Advanced Solution Methods for Microkinetic Models of Catalytic Reactions: A Methanol Synthesis Case Study

Patricia Rubert-Nason, Manos Mavrikakis, and Christos T. Maravelias
Dept. of Chemical and Biological Engineering, University of Wisconsin–Madison, Madison, WI 53706

Lars C. Grabow
Dept. of Chemical and Biomolecular Engineering, University of Houston, Houston, TX 77204

Lorenz T. Biegler
Dept. of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213

DOI 10.1002/aic.14322
Published online December 31, 2013 in Wiley Online Library (wileyonlinelibrary.com)

Microkinetic models, combined with experimentally measured reaction rates and orders, play a key role in elucidating detailed reaction mechanisms in heterogeneous catalysis and have typically been solved as systems of ordinary differential equations. In this work, we demonstrate a new approach to fitting those models to experimental data. For the specific example treated here, by reformulating a typical microkinetic model for a continuous stirred tank reactor to a system of nonlinear equations, we achieved a 1000-fold increase in solution speed. The reduced computational cost allows a more systematic search of the parameter space, leading to better fits to the available experimental data. We applied this approach to the problem of methanol synthesis by CO/CO_2 hydrogenation over a supported-Cu catalyst, an important catalytic reaction with a large industrial interest and potential for large-scale CO_2 chemical fixation. © 2013 American Institute of Chemical Engineers *AIChE J*, 60: 1336–1346, 2014
Keywords: density functional theory, nonlinear programming, parameter estimation, parallel computing

Introduction

Catalysts play a vital role in chemical industry. Over 90% of industrial chemical processes are catalyzed, and catalysts allow chemicals to be produced under less stringent conditions, thus reducing energy consumption, an essential goal in today's environmentally conscious and highly regulated environment.[1] However, for many reactions, the catalyzed reaction mechanism is still poorly understood, hindering the ability to develop improved catalysts.

Density functional theory (DFT) provides an important computational methodology for understanding catalytic reactions and for the design of new catalysts. DFT uses the principles of quantum mechanics to predict properties of materials at the atomic and molecular scale.[2] Specifically, an approximation of the Schrödinger equation is solved and can be used to predict the binding sites and binding energies of atoms and molecules on catalytically active surfaces and the activation energy barriers to various potential elementary reaction steps on the surface. DFT has provided us with unprecedented insights into the detailed reaction mechanism of various catalytic processes.[2]

To use the information derived from DFT most effectively, we need to translate our new understanding of the elementary steps into predictions of macroscopic outcomes such as the rate of production of products and byproducts. This can be accomplished using a microkinetic model.[3,4] Microkinetic models provide a bridge between the elementary steps that occur at the molecular scale and production rates at the macroscopic scale. They depend on the assumption that adsorbates are evenly distributed across the surface, that is, the mean field approximation.[4–6] Microkinetic modeling predicts reaction rates and surface coverages at reaction conditions based on a set of elementary steps and their rate constants.[6] This can lead us to a better understanding of what is happening on the surface of the catalyst. *A priori* assumptions about surface coverages, rate limiting steps, or quasi-equilibration are avoided within the framework of these models. This allows the model to be very general and predictive across a wide range of reaction conditions.[6]

Although DFT provides a good starting point for parameter values for microkinetic modeling,[6,7] production rates can be very sensitive to the binding and activation energies (*BE* and *Ea*) of the surface species and elementary steps, respectively.[8] The DFT values are subject to computational errors on the order of 0.1–0.2 eV and may also contain inaccuracies due to the effects of surface coverage or surface reconstruction under the reaction conditions or the selection of a specific surface to model, which may not be a good representation of the active site.[9] In practice, often times one performs detailed DFT calculations on a certain type of active site, which gives binding energies and activation energy barriers that predict, via the microkinetic model, reaction rates orders of magnitude different from the experimentally

Correspondence concerning this article should be addressed to C. T. Maravelias at christos@engr.wisc.edu. or M. Mavrikakis at manos@engr.wisc.edu.

measured ones. This situation points to the likely fact that the model used for the active site to derive the DFT parameters was incorrect. In those cases, having a set of optimized parameters, corresponding to minimizing the error from the experimentally measured quantities might be critical to guiding thinking toward revealing the correct nature of the active site. It is in that sense that improving on the standard manual approach to solving microkinetic models could have significant impact in the understanding of the fundamental mechanism. Therefore, parameter estimation is a necessary component of successful microkinetic modeling.

The parameter estimation problem is shown in Eqs. a–c. It is a nonlinear program (NLP), where we minimize a scalar objective function, $f(x)$, (Eq. a) subject to a set of constraints (Eqs. b and c).[10] Equality constraints are captured in the vector function $g(x)$, Eq. b, and inequality constraints in the vector function $h(x)$, Eq. c

$$\min f(x) \tag{a}$$

$$\text{s.t.} g(x) = 0 \tag{b}$$

$$h(x) \leq 0 \tag{c}$$

The constraints can be satisfied by more than one set of values for the variables. Through the solution of the NLP, we attempt to find variable values which satisfy the constraints and result in the lowest value of the objective function. When the objective function (Eq. a) or the feasible region is nonconvex, there may be more than one local minimum. NLP algorithms may converge to local minima which may be significantly poorer than the global minimum.[11]

For parameter estimation on microkinetic models, our objective is to minimize the difference between our model predictions and experimental data. The primary variables are *BE*, *Ea*, surface coverages, and partial pressures in the gas phase.* Additional variables, such as rate constants and reaction rates, are functions of the primary variables and parameters such as pressure, temperature, and feed composition.

In this work, we demonstrate a new approach to parameter estimation for microkinetic models that addresses the challenges of severe nonlinearity and ill-conditioning of the optimization model. We propose a number of techniques, including scaling, a simulation-based initialization, a nontrivial algorithm that allows us to consider multiple parameters while maintaining the robustness of the method, and a penalty-based relaxation from steady state. To our knowledge, this is the first time that the simultaneous method is adopted, along with the auxiliary techniques, for this type of problem.

We select methanol synthesis as a model reaction network to develop and demonstrate our approach. Worldwide, approximately 30 Mt of methanol are produced annually, primarily by steam reforming of natural gas.[12] Although the majority is used as feedstock for chemical production, it has significant potential as a liquid transportation fuel[13,14] and is a promising candidate for production from CO_2, potentially reducing greenhouse gas emissions.[15,16] However, efficient production from CO_2 in the absence of CO has not been established.[17] Microkinetic modeling has the potential to help us identify new catalysts and improved reaction conditions to produce methanol from effluent streams of CO_2.

*In general, parameters are fixed numbers within a model, whereas variables are allowed to change. *BE* and *Ea* are the parameters of the microkinetic model which are estimated in the parameter estimation problem. In the optimization model which estimates them, they are variables; that is, they are free to vary such that the error between the microkinetic model predictions and the experimental data is minimized. Thus, depending on context, we will refer to *BE* and *Ea* as both parameters and variables.

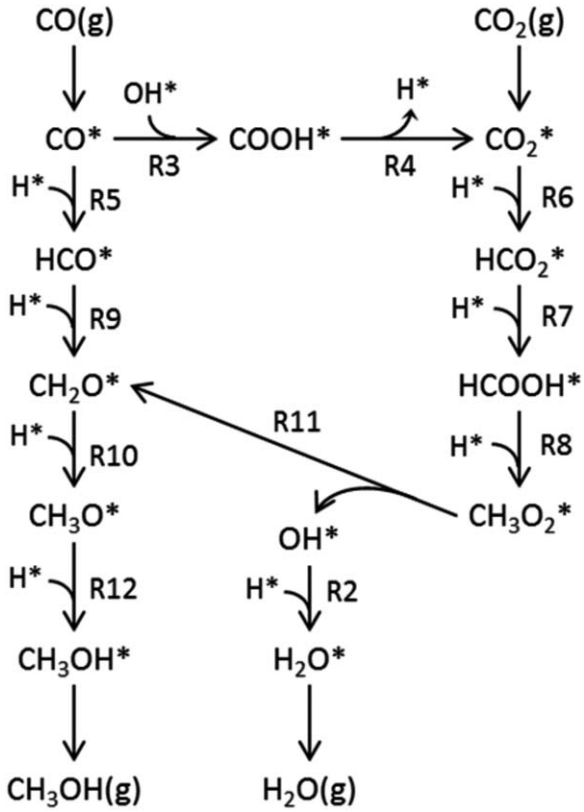

Figure 1. The reaction network for methanol synthesis, the example problem we have used to develop and demonstrate our approach.

Free sites are omitted for clarity. The labels R# refer to the reaction numbers provided in Table 3 and indicate the relevant steps to convert one intermediate to another. All reactions are reversible, but are shown in the active direction for clarity. H_2(g) adsorbs dissociatively onto the surface (not shown).

For the sake of simplicity, and in order to demonstrate our new approach to solving microkinetic models, we consider a methanol synthesis reaction network (Figure 1) with 16 elementary steps and 18 distinct species (three in the gas phase). It includes two main pathways; the synthesis of methanol from CO hydrogenation and the synthesis of methanol and water from CO_2 hydrogenation. It also includes the water gas shift reaction ($CO + H_2O \rightarrow CO_2 + H_2$) which connects the two pathways. This reaction network represents a subset of a more extensive model investigated by Grabow and Mavrikakis[18] using the standard microkinetic modeling approach.

Methods

Basic continuous stirred tank reactor microkinetic model

Our basic microkinetic model is developed for a steady-state stirred tank reactor. The model includes the set $\boldsymbol{I}$ of reactions (indexed over i), the set $\boldsymbol{J}$ of species (indexed over j), and the set $\boldsymbol{C}$ of experimental conditions (indexed over c). Set $\boldsymbol{J}$ includes the subsets $\boldsymbol{G}$ (gas-phase species) and $\boldsymbol{S}$ (surface species), with free sites (*) included in the set $\boldsymbol{S}$. For

each reaction, i, there exist subsets of $\boldsymbol{J}$ which correspond to reactants ($\boldsymbol{R_i}$) and products ($\boldsymbol{P_i}$).

As shown in Eq. 1, for each species (j) and condition (c), the rate of change $\left(\frac{dy_{jc}}{dt}\right)$ is equal to the difference between the flow rates into the system (F_{jc}^{in}) and out of the system (F_{jc}^{out}) plus the net generation (R_{jc}^{net}) for that species, where y_{jc} is the fractional coverage for surface species and partial pressure for gas-phase species. Because we are modeling a continuous stirred tank reactor, which is operated at steady state, the rate of change should be equal to zero

$$F_{jc}^{\text{in}} - F_{jc}^{\text{out}} + R_{jc}^{\text{net}} = \frac{dy_{jc}}{dt} = 0, \forall j, c \tag{1}$$

The net generation and consumption of each species (j) is a function of rates of the reactions (r_{ic}) and the stoichiometric coefficients (v_{ij}) of the species (j) in each reaction (i)

$$R_{jc}^{\text{net}} = \sum_i v_{ij} r_{ic}, \forall j, c \tag{2}$$

There is no flow in or out of the system for surface species (Eq. c)

$$F_{jc}^{\text{in}} = F_{jc}^{\text{out}} = 0, \forall c, j \in \boldsymbol{S} \tag{3}$$

For gas-phase species, F_{jc}^{in} (the flow rate in) is a constant specific to each experimental condition. The flow rate out (F_{jc}^{out}) is proportional to the partial pressure of the species in the gas phase ($y_{jc}, j \in \boldsymbol{G}$) divided by the total pressure (P_c), as shown in Eq. 4. F_{jc}^{out} is equivalent to the turnover frequency (TOF) for species j in condition c (with units of s^{-1})

$$F_{jc}^{\text{out}} = \frac{y_{jc}}{P_c} \sum_{j' \in \text{G}} \left(F_{j'c}^{\text{in}} + R_{j'c}^{\text{net}}\right), \forall c, j \in \boldsymbol{G} \tag{4}$$

All surface sites must be either occupied or free, so the surface coverages ($y_{jc}, j \in \boldsymbol{S}$) must sum to one (Eq. 5)

$$\sum_{j \in S} y_{jc} = 1, \forall c \tag{5}$$

The reaction rates (r_{ic}) are calculated using the forward (kf_{ic}) and reverse (kr_{ic}) rate constant, the concentrations (y_{jc}), and stoichiometric coefficients (v_{ij}) of the species involved in the reaction (Eq. 6)

$$r_{ic} = kf_{ic} \prod_{j \in R_i} y_{jc}^{|v_{ij}|} - kr_{ic} \prod_{j \in P_i} y_{jc}^{|v_{ij}|}, \forall i, c \tag{6}$$

where $\boldsymbol{R_i} \subset \boldsymbol{J}$ and $\boldsymbol{P_i} \subset \boldsymbol{J}$ are the sets of reactants and products for reaction i, respectively.

Equations 7–9 calculate the rate constants for all reactions. The forward rate constant (kf_{ic}) is calculated using the Arrhenius expression as a function of the pre-exponential factor (A_i) and activation energy (Ea_i) of the reaction (Eq. 7). The equilibrium rate constant (Keq_{ic}) is a function of the enthalpy (ΔH_{ic}) and entropy (ΔS_{ic}) of the reaction (Eq. 8). The reverse rate constant (kr_{ic}) is thermodynamically consistently calculated as the ratio of kf_{ic} and Keq_{ic} (Eq. 9). The rate constants are all dependent on temperature (T_c) and include the gas constant (R)

$$kf_{ic} = A_i \exp\left(-\frac{Ea_i}{RT_c}\right), \forall i, c \tag{7}$$

$$Keq_{ic} = \exp\left(-\frac{\Delta H_{ic}}{RT_c} + \frac{\Delta S_{ic}}{R}\right), \forall i, c \tag{8}$$

$$kr_{ic} = \frac{kf_{ic}}{Keq_{ic}}, \forall i, c \tag{9}$$

The enthalpy change of the reaction (ΔH_{ic}) is a function of the heats of formation (ΔHf_j) of the species in the gas phase and their binding energies (BE_j)

$$\Delta H_{ic} = \sum_j v_{ij} (\Delta Hf_j + BE_j), \forall i, c \tag{10}$$

The enthalpy may also include a correction for the temperature. To be thermodynamically consistent, the activation energy for each reaction must be greater than the enthalpy change of that reaction (Eq. 11). In absence of this constraint, it would be possible for an endothermic reaction to be represented as nonactivated, a scenario which is not physically possible

$$Ea_i \geq \Delta H_{ic}, \forall i, c \tag{11}$$

Likewise, all activation energies must be greater than zero and all binding energies less than zero

$$Ea_i \geq 0, \forall i \tag{12}$$

$$BE_j \leq 0, j \in \boldsymbol{S} \tag{13}$$

The parameters that we are fitting to the experimental data are BE_j and Ea_i. This gives a total of 26 fitted parameters in this work. Other parameters (e.g., A_i, ΔS_{ic}) could also be fit if desired.

Ordinary differential equation formulation

Because microkinetic models are nonlinear and poorly conditioned they can be quite difficult to solve. Typically, they have been formulated as systems of ordinary differential equations (ODEs).[3,4,19–25] An initial guess is made for the surface coverage which is consistent with Eq. 5 and Eq. 5 is removed from the system, since as long as the initial point satisfies Eq. 5, so will every future time point. The initial guess will not conform to the constraint that Eq. 1 be equal to zero. The system of ODEs is integrated from the initial guess to a steady state solution, which then satisfies Eq. 1.

This approach is simple to set up and can be run in readily available software such as Matlab. However, the ODE version of the model is computationally intensive to solve. Furthermore, the ODE-solver operates as a black box (with respect to the optimization algorithm), making the gradient of the objective function with respect to the parameters difficult to obtain. A common solution is to use the finite difference approximation to calculate the gradient. For n parameters this requires $n+1$ solutions of the system of ODEs, a significant computational expense. Furthermore, the error in the solution to the system of ODEs compounds with the finite difference error and makes it very difficult to accurately calculate the gradient.[26] This significantly impairs the ability of most optimization algorithms to solve the problem effectively and results in the need for iterative adjustment of the parameter values by the user. Consequently, it requires considerable physical insight to obtain good fits to the experimental data.

There are more sophisticated approaches to approximating the gradient of a system of ODEs, such as adjoint equations[27–30] or direct sensitivity equations,[27,30] which give more accurate results. However, these still require multiple solutions of the system of ODEs at each iteration of the

optimization and they are more difficult to implement. Overall, parameter estimation with the ODE model is expensive in both computational and user time. However, we are primarily interested in the steady state solution rather than the transient behavior. Therefore, we can set the rate of change for all species to zero and solve the NLP instead (see Eq. 1).

NLP formulation

There are several advantages to solving the model (Eqs. 1–13) as a system of nonlinear equations. First, solving a system of nonlinear equations is generally more computationally efficient than solving an equivalent system of ODEs. Second, when the problem is solved as a system of nonlinear equations, the gradient is directly available to the optimizer. This allows the gradients to be calculated very accurately at greatly reduced computational time, improving the performance of the optimization algorithm. Third, whereas in the ODE formulation solving the model and performing parameter estimation are separate steps, in the NLP formulation, we are able to simultaneously solve and optimize the model. Finally, having removed the necessity of having a high-quality ODE solver available, we are able to move the problem to a platform with more powerful optimization algorithms.

Despite the advantages, formulating the problem as a system of nonlinear equations is not as straightforward as the ODE formulation. Microkinetic models are naturally ill-conditioned (where the magnitudes of the variables are very different), a major cause of difficulties in optimization.[31–33] Therefore, solving the problem as a system of nonlinear equations requires significant reformulation, but results in dramatically decreased computational time. Such models are often overparametrized, so that parameter reduction is needed to yield well-conditioned systems. For this task, global sensitivity provides a rigorous way to determine which parameters are insensitive and can be fixed in the model. Moreover, there is a broad literature that shows global sensitivity to be very effective in determining a set of observable parameters from experimental data. In this study, we also considered a global sensitivity approach but found that severe nonlinearities induced by changing kinetic mechanisms showed sensitivities that vary greatly from (local) solution to solution. In particular, we found parameters that did not appear to affect the fit at a local minimum, have an important effect at different points in the parameter space. Therefore, we retained all parameters in our analysis so that their influence would not be ignored. As shown in this section, our overall solution strategy considers well-defined optimization subproblems with reduced parameter sets instead, so we were able to obtain the same advantages as with global sensitivity.

Conditioning and scaling

Both surface coverages and rate constants naturally span many orders of magnitude. For instance, the rate constants take on values ranging from approximately 10^{-20} to 10^{20}, 40 orders of magnitude. This results in poor conditioning, where the gradient of the objective function is much steeper with respect to one variable than another, leading to poor optimization performance. To address this, we replace our original variable (*old*) with a new variable (*new*) times a scaling constant ($old = new * scale$), where the constant, $scale$, is selected so that our new variable is on the order of 1. This rescaling is applied to variables kf_{ic}, Keq_{ic}, kr_{ic}, and y_{jc} and improves the conditioning of the problem. The scaled variables are substituted into Eqs. 1–13 to obtain a rescaled model, which no longer includes the original variables.

Another key challenge is the presence of exponential functions in the calculation of the rate constants. The gradient of the exponential function increases dramatically as the argument of the exponential increases. Consequently, for effective optimization, it is essential that the argument of the exponential remains small (approximately less than 5). However, when calculating the equilibrium constant, the heat of reaction can naturally take on a range of values that would violate this restriction. This issue cannot be addressed by using typical scaling approaches, with a multiplicative scaling factor, as the scaling factor would remain within the argument of the exponential. Instead, the equation is split in two. We replace Eq. 8 with two new Eqs. 14 and 15. Equation 14 calculates Keq_{ic} as function of a new variable, arg_{ic}, and a new constant, $argscale_{ic}$. These are related to one another and to the enthalpy of the reaction in Eq. 15. The value of the constant, $argscale_{ic}$, is selected such that the value of arg_{ic} is approximately zero at the initial point

$$Keq_{ic} = \exp(arg_{ic}) * \exp(argscale_{ic}), \forall i, c \tag{14}$$

$$arg_{ic} = -\frac{\Delta H_{ic}}{RT_c} + \frac{\Delta S_{ic}}{R} - argscale_{ic}, \forall i, c \tag{15}$$

The heat of reaction is then bounded so that arg_{ic} remains in the acceptable range. This improves the conditioning of the equation resulting in improved optimization performance. The bound is reset at the beginning of each optimization run. This approach is not required for kf_{ic} as the argument of the exponential is always negative.

Due to the nonlinearity (and large gradient) of the exponential terms for calculating the rate constants, modest changes in BE_j and Ea_i result in large changes in the rate constants, requiring frequent rescaling. To address this, we recondition the problem automatically during optimization (see solution algorithm).

Objective function

Our goal is to fit the parameters, BE_j and Ea_i, of the microkinetic model to the experimental data. We fitted our model to a comprehensive kinetic dataset published by Graaf et al.[34] The data was collected in a spinning basket reactor at 483 to 547 K and 15 to 50 atm over a $Cu/ZnO/Al_2O_3$ catalyst with various $H_2/CO_2/CO$ feed compositions. After removing points with relative exit mole fraction error greater than 40% (indicating that they were not obtained at steady state) and those obtained above 518 K where diffusion limitations were observed (as discussed in previous work[18]) we were left with 75 experimental data points.

A well-fitted model allows us to make accurate predictions about reaction rates, surface coverages, and so forth under different experimental conditions (pressure, temperature, feed composition, etc.). It also provides insight into the nature of the active site and the reaction mechanism. The selection of an appropriate objective function is very important, balancing the relative error for low-producing experimental conditions against the absolute error for high-producing experimental conditions. Equation 16 is a normalized sum of the squared errors (nSSR) objective function. It measures the percentage difference between the model predictions (F_{jc}^{out}) and the experimental data (E_{jc}^{out}). This

objective ensures that the relative error in low-producing experimental points is not too large

$$\text{error} = \sum_c \sum_{j \in G} \left(\frac{F_{jc}^{\text{out}}}{E_{jc}^{\text{out}}} - 1 \right)^2 \tag{16}$$

In contrast, the standard sum of the squared errors (SSR), shown in Eq. 17, measures the absolute difference between the model predictions and experimental data. This leads to better fitting of high-producing experimental points, but may effectively disregard low-producing experimental points

$$\text{error} = \sum_c \sum_{j \in G} \left(F_{jc}^{\text{out}} - E_{jc}^{\text{out}} \right)^2 \tag{17}$$

It is also possible to use heteroscedastic error functions which include aspects of both of these to better capture the structure of the experimental error.[24]

Penalty functions

To further improve optimization performance, we relax the constraint that Eq. 1 must be equal to zero, that is

$$F_{jc}^{\text{in}} - F_{jc}^{\text{out}} + R_{jc}^{\text{net}} = z_{jc}, \forall j, c \tag{18}$$

and introduce Eq. 19

$$d = \sum_j \sum_c |z_{jc}| \tag{19}$$

The sum of the deviations from steady state, d, is added to the objective function (see Eq. 21 below) as a penalty with a large leading coefficient ($\alpha = 10^4$), to insure that the solution remains very near steady state. The combination provides good control of even small deviations from steady state. The overall value of d in most solutions is $10^{-6}-10^{-4}$; this is a small error especially when spread over 75 conditions and 18 species. Moreover, the deviation from steady state at the individual points is generally much smaller than that obtained by solving the system of ODEs to "steady state."

With 26 parameters to fit 75 experimental points, there are regions of the parameter space where the response surface of the objective function is flat with respect to one or more parameters. In this case, we would like the values of the parameters to stay as close to the DFT values as possible without compromising the fit to experimental data. To that end, we have introduced a quadratic bias function, p, in Eq. 20, which acts like a prior probability in Bayesian formulation[35]

$$p = \sum\nolimits_j (BE_j - BE_j^{\text{DFT}})^2 + \sum\nolimits_i (Ea_i - Ea_i^{\text{DFT}})^2 \tag{20}$$

The quadratic form penalizes large deviations from the DFT values more strongly than small deviations, in accordance with what we expect about the experimental error. We add p to the objective function (Eq. 21), with a small leading coefficient ($\beta = 0.08$) to allow the parameters to move away from their DFT values when it improves the fit to the experimental data, but keep them near DFT when the improvement is minimal.

Our final, overall objective function, obj, is shown below, in Eq. 21

$$\min \text{obj} = \text{error} + \alpha d + \beta p \tag{21}$$

Solution of NLP

To solve the parameter estimation problem, we use the NLP-solver CONOPT, which is based on a generalized reduced gradient algorithm.[36] CONOPT is designed for large, sparse (where most variables are involved in only a few equations) systems and is particularly effective when the number of equations and number of variables is similar, as is the case here.[37] The optimization model is formulated and solved within the general algebraic modeling system (GAMS), an optimization modeling environment specifically designed to solve a wide range of optimization problems with a special emphasis on large systems of equations.[38] GAMS includes a modeling language which readily represents systems of equations, a compiler, automatic differentiation capabilities, and links to powerful optimization solvers.

Solution algorithm

We have developed a solution algorithm to get the best possible fits to the experimental data for our reformulated model (Figure 2). The algorithm is initialized by a single solution of the ODE version of the model and then the reformulated model is optimized with different subsets of parameters free (often only one) until we can obtain no further improvement. A formal statement of the algorithm is included at the end of this section.

Given a set of material parameters BE_j and Ea_i and values for y_{jc} (coverages and partial pressures) for all species and conditions, all the remaining values can be calculated. However, this means that an initial guess must be specified for the values of y_{jc}. Furthermore, due to the nonlinearity of the problem, the guess must be reasonably accurate to obtain feasible solutions. During initialization, the problem is solved once as a system of ODEs for each initial set of BE_j and Ea_i (Step 1) to obtain good initial guesses for the coverages and partial pressures of all species. The results from this solution are imported into GAMS where the problem is initially solved with BE_j and Ea_i fixed to ensure that the optimization is initialized with a minimal deviation from steady state (Steps 2–4).

To minimize the size of the problem, the optimization may be performed over subsets of the parameters. The model is optimized with a subset of the parameters free, while the remaining parameters are fixed resulting in a subset of the equations becoming fixed as well (Steps 5 and 9). These equations are removed from the system of equations. We have found that the best solutions are obtained by initially optimizing the model with one free parameter at a time. In the main loop, the parameters are freed in a predetermined order and the model is optimized with each parameter free until no further improvement is obtained. The specific order in which the parameters are freed may affect the final result; however, we try to minimize this effect with our algorithm's inner loop. The extent of improvement for each parameter is recorded and the problem is rescaled prior to each round of optimization. Each subsequent optimization is initialized at the final point of the prior optimization, unless the solution of the prior optimization is infeasible or the objective function (fit) has deteriorated (Step 6). In that case, the next optimization is initialized from the current best solution (which was stored in Step 7). When the model has been optimized with each of the parameters free (Step 8) we stop if there

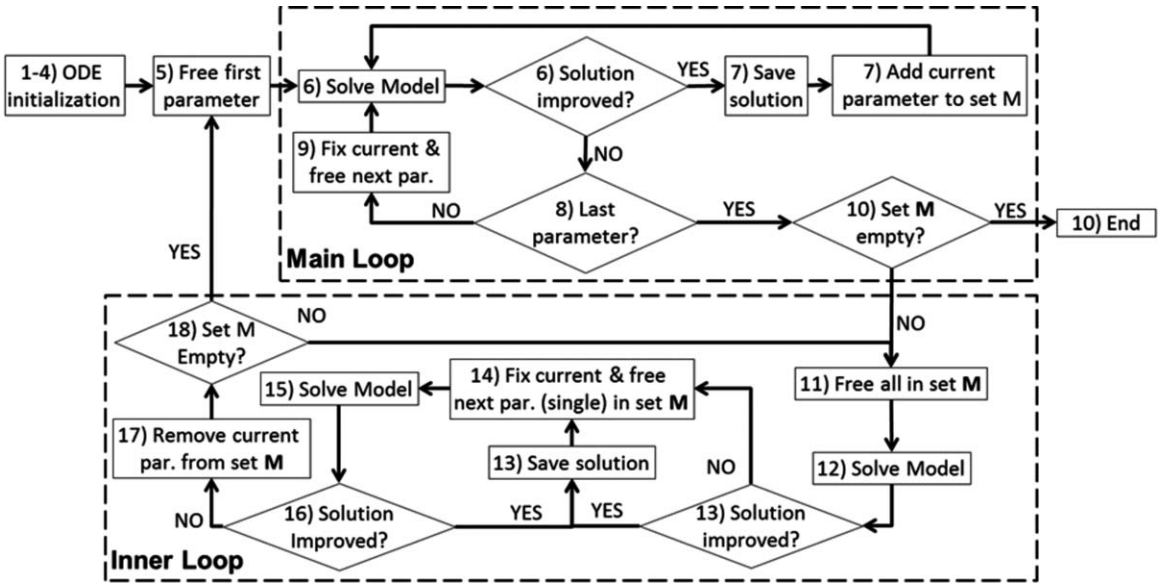

Figure 2. Flowchart of solution algorithm.

The problem is initialized by a single solution of the ODE formulation to obtain good initial guesses for the surface coverages and gas-phase concentrations. In the main loop, the parameters are optimized one at a time, in a fixed sequence. In the inner loop, the parameters which improved the objective in the main loop (set M) are optimized as a set and then individually. Parameters which do not improve the objective are removed from set M and when set M becomes empty we return to the main loop. The algorithm terminates when the objective does not improve in the main loop.

has been no improvement in the objective function. Otherwise, we move on to the inner loop.

In the inner loop, the model is initially optimized with all of the parameters in set **M** (parameters which improved in the main loop) free simultaneously (steps 11–13). The model is then optimized with each of the parameters in set **M** freed one at a time, in the order corresponding to the degree of improvement in the objective function in the main loop (Steps 14–16). When the fit no longer improves with a particular parameter free, that parameter is removed from set **M** (step 17) and the model is optimized with all of the parameters in the reduced set **M** free (Steps 11–13). When set **M** is empty (Step 18), we return to main loop at Step 5 and the process repeats. Optimization stops when the model has been optimized with each of the parameters freed sequentially (main loop) without further improvement in the objective function (Step 10). This gives us the best local minimum for the objective function. Therefore, we will find different solutions from different initial points.

Formal statement of parameter estimation algorithm

Initialization—Define **A** as the set of all parameters and establish good initial guesses for y_{jc}.

1. Solve the ODE model and store BE_j, Ea_i, and y_{jc}.
2. Fix all parameters in set **A**.
3. Initialize and scale all variables and solve NLP model (Eqs. 1–21).
4. Store BE_j, Ea_i, and y_{jc}. Set best = obj.

Main Loop—Optimize all parameters individually until they show no further improvement.

5. Free first parameter in **A** and fix all others.
6. Initialize and scale all variables and solve NLP model (Eqs. 1–21). If obj > best, the current best value, goto 8.
7. Store BE_j, Ea_i, and y_{jc}. Set best = obj. Add the current parameter to set **M** of improved parameters. Goto 6.
8. If current parameter is the last in the set of all parameters, **A**, goto 10.
9. Fix the current parameter and free the next parameter in **A**. Goto 6.
10. If the set **M** is empty, end.

Inner Loop—Further optimize the parameters that showed improvement in main loop (set **M**).

11. Free all parameters in set **M**.
12. Initialize and scale all variables and solve NLP model (Eqs. 1–21).
13. If obj < best, store BE_j, Ea_i, and y_{jc}, set best = obj.
14. Fix current parameter(s) and free the next parameter in set **M**.
15. Initialize and scale all variables and solve NLP model (Eqs. 1–21).
16. If obj < best, goto 13.
17. Remove the current parameter from set **M**.
18. If the set **M** is empty, goto 5. Otherwise, goto 11.

Performance evaluation

The above algorithm and the reformulation of the model resulted in a dramatic decrease in the computational time required for parameter estimation. For the reaction network in Figure 1, on average, a single simulation of the model as a system of ODEs takes approximately 5 CPU min on a 3-GHz Intel Core Duo CPU. To optimize the system of ODEs, over 2 h would be required just for one gradient approximation; a significant computational load which must be completed at every iteration of the optimization algorithm. In all, it would take approximately 2 weeks to complete an optimization of the ODE model with 150 optimization iterations. In contrast, solving the system with CONOPT requires an average of just 15 CPU min for the full optimization, a 1000-fold increase in speed. The increase in speed is dependent on the number of parameters optimized, the size of the reaction network and the number of experimental points to be fitted. Broadly, the speed-up increases as the size of the problem increases. However, as the problem becomes very large additional improvements in the model

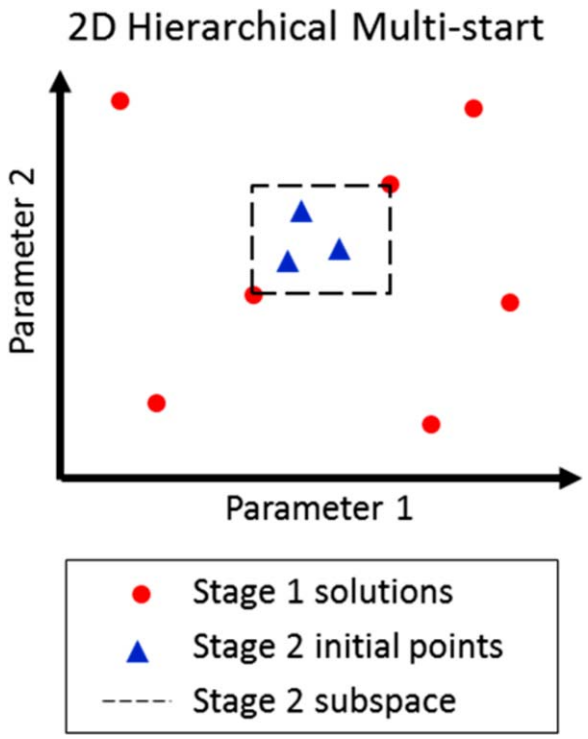

Figure 3. Hierarchical multistart approach.

An initial (Stage 1) sample of the parameter space is taken and the model is optimized. The best solutions in Stage 1 are used to define one or more parameter subspaces for Stage 2. The subspaces are sampled and the model is optimized from these points giving a higher-resolution sampling of promising parts of the parameter space. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

and/or algorithm may be required to solve the problem as a NLP.

Multistart approach

As microkinetic models are nonconvex and have multiple local minima, the final solution of any local optimization depends on the initial guess provided for the parameter values. Because the reformulation of the model dramatically reduces solution times, we can implement a multistart approach to efficiently and systematically search the parameter space for parameter values which provide a good fit to the experimental data. However, the number of points required to sample the space on a grid grows as n^p, where p is the number of parameters and n is the number of different values for each parameter we want to test. It follows that with 26 parameters, and intervals around the DFT values of approximately 1 eV (+/−0.5 eV), if we want to sample the space at 0.1-eV intervals it would require approximately 10^{26} points. If we increased our interval to 0.2 eV, we would still require approximately 10^{18} points. Therefore, our initial (Stage 1) sample is necessarily low-resolution. To obtain better solutions, we introduce a second sampling stage (Figure 3). After solving the model from the initial points in Stage 1, we identify the best solutions. Two or more of these solutions (preferably relatively close together in the parameter space) are used to define a subspace, which includes parameter values intermediate between the Stage 1 solutions. We sample the subspace (Stage 2) and use the new points to initialize the model. Additional sampling phases are included as needed based on the quality of the fits obtained and the Stage 2 sample resolution.

Other groups have also combined multistart approaches with NLP solvers to obtain greater reliability and improved solutions to nonlinear optimization problems. A notable example is OQNLP[39,40] a multistart NLP solver available in GAMS.

Results and Discussion

The hierarchical multistart approach was used to provide initial points for optimization. For our methanol synthesis example, our Stage 1 sample was a 10,000 point latin hypercube sample (LHS)[41] of the parameter space with bounds 0.65 eV on either side of the DFT values, or as constrained by the thermodynamics of the problem. (The activation energies must be positive and satisfy Eq. 11. The binding energies must be negative.) In Stage 2, we sampled one

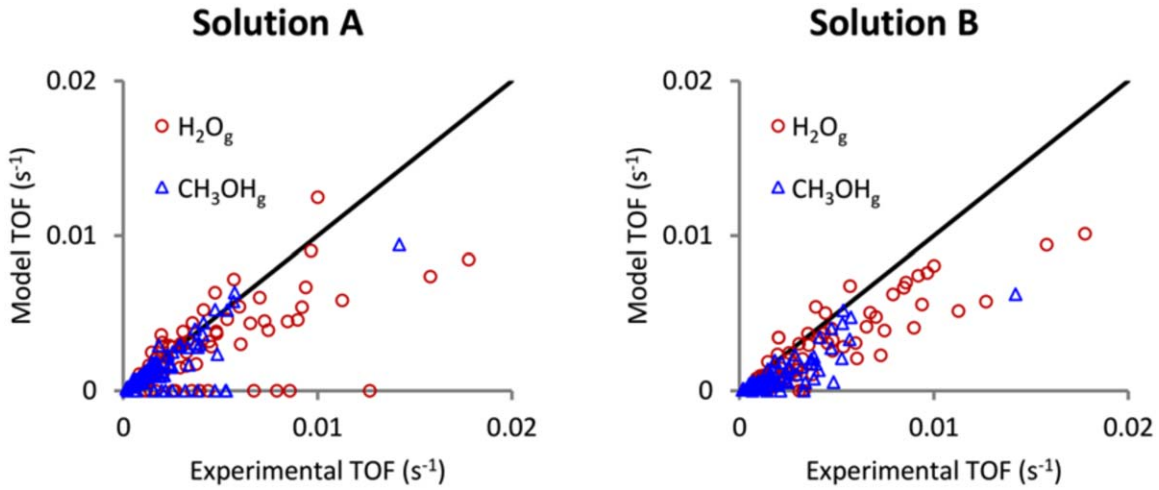

Figure 4. Two of the better solutions generated from the Stage 1 sampling; Solutions A and B have objective function values of 37 and 50, respectively.

These points are used to define the parameter subspace for Stage 2 sampling. Model-predicted TOF is plotted against experimentally measured TOF. Blue triangles represent methanol production rates; red circles, water production rates. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

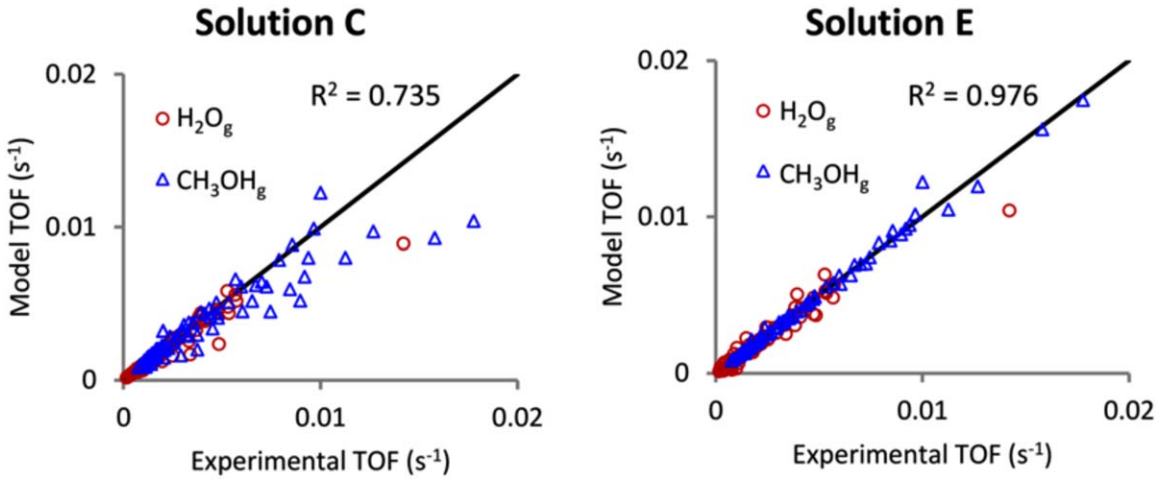

Figure 5. A Stage 2 LHS with 100 points bounded by Solutions A and B from Stage 1 was generated and used as initial points for optimization.

The best fits using the normalized and standard SSR in the objective function are Solution C and E, respectively. As would be expected, Solution C is a better fit for the data in the low-production range, whereas Solution E is a better fit in the high production. Both solutions represent an improvement over the fit obtained in our prior work[18] in their respective objective functions. Solution E is an improvement by all measures calculated. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

parameter subspace defined by two of the best points from Stage 1 with a smaller LHS (100 points). Each of the initial points generated was optimized using our solution algorithm to obtain our final results.

With two data points (methanol and water production) at each of 75 experimental conditions, the error using nSSR at steady state is 150 when the model predicts an inactive catalyst (no production). A perfect fit to the data would have zero (nSSR) error. In our prior work[18] using the ODE formulation of the same problem, we found a parameter set with an error (nSSR) of 13. Using the best initial points from the Stage 1 LHS, the algorithm yielded Solutions A and B, with objective function (nSSR) values of 37 and 50, respectively, which are shown in Figure 4. Although these fits are a dramatic improvement over the unfitted DFT parameter values (which predict no production), the fit remains poor. Solution B systematically underpredicts the production of both methanol and water. Solution A does a better job of predicting methanol production, but remains a poor predictor for water production and fails to predict any production at all for a significant number of points.

We used Solutions A and B to define a parameter subspace for Stage 2 sampling. Of the hundred Stage 2 solutions, 15 fit the data (based on nSSR) better than our previous work.[18] This shows that a hierarchical multistart approach can be an effective way to investigate large parameter spaces, which will become more important for larger models with greater numbers of parameters. The best solution from Stage 2 is shown in Figure 5 as Solution C, which has an objective function (nSSR) value of 5.04. The fit is much better than Solutions A and B; however, there is still some underprediction of the production of methanol, especially for the higher-producing experimental conditions. Meanwhile, the low-producing experimental conditions are very accurately modeled. This is partly a consequence of our selection of nSSR (Eq. 16) in our objective function. Since what we are minimizing is the percentage difference between the model predictions and the experimental data, the absolute deviation will naturally be larger for the higher-producing experimental conditions.

Using another starting point from the second stage LHS, we also obtained Solution D from the algorithm; see Tables 1–3 for comparison. Like Solution C, this solution underpredicts the production rates of high-producing experimental points. For some parameters, such as the activation energy for hydrogen disassociation, the values are very similar. However, there are differences in parameter values of up to 0.5 eV (*BE* of COOH), indicating these are distinct solutions.

To obtain a better fit to the high-producing points, we reran the optimization for the 100-point LHS, replacing Eq. 16 with Eq. 17 (the standard SSR) and scaling the leading coefficients, α and β, so that the relative influence of the penalties was unchanged. Seventeen of the resulting solutions had a standard SSR less than that found in our prior work.[18] The best of these solutions, with an R^2 value of

Table 1. A Comparison of the Quality Fit for Various Solutions to the Model

			Stage 1 w/ nSSR		Stage 2 w/ nSSR		Stage 2 w/ SSR		
	DFT	Prior Fit[18]	*A*	*B*	*C*	*D*	*E*	*F*	*G*
Normalized SSR	149.99	13.15	37.46	50.12	5.04	5.56	6.16	6.51	10.21
Standard SSR (10^{-4})	29.60	1.38	8.97	6.31	2.22	1.55	0.34	0.54	0.61

Our Stage 2 solutions fit using a standard SSR achieved improved fits to the data (compared to our prior work) based on all measures. Our prior work[18] used the standard SSR in the objective function.

Table 2. *BE* Values (eV) for DFT, Best Fit from Our Prior Work, and Selected Fits from this Work

			10^4-point LHS w/ nSSR		100-point LHS w/ nSSR		100-point LHS w/ SSR		
	DFT	Prior Fit[18]	*A*	*B*	*C*	*D*	*E*	*F*	*G*
CO	−0.9	−0.8	**−0.9**	**−1.1**	**−1.0**	−1.0	**−1.0**	−1.3	−1.0
H_2O	−0.2	−0.2	**−0.3**	**−0.3**	**−0.2**	−0.4	**−0.3**	−0.5	−0.4
H	−2.4	−2.5	**−2.5**	**−2.4**	**−2.4**	−2.5	**−2.5**	−2.5	−2.4
CO_2	−0.1	−0.1	**−0.1**	**−0.3**	**−0.2**	−0.4	**−0.3**	−0.1	−0.2
OH	−2.8	−3.2	**−3.1**	**−3.3**	**−3.1**	−3.2	**−3.3**	−3.0	−3.2
COOH	−1.5	−1.8	**−1.5**	**−1.5**	**−1.5**	−2.0	**−1.5**	−1.5	−1.5
HCOOH	−0.2	−0.7	**−0.9**	**−0.8**	**−0.8**	−1.2	**−0.9**	−1.3	−1.1
HCO	−1.2	−1.7	**−1.7**	**−2.2**	**−2.0**	−2.0	**−2.3**	−1.9	−2.1
HCO_2	−2.7	−3.3	**−3.1**	**−3.0**	**−3.1**	−3.2	**−3.2**	−3.3	−3.2
CH_3O_2	−2.0	−2.6	**−3.1**	**−2.5**	**−3.0**	−2.9	**−2.8**	−3.2	−2.9
CH_2O	−0.0	−0.5	**−0.8**	**−0.1**	**−0.6**	−0.3	**−0.3**	−0.8	−0.4
CH_3O	−2.5	−3.1	**−3.1**	**−2.5**	**−2.9**	−2.5	**−2.8**	−2.7	−2.7
CH_3OH	−0.3	−0.5	**−0.5**	**−0.5**	**−0.5**	−0.3	**−0.3**	−0.4	−0.3

There are differences of more than 1 eV between *BE* values in our present work and DFT predictions (e.g., HCO). There are also differences of more than 0.5 eV between solutions from our current work (e.g., CH_2O). There are other species for which all solutions have very similar *BE* values (e.g., H, H_2O, and CH_3OH). Solutions from Figures 4 and 5 in bold.

0.98, (SSR = 3.4e-5) is shown in Figure 5 as Solution E. (In comparison, our prior work[18] had an R^2 value of 0.88 and an SSR of 1.4e-4.) The fit is very good, with nearly all of the data points lying on or very near the parity line. We also include Solutions F and G from this set of solutions in Tables 1–3, to allow comparisons between their parameter values.

Table 1 includes the normalized and standard sum of squared errors for each solution, allowing us to compare and contrast the different solutions. Overall our Stage 2 sampling using the standard SSR provided the best solutions. They have the highest R^2 values and the lowest values for SSR. Their nSSR values are lower than those from our prior work.[18] The Stage 2 sampling optimized with nSSR in the objective function had lower values of nSSR, but the fit is poor for high-producing experimental points (Figure 5).

Tables 2 and 3 contain the *BE* and *Ea* values for selected fits, respectively. The fits are distinct in their parameter values. Certain parameters are fairly consistent across all solutions (such as *Ea* for $HCOOH^* + H^* \rightarrow CH_3O_2^* + *$ and *BE* of CO). Other parameters vary widely from solution to solution (such as the *BE* HCO and *Ea* $HCO^* + H^* \rightarrow CH_2O^* + *$). Because we have identified several solutions of similar quality, with different parameter sets, it is necessary to consider additional factors to identify the best fit to the data. These may include considering more than one measure of fit (such as standard and normalized SSR and R^2 value), privileging solutions whose parameters most resemble DFT, and examining subsets of the data and evaluating the model's ability to accurately predict trends in production based on temperature, pressure, feed composition, reaction orders with respect to reactants and products, and so forth. The model may also be validated by comparing the predictions of the fitted model to a dataset not used in the fitting. We will further investigate these directions in future work.

Conclusions

Reformulating microkinetic models as systems of nonlinear equations requires careful scaling and bounding to produce meaningful results. However, it pays off in dramatic improvements in computational speed. This allows more comprehensive searches of the parameter space than were previously feasible. In the past, the initial values of the

Table 3. *Ea* Values (eV) for DFT, Best Fit from Our Prior Work, and Selected Fits from this Work

			10^4-point LHS w/ nSSR		100-point LHS w/ nSSR		100-point LHS w/ SSR		
	DFT	Prior Fit[18]	*A*	*B*	*C*	*D*	*E*	*F*	*G*
R1 $H_2 + 2^* \rightarrow 2\ H^*$	0.5	0.4	**0.5**	**0.5**	**0.5**	0.5	**0.5**	0.5	0.5
R2 $H_2O^* + * \rightarrow H^* + OH^*$	1.1	0.7	**1.0**	**0.6**	**0.9**	1.1	**0.9**	1.0	1.0
R3 $CO^* + OH^* \rightarrow COOH^* + *$	0.5	0.5	**1.0**	**0.9**	**1.0**	1.0	**0.9**	1.0	0.9
R4 $COOH^* + * \rightarrow CO_2^* + H^*$	1.0	1.2	**1.0**	**1.0**	**1.0**	0.5	**1.0**	1.0	1.0
R5 $CO^* + H^* \rightarrow HCO^* + *$	0.9	0.5	**1.1**	**1.1**	**1.1**	1.1	**1.1**	1.4	1.1
R6 $CO_2^* + H^* \rightarrow HCO_2^*$	0.8	0.4	**0.6**	**0.6**	**0.6**	1.1	**0.6**	0.8	0.7
R7 $HCO_2^* + H^* \rightarrow HCOOH^* + *$	0.8	0.9	**1.3**	**0.8**	**1.0**	1.0	**1.2**	1.1	1.1
R8 $HCOOH^* + H^* \rightarrow CH_3O_2^* + *$	1.0	1.0	**0.9**	**0.8**	**1.0**	1.1	**1.0**	1.0	1.0
R9 $HCO^* + H^* \rightarrow CH_2O^* + *$	0.4	0.5	**0.5**	**0.6**	**0.5**	0.6	**1.0**	0.6	0.6
R10 $CH_2O^* + H^* \rightarrow CH_3O^* + *$	0.2	0.2	**0.4**	**0.1**	**0.4**	0.2	**0.3**	0.6	0.4
R11 $CH_3O_2^* + * \rightarrow CH_2O^* + OH^*$	0.6	0.4	**0.9**	**0.6**	**0.8**	0.8	**0.6**	1.0	0.8
R12 $CH_3O^* + H^* \rightarrow CH_3OH^* + *$	1.1	1.4	**1.2**	**0.6**	**1.1**	0.8	**1.1**	1.1	0.7

There are differences of more than 0.5 eV between *Ea* values in our present work and DFT predictions (e.g., R3, R4, R5, and R9). There are also differences of more than 0.4 eV between solutions from our current work (e.g., R9, R11, and R12). There are other reactions for which all solutions in our current work have very similar *Ea* values (e.g., R1, R2, and R3). Solutions from Figures 4 and 5 in bold.

parameters were adjusted manually based on intuition, knowledge of the system and the gradient at the current point. This approach was very time-consuming and yielded results that were heavily dependent upon the skill and knowledge of the user. Moreover, as the process was guided by prior assumptions about the system, the results were less likely to provide novel insights. Our improved approach allowed us to identify multiple fits to the data which are significantly better than those previously identified. This approach can be extended to reaction networks including multiple reaction pathways to identify which are feasible and most probable. We can further improve our approach by using more refined, statistically based, objective functions. Greater insight into the reaction on industrially relevant catalysts can help us develop better catalysts and better conditions for operating existing catalysts. Moreover, the techniques described herein can be directly extended to the problem of designing better catalysts, by allowing us to model new catalysts *in silico* much more efficiently than was previously possible.

Acknowledgments

This contribution is dedicated to the 90th birthday celebration of R. B. Bird. His legacy of excellence and scientific rigor has inspired generations of Chemical Engineers, including MM, CTM, LTB, and LCG. The authors acknowledge DOE-Basic Energy Sciences and 3M for partial financial support. LTB acknowledges the Hougen Visiting Professorship Fund. The computational work was performed in part using supercomputing resources from the following institutions: EMSL, a National scientific user facility at Pacific Northwest National Laboratory (PNNL); the Center for Nanoscale Materials at Argonne National Laboratory (ANL); and the National Energy Research Scientific Computing Center (NERSC). EMSL is sponsored by the Department of Energy's Office of Biological and Environmental Research located at PNNL. CNM and NERSC are supported by the U.S. Department of Energy, Office of Science, under contracts DE-AC02-06CH11357, and DE-AC02-05CH11231, respectively.

Literature Cited

1. Bartholomew CH, Farrauto RJ. *Fundamentals of Industrial Catalytic Processes*, 2nd ed. Hoboken, NJ: Wiley, 2006.
2. Greeley J, Norskov JK, Mavrikakis M. Electronic structure and catalysis on metal surfaces. *Annu Rev Phys Chem*. 2002;53:319–348.
3. Donazzi A, Maestri M, Michael BC, Beretta A, Forzatti P, Groppi G, Tronconi E, Schmidt LD, Vlachos DG. Microkinetic modeling of spatially resolved autothermal CH4 catalytic partial oxidation experiments over Rh-coated foams. *J Catal*. 2010;275(2):270–279.
4. Dumesic JA. *The Microkinetics of Heterogeneous Catalysis*. Washington, D.C.: American Chemical Society, 1993.
5. Cortwright R, Dumesic J. Kinetics of heterogeneous catalytic reactions: analysis of reaction schemes. *Adv Catal*. 46;2001:161–264.
6. Gokhale AA, Kandoi S, Greeley JP, Mavrikakis M, Dumesic JA. Molecular-level descriptions of surface chemistry in kinetic models using density functional theory. *Chem Eng Sci*. 2004;59(22–23):4679–4691.
7. Mhadeshwar AB, Wang H, Vlachos DG. Thermodynamic consistency in microkinetic development of surface reaction mechanisms. *J Phys Chem B*. 2003;107(46):12721–12733.
8. Lynggaard H, Andreasen A, Stegelmann C, Stoltze P. Analysis of simple kinetic models in heterogeneous catalysis. *Prog Surf Sci*. 2004;77(3–4):71–137.
9. Ovesen CV, Clausen BS, Schiotz J, Stoltze P, Topsoe H, Norskov JK. Kinetic implications of dynamical changes in catalyst morphology during methanol synthesis over Cu/ZnO catalysts. *J Catal*. 1997;168(2):133–142.
10. Biegler LT. *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2010.
11. Grossmann IE, Biegler LT. Part II. Future perspective on optimization. *Comput Chem Eng*. 2004;28(8):1193–1218.
12. Herder PM, Stikkelman RM. Methanol-based industrial cluster design: a study of design options and the design process. *Ind Eng Chem Res*. 2004;43(14):3879–3885.
13. Fang K, Li D, Lin M, Xiang M, Wei W, Sun Y. A short review of heterogeneous catalytic process for mixed alcohols synthesis via syngas. *Catal Today*. 2009;147(2):133–138.
14. Specht M, Staiss F, Bandi A, Weimer T. Comparison of the renewable transportation fuels, liquid hydrogen and methanol, with gasoline-energetic and economic aspects. *Int J Hydrogen Energy*. 1998;23(5):387–396.
15. Ma J, Sun NN, Zhang XL, Zhao N, Xiao F, Wei W, Sun Y. A short review of catalysis for CO2 conversion. *Catal Today*. 2009;148(3–4):221–231.
16. Olah GA, Goeppert A, Prakash GKS. Chemical recycling off carbon dioxide to methanol and dimethyl ether: from greenhouse gas to renewable, environmentally carbon neutral fuels and synthetic hydrocarbons. *J Org Chem*. 2009;74(2):487–498.
17. Lim HW, Park MJ, Kang SH, Chae HJ, Bae JW, Jun KW. Modeling of the kinetics for methanol synthesis using Cu/ZnO/Al2O3/ZrO2 catalyst: influence of carbon dioxide during hydrogenation. *Ind Eng Chem Res*. 2009;48(23):10448–10455.
18. Grabow L, Mavrikakis M. On the mechanism of methanol synthesis on Cu through CO and CO2 hydrogenation. *ACS Catal*. 2011;1:365–384.
19. Rostamikia G, Mendoza AJ, Hickner MA, Janik MJ. First-principles based microkinetic modeling of borohydride oxidation on a Au(111) electrode. *J Power Sources*. 2011;196(22):9228–9237.
20. Cao XM, Burch R, Hardacre C, Hu P. Reaction mechanisms of crotonaldehyde hydrogenation on Pt(111): density functional theory and microkinetic modeling. *J Phys Chem C*. 2011;115(40):19819–19827.
21. Madon RJ, Braden D, Kandoi S, Nagel P, Mavrikakis M, Dumesic JA. Microkinetic analysis and mechanism of the water gas shift reaction over copper catalysts. *J Catal*. 2011;281(1):1–11.
22. Thybaut JW, Sun JJ, Olivier L, Van Veen AC, Mirodatos C, Marin GB. Catalyst design based on microkinetic models: oxidative coupling of methane. *Catal Today*. 2011;159(1):29–36.
23. Liu K, Wang A, Zhang W, Wang J, Huang Y, Wang X, Shen J, Zhang T. Microkinetic study of CO oxidation and PROX on Ir-Fe Catalyst. *Ind Eng Chem Res*. 2011;50(2):758–766.
24. Bligaard T, Norskov JK, Dahl S, Matthiesen J, Christensen CH, Sehested J. The Bronsted-Evans-Polanyi relation and the volcano curve in heterogeneous catalysis. *J Catal*. 2004;224(1):206–217.
25. Hansgen DA, Vlachos DG, Chen JGG. Using first principles to predict bimetallic catalysts for the ammonia decomposition reaction. *Nat Chem*. 2010;2(6):484–489.
26. Nocedal J, Wright SJ. *Numerical Optimization*, 2nd ed. New York: Springer; 2006.
27. Alexe M, Sandu A. Forward and adjoint sensitivity analysis with continuous explicit Runge-Kutta schemes. *Appl Math Comput*. 2009;208(2):328–346.
28. Cao Y, Li ST, Petzold L. Adjoint sensitivity analysis for differential-algebraic equations: algorithms and software. 15th Toyota Conference on Scientific and Engineering Computations for the 21st Century. Mikkabi, Japan, October 28–31, 2001.
29. Cao Y, Li ST, Petzold L, Serban R. Adjoint sensitivity analysis of differential-algebraic equations: the adjoint DAE system and its numerical solution. *SIAM J Sci Comput*. 2002;24(3):1076–1089.
30. Sandu A, Daescu DN, Carmichael GR. Direct and adjoint sensitivity analysis of chemical kinetic systems with KPP: part I - theory and software tools. *Atmos Environ*. 2003;37(36):5083–5096.
31. Tanartkit P, Biegler LT. Reformulating ill-conditioned differential-algebraic equation optimization problems. *Ind Eng Chem Res*. 1996;35(6):1853–1865.
32. Vassiliadis VS, Floudas CA. The modified barrier function approach for large-scale optimization. *Comput Chem Eng*. 1997;21(8):855–874.
33. Wright MH. Ill-conditioning and computational error in interior methods for nonlinear programming. *SIAM J Optim*. 1998;9(1):84–111.

34. Graaf GH, Stamhuis EJ, Beenackers A. Kinetics of low-pressure methanol synthesis. *Chem Eng Sci.* 1988;43(12):3185–3195.
35. Rubin DB, Gelman A, Carlin JB, Stern H. *Bayesian Data Analysis*, 2nd ed. Boca Raton: Chapman & Hall/CRC, 2003.
36. Drud A. Conopt - a grg code for large sparse dynamic nonlinear optimization problems. *Math Program.* 1985;31(2):153–191.
37. Drud A, CONOPT. ARKI Consulting and Development A/S, Bagsvaerd, Denmark, GAMS Solver Manual. http://www.gams.com/solvers/index.htm (Last accessed: Dec 2013).
38. Brooke A, Kendrick D, Meeraus A. (1988). GAMS: A User's Guide. The Scientific Press, San Francisco, CA.
39. Ugray Z, Lasdon L, Plummer J, Glover F, Kelly J, Marti R. Scatter search and local NLP solvers: a multistart framework for global optimization. *Inform J Comput.* 2007;19(3):328–340.
40. Ugray Z, Lasdon L, Plummer JC, Bussieck M. Dynamic filters and randomized drivers for the multi-start global optimization algorithm MSNLP. *Optim Methods Softw.* 2009;24(4–5):635–656.
41. Stein M. Large sample properties of simulations using latin hypercube sampling. *Technometrics.* 1987;29(2):143–151.

Manuscript received Sept. 18, 2013, and revision received Nov. 18, 2013.