion

The main contribution in this thesis is addressing the challenges in developing microkinetic models for complex reaction systems. The two key challenges involving stiff and size are addressed. In Chapter 3, a graph-theoretic framework is developed to generate non-stiff non-linear reduced models. Within this framework, a set of pseudo-species that evolve only in the slow time scale are generated as a linear combination of original species via a cycle identification procedure. A reduced model is formulated using these pseudo-species and algebraic constraints arising from fast/equilibrated reactions. The incorporation of complete conversion or quasi-equilibrium constraints allows a reduction in the number of model parameters. The efficacy of the developed framework is illustrated through application on two chemical systems. The cracking reaction scheme of 1-butene over zeolite acids was studied and an order of magnitude reduction in the number of integration steps was observed by incorporating quasi-equilibrium constraints. Further, the trade-of between the accuracy and the computational complexity of the resulting reduced models for carbon metabolism in erythrocytes system was studied by gradually relaxing the criteria for identifying fast/equilibrated reactions. The developed graph-theoretic framework is an automatic, generic procedure that generates non-stiff reduced models of isothermal reaction systems.

In Chapter 4, the application of RING in the context of biochemical reaction systems is discussed. It is shown that RING can be adopted to model a variety of complex biochemical reaction networks. With the capability of molecule symbolization, the framework can be equably and flexibly be applied for network generation and enumeration of pathways for biochemical reaction networks involving organelle and cellular-level chemistries. These features are demonstrated through three case studies. In the first case study, we generate an exhaustive reaction network for cell metabolism in Escherichia coli. The pathway identification feature in RING generates distinct pathways from Xylose to 2KG, of which one corresponds to a novel pathway recently reported in the literature. In the other case studies, we generate reaction networks for N-glycosylation in mammalian cells using a set of reaction rules reported in the literature. The exhaustiveness and robustness of the reaction network generated is demonstrated through multiple enzyme knockout studies. Path finding was utilized to examine possible routes to synthesize a product glycan. The symbolization of molecule sub-fragments into abstract atoms in the reaction language along with the topological network analysis features incorporated in RING, enable its generic implementation to generate different biochemical reaction networks.

In Chapter 5, a microkinetic model for an olefin interconversion reaction system is developed. Olefin interconversion reaction chemistry is defined through a set of reaction rules that are coded into RING. The reaction network generated is then lumped based on chemical functionality-based lumping while incorporating constraints on the lump representative. The mathematical model generated is parameterized into 29 parameters. Sequential optimization is used to find the optimum set of parameters. The best solution shows good agreement with the experimental datasets with an objective function value of 27.21. Complete conversion of C_8 and C_9 aliphatic species is associated with high values of k_4 estimated within each mode respectively. High space velocity experiments are required to estimate the initial concentration profiles of C_8 and C_9 aliphatic species resulting in estimation of parameter k_4 with high certainty. A schematic representation of this research is shown in Fig. 6.1.
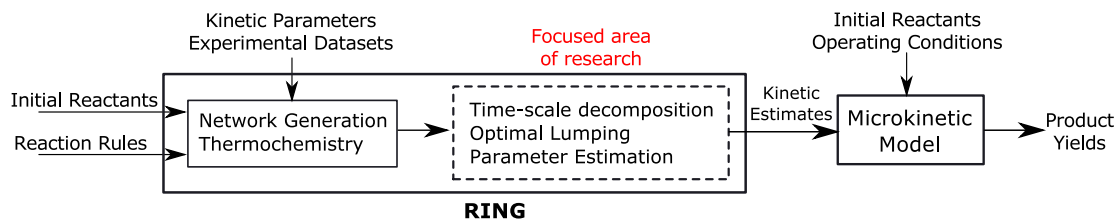
Figure 6.1: Schematic representation of the research

6.2 Future directions

6.2.1 Multi-time scale analysis of complex reaction systems

As noted in Chapter 3, the existing model reduction methods like singular perturbations have been developed to generate reduced models for multi-time scale complex reaction systems. For multi-time scale systems, the singular perturbation theory uses a nested application of two time scale analysis over the multiple time scales [87]. In an analogous way, the steps presented in the manuscript for a two-time scale system can be repeated for multiple time scales. For successive time scales, the equilibrium tolerance or kinetic threshold can be increased or decreased respectively to identify fast reactions for each time scale. The implementation of a framework to address multi-time scale still needs to be worked out.

6.2.2 Microkinetic modeling of olefin interconversion reaction system

As noted in Chapter 5, a set of parameter values are estimated that fit the experimental data at temperature 723K for the olefin interconversion reaction system having 127 species and 7802 reactions. Similar steps can be followed to fit experimental datasets at temperatures at 623 and 673K to estimate a set of activation energies along with a set of kinetic rate constants at a reference temperature, T_{ref}. Preliminary work has been carried in estimating the activation energies, however, due to limited experimental datasets, reasonable activation energies comparable to the published numbers were not estimable.

6.2.3 Microkinetic modeling of methanol-to-hydrocarbons

The conversion of methanol to hydrocarbons (MTH) has been shown to involve aromatic- and olefins-based catalytic cycles involving reactions occurring in a hydrocarbon pool [7]. The reaction system for MTH involves a set of six major chemistries - olefin methylation, olefin cracking, hydrogen transfer, cyclization, aromatic methylation, and aromatic dealkylation – occurring within the system. The olefin catalytic cycle is driven by successive methylation of propylene to form higher hydrocarbons and these higher hydrocarbons can crack to form smaller olefins. The aromatic catalytic cycle is driven through successive methylation of aromatic species to form higher aromatics like hexa-methylbenzene. The two cycles are connected through hydrogen transfer and dealkylation reactions that allow species to switch between the two cycles. MTH production has been researched extensively in the Bhan group. The experimental data generated by Rachit Khare on Self Pillared Pentasil (SPP) Mordenite Framework Inverted (MFI) catalysts can be used for development of parameter estimation module in RING. A complex reaction system like MTH has $\sim 10,000+$ species and $\sim 10^6$ reactions and reaction rate constants in the system vary over 30 orders in magnitude.
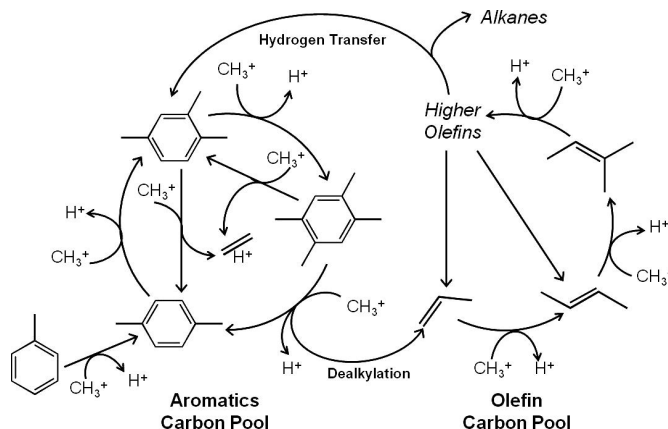
Figure 6.2: Dual Olefin and Aromatic Methylation Catalytic Cycle for Methanol to Hydrocarbons on H-ZSM-5. Adapted from [7]

6.2.4 Optimal lumping schemes in large reaction networks through systematic error incorporation

Developing a kinetic model for a complex reaction system requires knowing the full reaction network apriori. This reaction network is generated through a set of initial reactants and reaction rules. However, as noted in preliminary work, such networks generally involve 10,000+ species and 100,000+ reactions and require large computational times for integrating the resulting DAEs. Assuming a linear scale for extrapolation, a 200 species network (for methanol conversion to hydrocarbons system) requires ~ 1 minute for integration, a 1087 species network requires ~ 5 minutes for integration, therefore, a 10,000+ species network would require ~ 1 hour for just forward simulation. Lumping of species is necessary to keep the parameter estimation computationally tractable. The computational time mentioned above is irrespective of the system and is due to the stiffness and the size of the system.

The concept of lumping involves grouping certain species in the reaction network into few equivalence classes where each class represents an independent entity. The vector-based representation of a structure-oriented lumping (SOL) [156] offers a natural framework for lumping structural isomers that have the same set and number of different functional groups but have a different order or position of these groups in the molecule. However, molecules can only be represented as lumps and it is not always possible to get the structure of the individual molecules that cosntitue the lump from the vector. Combustion and pyrolysis systems follow reduction of the chemical reaction network through rate estimation of different reactions and retaining reactions with rates above a characteristic reaction rate [157, 158, 159]. Loss of information is observed as different composition of the original species can result in the same composition of the representative lump in the lumped system.

A theoretical study on exact lumping of a unimolecular reaction system was presented in [160, 161], formulating a linear transformation of the set of original species to a set of lumped species such that the lumped system described the behavior of the original system and was invariant with different compositions of original species. Lumping of a 10-species model was proposed in [162, 163] where two species were lumped at a time and the overall error introduced in the model was calculated for each pair of species. The lump with the least error was chosen at each level and further lumping was continued. An error tolerance was used as a parameter to identify a 6-species lumping scheme that predicted the behavior of the original 10-species system.

The above model reduction schemes identify the challenge of optimum lumping but do not specify any method for performing lumping for large complex reaction systems. A systematic way of reducing the size of the network through lumping is proposed. In RING, the lumping scheme consists of three steps: (1) identifying molecules with the same number of different types of functional groups and grouping them into one lump (exact lumping), (2) defining a representative molecule to each lump based on users specification for cyclic and acyclic species, and (3) PONA lumping based on molecular formula.
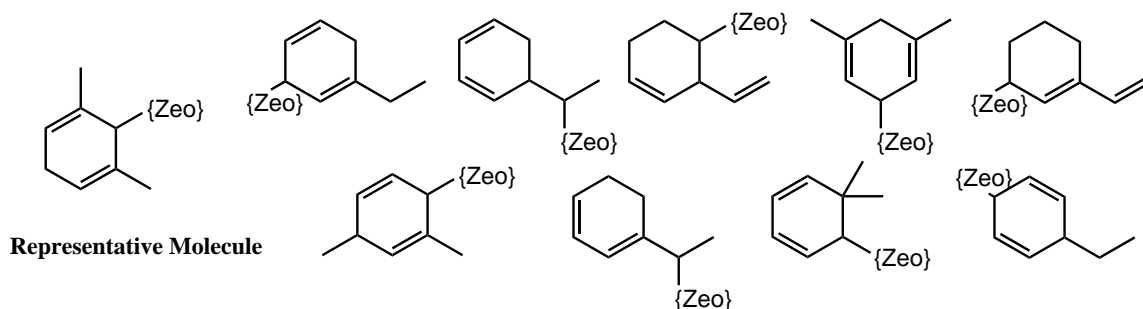
Figure 6.3: A lump representation illustrating the representative molecule

Identification of the representative molecule is determined on the basis of user-defined criteria. An example of a lump is illustrated below. The representative molecule is supposed to have similar thermodynamic and kinetic functionalities as every molecule in the lump. The thermodynamic values like enthalpy, entropy, free energy are calculated on-the-fly using group additivity values while the kinetic values are based on molecular characteristics such as carbon number, primary, secondary, or tertiary, coordination, double bonds etc.

The objective of this task is to help the modeler identify the optimum lumping scheme through systematic error incorporation. The above represented lump contains species with Gibbs free energies varying over a range of -56.28 to -83.09 kJ/mol. Heterogeneous catalysis is a surface driven phenomenon where thermochemistry of the surface species affects the rate of reactions. Lumping species with a wide range of Gibbs free energy causes error in the kinetics of the model. An algorithm for generating optimum lumps is presented below.

The inputs to the algorithm are the graph G (N, E) and a vector $\mathbf{W}$ that contains the Gibbs free energies of the species eligible for lumping. The eligibility criteria are based on characteristics like carbon number; paraffins, olefins, aromatics and naphthenes;

Algorithm 1 EfficientLumpingScheme($\boldsymbol{G}$,$\mathbf{W}$)

Sort $\mathbf{W}$ on basis of decreasing Gibbs free energy (ΔG)

Assume $\alpha = \pm$ 2kJ/mol;

while (error<tolerance δ)

if $\left(\left|\frac{\Delta G_{i+1}-\Delta G_i}{\Delta G_{rep}}\right| \leq \alpha\right)$; $\forall$ i ϵ A,

$A_i \rightarrow A_{i+1}$,

rep. molecule = LumpingScheme(A_i,A_{i+1})

Add rep. molecule to $\mathbf{E}$

Calculate error = exp(-Δ(ΔG)/RT)

α++; //increment α

end while

return $\mathbf{E}$
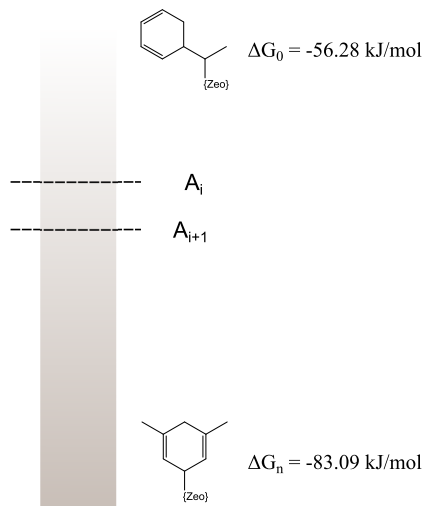
Figure 6.4: Scheme representing range of Gibbs free energy for species eligible for lumping

primary, secondary and tertiary alkoxides. We initiate the lumping process using an initial range of $G_i \pm \alpha$ (kJ/mol) as a criterion for lumping. Lumping species over a range of Gibbs free energy accumulates an error of exp(-Δ(ΔG)/RT) where Δ(ΔG) = 2α. If the error calculated is below a specified tolerance δ, the species lying within the range specified can be lumped. While lumping two species, either (1) the users lumping scheme, or (2) the more stable of the two species, or (3) species observed in the experimental data would be chosen as the representative molecule. The representative molecule is stored in the vector $\mathbf{E}$. At the end, we generate the set of representative molecules and the size of the vector $\mathbf{E}$ governs the size of the reaction system. This process will be carried out for all the species in each group. As we start relaxing the tolerance and allow multiple species to be further lumped, the error in the kinetic formulation increases. There is a trade-off between the size of the kinetic model vs its ability to predict the kinetics accurately. Through this task, we identify an optimal lumping scheme based on the size of the system and error incorporated in kinetic and thermodynamic values. Lumping rules specified by the user deviating from the above optimal lumping scheme will be identified and reported to the user as feedback.

6.2.5   Model-based design of experiments

In complex reaction systems, there is a possibility of multiple mechanisms (or reaction routes) existing between reactants and products. A major challenge in parameter estimation could arise as a result – multiple sets of kinetic parameters could lead to similar predictions or multiple models can be proposed for a reaction system. Further, the confidence interval of certain parameter estimates may be unacceptably large because experimental data did not cover regions of the parameter space most sensitive to those parameters. Additional experimental data will, therefore, be required for performing model discrimination and improving accuracy. To this end, RING can be additionally equipped to do model-based experimental design by pursuing two approaches. First, each of the multiple models can be solved, at different operating conditions (concentrations, flow rates, temperature, space velocity, etc.) and identify where and how the predictions of the different models diverge. This will help identify new experimental conditions at which additional data can be obtained. Second, for those parameters that have a large confidence interval, state-of-the-art experimental design methods such as the A and D optimality criteria [164] can be used to pinpoint the "best" operating conditions at which new experiments must be conducted. The additional data in both cases will be used for improving the estimates of the kinetic parameters. Thus, a strong feedback between experimentation and computations can be established.

Bibliography

[1] S. Rangarajan, A. Bhan, and P. Daoutidis. Rule-based generation of thermochemical routes to biomass conversion. *Ind. Eng. Chem. Res.*, 49(21):10459–10470, 2010.

[2] S. Rangarajan, A. Bhan, and P. Daoutidis. Language-oriented rule-based reaction network generation and analysis: Description of RING. *Computers & Chemical Engineering*, 45:114–123, 2012.

[3] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. Sundials: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software*, 31(3):363–396, 2005.

[4] P. Hossler, L.T. Goh, and W.-S. Lee, M. M.and Hu. Glycovis: visualizing glycan distribution in the protein N-glycosylation pathway in mammalian cells. *Biotechnol Bioeng*, 95:946–960, 2006.

[5] S. Rangarajan, A. Bhan, and P. Daoutidis. Language-oriented rule-based reaction network generation and analysis: Description of RING. *Computers and Chemical Engineering*, 45:114–123, 2012.

[6] Z. P. Gerdtzen, P. Daoutidis, and W. S. Hu. Non-linear reduction for kinetic models of metabolic reaction networks. *Metabolic Engineering*, 6:140–154, 2004.

[7] S. Ilias and A. Bhan. The mechanism of the catalytic converion of methanol-to-hydrocarbons. *ACS Catal.*, 3:18 – 31, 2013.

[8] R. Vinu and Linda J. Broadbelt. Unraveling reaction pathways and specifying reaction kinetics for complex systems. *Annual Review of Chemical and Biomolecular Engineering*, 3(1):29–54, 2012.

[9] Linda J. Broadbelt and Jim Pfaendtner. Lexicography of kinetic modeling of complex reaction networks. *AIChE Journal*, 51(8):2112–2121, 2005.

[10] G. P. Froment, B. O. Van de Steene, P. S. Van Damme, S. Narayanan, and A. G. Goossens. Thermal cracking of ethane and ethane-propane mixtures. *Industrial & Engineering Chemistry Process Design and Development*, 15(4):495–504, 1976.

[11] D. Mohan, C. U. Pittman Jr., and P. H. Steele. Pyrolysis of wood/biomass for bio-oil: A critical review. *Energy & Fuels*, 20(3):848–889, 2006.

[12] G. Yaluris, R. J. Madon, and J. A. Dumesic. 2-methylhexane cracking on y zeolites: Catalytic cycles and reaction selectivity. *Journal of Catalysis*, 165:205–220, 1997.

[13] A. Corma, G. W. Huber, L. Sauvanaud, and P. O'Connor. Biomass to chemicals: Catalytic conversion of glycerol/water mixtures into acrolein, reaction network. *Journal of Catalysis*, 257(1):163–171, 2008.

[14] Charles K. Westbrook, William J. Pitz, Olivier Herbinet, Henry J. Curran, and Emma J. Silke. A comprehensive detailed chemical kinetic reaction mechanism for combustion of n-alkane hydrocarbons from n-octane to n-hexadecane. *Combustion and Flame*, 156(1):181 – 199, 2009.

[15] H-W Wong, X Li, M. T. Swihart, and L. J. Broadbelt. Detailed kinetic modeling of silicon nanoparticle formation chemistry via automated mechanism generation. *The Journal of Physical Chemistry A*, 108(46):10122 – 10132, 2004.

[16] Shumaila S. Khan, Qizhi Zhang, and Linda J. Broadbelt. Automated mechanism generation. part 1: mechanism development and rate constant estimation for voc chemistry in the atmosphere. *Journal of Atmospheric Chemistry*, 63(2):125–156, 2009.

[17] A. N. Mayeno, R. S. H. Yang, and B. Reisfeld. Biochemical reaction network modeling: predicting metabolism of organic chemical mixtures. *Environmental Science and Technology*, 39:5363 – 5371, 2005.

[18] M. Rizzi, M. Baltes, U. Theobald, and M. Reuss. In vivo analysis of metabolic dynamics in saccharomyces cerevisiae: Ii. mathematical model. *Biotechnology and Bioengineering*, 55:592–608, 1997.

[19] J. L. Reed, T. D. Vo, C. H. Schilling, and B. O. Palsson. An expanded genome-scale model of escheria coli K-12 (iJR904 GSM/ GPR). *Genome Biology*, 4(9):R54, 2003.

[20] Prodromos Daoutidis, W. Alex Marvin, Srinivas Rangarajan, and Ana I. Torres. Engineering biomass conversion processes: A systems perspective. *AIChE Journal*, 59(1):3–18, 2013.

[21] Michael E. Jenkin, Sandra M. Saunders, and Michael J. Pilling. The tropospheric degradation of volatile organic compounds: a protocol for mechanism development. *Atmospheric Environment*, 31(1):81 – 104, 1997.

[22] R. J. Quann and S. B. Jaffe. Building useful models of complex reaction systems in petroleum refining. *Chemical Engineering Science*, 51(10):1615 – 1631, 1996.

[23] T. C. Ho. Kinetic modeling of large-scale reaction systems. *Catalysis Reviews*, 50(3):287–378, 2008.

[24] William H. Green Jr. Predictive kinetics: A new approach for the 21st century. *Advances in Chemical Engineering*, 24(32):1–50, 2007.

[25] Mavrovouniotis M.L. Okino, M. S. Simplification of mathematical models of chemical reaction systems. *Chemical Reviews*, 98(2):1–50, 1998.

[26] L. Wang, S. Dash, C. Y. Ng, and C. D. Maranas. A review of computational tools for design and reconstruction of metabolic pathways. *Synthetic and Systems Biotechnology*, 2:243 – 252, 2017.

[27] J. W. Lee, D. Na, J. M. Park, J. Lee, S. Choi, and S. Y. Lee. Systems metabolic engineering of microorganisms for natural and non-natural chemicals. *Nature chemical biology.*, 8:536 – 546, 2012.

[28] J. Nielsen and J. D. Keasling. Engineering cellular metabolism. *Cell.*, 164:1185 – 1197, 2016.

[29] B. M. Woolston, S. Edgar, and G. Stephanopoulos. Metabolic engineering: past and future. *Annual review of chemical and biomolecular engineering.*, 4:259 – 288, 2013.

[30] N. Hadadi and V. Hatzimanikatis. Design of computational retrobiosynthesis tools for the design of de novo synthetic pathways. *Current opinion in chemical biology.*, 28:99 – 104, 2015.

[31] M. R. Long, W. K. Ong, and J. L. Reed. Computational methods in metabolic engineering for strain design. *Current opinion in biotechnology.*, 34:135 – 141, 2015.

[32] M. H. Medema, R. van Raaphorst, E. Takano, and R. Breitling. Computational tools for the synthetic design of biochemical pathways. *Nature reviews. Microbiology.*, 10:191 – 202, 2012.

[33] A.S. Tomlin and T. Turanyi. *Piling, M. J., Low-Temperature Combustion and Auto-Ignition*, volume 35. Elsevier, Amsterdam, 1997. Chapter 4.

[34] L. J. Broadbelt, S. M. Stark, and M. T. Klein. Computer-generated pyrolysis modeling - on the fly generation of species, reactions and rates. *Industrial & Engineering Chemistry Research*, 33(4):790–799, 1994.

[35] S. E. Prickett and M. L. Mavrovouniotis. Construction of complex reaction systems .1. Reaction description language. *Computers & Chemical Engineering*, 21(11):1219–1235, 1997.

[36] A. Ratkiewicz and T. N. Truong. Application of chemical graph theory for automated mechanism generation. *Journal of Chemical Information and Modeling*, 43:36 – 44, 2003.

[37] J. Dugundji and I. Ugi. An algebraic model of constitutional chemistry as a basis for chemical computer programs. *Springer Berlin Heidelberg*, pages 19 – 64, 1973.

[38] S. D. Finley, L. J. Broadbelt, and V. Hatzimanikatis. Computational framework for predictive biodegradation. *Biotechnology and Bioengineering*, 6(104):1086–1097, 2009.

[39] C. S. Henry, L. J. Broadbelt, and V. Hatzimanikatis. Thermodynamics-based metabolic flux analysis. *Biophysical Journal*, 5(92):1792–1805, 2007.

[40] C. S. Henry, L. J. Broadbelt, and V. Hatzimanikatis. Discovery and analysis of novel metabolic pathways for the biosynthesis of industrial chemicals: 3-hydroxypropanoate. *Biotechnology and Bioengineering*, 3(106):462–473, 2010.

[41] C. Li, Henry C. S., M. D. Jankowski, J. A. Ionita, V. Hatzimanikatis, and L. J. Broadbelt. Computational discovery of biochemical routes to specialty chemicals. *Chemical Engineering Science*, 59:5051–5060, 2004.

[42] Miguel A. Baltanas and Gilbert F. Froment. Computer generation of reaction networks and calculation of product distributions in the hydroisomerization and hydrocracking of paraffins on pt-containing bifunctional catalysts. *Computers & Chemical Engineering*, 9(1):71 – 81, 1985.

[43] S.H. Hsu, B. Krishnamurthy, P. Rao, C. H. Zhao, S. Jagannathan, and V. Venkatasubramanian. A domain-specific compiler theory based framework for automated reaction network generation. *Computers & Chemical Engineering*, 32(10):2455–2470, 2008.

[44] Jean-Loup Faulon and Allen G. Sault. Stochastic generator of chemical structure. 3. reaction network generation. *Journal of Chemical Information and Computer Sciences*, 41(4):894–908, 2001, http://pubs.acs.org/doi/pdf/10.1021/ci000029m. PMID: 11500106.

[45] R. G. Susnow, A. M. Dean, W. H. Green, P. Peczak, and L. J. Broadbelt. Rate-Based Construction of Kinetic Models for Complex Systems. *The Journal of Physical Chemistry A*, 101(20):3731–3740, 1997.

[46] L. J. Broadbelt, S. M. Stark, and M. T. Klein. Termination of computer-generated reaction mechanisms: Species rank-based convergence criterion. *Industrial & Engineering Chemistry Research*, 34(8):2566 – 2573, 1995.

[47] K.M. Van Geem, M-F. Reyniers, G.B. Marin, J. Song, W.H. Green, and D. M. Matheu. Automatic reaction network generation using RMG for steam cracking of n-hexane. *AIChE Journal*, 52(2):718–730, 2006.

[48] James R. Faeder, William S. Hlavacek, Ilona Reischl, Michael L. Blinov, Henry Metzger, Antonio Redondo, Carla Wofsy, and Byron Goldstein. Investigation of early events in fceri-mediated signaling using a detailed mathematical model. *The Journal of Immunology*, 170(7):3769–3781, 2003, http://www.jimmunol.org/content/170/7/3769.full.pdf+html.

[49] M. Arita. The metabolic world of escherichia coli is not small. *Proceedings of the National Academy of Sciences*, 101(6):1543 – 1547, 2004.

[50] J. Gonzalez-Lergier, L. J. Broadbelt, and V. Hatzimanikatis. Theoretical considerations and computational analysis of the complexity in polyketide synthesis pathways. *Journal of the American Chemical Society*, 127(27):9930 – 9938, 2005.

[51] L.T. Fan, B. Bertok, and F. Friedler. A graph-theoretic method to identify candidate mechanisms for deriving the rate law of a catalytic reaction. *Computers and Chemistry*, 26:265 – 292, 2002.

[52] Ilie Fishtik, Caitlin A. Callaghan, and Ravindra Datta. Reaction route graphs. i. theory and algorithm. *The Journal of Physical Chemistry B*, 108(18):5671–5682, 2004, http://pubs.acs.org/doi/pdf/10.1021/jp0374004.

[53] Y-C. Lin, L. T. Fan, S. Shafie, B. Bertok, and F. Friedler. Generation of light hydrocarbons through Fischer-Tropsch synthesis: Identification of potentially dominant catalytic pathways via the graph-theoretic method and energetic analysis. *Computers and Chemical Engineering*, 33:1182 – 1186, 2009.

[54] A. Kummel, S. Panke, and M. Heinemann. Putative regulatory sites unraveled by network-embedded thermodynamic analysis of metabolome data. *Molecular Systems Biology*, 2:0034, 2006.

[55] James J. P. Stewart. MOPAC-2009, stewart computational chemistry. `OpenMOPAC.net(2008)`, accessed Dec 2010.

[56] L. J. Broadbelt, S. M. Stark, and M. T. Klein. Computer generated reaction networks: on-the-fly calculation of species properties using computational quantum chemistry. *Chemical Engineering Science*, 49(24(2)):4991 – 5010, 1994.

[57] L. J. Broadbelt, S. M. Stark, and M. T. Klein. Computer generated reaction modelling: Decomposition and encoding algorithms for determining species uniqueness. *Computers & Chemical Engineering*, 20(2):113 – 129, 1996.

[58] S. J. Chinnick, D. L. Baulch, and P. B. Asyscough. An expert system for hydrocarbon pyrolysis reactions. *Chemometrics and Intelligent laboratory Systems*, 5:39–52, 1988.

[59] B. Heyberger, F. Battin-Leclerc, V. Warth, R. Fournet, G. M. Come, and G. Scacchi. Comprehensive mechanism for the gas-phase oxidation of propene. *Combustion and Flame*, 126:1780–1802, 2001.

[60] V. Warth, F. Battin-Leclerc, R. Fournet, P.A. Glaude, G.M. Come, and G. Scacchi. Computer based generation of reaction mechanisms for gas-phase oxidation. *Computers and chemistry*, 24:541–560, 2000.

[61] E. S. Blurock. Reaction system for modeling chemical reactions. *Journal of Chemical Information and Computer Science*, 35:607 – 616, 1994.

[62] W.H. Green, B. Bhattacharjee, O. Oluwole, J. Song, R. Sumathi, C. D. Wijaya, Wong H-W., P. E. Yelvington, and J. Yu. New methods for predictive chemical kinetics. *Prepr. Pap.-Am. Chem. Soc. Div. Fuel Chem.*, 49(1):323, 2004.

[63] J. Song. Massachusetts Institute of Technology, 2004. PhD. Dissertation.

[64] S. E. Prickett and M. L. Mavrovouniotis. Construction of complex reaction systems .2 Molecule manipulation and reaction application algorithms. *Computers & Chemical Engineering*, 21(11):1237 – 1254, 1997.

[65] S. E. Prickett and M. L. Mavrovouniotis. Construction of complex reaction systems.3. An example: alkylation of olefins. *Computers & Chemical Engineering*, 21(12):1325 – 1337, 1997.

[66] S. Katare, A. Bhan, J. M. Caruthers, W. N. Delgass, and V. Venkatasubramanian. A hybrid genetic algorithm for efficient parameter estimation of large kinetic models. *Computers & Chemical Engineering*, 28(12):2569–2581, 2004.

[67] A. Bhan, S.-H. Hsu, V. Venkatasubramanian, J. M. Caruthers, and W. N. Delgass. Microkinetic modeling of propane aromatization over HZSM-5. *Journal of Catalysis*, 235:35, 2005.

[68] J.M. Caruthers, J.A. Lauterbach, K.T. Thomson, V. Venkatasubramanian, C.M. Snively, A. Bhan, S. Katare, and G. Oskarsdottir. Catalyst design: knowledge extraction from high-throughput experimentation. *Journal of Catalysis*, 216(1-2):98–109, MAY 15 2003.

[69] F. P. Di Maio and P. G. Lignola. KING, a kinetic network generator. *Chemical Engineering Science*, 47((9-11)):2713 – 2718, 1992.

[70] M. L. Blinov, J. Yang, J. R. Faeder, and W. S. Hlavacek. Graph theory for rule-based modeling of biochemical networks. *Transactions on Computational Systems Biology VII, Lecture notes in Computer Sceince*, 4230:89 – 106, 2006.

[71] J. R. Faeder, M. L. Blinov, B. Goldstein, and W. S. Hlavacek. Rule-based modeling of biochemical networks. *Complexity*, 10:22–41, 2005.

[72] M.L. Blinov, J. R. Faeder, J. Yang, B. Goldstein, and W. S. Hlavacek. 'On-the-fly' or 'generate-first' modeling? *Nature Biotechnology*, 23(11):1344 – 1345, 2005.

[73] C. S. Henry, L. J. Broadbelt, and V. Hatzimanikatis. Discovery and analysis of novel metabolic pathways for the biosynthesis of industrial chemicals: 3-hydroxypropanoate. *Biotechnology and Bioengineering*, 106(3):462 – 473, 2010.

[74] A. D. Hill, J. R. Tomshine, E. M. B. Weeding, V. Sotiropoulos, and Y. N. Kaznessis. Synbioss: The synthetic biology modeling suite. *Bioinformatics*, 24(51):2551–2553, 2008.

[75] Arthur N. Mayeno, Raymond S. H. Yang, and Brad Reisfeld. Biochemical reaction network modeling:? predicting metabolism of organic chemical mixtures. *Environmental Science & Technology*, 39(14):5363–5371, 2005, http://pubs.acs.org/doi/pdf/10.1021/es0479991. PMID: 16086453.

[76] G. Rodrigo, J. Carrera, K. J. Prather, and A. Jaramillo. Desharky: automatic design of metabolic pathways for optimal cell growth. *Bioinformatics.*, 24:2554 – 2556, 2008.

[77] K. L. Prather and C. H. Martin. De novo biosynthetic pathways: rational design of microbial chemical factories. *Current opinion in biotechnology.*, 19:468 – 474, 2008.

[78] S. H. Lam and D. A. Goussis. Understanding complex chemical kinetics with computational singular perturbation. *Proc. Combus. Inst.*, 22(1):931–941, 1989.

[79] S. H. Lam and D. A. Goussis. The CSP method for simplifying kinetics. *International Journal of Chemical Kinetics*, 26(4):461–486, 1994.

[80] A. Massias, D. Diamantis, E. Mastorakos, and D. A. Goussis. An algorithm for the construction of global reduced mechanisms with CSP data. *Combustion and Flame*, 117(4):685–708, 1999.

[81] N. Fenichel. Geometric singular perturbation theory for ordinary differential equations. *Journal of Differential Equations*, 31(1):53–98, 1979.

[82] U. Maas and S. B. Pope. Simplifying chemical kinetics: Intrinsic low-dimensional manifolds in composition space. *Combustion and Flame*, 88(3-4):239–264, 1992.

[83] S. B. Pope. Computationally efficient implementation of combustion chemistry using in situ adaptive tabulation. *Combustion Theory and Modelling*, 1(1):41–63, 1997.

[84] B. Yang and S. B. Pope. Treating chemistry in combustion with detailed mechanismsIn situ adaptive tabulation in principal directionsPremixed combustion. *Combustion and Flame*, 112(1-2):85–112, 1998.

[85] S. J. Fraser. The steady state and equilibrium approximations: A geometrical picture. *The Journal of Chemical Physics*, 88(8):4732–4738, 1988.

[86] M. R. Roussel and S. J Fraser. Accurate steady-state approximations: implications for kinetics experiments and mechanism. *The Journal of Physical Chemistry*, 95(22):8762–8770, 1991.

[87] N. Vora and P. Daoutidis. Nonlinear model reduction of chemical reaction systems. *AIChE Journal*, 47(10):2320–2332, 2001.

[88] R. A. Adomaitis. Dynamic order reduction of thin-film deposition kinetics models: A reaction factorization approach. *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films*, 34(1), 2016.

[89] E. M. Remmers, C. D. Travis, and R. A. Adomaitis. Reaction factorization for the dynamic analysis of atomic layer deposition kinetics. *Chemical Engineering Science*, 127:374–391, 2015.

[90] C. H. Lee and H. G. Othmer. A multi-time-scale analysis of chemical reaction networks: I. Deterministic systems. *Journal of Mathematical Biology*, 60(3):387–450, 2010.

[91] T. P. Prescott and A. Papachristodoulou. Layered decomposition for the model order reduction of timescale separated biochemical reaction networks. *Journal of Theoretical Biology*, 356:113–122, 2014.

[92] D. Lebiedz. Computing minimal entropy production trajectories: An approach to model reduction in chemical kinetics. *The Journal of Chemical Physics*, 120(15):6890, 2004.

[93] T. Lu and C. K. Law. On the applicability of directed relation graphs to the reduction of reaction mechanisms. *Combustion and Flame*, 146(3):472–483, 2006.

[94] T. Lu, C. K. Law, C. S. Yoo, and J. H. Chen. Dynamic stiffness removal for direct numerical simulations. *Combustion and Flame*, 156(8):1542–1551, 2009.

[95] P. Pepiot-Desjardins and H. Pitsch. An efficient error-propagation-based reduction method for large chemical kinetic mechanisms. *Combustion and Flame*, 154(1-2):67–81, 2008.

[96] M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The KEGG resource for deciphering the genome. *Nucleic Acids Res.*, 32:D277–D280, 2004.

[97] L. B. Ellis, D. Roe, and L. P. Wackett. The university of minnesota biocatalysis/biodegradation database: the first decade. *Nucleic acids research*, 34:D517–521, 2006.

[98] L. B. Ellis and L. P. Wackett. Use of the university of minnesota biocatalysis/biodegradation database for study of microbial degradation. *Microbial Informatics and Experimentation.*, 2:1, 2012.

[99] F. J. Krambeck and M. J. Betenbaugh. A mathematical model of N-linked glycosylation. *Biotechnol Bioeng*, 92:711–728, 2005.

[100] G. Liu and S. Neelamegham. A computational framework for the automated construction of glycosylation reaction networks. *Plos One.*, 9:1 – 10, 2014.

[101] A. G. McDonald, K. F. Tipton, and G. P. Davey. A knowledge-based system for display and prediction of o-glycosylation network behaviour in response to enzyme knockouts. *Plos Comput Biol.*, 12:1 – 10, 2016.

[102] S. Rangarajan, T. Kaminski, E. V. Wyk, A. Bhan, and P. Daoutidis. Language-oriented rule-based reaction network generation and analysis: Algorithms of RING. *Computers and Chemical Engineering*, 64:124–137, 2014.

[103] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. Sundials: Suite of nonlinear and differential/algebraic equation solvers. *ACM Trans. Math. Softw.*, 31(3):363 – 396, 2005.

[104] R. Faber, P. Li, and G. Wozny. Sequential parameter estimation for large-scale systems with multiple data sets. 1. computational framework. *Ind. Eng. Chem. Res.*, 42(23):5850 – 5860, 2003.

[105] A. Wachter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25 – 57, 2006.

[106] W.R. Esposito and C.A. Floudas. Global optimization for the parameter estimation of differential-algebraic systems. *Ind. Eng. Chem. Res.*, 39:1291 – 1310, 2000.

[107] M. Stein. Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143 – 151, 1987.

[108] K. A. P. McLean and K. B. McAuley. Mathematical modelling of chemical processes obtaining the best model predictions and parameter estimates using identifiability and estimability procedures. *Can. J. Chem. Eng*, 90:351 – 366, 2011.

[109] S. Wu, K. A. P. McLean, T. J. H. Harris, and K. B. McAuley. Selection of optimal parameter set using estimability analysis and MSE-based model-selection criterion. *Int. J. Adv. Mechatronic Syst.*, 3:188 – 197, 2011.

[110] C. Sun and J. Hahn. Parameter reduction for stable dynamical systems based on hankel singular values and sensitivity analysis. *Chemical Engineering Science*, 61:5393 – 5403, 2006.

[111] K. Z. Yao, B. M. Shaw, B. Kou, K. B. McAuley, and D. W. Bacon. Modeling ethylene/butene copolymerization with multi-site catalysts: parameter estimability analysis and experimental design. *Polym. React. Eng.*, 11:563 – 588, 2003.

[112] J. R. Bowen, A. Acrivos, and A. K. Oppenheim. Singular perturbation refinement to quasi-steady state approximation in chemical kinetics. *Chemical Engineering Science*, 18(3):177–188, 1963.

[113] M. Mincheva and M. R. Roussel. Graph-theoretic methods for the analysis of chemical and biochemical networks. I. Multistability and oscillations in ordinary differential equation models. *Journal of Mathematical Biology*, 55(1):61–86, 2007.

[114] P. Holme, M. Huss, and H. Jeong. Subnetwork hierarchies of biochemical pathways. *Bioinformatics*, 19(4):532–538, 2003.

[115] M. Domijan and M. Kirkilionis. Graph theory and qualitative analysis of reaction networks. *Networks and Heterogeneous Media*, 3(2):295–322, 2008.

[116] R. Heinrich, S. M. Rapoport, and T. A. Rapoport. Metabolic regulation and mathematical models. *Progress in Biophysics and Molecular Biology*, 32:1–82, 1978.

[117] T. H Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms, Second Edition*, volume 7. The MIT Press, England, 2001.

[118] R. Tarjan. Enumeration of the Elementary Circuits of a Directed Graph. *SIAM Journal on Computing*, 2(3):211–216, 1973.

[119] A. Kumar, P. D. Christofides, and P. Daoutidis. Singular perturbation modeling of nonlinear processes with nonexplicit time-scale multiplicity. *Chemical Engineering Science*, 53(8):1491–1504, 1998.

[120] C.-J. Chen, S. Rangarajan, I. M. Hill, and A. Bhan. Kinetics and Thermochemistry of C_4C_6 Olefin Cracking on H-ZSM-5. *ACS Catal.*, 4(7):2319 – 2327, 2014.

[121] C. M. Nguyen, B. A. De Moor, M.-F. Reyniers, and G. B. Marin. Physisorption and Chemisorption of Linear Alkenes in Zeolites: A Combined QM-Pot(MP2//B3LYP:GULP)–Statistical Thermodynamics Study. *The Journal of Physical Chemistry C*, 115(48):23831–23847, 2011.

[122] C. M. Nguyen, B. A. De Moor, M.-F. Reyniers, and G. B. Marin. Isobutene Protonation in H-FAU, H-MOR, H-ZSM-5, and H-ZSM-22. *The Journal of Physical Chemistry C*, 116(34):18236–18249, 2012.

[123] Z. P. Gerdtzen. *Modeling, analysis and theoretical exploration of the metabolism of mammalian cells in culture.* PhD thesis, University of Minnesota, 2005.

[124] G. H Golub and C. F. Van Loan. *Matrix Computations, 4th Edition.* Johns Hopkins University Press, Baltimore, MD, 2013.

[125] N. Wu, M. Y. Yang, U. Gaur, H. L. Xu, Y. F. Yao, and D. Y. Li. Alpha-ketoglutarate: Physiological functions and applications. *Biomol Ther.*, 24:1 – 8, 2016.

[126] B. Zdzisinska, A. Zurek, and M. Kandefer-Szerszen. Alpha-ketoglutarate as a molecule with pleiotropic activity: Well-known and novel possibilities of therapeutic use. *Arch Immunol Ther Ex.*, 65:21 – 36, 2017.

[127] U. Stottmeister, A. Aurich, H. Wilde, J. Andersch, S. Schmidt, and D. Sicker. White biotechnology for green chemistry: fermentative 2-oxocarboxylic acids as novel building blocks for subsequent chemical syntheses. *J Ind Microbiol Biot.*, 32:651 – 664, 2005.

[128] D. G. Barrett and M. N. Yousaf. Poly(triol alpha-ketoglutarate) as biodegradable, chemoselective, and mechanically tunable elastomers. *Macromolecules*, 41:6347 – 6352, 2008.

[129] Y. S. Tai, M. Xiong, P. Jambunathan, J. Wang, C. Stapleton, and K. Zhang. Engineering nonphosphorylative metabolism to generate lignocellulose-derived products. *Nature chemical biology.*, 12:247 – 253, 2016.

[130] Daylight. *Daylight Theory Manual.*, 2008.

[131] M. H. Medema and M. A. Fischbach. Computational approaches to natural product discovery. *Nature chemical biology.*, 11:639 – 648, 2015.

[132] S.A. Tabak, F. J. Krambeck, and W. E. Garwood. Conversion of propylene and butylene over ZSM-5 catalyst. *AIChE J*, 32:1526 – 1531, 1986.

[133] R. J. Quann, L. A. Green, S.A. Tabak, and F. J. Krambeck. Chemistry of olefin oligomerization over ZSM-5 catalyst. *Ind. Eng. Chem. Res.*, 27:565 – 570, 1988.

[134] M. Boronat, C. M. Zicovich-Wilson, P. Viruela, and A. Corma. Influence of the local geometry of zeolite active sites and olefin size on the stability of alkoxide intermediates. *J. Phys. Chem. B*, 105(45):11169 – 11177, 2001.

[135] A. Bhan, Y. V. Joshi, W. N. Delgass, and K. T. Thomson. DFT Investigation of Alkoxide Formation from Olefins in H-ZSM-5. *J. Phys. Chem. B*, 107(38):10476 – 10487, 2003.

[136] C. M. Nguyen, B. A. De Moor, M.-F. Reyniers, and G. B. Marin. Physisorption and chemisorption of linear alkenes in zeolites: A combined QM-Pot(MP2//B3LYP:GULP)statistical thermodynamics study. *The Journal of Physical Chemistry C*, 115(48):23831 – 23847, 2011.

[137] P. Cnudde, K.D. Wispelaere, J.V. Mynsbrugge, M. Waroquier, and V. V. Speybroeck. Effect of temperature and branching on the nature and stability of alkene cracking intermediates in H-ZSM-5. *Journal of Catalysis*, 345:53 – 69, 2017.

[138] D. M. McCann, D. Lesthaeghe, P. W. Kletnieks, D. R. Guenther, M. J. Hayman, V. V. Speybroeck, M. Waroquier, and J. F. Haw. A complete catalytic cycle for supramolecular methanol-to-olefins conversion by linking theory with experiment. *Angew. Chem. Int. Ed.*, 47:5179 – 5182, 2008.

[139] J. Weitkamp, P. A. Jacobs, and J. A. Martens. Isomerization and Hydrocracking of C_9 through C_{16} n-Alkanes on Pt/HZSM-5 zeolite. *Applied Catalysis*, 8:123 – 141, 1983.

[140] J. S. Buchanan, J. G. Santiesteban, and W. O. Haag. Mechanistic considerations in acid-catalyzed cracking of olefins. *Journal of Catalysis*, 158:279 – 287, 1996.

[141] F. C. Whitmore. The common basis of intramolecular rearrangements. *J. Am. Chem. Soc.*, 54(8):3274 – 3283, 1932.

[142] M. L. Sarazen, E. Doskocil, and E. Iglesia. Effects of void environment and acid strength on alkene oligomerization selectivity. *ACS Catal.*, 6:7059 – 7070, 2016.

[143] J. F. Haw, J. B. Nicholas, W. Song, F. Deng, Z. Wang, T. Xu, and C. S. Heneghan. Roles for Cyclopentenyl Cations in the Synthesis of Hydrocarbons from Methanol on Zeolite Catalyst HZSM-5. *J. Am. Chem. Soc.*, 122(19):4763 – 4775, 2000.

[144] G. M. Mullen and M. J. Janik. Density Functional Theory Study of Alkane-Alkoxide Hydride Transfer in Zeolites. *ACS Catal.*, 1(2):105 – 115, 2011.

[145] W. O. Haag, R. M. Lago, and P. B. Weisz. Transport and reactivity of hydrocarbon molecules in a shape-selective zeolite. *Faraday Discussions of the Chemical Society*, 72:317 – 330, 1981.

[146] M. N. Mazar, S. Al-Hashimi, M. Cococcioni, and A. Bhan. Beta-scission of olefins on acidic zeolites: A periodic PBE-D study in H-ZSM-5. *Journal of Physical Chemistry C*, 117:23609, 2013.

[147] T. v. Aretin, S. Schallmoser, S. Standl, M. Tonigold, J. A. Lercher, and O. Hinrichsen. Single-Event Kinetic Model for 1-Pentene Cracking on ZSM-5. *Ind. Eng. Chem. Res.*, 54:11792 – 11803, 2015.

[148] T. v. Aretin and O. Hinrichsen. Single-Event Kinetic Model for Cracking and Isomerization of 1-Hexene on ZSM-5. *Ind. Eng. Chem. Res.*, 53:19460 – 19470, 2014.

[149] S. Standl, M. Tonigold, and O. Hinrichsen. Single-Event Kinetic Modeling of Olefin Cracking on ZSM-5: Proof of Feed Independence. *Ind. Eng. Chem. Res.*, 56:13096 – 13108, 2017.

[150] P. Oliveira, P. Borges, Ricardo Ramos Pinto, Amlia Lemos, Flvio Lemos, Jacques Vdrine, and F. Rama Ribeiro. Light olefin transformation over ZSM-5 zeolites with different acid strengths A kinetic model. *Applied Catalysis A General*, 384:177 – 185, 2010.

[151] Y. V. Joshi and K. T. Thomson. Embedded cluster (QM/MM) investigation of C_6 diene cyclization in HZSM-5. *Journal of Catalysis*, 230(2):440 – 463, 2005.

[152] Y. V. Joshi and K. T. Thomson. Brønsted acid catalyzed cyclization of C_7 and C_8 dienes in HZSM-5: A hybrid QM/MM study and comparison with C_6 diene cyclization. *The Journal of Physical Chemistry C*, 112(33):12825 – 12833, 2005.

[153] M. L. Sarazen and E. Iglesia. Experimental and theoretical assessment of the mechanism of hydrogen transfer in alkane-alkene coupling on solid acids. *J. Catal.*, 354:287 – 298, 2017.

[154] L. Ying, J. Zhu, Y. Cheng, L. Wang, and X. Li. Kinetic modeling of C_2-C_7 olefins interconversion over ZSM-5 catalyst. *J. Ind. Eng. Chem.*, 33:80 – 90, 2016.

[155] X. Huang, D. Aihemaitijiang, and W.-D. Xiao. Reaction pathway and kinetics of C_3-C_7 olefin transformation over high-silicon HZSM-5 zeolite at 400-490. *Chem. Eng. J*, 280:222 – 232, 2015.

[156] R. J. Quann and S. B. Jaffe. Structure-oriented lumping: describing the chemistry of complex hydrocarbon mixtures. *Industrial & Engineering Chemistry Research*, 31:2483 – 2497, 1992.

[157] W. H. Green, P. I. Barton, B. Bhattacharjee, D. M. Matheu, D. A. Schwer, J. Song, R. Sumathi, H. H. Carstensen, A. M. Dean, and J. M. Grenda. Computer construction of detailed chemical kinetic models for gas-phase reactors. *Industrial & Engineering Chemistry Research*, 40:5362 – 5370, 2001.

[158] R. G. Susnow, A. M. Dean, W. H. Green, P. Peczak, and L. J. Broadbelt. Rate-based construction of kinetic models for complex systems. *The Journal of Physical Chemistry A*, 101:3731 – 3740, 1997.

[159] E. Ranzi, A. Frassoldati, S. Granata, and T. Faravelli. Wide-range kinetic modeling study of the pyrolysis, partial oxidation, and combustion of heavy n-alkanes. *Industrial & Engineering Chemistry Research*, 44:5170 – 5183, 2005.

[160] J Wei and J. C. W. Kuo. A lumping analysis in monomolecular reaction systems: Analysis of the exactly lumpable system. *Ind. Eng. Chem. Fundam.*, 8:114 – 123, 1969.

[161] J. C. W. Kuo and J. Wei. A lumping analysis in monomolecular reaction systems: Analysis of approximately lumpable system. *Ind. Eng. Chem. Fundam.*, 8:124 – 133, 1969.

[162] P. G. Coxson and K. B. Bischoff. Lumping strategy. 1. introductory techniques and applications of cluster analysis. *Ind. Eng. Chem. Res.*, 26:1239 – 1248, 1987.

[163] P. G. Coxson and K. B. Bischoff. Lumping strategy. 2. a system theoretic approach. *Ind. Eng. Chem. Res.*, 26:2151 – 2157, 1987.

[164] C. Kravaris, J. Hahn, and Y. Chu. Advances and selected recent developments in state and parameter estimation. *Computers & Chemical Engineering*, 51(0):111 – 123, 2013.

[165] X. Zhang, D. Liu, D. Xu, S. Asahina, K. A. Cychosz, K. V. Agrawal, Y. A. Wahedi, A. Bhan, S. A. Hashimi, O. Terasaki, M. Thommes, and M. Tsapatsis. Synthesis of self-pillared zeolite nanosheets by repititive branching. *Science*, 336:1684, 2012.

Appendix

A   Inputs into RING for studying biosynthesis of 2KG from Xylose

```
input reactant "[{NAD}+]"
input reactant "P(=O)(O)(O)O"
input reactant "[H+]"
input reactant "[{NAD}H]"
input reactant "C(O)C(O)C(O)C(O)C=O"
input reactant "C(=O)(O)O"
input reactant "O"
input reactant "{CoA}SH"


define composite atom NAD
define composite atom NADH
define composite atom CoA


group CdoubleC (c1,c2){
    C labeled c1
    C labeled c2 double bond to c1}


define characteristic olefinicMol on Molecule{
    Molecule contains >= 1 of group CdoubleC
}


define characteristic carboxylated on mol
{
        fragment b{
        C labeled c1 {connected to 3 X with any bond}
        O labeled o1 double bond to c1
        O labeled o2 single bond to c1
        }
        mol contains 1 of b
}


define characteristic dicarboxylated on mol
{
        fragment b{
        C labeled c1 {connected to 3 X with any bond}
```

```
        O labeled o1 double bond to c1
        O labeled o2 single bond to c1
        }
        mol contains 2 of b
}


define characteristic tricarboxylated on mol
{
        fragment b{
        C labeled c1 {connected to 3 X with any bond}
        O labeled o1 double bond to c1
        O labeled o2 single bond to c1
        }
        mol contains 3 of b
}


define characteristic C6 on mol
{
        fragment b{
        C labeled c1
        C labeled c2 single bond to c1
        }
        mol contains <=5 of b
}


define characteristic containsPhosphate on mol
{
        fragment b{
        P labeled p1
        O labeled o1 double bond to p1
        O labeled o2 single bond to p1
        C labeled c1 single bond to o2
        }
        mol contains >= 1 of b
}


define characteristic phosphorylated on mol
{
        fragment b{
```

```
        P labeled p1
        O labeled o1 double bond to p1
        O labeled o2 single bond to p1
        C labeled c1 single bond to o2
        }
        mol contains 1 of b
}


define characteristic diphosphorylated on mol
{
        fragment b{
        P labeled p1
        O labeled o1 double bond to p1
        O labeled o2 single bond to p1
        C labeled c1 single bond to o2
        }
        mol contains 2 of b
}


//global constraints specification
global constraints on Molecule
{
        //declaration of a fragment named 'a'
        fragment a
        {
                C+ labeled 1
                $ labeled 2 double bond to 1
        }
        //molecule does not contain C+=$, where $ is any atom
        ! Molecule contains a

        //cannot have C-O-P-O-C linkage
        fragment b
        {
        C labeled c1
        O labeled o1 single bond to c1
        P labeled p1 single bond to o1
        O labeled o2 single bond to p1
        C labeled c2 single bond to o2
```

```
        }
        ! Molecule contains b

        //Molecule.size < 15 //molecule size is less than 10 (number of heavy,
            non hydrogen atoms is less than 10)
        Molecule.size between 1 and 24

        //Molecule.charge >-2 && Molecule.charge <2 //(charge is -1, 0, or 1)
        Molecule.charge between -2 and 2

        fragment b
        {
                C labeled c1
                C labeled c2 double bond to c1
                X labeled x1 double bond to c2
        }
        ! Molecule contains >= 1 of b

        fragment c{
        P labeled p1
        O labeled o1 double bond to p1
        O labeled o2 single bond to p1
        C labeled c1 single bond to o2
        }
        ! Molecule contains > 2 of c

        fragment d{
        CoA labeled c1
        S labeled s1 single bond to c1
        }
        ! Molecule contains > 1 of d
}

//Define reaction rules

//reaction rule 111a
rule Oxido111a{
        neutral reactant r1{
        C labeled c1
```

```
        H labeled h1 single bond to c1
        O labeled o1 single bond to c1
        H labeled h2 single bond to o1
        }
        reactant r2{
        NAD+ labeled n1}
        constraints{
                r1.size <= 10 && r1 is C6}
        increase bond order(c1, o1)
        break bond (c1, h1)
        break bond (o1, h2)
        form bond (n1, h1)
        modify atomtype (h2, H+)
        modify atomtype (n1, NAD)
}

//reverse reaction rule 111a
rule ReverseOxido111a{
        linear reactant r1{
        C labeled c1 {connected to 1 O with any bond}
        O labeled o1 double bond to c1
        }
        reactant r2{
        NAD labeled n1
        H labeled h1 single bond to n1
        }
        positive reactant r3{
        H+ labeled h2
        }
        constraints{
        ! r1 is containsPhosphate && r1.size >= 6 && r1.size <=12
        }
        break bond (n1, h1)
        decrease bond order (c1, o1)
        form bond (c1, h1)
        form bond (o1, h2)
        modify atomtype (h2, H)
        modify atomtype (n1, NAD+)
}
```

```
//reaction rule 111b
rule Oxido111b{
      linear reactant r1{
      C labeled c1
      O labeled o1 single bond to c1
      H labeled h1 single bond to c1
      C labeled c2 single bond to c1
      C labeled c3 single bond to c2
      C labeled c4 single bond to c3
      O labeled o2 double bond to c4
      }
      break bond (c1, h1)
      decrease bond order (c4, o2)
      form bond (h1, c4)
      form bond (o2, c1)
}

rule ReverseOxido111b{
      cyclic reactant r1{
            C labeled c1
            O labeled o1 single bond to c1
            H labeled h1 single bond to c1
            C labeled c2 single bond to o1 {connected to 4 $ with single
                 bond}
            O labeled o2 single bond to c2
      }
      constraints{
      r1.maxringsize = 5}
      break bond (o1, c2)
      break bond (c1, h1)
      form bond (c2, h1)
      increase bond order (c1, o1)
}


//Reaction Rule for oxidoreductase EC class
//EC 1.1.1 Oxidoreductases acting on the alcoholic group CH-OH of donors
```

```
//1.1.1.c KEGG: R05698

rule Oxido111c{
        linear reactant r1{
        C labeled c1
        O labeled o1 single bond to c1
        H labeled h1 single bond to c1
        C labeled c2 single bond to c1
        C labeled c3 single bond to c2
        C labeled c4 single bond to c3
        C labeled c5 single bond to c4
        O labeled o2 double bond to c5
        }
        break bond (c1, h1)
        decrease bond order (c5, o2)
        form bond (h1, c5)
        form bond (o2, c1)
}

rule ReverseOxido111c{
        cyclic reactant r1{
                C labeled c1
                O labeled o1 single bond to c1
                H labeled h1 single bond to c1
                C labeled c2 single bond to o1 {connected to 4 $ with single
                    bond}
                O labeled o2 single bond to c2
        }
        constraints{
        r1.maxringsize <= 6 && r1.minringsize > 5}
        break bond (o1, c2)
        break bond (c1, h1)
        form bond (c2, h1)
        increase bond order (c1, o1)
}

rule Oxido111d{
        reactant r1{
                C labeled c1
```

```
                    C labeled c2 single bond to c1
                    O labeled o1 double bond to c2
                    C labeled c3 single bond to c2
                    C labeled c4 single bond to c3
                    O labeled o2 single bond to c3
                    H labeled h1 single bond to o2
            }
            constraints{
            r1 is carboxylated && r1.size <= 10}
            break bond (c3, c4)
            form bond (c4, c2)
            decrease bond order (c2, o1)
            break bond (o2, h1)
            form bond (o1, h1)
            increase bond order (c3, o2)
}


//different reaction all together... combined mechanism for 2 reactions
rule Dehydrogenase111d{
            neutral reactant r1{
            C labeled c1 {connected to 3 X with any bond}
            O labeled o1 single bond to c1
            H labeled h1 single bond to o1
            H labeled h2 single bond to c1
            C labeled c2 single bond to c1
            C labeled c3 single bond to c2
            O labeled o2 double bond to c3
            O labeled o3 single bond to c3
            H labeled h3 single bond to o3
            }
            positive reactant r2{
            NAD+ labeled n1
            }
            constraints{
            r1 is tricarboxylated
            }
            break bond (h3, o3)
            modify atomtype (n1, NAD)
```

```
        form bond (n1, h3)
        break bond (c3, c2)
        increase bond order (c3, o3)
        break bond (h2, c1)
        break bond (o1, h1)
        modify atomtype (h1, H+)
        increase bond order (c1, o1)
        form bond (c2, h2)
}


//oxidises an aldehyde to an acid with presence of water
rule Oxido121a{
        neutral reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
                H labeled h1 single bond to c1
        }
        positive reactant cofactor{
                NAD+ labeled n1
        }
        reactant r3{
                O labeled o2
                H labeled h2 single bond to o2
                H labeled h3 single bond to o2
        }
        constraints{
                r1.size < 13
        }
        break bond (c1, h1)
        form bond (n1, h1)
        break bond (h2, o2)
        form bond (c1, o2)
        modify atomtype (h2, H+)
        modify atomtype (n1, NAD)
}


//phosphorylates an aldehyde with the presence of phosphate group
rule Oxido121b{
        neutral reactant r1{
```

```
        C labeled c1
        H labeled h1 single bond to c1
        O labeled o1 double bond to c1
        }
        reactant r2{
        P labeled p1
        O labeled o2 single bond to p1
        H labeled h2 single bond to o2
        }
        positive reactant cofactor{
                NAD+ labeled n1
        }
        constraints{
        r1 is phosphorylated && r2.size <= 5}
        break bond (c1, h1)
        break bond (o2, h2)
        form bond (c1, o2)
        form bond (n1, h1)
        modify atomtype (h2, H+)
        modify atomtype (n1, NAD)
}


// //reaction rule 2.2.1.a Transketolase
rule Transferase221a{
        linear reactant r1{
        C labeled c1
        O labeled o1 double bond to c1
        C labeled c2 single bond to c1 {connected to 2 X with single bond}
        O labeled o2 single bond to c2
        C labeled c3 single bond to c1
        H labeled h1 single bond to c3
        O labeled o3 single bond to c3
        H labeled h2 single bond to o3
        }
        linear reactant r2{
        C labeled c4 {connected to 2 X with any bond}
        O labeled o4 double bond to c4
        H labeled h3 single bond to c4
        C labeled c5 single bond to c4 {connected to 3 X with any bond}
```

```
        O labeled o5 single bond to c5
        }
        constraints{
        ! r1 is olefinicMol && ! r2 is olefinicMol && r1 is phosphorylated &&
            r2 is phosphorylated && r1.size >= 12 && r2.size >= 12 && r1.size <=
             16 && r2.size <= 16
        }
        break bond (c1, c3)
        break bond (o3, h2)
        increase bond order (c3, o3)
        decrease bond order (c4, o4)
        form bond (c1, c4)
        form bond (o4, h2)
        product constraints on mol{
                mol is phosphorylated
        }
}

//reaction rule 2.2.1.b Transaldolase
rule Transferase221b{
        linear reactant r1{
        C labeled c1 {connected to 2 X with any bond}
        O labeled o1 single bond to c1
        C labeled c2 single bond to c1
        O labeled o2 double bond to c2
        C labeled c3 single bond to c2
        O labeled o3 single bond to c3
        C labeled c4 single bond to c3
        O labeled o4 single bond to c4
        H labeled h1 single bond to o4
        }
        linear reactant r2{
        C labeled c5 {connected to 2 X with any bond}
        O labeled o5 double bond to c5
        H labeled h2 single bond to c5
        C labeled c6 single bond to c5 {connected to 3 X with any bond}
        O labeled o6 single bond to c6
        }
        constraints{
```

```
        ! r1 is olefinicMol && ! r2 is olefinicMol && r1 is phosphorylated &&
            r2 is phosphorylated && r1.size >= 16 && r2.size >= 10 && r1.size <=
             18 && r2.size <= 12
        }
        break bond (c3, c4)
        break bond (o4, h1)
        increase bond order (c4, o4)
        decrease bond order (c5, o5)
        form bond (c3, c5)
        form bond (o5, h1)
        product constraints on mol{
                mol is phosphorylated
        }
}


//reaction rule 2.3.3
//this reaction rule is reaction of CoA with oxaloacetate
rule Transferase233a{
        neutral reactant r1{
        C labeled c1
        O labeled o1 double bond to c1
        S labeled s1 single bond to c1
        CoA labeled z1 single bond to s1
        C labeled c2 single bond to c1 {connected to 1 X with any bond}
        H labeled h1 single bond to c2
        }
        neutral reactant r2{
        C labeled c3 {connected to 1 O with any bond}
        O labeled o2 double bond to c3
        }
        reactant r3{
        O labeled o3
        H labeled h2 single bond to o3
        H labeled h3 single bond to o3
        }
        constraints{
        r2 is dicarboxylated && r1.size = 5 && r2.size = 9
        }
        break bond (c1, s1)
```

```
          break bond (h2, o3)
          break bond (c2, h1)
          form bond (h1, s1)
          decrease bond order (c3, o2)
          form bond (c1, o3)
          form bond (o2, h2)
          form bond (c2, c3)
}


//reaction rule phosphotransferase 2.7.1
rule Transferase271a{
          linear reactant r1{
          C labeled c1 {connected to 2 X with any bond} //needs to be terminal
               alcohol group
          O labeled o1 single bond to c1
          H labeled h1 single bond to o1
          }
          reactant r2{
          P labeled p1
          O labeled o2 single bond to p1
          H labeled h2 single bond to o2
          }
          break bond (p1, o2)
          break bond (o1, h1)
          form bond (h1, o2)
          form bond (o1, p1)
}


//reverse reaction rule phosphotransferase 2.7.1
//added this reaction rule so that species which are not terminal alcohols can
     also react
rule ReverseTransferase271b{
          linear reactant r1{
          C labeled c1 {connected to 1 O with any bond}
          O labeled o1 single bond to c1
          P labeled p1 single bond to o1
          }
          reactant r2{
          O labeled o2
```

```
        H labeled h1 single bond to o2
        H labeled h2 single bond to o2
        }
        break bond (p1, o1)
        break bond (o2, h1)
        form bond (p1, o2)
        form bond (o1, h1)
}

//reaction rule phosphotransferase 2.7.2 //requires a carboxylic acid group at
    the end
rule Transferase272b{
        linear reactant r1{
        C labeled c1 {connected to 3 X with any bond} //needs to be terminal
            acid group
        O labeled o1 single bond to c1
        O labeled o2 double bond to c1
        H labeled h1 single bond to o1
        }
        reactant r2{
        P labeled p1
        O labeled o3 single bond to p1
        H labeled h2 single bond to o3
        }
        constraints{
                r1 is phosphorylated && r2.size <= 5
        }
        break bond (p1, o3)
        break bond (o1, h1)
        form bond (h1, o3)
        form bond (o1, p1)
}

//reverse reaction rule for phosphotransferase 2.7.2
rule ReverseTransferase272b{
        linear reactant r1{
        C labeled c1
        O labeled o1 double bond to c1
        O labeled o2 single bond to c1
```

```
        P labeled p1 single bond to o2
        }
        reactant r2{
        O labeled o3
        H labeled h1 single bond to o3
        H labeled h2 single bond to o3
        }
        constraints{
                r1 is diphosphorylated
        }
        break bond (o2, p1)
        break bond (o3, h1)
        form bond (p1, o3)
        form bond (o2, h1)
}


//h311a
rule Hydrolaseeester311a{
        neutral reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
                O labeled o2 single bond to c1
                C labeled c2 single bond to o2
                C labeled c3 single bond to c1
        }
        reactant r2{
                O labeled o3
                H labeled h1 single bond to o3
                H labeled h2 single bond to o3
        }
        constraints{
                r1.size <=5
        }
        break bond (c1, o2)
        break bond (h1, o3)
        form bond (c1, o3)
        form bond (o2, h1)
}
```

```
//h311b
rule Hydrolaseester311b{
        neutral reactant r1{
                C labeled c1
                C labeled c2 double bond to c1
                O labeled o2 single bond to c1
                C labeled c3 single bond to o2
                C labeled c4 single bond to c1
        }
        reactant r2{
                O labeled o3
                H labeled h1 single bond to o3
                H labeled h2 single bond to o3
        }
        constraints{
                r1.size <=5
        }
        break bond (c1, o2)
        break bond (h1, o3)
        form bond (c1, o3)
        form bond (o2, h1)
}


//h312c
rule Hydrolaseester312c{
        neutral reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
                S labeled s1 single bond to c1
                CoA labeled z1 single bond to s1
                C labeled c3 single bond to c1
        }
        reactant r2{
                O labeled o3
                H labeled h1 single bond to o3
                H labeled h2 single bond to o3
        }
        constraints{
                r1.size <=5
```

```
        }
        break bond (c1, s1)
        break bond (h1, o3)
        form bond (c1, o3)
        form bond (s1, h1)
}


//h312d
rule Hydrolaseester312d{
        neutral reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
                S labeled s1 single bond to c1
                CoA labeled z1 single bond to s1
                C labeled c3 single bond to c1
        }
        reactant r2{
        NAD labeled n1
        H labeled h1 single bond to n1
        }
        positive reactant r3{
        H+ labeled h2
        }
        constraints{
                r1.size <=5
        }
        break bond (c1, s1)
        break bond (n1, h1)
        form bond (c1, h1)
        form bond (s1, h2)
        modify atomtype (h2, H)
        modify atomtype (n1, NAD+)
}


//r312d
rule ReverseHydrolaseester312d{
        neutral reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
```

```
                        C labeled c2 single bond to c1
                        H labeled h1 single bond to c1
                }
                reactant r2{
                        CoA labeled z1
                        S labeled s1 single bond to z1
                        H labeled h2 single bond to s1
                }
                reactant cofactor{
                        NAD+ labeled n1
                }
                constraints{
                        r1.size <=4
                }
                break bond (s1, h2)
                break bond (h1, c1)
                form bond (s1, c1)
                form bond (h1, n1)
                modify atomtype (n1, NAD)
                modify atomtype (h2, H+)
        }


        //h371a
        rule Hydrolaseeester371a{
                neutral reactant r1{
                        C labeled c1
                        O labeled o1 double bond to c1
                        C labeled c2 single bond to c1
                        C labeled c3 single bond to c1
                        C labeled c4 single bond to c2
                        O labeled o2 double bond to c4
                }
                reactant r2{
                        O labeled o3
                        H labeled h1 single bond to o3
                        H labeled h2 single bond to o3
                }
                constraints{
                        r1.size <=6
```

```
        }
        break bond (c1, c2)
        break bond (h1, o3)
        form bond (c1, o3)
        form bond (c2, h1)
}

//reaction rule 4.1.1 Lyase EC class
rule Lyase411a{
        neutral reactant r1{
        C labeled c1
        O labeled o1 double bond to c1
        C labeled c2 single bond to c1
        O labeled o2 double bond to c2
        O labeled o3 single bond to c2
        H labeled h1 single bond to o3
        }
        reactant r2{
        CoA labeled z1
        S labeled s1 single bond to z1
        H labeled h2 single bond to s1
        }
        reactant cofactor{
        NAD+ labeled n1
        }
        constraints{
        ! r1 is phosphorylated}
        break bond (o3, h1)
        form bond (n1, h1)
        modify atomtype (n1, NAD)
        break bond (c1, c2)
        increase bond order (c2, o3)
        break bond (s1, h2)
        form bond (s1, c1)
        modify atomtype (h2, H+)
        product constraints on mol {
        mol.size <= 6}
}
```

```
rule Lyase412a{
        linear reactant r1{
                C labeled c1 {connected to 3 X with any bond}
                O labeled o1 double bond to c1
                C labeled c2 single bond to c1
                O labeled o2 single bond to c2
                C labeled c3 single bond to c2
                O labeled o3 single bond to c3
                H labeled h1 single bond to o3
        }
        constraints{
                r1 is diphosphorylated
        }
        break bond (c2, c3)
        break bond (o3, h1)
        increase bond order (c3, o3)
        form bond (h1, c2)
        product constraints on mol{
                mol is phosphorylated
        }
}


//reaction rule 4.2.1a
//deHydratase - removes a water molecule from 2 adjacent alcohol groups and
    results in a ketone or aldehyde group
rule Lyase421a{
        neutral reactant r1{
        C labeled c1
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1 //to ensure it is an alcohol group
        C labeled c2 single bond to c1
        O labeled o2 single bond to c2
        H labeled h2 single bond to o2 //to ensure it is an alcohol group
        H labeled h3 single bond to c2
        }
        constraints{
        ! r1 is containsPhosphate && r1 is C6}
        break bond (c1, o1)
        break bond (o2, h2)
```

```
        break bond (h3, c2)
        form bond (h3, o1)
        form bond (h2, c1)
        increase bond order (c2, o2)
}


//reaction rule 4.2.1b
//enolase - terminal alcohol group can dehydrate if beta-carbon has a hydrogen
//can add constraint that the molecule is carboxylated and phosphorylated
rule Lyase421b{
        neutral reactant r1{
        C labeled c1 {connected to 2 X with any bond}
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1
        C labeled c2 single bond to c1
        H labeled h2 single bond to c2
        }
        constraints{
        r1 is phosphorylated && r1 is carboxylated}
        break bond (c1, o1)
        break bond (c2, h2)
        form bond (o1, h2)
        increase bond order (c1, c2)
}


//reaction rule 4.2.1c
//aconitase - this reaction rule switches alcohol group .. making it more
    specific to prevent reaction network explosion
rule Lyase421c{
        neutral reactant r1{
        C labeled c1
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1
        C labeled c2 single bond to c1
        H labeled h2 single bond to c2
        }
        constraints{
        r1 is tricarboxylated && r1.size <= 13
        }
```

```
        break bond (c1, o1)
        break bond (c2, h2)
        form bond (c1, h2)
        form bond (c2, o1)
}


//reaction rule 5.3.2a
//this reaction rule converts an alcohol with an adjacent double bond (C=C)
    into an enol form
rule Isomerase532a{
        neutral reactant r1{
        C labeled c1 {connected to 3 X with any bond}
        C labeled c2 double bond to c1
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1
        }
        constraints{
        r1.size <= 6 && r1 is carboxylated
        }
        decrease bond order (c1, c2)
        increase bond order (c1, o1)
        break bond (o1, h1)
        form bond (h1, c2)
}


//reaction rule 5.3.2.b
//this reaction rule does similar steps as hydrogen transfer
rule Isomerase532b{
        neutral reactant r1{
                C labeled c1 {connected to 3 X with any bond}
                C labeled c2 double bond to c1
                C labeled c3 single bond to c2
                H labeled h1 single bond to c3
        }
        constraints{
                r1.size <= 10 && r1 is carboxylated
        }
        decrease bond order (c1, c2)
        break bond (h1, c3)
```

```
        increase bond order (c2, c3)
        form bond (h1, c1)
}


//reaction rule 5.4.2a
//the alcohol group undergoes phosphorylation and then successive
    dephosphorylation of the terminal phosphate group
rule Isomerase542a{
        neutral reactant r1{
        C labeled c1
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1
        C labeled c2 single bond to c1 {connected to 2 X with any bond}
        O labeled o2 single bond to c2
        P labeled p1 single bond to o2
        }
        constraints{
        r1 is phosphorylated && r1 is carboxylated}
        break bond (o2, p1)
        break bond (o1, h1)
        form bond (o2, h1)
        form bond (o1, p1)
}


//reverse reaction rule for 5.4.2
rule ReverseIsomerase542a{
        neutral reactant r1{
        C labeled c1 {connected to 2 X with any bond}
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1
        C labeled c2 single bond to c1
        O labeled o2 single bond to c2
        P labeled p1 single bond to o2
        }
        constraints{
        r1 is phosphorylated && r1 is carboxylated}
        break bond (o1, h1)
        break bond (o2, p1)
        form bond (o1, p1)
```

```
        form bond (o2, h1)
}


//Ligase621a
rule Ligase621a{
        neutral reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
                O labeled o2 single bond to c1
                H labeled h1 single bond to o2
                C labeled c2 single bond to c1
        }
        reactant r2{
                CoA labeled z1
                S labeled s1 single bond to z1
                H labeled h2 single bond to s1
        }
        constraints{
        r1.size <= 4 && r1.size <= 3 && r1 is carboxylated}
        break bond (c1, o2)
        break bond (h2, s1)
        form bond (s1, c1)
        form bond (h2, o2)
}


//Ligase641a
rule Ligase641a{
        reactant r1{
                C labeled c1
                O labeled o1 double bond to c1
                O labeled o2 single bond to c1
                H labeled h1 single bond to o2
                O labeled o3 single bond to c1
                H labeled h2 single bond to o3
        }
        reactant r2{
                C labeled c2
                O labeled o4 double bond to c2
                C labeled c3 single bond to c2
```

```
                    C labeled c4 single bond to c2
                    H labeled h3 single bond to c4
            }
            constraints{
            r1.size <= 4 && r2.size <= 4 && r1 is carboxylated}
            break bond (h3, c4)
            break bond (c1, o3)
            form bond (c1, c4)
            form bond (h3, o3)
}


//Ligase641b
rule Ligase641b{
            reactant r1{
                    C labeled c1
                    O labeled o1 double bond to c1
                    O labeled o2 single bond to c1
                    H labeled h1 single bond to o2
                    O labeled o3 single bond to c1
                    H labeled h2 single bond to o3
            }
            reactant r2{
                    C labeled c2
                    O labeled o4 double bond to c2
                    C labeled c3 single bond to c2
                    C labeled c4 single bond to c2
                    C labeled c5 double bond to c4
                    C labeled c6 single bond to c5
                    H labeled h3 single bond to c6
            }
            constraints{
            r1.size <= 4 && r2.size <= 6 && r1 is carboxylated}
            break bond (h3, c6)
            break bond (c1, o3)
            form bond (c1, c6)
            form bond (h3, o3)
}

rule Ligase641c{
```

```
        neutral reactant r1{
        C labeled c1 {connected to 1 X with single bond}
        H labeled h1 single bond to c1
        C labeled c2 single bond to c1
        O labeled o1 double bond to c2
        C labeled c3 single bond to c2
        }
        reactant r2{
        C labeled c4
        O labeled o2 double bond to c4
        O labeled o3 single bond to c4
        H labeled h2 single bond to o3
        O labeled o4 single bond to c4
        }
        constraints{
        r2.size <= 4 && r1.size <= 6 && r1 is carboxylated}
        break bond (c4, o4)
        break bond (h2, o3)
        form bond (h2, o4)
        break bond (h1, c1)
        form bond (c4, c1)
        form bond (h1, o3)
}

find pathways to mol{
        mol is "OC(=O)C(=O)CCC(=O)O"
        } constraints {
        maximum length 10
        eliminate similar pathways
        } store in "2KG.txt"
```

B   Inputs into RING for studying N-Glycosylation system

```
// Input glycan: Man9(GlcNAc2) or {Man}9{GlcNAc}2
input reactant "N[{GlcNAc}][{B14}][{GlcNAc}][{B14}][{Man}]([{A13}][{Man}][{A12
    }][{Man}][{A12}][{Man}])[{A16}][{Man}]([{A13}][{Man}][{A12}][{Man}])[{A16
    }][{Man}][{A12}][{Man}]"
//The following reactants are substrates needed to initiate reactions (a.k.a.
    nucleotide sugars)
input reactant "[{B12}][{GlcNAc}]" // UDP-GlcNAc (for beta1,2 linkage)
input reactant "[{A16}][{Fuc}]" // GDP-Fuc
input reactant "[{B14}][{Gal}]" // UDP-Gal
input reactant "[{B14}][{GlcNAc}]" // UDP-GlcNAc (for beta1,4 linkage)
input reactant "[{B16}][{GlcNAc}]" // UDP-GlcNAc (for beta1,6 linkage)
input reactant "[{A23}][{Sia}]" // CMP-Sia
input reactant "[{GlcNAc}]"
input reactant "[{Man}]"
input reactant "[{Fuc}]"
input reactant "[{Gal}]"
input reactant "[{Sia}]"

define composite atom GlcNAc // boxes (GlcNAc)
define composite atom Man // circles (Man)
define composite atom Fuc // triangles (Fuc)
define composite atom Gal // empty circles (Gal)
define composite atom Sia // diamonds (Sia)
//defining the bonds
define composite atom A12 // alpha1,2
define composite atom A13 // alpha1,3
define composite atom A16 // alpha1,6
define composite atom A23 // alpha2,3
//define composite atom A26 (if needed later)
define composite atom B12 // beta1,2
define composite atom B14 // beta1,4
define composite atom B16 // beta1,6

      // Man-a12-Man (Man-a12-Man)
group alpha12 (s1, s2, s3){
      Man labeled s1
      A12 labeled s2 single bond to s1
```

```
        Man labeled s3 single bond to s2
        }


        // Man-a13-Man (Man-a13-Man)
group alpha13 (s1, s2, s3){
        Man labeled s1
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2
        }


        // Man-a16-Man (Man-a16-Man)
group alpha16 (s1, s2, s3){
        Man labeled s1
        A16 labeled s2 single bond to s1
        Man labeled s3 single bond to s2
        }


group alpha16alpha16 (s1, s2, s3, s4, s5){
        Man labeled s1 {connected to 1 $ with any bond}
        A16 labeled s2 single bond to s1
        Man labeled s3 single bond to s2
        A16 labeled s4 single bond to s3
        Man labeled s5 single bond to s4
}


group alpha16alpha16tri (s1, s2, s3, s4, s5){
        Man labeled s1
        A16 labeled s2 single bond to s1
        Man labeled s3 single bond to s2
        A16 labeled s4 single bond to s3
        Man labeled s5 single bond to s4
}


group alpha13alpha16 (s1, s2, s3, s4, s5){
        Man labeled s1 {connected to 1 $ with any bond}
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2
        A16 labeled s4 single bond to s3
        Man labeled s5 single bond to s4
```

```
}


        //Man-b12-GlcNAc (Man-b12-GlcNAc)
group beta12 (s1, s2, s3){
        GlcNAc labeled s1
        B12 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <= 3 $ with any bond}
        }


        // Man-b14-GlcNAc (Man-b14-GlcNAc)
group beta14 (s1, s2, s3){
        GlcNAc labeled s1
        B14 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <= 3 $ with any bond}
        }


        // Man-b16-GlcNAc (Man-b16-GlcNAc)
group beta16 (s1, s2, s3){
        GlcNAc labeled s1
        B16 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <= 3 $ with any bond}
        }


        // Man-b14m-GlcNAc (Man-b14m-GlcNAc) - Bisecting GlcNAc
        //G Does this rule in reality refers to bisecting one? it seems to wide
group beta14m (s1, s2, s3){
        GlcNAc labeled s1
        B14 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to 4 $ with any bond}
        }


        // GlcNAc-b14cb-Gal (GlcNAc-b14cb-Gal) Gal capping
group beta14cb (s1, s2, s3){
        Gal labeled s1
        B14 labeled s2 single bond to s1
        GlcNAc labeled s3 single bond to s2
        }


        // GlcNAc-a16t-Fuc (GlcNAc-a16t-Fuc) Fucosylation
```

```
group alpha16t (s1, s2, s3){
       Fuc labeled s1
       A16 labeled s2 single bond to s1
       GlcNAc labeled s3 single bond to s2
       }


       // Gal-a23-Sia (Gal-a23-Sia) Sialylation (alpha2,3 linkage)
group alpha23 (s1, s2, s3){
       Sia labeled s1
       A23 labeled s2 single bond to s1
       Gal labeled s3 single bond to s2
       }


group freeGlcNAc (s1){
       GlcNAc labeled s1 {connected to 1 $ with any bond}
}


group tetraAntennary (s1, s2, s3, s4, s5)
{
       Man labeled s1
       A13 labeled s2 single bond to s1
       Man labeled s3 single bond to s2 {connected to 3 $ with single bond}
       A16 labeled s4 single bond to s1
       Man labeled s5 single bond to s4 {connected to 3 $ with single bond}
}


group triAntennary1 (s1, s2, s3, s4, s5) // plus no group alpha16alpha16 //two
    branches in upper side
{
       Man labeled s1
       A13 labeled s2 single bond to s1
       Man labeled s3 single bond to s2 {connected to 2 $ with single bond}
       A16 labeled s4 single bond to s1
       Man labeled s5 single bond to s4 {connected to 3 $ with single bond}
}


group triAntennary2 (s1, s2, s3, s4, s5) // plus no group alpha16alpha16 //two
    branches in lower side
{
```

```
        Man labeled s1
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to 3 $ with single bond}
        A16 labeled s4 single bond to s1
        Man labeled s5 single bond to s4 {connected to 2 $ with single bond}
}


group biAntennary (s1, s2, s3, s4, s5, s6)
{
        Man labeled s1
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to 2 $ with single bond}
        B12 labeled s6 single bond to s3
        A16 labeled s4 single bond to s1
        Man labeled s5 single bond to s4 {connected to 2 $ with single bond}
}


group hybridA13Arm (s1, s2, s3, s4)
{
        Man labeled s1
        // alpha1,3-arm
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to 2 $ with single bond}
        B12 labeled s4 single bond to s3
}


group nonhybridA13Arm (s1, s2, s3, s4)
{
        Man labeled s1
        // alpha1,3-arm
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to 2 $ with single bond}
        B14 labeled s4 single bond to s3
}


group nonhybridA16Arm1 (s1, s2, s3, s4, s5)
{
        Man labeled s1
        // alpha1,6-arm
```

```
          A16 labeled s2 single bond to s1
          Man labeled s3 single bond to s2
          B12 labeled s4 single bond to s3
          GlcNAc labeled s5 single bond to s4
}


group nonhybridA16Arm2 (s1, s2, s3, s4, s5)
{
          Man labeled s1
          // alpha1,6-arm
          A16 labeled s2 single bond to s1
          Man labeled s3 single bond to s2
          B16 labeled s4 single bond to s3
          GlcNAc labeled s5 single bond to s4
}


//new rule

// Rule for each enzyme: Find reactant r1 and "break bond, establish a new bond
      if needed"
//G MAN I
rule enzyme1{
          reactant r1{
                  Man labeled s1 {connected to 1 $ with any bond} //primary Man
                  A12 labeled s2 single bond to s1
                  Man labeled s3 single bond to s2 {connected to 2 $ with any bond
                      }
                  }
          break bond (s2, s3)
}
//G MAN IIa - Perhaps we will need to rewrite this rule in order to make the
    mechanism remove two mannoses (alpha13 and 16 at once and not in two
    separeted steps)
rule enzyme2{
          reactant r1{
          Man labeled s1 {connected to 1 $ with any bond} //primary Man
          A13 labeled s2 single bond to s1
          Man labeled s3 single bond to s2 {!connected to B14 with any bond}
          A16 labeled s4 single bond to s3
```

```
        Man labeled s5 single bond to s4 {connected to 1 $ with any bond}
        }
        constraints{
        r1 contains = 1 of group beta12
        }
        break bond (s2, s3)
        break bond (s4, s3)
}


//addition of GlcNAc
//G GnT I
rule enzyme3{
        reactant r1{
        Man labeled s1 {connected to 1 $ with any bond} //primary Man
        A13 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to B14 with any bond}
        }
        reactant r2{
        B12 labeled s4 {connected to 1 $ with any bond}
        GlcNAc labeled s5 single bond to s4
        }
        constraints{
        r1 contains = 2 of group alpha13 && r1 contains = 2 of group alpha16 &&
            ! r1 contains >= 1 of group alpha12 && ! r1 contains >= 1 of group
           alpha16t && ! r1 contains >=1 of group beta14cb && ! r1 contains >=1
            of group alpha23
        }
        form bond (s4, s1)
}
//G GnT II
rule enzyme4{
        reactant r1{
        Man labeled s1 {connected to 1 $ with any bond}
        A16 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to B14 with single bond}
        }
        reactant r2{
        B12 labeled s4 {connected to 1 $ with any bond}
        GlcNAc labeled s5 single bond to s4
```

```
        }
        constraints{
        ! r1 contains >= 1 of group beta14m && ! r1 contains >=1 of group
            beta14cb && ! r1 contains >=1 of group alpha23}
        form bond (s4, s1)
}
//G GnT III
rule enzyme5{
        reactant r1{
        Man labeled s1 {connected to 3 $ with any bond}
        B14 labeled s2 single bond to s1
        GlcNAc labeled s3 single bond to s2
        B14 labeled s4 single bond to s3
        GlcNAc labeled s5 single bond to s4
        }
        reactant r2{
        B14 labeled s6 {connected to 1 $ with any bond}
        GlcNAc labeled s7 single bond to s6
        }
        constraints{
        r1 contains >=1 of group beta12 && ! r1 contains >=1 of group beta14cb
            && ! r1 contains >=1 of group alpha23}
        form bond (s6, s1)
}
//G GnT IV
rule enzyme6{
        reactant r1{
        GlcNAc labeled s1 {connected to 1 $ with any bond} // This contain the
            restriction for galactose ("Prior addition of Gal precludes activity
             ")
        B12 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <=2 $ with any bond}
        A13 labeled s4 single bond to s3
        }
        reactant r2{
        B14 labeled s5 {connected to 1 $ with any bond}
        GlcNAc labeled s6 single bond to s5
        }
        constraints{
```

```
        ! r1 contains >=1 of group beta14m && r1 contains >= 1 of group
           alpha16t}
        form bond (s5, s3)
}
//G GnT V
rule enzyme7{
        reactant r1{
        GlcNAc labeled s1 {connected to 1 $ with any bond} // This contain the
           restriction for galactose ("Prior addition of Gal precludes activity
            ")
        B12 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <=2 $ with any bond}
        A16 labeled s4 single bond to s3
        }
        reactant r2{
        B16 labeled s5 {connected to 1 $ with any bond}
        GlcNAc labeled s6 single bond to s5
        }
        constraints{
        ! r1 contains >=1 of group beta14m && r1 contains >= 1 of group
           alpha16t}
        form bond (s5, s3)
}
//G FucT
rule enzyme8{
        reactant r1{
        N labeled s1
        GlcNAc labeled s2 single bond to s1 {connected to <= 2 $ with any bond}
        B14 labeled s3 single bond to s2
        GlcNAc labeled s4 single bond to s3
        }
        reactant r2{
        A16 labeled s5 {connected to 1 $ with any bond}
        Fuc labeled s6 single bond to s5
        }
        constraints{
```

```
        r1 contains <=2 of group beta12 && r1 contains <=1 of group alpha13 &&
            r1 contains <=1 of group alpha16 && ! r1 contains >=1 of group
            beta14m && ! r1 contains >=1 of group alpha12 && ! r1 contains >=2
            of group beta14cb
        }
        form bond (s5, s2)
}
//G GalT a
rule enzyme9a{
        reactant r1{
        GlcNAc labeled s1 {connected to 1 $ with any bond}
        B12 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <=3 $ with any bond}
        }
        reactant r2{
        B14 labeled s4 {connected to 1 $ with any bond}
        Gal labeled s5 single bond to s4
        }
    constraints{
    r1 contains < 1 of group alpha16alpha16 && r1 contains < 1 of group
        alpha13alpha16}
        form bond (s4, s1)
}
//G GalT b
rule enzyme9b{
        reactant r1{
        GlcNAc labeled s1 {connected to 1 $ with any bond}
        B14 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <=3 $ with any bond}
        }
        reactant r2{
        B14 labeled s4 {connected to 1 $ with any bond}
        Gal labeled s5 single bond to s4
        }
        constraints{
    r1 contains < 1 of group alpha16alpha16 && r1 contains < 1 of group
        alpha13alpha16}
        form bond (s4, s1)
}
```

```
//G GalT c
rule enzyme9c{
        reactant r1{
        GlcNAc labeled s1 {connected to 1 $ with any bond}
        B16 labeled s2 single bond to s1
        Man labeled s3 single bond to s2 {connected to <=3 $ with any bond}
        }
        reactant r2{
        B14 labeled s4 {connected to 1 $ with any bond}
        Gal labeled s5 single bond to s4
        }
        constraints{
    r1 contains < 1 of group alpha16alpha16 && r1 contains < 1 of group
        alpha13alpha16}
        form bond (s4, s1)
}


//G SiaT
rule enzyme10a{
        reactant r1{
        Gal labeled s1 {connected to 1 $ with any bond}
        B14 labeled s2 single bond to s1
        GlcNAc labeled s3 single bond to s2
        }
        reactant r2{
        A23 labeled s4 {connected to 1 $ with any bond}
        Sia labeled s5 single bond to s4
        }
        form bond (s4, s1)
}

find all gtetraAntennary{
        gtetraAntennary contains group tetraAntennary
} store in "tetraAntennary.txt"

find all gtriAntennary{
        (gtriAntennary contains group triAntennary1 && !gtriAntennary contains
            group alpha16alpha16tri) || (gtriAntennary contains group
            triAntennary2 && !gtriAntennary contains group alpha16alpha16tri)
```

```
} store in "triAntennary.txt"

find all gbiAntennary{
       gbiAntennary contains group biAntennary
} store in "biAntennary.txt"

find all ghybrid{
       !(ghybrid contains group nonhybridA16Arm1 || ghybrid contains group
           nonhybridA16Arm2) && (ghybrid contains group hybridA13Arm)
} store in "hybrid.txt"

find all ghighmannose{
       ghighmannose.size > 2 && !(ghighmannose contains group hybridA13Arm ||
           ghighmannose contains group nonhybridA13Arm || ghighmannose contains
            group nonhybridA16Arm1 || ghighmannose contains group
           nonhybridA16Arm2)
   ghighmannose contains <= 2 of group freeGlcNAc
} store in "highmannose.txt"

find all speciestogether{
       speciestogether.size > 2
} store in "speciestogether.txt"
```

C   Inputs into RING for studying Olefin Interconversion reaction system

```
input reactant "C=CC"
input reactant "[{Zeo}H]"//representing a bronsted acid
input reactant "[{HTA}H]"
input reactant "N#N"
input reactant "CC(C)C"
input temperature 723 K
preferred units mmol hr cc

define composite atom Zeo (heterogeneous site)
define composite atom HTA

import "GroupAdditivity2.txt"
import "GroupCorrections2.txt"
import "MTH_SPP_PE.txt"

store lumped network in "storedRxnsLumpedNewModel.txt", species in "
    storedSpeciesLumpedNewModel.txt"

group CdoubleC (c1,c2){
    C labeled c1
    C labeled c2 double bond to c1}

group CanyC (c1,c2){
      nonringatom C labeled c1
      nonringatom C labeled c2 any bond to c1
}

group prisecdouble (c1, c2){
      C labeled c1 {connected to 1 C with double bond}
      C labeled c2 double bond to c1 {connected to 2 C with any bond}
}

group secsecdouble (c1, c2){
      C labeled c1 {connected to 2 C with any bond}
```

```
        C labeled c2 double bond to c1 {connected to 2 C with any bond}
        }


group primterdouble (c1, c2){
      C labeled c1 {connected to 1 C with double bond}
      C labeled c2 double bond to c1 {connected to 3 C with any bond}
}


group sectertdouble (c1, c2){
      C labeled c1 {connected to 2 C with any bond}
      C labeled c2 double bond to c1 {connected to 3 C with any bond}
}


group terterdouble (c1, c2){
      C labeled c1 {connected to 3 C with any bond}
      C labeled c2 double bond to c1 {connected to 3 C with any bond}
}


define characteristic allylicMol on Molecule{
    fragment f{
        C labeled c1 {! connected to >1 C with any bond}
        C labeled c2 double bond to c1
        C labeled c3 single bond to c2}
    Molecule contains >=1 of f
}


define characteristic primaryCarbeniumIon on Mol{
      fragment f{
              C labeled c1 {connected to <2 C with single bond}
              Zeo labeled z1 single bond to c1
              }
       Mol contains 1 of f
}


define characteristic secondaryCarbeniumIon on Mol{
      fragment f{
              C labeled c1 {connected to 2 C with single bond}
              Zeo labeled z1 single bond to c1
              }
```

```
      Mol contains 1 of f
}


define characteristic tertiaryCarbeniumIon on Mol{
      fragment f{
            C labeled c1 {connected to 3 C with single bond}
            Zeo labeled z1 single bond to c1
            }
      Mol contains 1 of f
}


define characteristic allylicCarbeniumIon on Mol{
      fragment f{
            C labeled c1 {connected to 1 group CdoubleC with single bond}
            Zeo labeled z1 single bond to c1
            }
      Mol contains 1 of f
}


define characteristic alcoholMol on Molecule{
    fragment f{
        C labeled c1
        O labeled o1 single bond to c1
        H labeled h1 single bond to o1
        }
        Molecule contains >=1 of f
}


define characteristic etherMol on Molecule{
    fragment f{
        C labeled c1
        O labeled o1 single bond to c1
        C labeled c2 single bond to o1}

        Molecule contains >=1 of f
}


define characteristic adsorbed on Molecule{
   fragment f{
```

```
      Zeo labeled z1 {connected to >=1 C with any bond}
   }
   Molecule contains >=1 of f
}


define characteristic gasPhase on Molecule {
   ! Molecule is adsorbed
}


define characteristic paraffinicMol on Molecule{
    ! Molecule is aromatic
        Molecule is gasPhase
    ! Molecule contains >=1 of group CdoubleC
}


define characteristic olefinicMol on Molecule{
    ! Molecule is aromatic
        Molecule is gasPhase
    Molecule contains 1 of group CdoubleC
}


define characteristic surfaceOlefinicMol on Molecule{
        ! Molecule is oxygenate
    ! Molecule is aromatic
        Molecule is adsorbed
    Molecule contains >=1 of group CdoubleC
}


define characteristic surfacewithoutOlefinicMol on Molecule{
        ! Molecule is oxygenate
    ! Molecule is aromatic
        Molecule is adsorbed
    Molecule contains <1 of group CdoubleC
}


define characteristic dieneMol on Molecule{
    ! Molecule is aromatic
        Molecule is gasPhase
    Molecule contains >=2 of group CdoubleC
```

```
}

define characteristic dienesurfaceMol on Molecule{
        ! Molecule is oxygenate
        ! Molecule is aromatic
        Molecule contains 2 of group CdoubleC
}

define characteristic enesurfaceMol on Molecule{
        ! Molecule is oxygenate
        ! Molecule is aromatic
        Molecule contains 1 of group CdoubleC
}

define characteristic branchedSpecies on Molecule {
   fragment f{
      C labeled c1 {connected to >2 C with any bond}
   }
   Molecule contains >=1 of f
}

define characteristic methylbranch on Molecule {
        fragment f{
                C labeled c1 {connected to 3 H with single bond}
                C labeled c2 single bond to c1 {connected to >2 C with any bond}
        }
        Molecule contains >=1 of f
}

define characteristic dimethylbranch on Molecule {
        fragment f{
                C labeled c1 {connected to 3 H with single bond}
                C labeled c2 single bond to c1 {connected to >2 C with any bond}
        }
        Molecule contains 2 of f
}

define characteristic SurfaceMethyl on Molecule {
   fragment f{
```

```
      C labeled c1
          Zeo labeled z1 single bond to c1}
   Molecule.size=2 && Molecule contains >=1 of f
}


//branching beta
define characteristic betabranching on Molecule{
      fragment f{
              C labeled c1
              Zeo labeled z1 single bond to c1
              C labeled c2 single bond to c1 {connected to 3 C with any bond}
      }
      Molecule contains >=1 of f && Molecule.size >= 5
}


global constraints on Molecule
{
   //declaration of a fragment named 'a'
   fragment a
   {
      C labeled 1 {connected to 1 Zeo with single bond}
      C labeled 2 double bond to 1
   }
   ! Molecule contains a

      fragment carb{
         C labeled c1
      }

      fragment QuartenaryCarbon{
         C labeled c1 {connected to 4 C with single bond}
      }

      fragment AlkylFragments{
         nonringatom C labeled c1 {connected to >=1 nonringatom C with any
             bond}
         ringatom C labeled c2 any bond to c1
      }
```

```
    fragment exocyclicCarbon{
       nonringatom C labeled c1 {!connected to >=1 ringatom C with any bond}
    }
    fragment exocyclicDoubleBond{
       nonringatom C labeled c1
       ringatom C labeled c2 double bond to c1
    }

    (Molecule is cyclic && Molecule.minringsize = 6 && Molecule contains
        <=13 of carb && Molecule contains <=1 of QuartenaryCarbon &&
        Molecule contains < 1 of AlkylFragments && Molecule contains < 1 of
        exocyclicCarbon) || (! Molecule is cyclic && Molecule contains <=9
        of carb)
    (! Molecule contains exocyclicDoubleBond)
    //cannot have a C=C=C or a C=C=O
fragment b
{
    C labeled c1
    C labeled c2 double bond to c1
    X labeled x1 double bond to c2 // note X represents a heavy atom
}
! Molecule contains b

    fragment d{
    C labeled c1
    C labeled c2 double bond to c1
    C labeled c3 single bond to c2
    O labeled o1 single bond to c3
    }
    ! Molecule contains d

    fragment e{
    C labeled c1
    O labeled o1 single bond to c1
    H labeled h1 single bond to o1
    C labeled c2 single bond to c1
    C labeled c3 single bond to c2
    C labeled c4 single bond to c2
    C labeled c5 single bond to c2
```

```
        }
        ! Molecule contains e

    (! Molecule contains >=3 of group CdoubleC) || (Molecule is cyclic)
}


//adsorption of olefins
rule OleAds{
gasPhase reactant r1
{
    C labeled c1
    C labeled c2 double bond to c1
}
reactant r2
{
   Zeo labeled z1 {! connected to >=1 C with any bond}
   H labeled h1 single bond to z1
 }
constraints { ! r1 is aromatic}
form bond (c1,h1)
decrease bond order (c1,c2)
form bond (c2,z1)
break bond (z1,h1)
}


//adsorption of aromatics
rule AromAds{
aromatic reactant r1
{
   c labeled c1
   c labeled c2 aromatic bond to c1
}
reactant r2
{
   Zeo labeled z1 {! connected to >=1 C with any bond}
   H labeled h1 single bond to z1
 }
form bond (c1,h1)
break bond (z1,h1)
```

```
modify atomtype (c1,C)
modify atomtype (c2,C)
modify bond (c1,c2,single)
form bond (c2,z1)
}


//desorption of carbenium ions to form olefins/aromatics
rule Desorption{
adsorbed reactant r1
{
    C labeled c1
    Zeo labeled z1 single bond to c1
    C labeled c2 single bond to c1 {! connected to any atom with double bond}
    H labeled h1 single bond to c2
}
break bond (c2,h1)
increase bond order (c1, c2)
break bond (c1,z1)
form bond (h1,z1)
product constraints on mol
{
   !(mol is cyclic && mol.minringsize < 6)
}
}


//oligomerization
rule Oligo{
gasPhase reactant r1
{
    C labeled c1
    C labeled c2 double bond to c1
}
linear reactant r2
{
    C labeled c3
    Zeo labeled z1 single bond to c3
}
constraints {
   (! r1 is cyclic && r1.size + r2.size <=11 && r2.size >=3 && r1.size >=2)
```

```
}
form bond (c1,c3)
decrease bond order (c1,c2)
break bond (c3,z1)
form bond (c2,z1)
}

//beta scission - 1 where the final alkoxide is not methyl
rule BetaSci1{
linear reactant r1
{
    C labeled c1
    Zeo labeled z1 single bond to c1
    C labeled c2 single bond to c1
    nonringatom C labeled c3 single bond to c2 {connected to >1 C with any bond
        }
}
break bond (c2,c3)
increase bond order (c1,c2)
break bond (c1,z1)
form bond (c3,z1)
}

//Cyclization with internal hydride shifts 1,6 cyclization
rule Cyclization{
linear reactant r1{
    nonringatom C labeled c1
    Zeo labeled z1 single bond to c1
    C labeled c2 single bond to c1
    C labeled c3 any bond to c2
    C labeled c4 any bond to c3
    C labeled c5 any bond to c4
    C labeled c6 double bond to c5}
        constraints{
                r1.size >= 7
        }
form bond (c1,c6)
decrease bond order (c5,c6)
break bond (c1,z1)
```

```
form bond (c5,z1)
}

rule Hshift {
linear reactant r1{
   C labeled c1 {connected to >=1 C with single bond}
   Zeo labeled z1 single bond to c1
   C labeled c2 single bond to c1 {connected to <=3 C with single bond}
   H labeled h1 single bond to c2
}
break bond (c1,z1)
break bond (c2,h1)
form bond (c1,h1)
form bond (c2,z1)
}

rule MethylShift {
linear reactant r1{
  nonringatom C labeled c1 {connected to <=3 C with single bond}
  nonringatom C labeled c2 single bond to c1 {connected to >=2 C with single
      bond}
  nonringatom C labeled c3 single bond to c2 {connected to 1 C with single bond
      }
  Zeo labeled z1 single bond to c1
}
break bond (z1,c1)
break bond (c2,c3)
form bond (c1,c3)
form bond (c2,z1)
}

rule RingMethylShift {
reactant r1{
  ringatom C labeled c1
  nonringatom C labeled c2 single bond to c1
  ringatom C labeled c3 single bond to c1
  Zeo labeled z1 single bond to c3
}
constraints {
```

```
    r1.minringsize = 6
    fragment f{
      ringatom C labeled c1
      ringatom C labeled c2 double ring bond to c1
    }
    r1 contains >=2 of f
}
break bond (c2,c1)
form bond (c3,c2)
break bond (c3,z1)
form bond (c1,z1)
}

rule RingAllylShift {
reactant r1 {
  ringatom C labeled c1
  Zeo labeled z1 single bond to c1
  ringatom C labeled c2 single bond to c1
  ringatom C labeled c3 double bond to c2
}
constraints {
    fragment f{
          ringatom C labeled c1
          ringatom C labeled c2 double ring bond to c1
        }
        r1 contains >=2 of f
        r1.minringsize >= 6
}
break bond (c1,z1)
form bond (c3,z1)
decrease bond order (c2,c3)
increase bond order (c1,c2)
}

rule ReconstructHydTransfer1newnoncyclic{
adsorbed reactant r1{
    C labeled c1
    Zeo labeled z1 single bond to c1}
linear reactant r2{
```

```
    C labeled cr1
    H labeled h1 single bond to cr1}
constraints {
        r2.size < r1.size && r1.size + r2.size <= 19
        (r2 is paraffinicMol && r2.size >= 2)||(r2 is olefinicMol && r2.size
            >=4)||(r2 is dieneMol && r2.size >= 6)
        ! r1 is cyclic && r1.size >= 3 && r1.size <= 10} //while reconstructing
            , we put a total size constraint of 12 and r2 can at min be ethane
break bond (cr1,h1)
form bond (h1,c1)
break bond (c1,z1)
form bond (cr1,z1)
}

rule ReconstructHydTransfer2newnoncyclic{
adsorbed reactant r1{
    C labeled c1
    Zeo labeled z1 single bond to c1}
linear reactant r2{
    C labeled cr1
    H labeled h1 single bond to cr1}
constraints {
        r2.size >= r1.size && r1.size + r2.size <= 19
        (r2 is paraffinicMol && r2.size >= 2)||(r2 is olefinicMol && r2.size
            >=4)||(r2 is dieneMol && r2.size >= 6)
        ! r1 is cyclic && r1.size >= 3 && r1.size <= 10} //while reconstructing
            , we put a total size constraint of 12 and r2 can at min be ethane
break bond (cr1,h1)
form bond (h1,c1)
break bond (c1,z1)
form bond (cr1,z1)
}

//rule Hydride transfer from {HTA}H //Here the adsorbed species is cyclic
    species
rule ReconstructHydTransfer2newcyclic{
adsorbed reactant r1{
    C labeled cr1
    Zeo labeled z1 single bond to cr1}
```

```
linear reactant r2{
    C labeled c1
    H labeled h1 single bond to c1}
constraints {
        ! r2 is cyclic && r1.size + r2.size <= 19
        r1 is cyclic && r1 contains <=2 of group CdoubleC && r1.minringsize>= 6
            && r1.size <= 10
        (r2 is paraffinicMol && r2.size >= 2)||(r2 is olefinicMol && r2.size
            >=4)||(r2 is dieneMol && r2.size >= 6)
        } //while reconstructing, we put a total size constraint of 12 and r2
            can at min be ethane
break bond (c1,h1)
form bond (h1,cr1)
break bond (cr1,z1)
form bond (c1,z1)
}


rule ReconstructHydTransfer2{
adsorbed reactant r1{
    C labeled cr1
    Zeo labeled z1 single bond to cr1}
gasPhase cyclic reactant r2{
    ringatom C labeled c1
    ringatom C labeled c2 single bond to c1
    H labeled h1 single bond to c1
    }
constraints {
        r1.size >= 3 && r1.size <= 10
    ! r1 is cyclic && r1.size + r2.size <= 19 && r2.minringsize>=6
    r2 contains <=2 of group CdoubleC && r2.size <= 12
}
break bond (c1,h1)
form bond (cr1,h1)
break bond (cr1,z1)
form bond (c1,z1)
}
```

D Experimental Procedure for Olefin Interconversion Work

D.1 Experimental Procedure

SPP (Self-Pillared Pentasil) MFI samples were synthesized by the methods reported previously in [165]. The samples were pressed into pellets, crushed, and sieved between 40- and 80-mesh sieves to obtain particles sized between 180 and 425 um. Typically quartz sand (Agros Organics) was loaded with the sample to dilute the bed (18-20% catalyst weight) to maintain isothermal conditions. The quartz sand was first washed with nitric acid (1 M) rinsed several times with deionized water and calcined at 1273 K for 4 h.

Flow of reactants was controlled by mass flow controllers (Brooks Instrument series) through a stainless-steel packed-bed reactor (0.25 in. O.D., 0.125 in. I.D) with a concentric thermowell (0.0625 in O.D., 0.0485 in I.D.) was used for the catalytic conversion of propylene. The reactor temperature was monitored by a K-type thermocouple (Omega Engineering) inserted into the thermowell and regulated using a heating coil (ARi Industries Inc., AeroRod heating assembly) and a Watlow 96 series temperature controller. Prior to every reaction, the catalyst was pretreated in situ in 1.67 cm^3s^{-1} zero grade air by heating from ambient to 823 K (1 K min^{-1}) and holding at 823 K for 4 h before cooling to the reaction temperature. The reactor was then flushed with helium (Minneapolis Oxygen, 99.995% purity) for 30 minutes prior to reaction. The reactant stream consisted of propene (Praxair, 50% propene, 50% argon / Matheson Tri-gas 99.95% purity), argon (Matheson Tri-gas 99.95% purity) that was used as an internal standard for chromatographic analysis, and helium (Minneapolis Oxygen, 99.995% purity). A mixture of methane and argon (Airgas, 10% methane, 90% argon) was used as an internal standard. Reaction effluents were analyzed using a gas chromatograph (Agilent 7890) with a 50 m x 320 um x 0.52 um dimethylpolysiloxane J&W HP-1 column connected to a flame ionization detector in parallel to a 50 m x 320 um J&W GS-GasPro column connected to a thermal conductivity detector. The product distribution shown in Section 6 includes hydrocarbon species above C_9 that were not separately identified but instead classified as C_9 Hydrocarbons.