Experimental Investigation and High Resolution Simulator of In-Situ Combustion Processes

Quarterly Report

Start date: April 2006
End date: June 2006

Margot Gerritsen
Anthony R. Kovscek

July 2006

DE-FC26-03 NT15405

Department of Petroleum Engineering
Stanford University
Green Earth Sciences Building
367 Panama Street
Stanford, CA 94305-2220

Disclaimer:
This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expresses herein do no necessarily state or reflect those of the United States Government or any agency thereof.

Abstract

Accurate simulation of in-situ combustion processes is computationally very challenging because the spatial and temporal scales over which the combustion process takes place are very small. In this current and **eleventh** report, we report on the development of a virtual kinetic cell (VKC) that aids the study of the interaction between kinetics and phase behavior. The VKC also provides an excellent tool for developing and testing specialized solvers for the stiff kinetics encountered in ISC processes.

Table of Contents

List of graphical materials

1. Introduction

2. Executive Summary

   2.1. Personnel
   2.2. Important accomplishments

3. Experimental

4. Results and discussion

5. Conclusion

References

List of graphical materials

Figure 1. Illustration of discontinuity sticking resulting from inconsistent algebraic variables.

Figure 2. Simulation results from a ramped temperature experiment using the minimal reaction model. The temperature is raised from 373K to 873K over a period of 10 hours. Cell temperature, oxygen consumption rate and component concentrations are shown for 3 different oxygen feed concentrations.

Figure 3. Simulation results from a ramped temperature experiment using the minimal reaction model. The temperature is raised from 373K to 873K over a period of 10 hours. Cell temperature, oxygen consumption rate and component concentrations are shown for at 3 different reaction activation energies.

Figure 4. The temperatures for the integration subintervals in the performance comparison are obtained by sampling the characteristic ISC temperature profile. The continuous temperature profile is shown along with a zero-order parametrization corresponding to $N = 25$ equidistant subintervals.

Figure 5. Work-precision diagrams for the SARA based ISC reaction model. The tailored ESDIRK implementations are compared to the widely used DASSL and LSODE codes. Comparisons are made for 25 and 100 subintervals

1. Introduction

In-situ combustion, or air injection, is the process of injecting oxygen into oil reservoirs to oxidize the heaviest components of the crude oil and enhance oil recovery through the heat and pressure produced. The emphasis of this work is to study and model numerically in situ combustion processes. The ultimate objectives are to provide a working accurate, parallel in situ combustion numerical simulator and to better understand the in-situ combustion process when using metallic additives and/or solvents combined with in situ combustion. For this purpose, experimental, analytical and numerical studies are conducted.

This report presents results of the third quarter of the third year of this project.

2. Executive Summary

2.1. Personnel

Current personnel include Prof. Margot Gerritsen (PI), Prof. Tony Kovscek (Co-PI), Dr. Louis Castanier (Technical manager), Mr. Rami Younis (PhD student). Mr. Rotimi Awoleke (MSc student) is also working on the project, albeit paid from departmental sources. We have also continued collaboration with Mr. Morten Kristensen (PhD student) from the Technical University of Denmark, and his advisors Prof. Erling Stenby and Prof. Michael Michelsen. Mr. Kristensen's work is the focus of this quarterly report. Finally, research associate Dr. Jim Lambers has been instrumental in the development of the integrated upscaling and adaptivity methodology. In the next quarterly report we'll report on the development of several new upscaling methods for adaptive grids by Dr. Lambers and Prof. Gerritsen.

2.2. Important accomplishments

As discussed in previous quarterly reports, we have proposed the use of implicit one-step ESDIRK (Explicit Singly Diagonal Implicit Runge-Kutta) methods for integration of the stiff kinetics in reactive, compositional and thermal processes that are solved using operator-splitting type approaches. In this report, we discuss the chosen implementation.

To facilitate the algorithmic development we constructed a virtual kinetic cell model. The model serves both as a tool for the development and testing of tailored solvers as well as a testbed for studying the interactions between chemical kinetics and phase behavior. The idea of the VKC was proposed last year. We now have results for two chemical kinetics models for the in-situ combustion process. The models have 6 and 14 components, respectively. Through benchmark studies using the 14 component reaction model the new ESDIRK solvers are shown to improve computational speed when compared to the widely used multi-step BDF methods DASSL and LSODE.

Phase changes are known to cause convergence problems for the integration method. We propose an algorithm for detection and location of phase changes based on discrete event system theory. Experiments show that the algorithm improves the robustness of the integration process near phase boundaries by lowering the number convergence and error test failures by more than 50% compared to direct integration without the new algorithm. Reactive transport processes stiff ODE solvers ESDIRK methods discrete event systems phase change detection differential-algebraic equations multi-scale methods operator splitting enhanced oil recovery in-situ combustion

In this work, we assume that the operator splitting method we developed in (Younis & Gerritsen, 2006) is used for the integration of the ISC process. In a reaction substep, each grid block is now effectively treated as a small kinetic cell with homogeneous pressure and temperature and well mixed fluids. To solve the kinetics, high computational efficiency is very desirable as a typical simulation may involve millions of reaction substeps.

2.2.2 Reaction models

The minimal model

This model includes a minimal realistic set of components and reactions to represent ISC

behavior. The components are in the model are light oil (LO), heavy oil (HO), coke (C), oxygen (O_2), water (W) and inert gas (IG). The model reactions are given by

$$
\begin{aligned}
\mathrm{LO} + \mathrm{O}_2 &\rightarrow \mathrm{IG} + \mathrm{W} \\
\mathrm{HO} + \mathrm{O}_2 &\rightarrow \mathrm{IG} + \mathrm{W} \\
\mathrm{HO} &\rightarrow \mathrm{LO} + \mathrm{C} + \mathrm{IG} \\
\mathrm{C} + \mathrm{O}_2 &\rightarrow \mathrm{IG} + \mathrm{W}
\end{aligned}
$$

The first two reactions model oxidation of the light and heavy oil, respectively. The cracking of heavy oil is modeled with the third reaction, while the last reaction models complete oxidation of the deposited fuel (coke). This minimal model is the common model used in many ISC simulations and it is the default model available in the commercial simulator CMG STARS (STARS).

The SARA based model

In the SARA (saturates, aromatics, resins and asphaltenes) approach, chemical reactivity and phase behavior is taken into account when grouping the components. Saturates are saturated hydrocarbons with straight or branched chains, little ring-structure, and only little nitrogen, sulfur and oxygen content. Aromatics contain one or more aromatic rings. Resins are the second heaviest fraction with high polarity due to considerable nitrogen, sulfur and oxygen content. Asphaltenes are defined as the fraction of the crude insoluble in n-heptane. Within each fraction, groups are distinguished based on reactivity. A further subdivision of the fractions based on boiling point may be used to obtain improved representation of phase behavior.

The model we consider in this work contains 14 (pseudo) components, listed in Table 1, and the 14 reactions following. Details of this model can be found in (Freitag & Verkozcy, 2005, Freitag & Exelby, 2006, and Ren *et al*., 2005. The reactions include pyrolysis, LTO and HTO of the SARA fractions along with HTO of partially oxidized LTO residue and the coke formed by pyrolysis. The LTO residues of resins, aromatics and saturates are non-volatile oil components whereas the LTO residue of asphaltenes is solid.

Component Name	*Abbreviation*	*Phase*
Water	$\mathrm{H}_2\mathrm{O}$	water
Inert oil	InertOil	oil
Oxidized resins/aromatics	OxdResAr	oil
Oxidized saturates	OxdSat	oil
Asphaltenes	Asph	oil
Resins	Resins	oil
Aromatics	Arom	oil/gas

Saturates	Sat	oil/gas
Light oil	Lites	oil/gas
Carbon dioxide	CO2	oil/gas
Nitrogen	N2	gas
Oxygen	O2	gas
Oxidized asphaltenes	OxdAsph	solid
Pyrolysis coke	PyrCoke	solid

Table 1. List of (pseudo) components for the SARA based model.

$$\text{cracking / pyrolysis} \quad \begin{cases} \text{Asph} & \rightarrow & \text{H2O} + \text{Arom} + \text{Sat} + \text{Lites} + \text{PyrCoke} \\ \text{Resins} & \rightarrow & \text{H2O} + \text{Asph} + \text{Arom} + \text{Sat} + \text{Lites} \\ \text{Arom} & \rightarrow & \text{Asph} + \text{Lites} \end{cases}$$

$$\text{LTO} \quad \begin{cases} \text{Asph} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 + \text{OxdAsph} \\ \text{Resins} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 + \text{OxdResAr} \\ \text{Arom} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 + \text{OxdResAr} \\ \text{Sat} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 + \text{OxdSat} \end{cases}$$

$$\text{HTO} \quad \begin{cases} \text{PyrCoke} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \\ \text{OxdAsph} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \\ \text{OxdResAr} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \\ \text{OxdSat} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \\ \text{Asph} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \\ \text{Resins} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \\ \text{Arom} + \text{O}_2 & \rightarrow & \text{H}_2\text{O} + \text{CO}_2 \end{cases}$$

2.2.3 Building the Virtual Kinetic Cell (VKC)

We model a closed system consisting of 3 fluid phases (oil, water and gas), an immobile solid phase and the porous medium. Temperature, pressure and component concentrations are assumed to be uniform in the cell. For convenience, we introduce the molar concentration c_{ij} of a component $i, i = 1, \ldots, n_c$, in a phase j, $j \in \{o, w, g\}$, and the overall molar concentration C_i of component i. They are given by

$$c_{ij} = \varphi_f x_{ij} \rho_j S_j,$$
$$C_i = \sum_{j \in \{o,w,g\}} c_{ij},$$

where S_j are the phase saturations, ρ_j the molar phase densities and x_{ij} the mole fraction of component i in phase j. φ_f is the fluid porosity which varies with solid fuel concentration in the pore space according to

$$\varphi_f = \varphi_v - \frac{C_s}{\rho_s},$$

where C_s and ρ_s are the density and concentration of the solid component, respectively. The solid concentration is defined with respect to total volume. The mass conservation of each chemical component can be expressed as

$$\frac{dC_i}{dt} = \sum_{k=1}^{n_r} A_{ik} r_k + \frac{F_i^{in} - F_i^{out}}{V}, \qquad i = 1, \ldots, n_c. \tag{1}$$

Here, F_i^{in} and F_i^{out} are the molar flow rates of component i in and out of the cell, respectively, V is the cell volume, r_k the kinetic expression for the kth ($k = 1, \ldots, n_r$) chemical reaction and A_{ik} is the stoichiometric coefficient for component i in reaction k.
The energy conservation equation is given by

$$\frac{dU}{dt} = \sum_{k=1}^{n_r} (-\Delta H_k) r_k + \frac{H^{in} - H^{out} + Q^{ext}}{V}, \tag{2}$$

where H^{in} and H^{out} are the fluxes of enthalpy in and out of the cell, and Q^{ext} is a heat source/sink term due to external heating or cooling, which is given by

$$Q^{ext} = U_a (T_r - T),$$

where T_r is the heating/cooling temperature and U_a is an overall heat transfer coefficient.
Finally, the total internal energy of the system per unit volume, U, is given by

$$U = (1 - \varphi_v) \rho_r U_r + (\varphi_v - \varphi_f) \rho_s U_s + \varphi_f \sum_{j \in \{o,w,g\}} U_j \rho_j S_j. \tag{3}$$

Since the cell volume is assumed constant, gases are allowed to leak out of the cell when volume changes occur due to chemical reactions and temperature changes. When only components in the gas phase can leave the cell, the molar flow rates in Eqn. (1) may be expressed as

$$F_i^{out} = Q^{out} \rho_g x_{ig},$$

where ρ_g is the gas density and Q^{out} is the volumetric flow rate, given by

$$Q^{out} = k_v \left(P - P^{ext} \right).$$

Here, k_v denotes a valve coefficient.

Since pressure and temperature variations and chemical reactions cause volume changes, a constraint is needed in order for the fluid and solid volumes, V_f and V_s, to match the void volume, V_v. We set

$$V_v = V_f + V_s. \tag{4}$$

The chemical reactions occurring are assumed to be kinetically driven. They are modeled using the Arrhenius rate relations

$$k_k = \alpha_k \exp\left(-\frac{E_k}{RT} \right),$$

where α_k and E_k are the pre-exponential factor and activation energy for reaction k, respectively, and R is the universal gas constant. We will assume that the reactions are first order in all reactants. Oxidation reaction rates for components in the oil phase are modeled as

$$r_k = k_k \cdot P x_g^{O_2} \cdot C_i.$$

For solid components the rate expression is taken to be

$$r_k = k_k \cdot P x_g^{O_2} \cdot C_s.$$

Finally, for cracking/pyrolysis reactions, we use

$$r_k = k_k \cdot C_i.$$

Reaction kinetic parameters for ISC models are normally obtained from laboratory experiments. It is still an open question how to upscale kinetics from the laboratory scale to the reservoir scale. Lacking better alternatives, we will use the laboratory parameters directly, as is generally done.

We assume that components partition into at most two phases, that components in the solid phase exist exclusively in this phase, and that the water phase consists of water only. For simplicity, the equilibrium between hydrocarbon components in the gas and oil phases is described by

$$K_i = \frac{x_{ig}}{x_{io}},$$

where the equilibrium factors are assumed to vary only with pressure and temperature according to

$$K_i = \frac{a_1}{P} \cdot \exp\left(\frac{a_2}{T - a_3} \right),$$

with a_1, a_2 and a_3 correlation constants. Using this simplified phase equilibrium description the flash calculation reduces to solving the Rachford-Rice equation for the molar gas phase fraction, β, given by

$$\sum_{i \in \{hc\}} (x_{ig} - x_{io}) = \sum_{i \in \{hc\}} \frac{C_i (K_i - 1)}{1 - \beta + \beta K_i}. \tag{5}$$

The assumption that the equilibrium factors do not vary with composition is generally valid away from critical regions. The simplified phase description is a good starting point for model and algorithm development. In future work, we plan to extend the model with an equation-of-state (EoS) based phase equilibrium description.

As primary variables we choose temperature, pressure and the n_c overall component concentrations. The VKC equations, (1), (2) and (4), comprise a set of differential-algebraic equations (DAEs). Including the total internal energy as a variable and Eqn. (3) as an extra constraint, we can write the DAE in semi-explicit form. In this formulation, temperature is aligned with the energy constraint (3) and pressure with the volume constraint (4). In multi-phase regions Eqn. (5) is appended to the equation system and aligned with the gas phase fraction. The two ISC reaction models discussed in the previous section are used for testing and benchmarking.

2.2.4 Choice of integration method for reactions

The kinetic cell equations consist of coupled ordinary differential equations (ODEs) and algebraic equations (AEs). We will consider the general form of the resulting differential algebraic equation (DAE) system

$$\mathbf{M} \frac{d\mathbf{u}}{dt} = \mathbf{f}(t, \mathbf{u}), \qquad \mathbf{u}(t_0) = \mathbf{u}_0, \tag{6}$$

where $\mathbf{u} \in \mathbb{R}^m$ is a vector of state variables depending on t, and $\mathbf{f}$ is a vector function mapping $\mathbb{R} \times \mathbb{R}^m$ into $\mathbb{R}^m$. For the kinetic cell model the mass matrix, $\mathbf{M} \in \mathbb{R}^m \times \mathbb{R}^m$, is simply a diagonal matrix. The diagonal elements M_{ii} are one when the ith equation is a differential equation, and zero when it is algebraic. The initial conditions are assumed consistent with the algebraic constraints. In the VKC model, the right-hand-side functions may be discontinuous because of phase changes. The number of elements in the state vector is typically between 10 and 20. We require results to have between 2 and 4 correct significant digits (relative accuracy between $10^{-2} - 10^{-4}$). This requirement is based on our expectations of the temporal and spatial accuracies in other parts of the equations. As we will show later, the performance of the proposed methods is not sensitive to small variations in the required accuracy.

The short integration intervals between operator splitting updates in the VKC and the stiff reaction kinetics are the primary characteristics guiding our choice of integration method. When

integrating from t_n to t_{n+1} one-step methods only use state information from t_n whereas multi-step methods use information from several previous integration steps, t_{n-j}, with $j = 0,1,\ldots,s$ (Hairer *et al.*, 1992). Multi-step methods are often implemented in a variable order formulation in which they start out at low order and with small time steps. They build up to higher order as more information becomes available, and are, at that stage, generally outperforming one-step methods. This suggests that multi-step methods are attractive for problems with long and smooth integration intervals, whereas one-step methods are preferable on shorter intervals and on problems with frequent discontinuities in the solution. Hence, we will focus here on one-step methods. Of the one-step methods, we limit ourselves to the family of Runge-Kutta (RK) methods, which are well studied in the literature. Other classes of methods could be considered also, such as the Rosenbrock methods (Rosenbrock, 1963) or the extrapolation methods (Hairer *et al.*, 1992), but we will consider Runge-Kutta methods exclusively in this paper. For stability reasons we shall further limit our search to implicit RK methods that allow efficient integration of the stiff kinetics.

A general s-stage Runge-Kutta scheme for solving the DAE system (6) may be expressed as

$$\begin{aligned}
\mathbf{M}\mathbf{U}_i &= \mathbf{M}\mathbf{u}_n + \Delta t_n \sum_{j=1}^{s} a_{ij}\mathbf{f}(t_n + c_j \Delta t_n, \mathbf{U}_j), \\
\mathbf{M}\mathbf{u}_{n+1} &= \mathbf{M}\mathbf{u}_n + \Delta t_n \sum_{i=1}^{s} b_i \mathbf{f}(t_n + c_i \Delta t_n, \mathbf{U}_i).
\end{aligned}$$

Here, $\mathbf{U}_i$ denotes the solution at the ith, $(i = 1,\ldots,s)$, internal stage of integration step n and Δt_n denotes the time step length.

Runge-Kutta methods are classified according to the structure of their Butcher tableau, given by

$$\begin{bmatrix}
c_1 & a_{11} & a_{12} & \ldots & a_{1s} \\
c_2 & a_{21} & a_{22} & \ldots & a_{2s} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
c_s & a_{s1} & a_{s2} & \ldots & a_{ss} \\
\mathbf{u}_{n+1} & b_1 & b_2 & \ldots & b_s
\end{bmatrix} = \begin{bmatrix} \mathbf{c} & \mathbf{A} \\ & \mathbf{b}^T \end{bmatrix}.$$

For explicit methods (ERK), the matrix $\mathbf{A}$ is strictly lower triangular, which implies that all the internal stages can be calculated explicitly, making ERK methods computationally fast and straightforward to implement. However, ERK methods generally have poor stability properties, which make them unsuitable for stiff problems. The four remaining subclasses of Runge-Kutta methods are all implicit, that is, the values of the internal stages are no longer calculated explicitly from the values of the previous stages. Each integration step of an implicit method requires the solution of a system of ms nonlinear equations. Normally, an iterative method, such as Newton's method, is applied. For diagonally implicit RK (DIRK) methods, the stage values can be calculated sequentially. Hence, the computational costs of DIRK methods are lower than those of fully implicit RK (FIRK) methods, for which all ms equations must be solved simultaneously. If all diagonal elements of $\mathbf{A}$ are identical and its upper diagonal elements are zero, the method is said to be a singly diagonally implicit (SDIRK). If the first stage of an SDIRK method is

explicit, the method is said to be explicit singly diagonally implicit (ESDIRK).

For our purposes, the ESDIRK methods, for which the general form of the Butcher tableau can be found in e.g. (Kvaerno, 2004), are attractive. The diagonal structure of these methods allows sequential evaluation of the internal stages, and because the diagonal elements are equal, the iteration matrix for solving the nonlinear stage equations need only be evaluated and factorized once per integration step. ESDIRK methods can be constructed such that they are both A- and L-stable as well as stiffly accurate (Hairer & Warner, 1996). Stiffly accurate methods avoid the order reduction phenomenon observed by Prothero & Robinson (1974) when applied to stiff ODEs. Also, the explicit first stage of ESDIRK methods ensures high stage order (≥ 2) which is important for the order of accuracy in the algebraic components of the DAE. An additional advantage of having high stage order is that it allows the construction of high order interpolants to be used for generating output between mesh-points. We will exploit these interpolants when constructing an algorithm for locating the discontinuities that occur due to phase changes.

The Butcher tableau for ESDIRK methods takes the form

$$\begin{bmatrix} 0 & 0 & 0 & & & & \\ c_2 & a_{21} & \gamma & 0 & & & \\ c_3 & a_{31} & a_{32} & \gamma & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \\ c_{s-1} & a_{s-1,1} & a_{s-1,2} & a_{s-1,3} & \cdots & \gamma & 0 \\ 1 & b_1 & b_2 & b_3 & \cdots & b_{s-1} & \gamma \\ \mathbf{u}_{n+1} & b_1 & b_2 & b_3 & \cdots & b_{s-1} & \gamma \end{bmatrix}.$$

Alexander (2003) proposed a four stage ESDIRK method of order 3 with an embedded error estimator of order 4. Williams *et al.* (2002) also published a four stage ESDIRK method suited for index 2 DAEs, but with orders 2 and 3 for the error estimator and advancing method, respectively. Finally, Kvaerno (2004) recently published a range of ESDIRK methods of orders 3 to 5 emphasizing strong stability properties of both the error estimator and the advancing method.

We implemented four ESDIRK methods of orders 2-5. The lowest order method is simply the trapezoidal rule. When combined with the implicit Euler method we have an embedded pair of methods of orders 1 and 2. We refer to this method as ESDIRK12 and use a similar notation for higher order ESDIRK schemes. The bulk of the computational costs in an ESDIRK method is in the solution of the nonlinear algebraic equations that arise in each internal stage. Efficient control of the iterative scheme applied to these equations and of the discretization error by step size adjustments are the two main design challenges.

The step size is adapted based on an error estimate for the lowest order method obtained by subtracting the two solutions of orders p and $p-1$. This error estimate is essentially free, since it involves no additional function evaluations or system solves. Adjusting the step size to meet an accuracy requirement is, in essence, a control problem. We implemented the predictive controller

$$\Delta t_n = \frac{\Delta t_{n-1}}{\Delta t_{n-2}} \left(\frac{\varepsilon}{r_n} \right)^{k_2/(p+1)} \left(\frac{r_{n-1}}{r_n} \right)^{k_1/(p+1)} \Delta t_{n-1},$$

suggested by Gustafsson (1992). k_1 and k_2 are the gain parameters for the controller, while ε is the desired tolerance (including a safety factor). Gustafsson suggests using $k_1 = k_2 = 1$ based on experiments, where the number of integration steps is measured for a given problem as a function of k_1 and k_2. The number of steps required is not sensitive to k_1 and k_2 for the interval $[k_1, k_2] \in 0.5, 1.5\,;\, 0.5, 1.5]$. Within this interval, variations in k_1 and k_2 result in changes in the number of integration steps of less than 5%. Our experiments using the VKC model show similar results, so we choose $k_1 = k_2 = 1$. The variable r_n is the norm of the estimated local error at t_n, given by

$$r_n = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\frac{err_i}{atol_i + rtol_i \cdot |u_i|}\right)^2},$$

in which $atol$ and $rtol$ are (componentwise) absolute and relative error tolerances specified by the user. Experiments have shown that the controller gives a small reduction in the number of failed integration steps along with a smoother variation of step sizes compared to the conventional control law $\Delta t_n = \left(\varepsilon / r_n\right)^{1/(p+1)} \Delta t_{n-1}$ that is commonly implemented in ODE solvers.

The nonlinear equations arising in each internal stage of the ESDIRK methods are solved using a modified Newton's method given by

$$\begin{aligned}
\left[\mathbf{M} - \Delta t \gamma \frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right] \Delta \mathbf{U}_i &= \mathbf{M}\mathbf{u}_n + h\sum_{j=1}^{i-1} a_{ij}\mathbf{f}(t_n + c_j \Delta t, \mathbf{U}_j^{(k)}) - \mathbf{M}\mathbf{U}_i^{(k)}, \\
\mathbf{U}_i^{(k+1)} &= \mathbf{U}_i^{(k)} + \Delta \mathbf{U}_i.
\end{aligned}$$

The iteration matrix $\mathbf{M} - h\gamma \partial \mathbf{f} / \partial \mathbf{u}$ is evaluated and factorized once per integration step. We construct an initial guess for the stage value at each internal stage by continuous extension of the method as suggested in Enright *et al.* (1986). The continuous extension is given by

$$\mathbf{u}(t_n + \theta \Delta t_n) = \mathbf{u}_n + \Delta t_n \sum_{i=1}^{s} b_i^*(\theta)\mathbf{f}(t_n + c_i \Delta t_n, \mathbf{U}_i),$$

where the quadrature weights, b_i^*, depend on θ according to

$$b_i^*(\theta) = b_{i,1}^*\theta + b_{i,2}^*\theta^2 + \ldots + b_{i,p}^*\theta^p.$$

The coefficients for the continuous extension are determined by requiring that the set of order conditions used to determine the coefficients of the ESDIRK method itself is satisfied. A prediction of the stage value at the ith internal stage is then obtained by substituting $\theta_i = 1 + c_i h_{n+1}/h_n$.

The iterations are terminated when

$$\| \mathbf{R}^{(k)} \|_2 \leq \kappa \cdot rtol,$$

with $\mathbf{R}$ the scaled vector of residuals, $rtol$ the relative error tolerance for the local discretization error. The parameter κ is normally chosen between 0.5 and 0.01. For reasons of robustness we chose $\kappa = 0.01$, such that the errors allowed in the iteration process are much smaller and do not interfere with the estimation and control of the local integration error. Experiments show that the cost of the extra iterations needed using $\kappa = 0.01$ compared to, for example, $\kappa = 0.1$, is minor, since the tighter value of κ also reduces the total number of integration steps.

During iterations the convergence rate is estimated as

$$\alpha_k = \frac{\| \mathbf{R}^{(k)} \|_2}{\| \mathbf{R}^{(k-1)} \|_2}.$$

If for some k during iterations $\alpha_k > 1$, the iterations are terminated and the step size decreased. The estimated convergence rate is also used by the step size controller to limit the step size if convergence in the previous step was too slow.

2.2.5 Phase changes

Phase changes in the VKC cause discontinuities in the right-hand-sides of the kinetics equations. Straightforward integration across these discontinuities may lead to non-physical phase changes, poor convergence and repeated step failures.We propose an algorithm for robust detection and location of phase changes by considering the kinetic cell as a discrete event problem. The appearance or disappearance of a fluid phase marks the occurrence of a " discrete event", e.g. a change from a single phase region to a two-phase region, or vice-versa. The time of the phase change can not be determined a priori. The detection of a phase change and subsequent location of the exact time of change are the main components of the proposed discrete event algorithm.

We will represent the discrete event DAE system in the form

$$\begin{aligned} \frac{d\mathbf{v}}{dt} &= \mathbf{f}(t, \mathbf{v}, \mathbf{w}), \qquad \mathbf{v}(t_0) = \mathbf{v}_0, \\ 0 &= \mathbf{g}(t, \mathbf{v}, \mathbf{w}), \qquad \mathbf{w}(t_0) = \mathbf{w}_0, \\ 0 &< q_j(t, \mathbf{v}, \mathbf{w}), \qquad j = 1, \ldots, n_{ev}. \end{aligned}$$

The DAE system has been reformulated in semi-explicit form. Now, $\mathbf{v} \in \mathbb{R}^{m_d}$ is the vector of differential variables and $\mathbf{w} \in \mathbb{R}^{m_a}$ is the vector of `algebraic' variables. $\mathbf{f}$ and $\mathbf{g}$ are vector functions mapping $\mathbb{R} \times \mathbb{R}^{m_d} \times \mathbb{R}^{m_a}$ into $\mathbb{R}^{m_d}$ and $\mathbb{R}^{m_a}$, respectively. We will assume that $\partial \mathbf{g} / \partial \mathbf{v}$ is non-singular (index one DAE). The q_j's are event functions associated with the current system state and n_{ev} denotes the number of event functions. In general, the system may change between many different states. If the attainable states are indexed as $p \in \mathcal{I}_{state}$ then each state is represented with $\{m_d, m_a, n_{ev}\}_p$.

To illustrate the concepts we consider the discrete event formulation of the VKC. We neglect changes associated with the water phase and consider only the two-phase equilibrium between hydrocarbon components in the oil and gas phases. Three attainable states exist for this system: (1) single phase oil, (2) single phase gas and (3) two-phase oil-gas. In the two-phase region, the active equations for the VKC are

$$
\begin{aligned}
\frac{dC_i}{dt} &= \sum_{k=1}^{n_r} A_{ik} r_k + \frac{F_i^{in} - F_i^{out}}{V}, \qquad i = 1, \ldots, n_c, \\
\frac{dU}{dt} &= \sum_{k=1}^{n_r} (-\Delta H_k) r_k + \frac{H^{in} - H^{out} + Q^{ext}}{V}, \\
0 &= U - (1-\varphi_v)\rho_r U_r + (\varphi_v - \varphi_f)\rho_s U_s + \varphi_f \sum_j U_j \rho_j S_j, \\
0 &= V_v - V_f - V_s, \\
0 &= \sum_{i \in \{hc\}} \frac{C_i (K_i - 1)}{1 - \beta + \beta K_i}.
\end{aligned}
$$

Here, $m_d = n_c + 1$, $m_a = 3$ and $n_{ev} = 2$. The two event functions associated with the two-phase region are

$$
\begin{aligned}
0 &< \sum_{i \in \{hc\}} C_i (1 - K_i), \\
0 &< \sum_{i \in \{hc\}} C_i \left(1 - \frac{1}{K_i}\right),
\end{aligned}
$$

corresponding to disappearance of the gas phase and the oil phase, respectively. Our event detection algorithm is based on the principle of discontinuity locking (Park & Barton, 1996). Within each integration step the system is locked in its current state, even if one or more event functions are satisfied. The event functions are evaluated at the end of each step, and if any of them are satisfied, the exact time of occurrence is located. The approach is based on the assumption that the system of equations is mathematically well behaved in a small neighborhood of the phase changes.

Since the changes often occur between mesh points, we need to interpolate the differential and algebraic variables of the DAE in order to evaluate the event functions. For differential variables we use the continuous extension provided by the ESDIRK methods. The algebraic variables could, in principle, also be interpolated by a suitable polynomial. However, after location of the event, a consistent initialization calculation is required to restart the integration in the new system state. That is, based on values for the differential variables, the algebraic constraints are solved to find consistent initial values for the algebraic variables. This consistency calculation may result in the numerical phenomenon referred to as *discontinuity sticking*. The problem is illustrated in Figure 1. Assume that the event function $q^*(t, \mathbf{v}, \mathbf{w}) > 0$ was satisfied in the integration step $[t_n, t_{n+1}]$. The exact event time, t^*, may than be located by solving the scalar equation

$$
q^*(t^*, \mathbf{v}^p(t^*), \mathbf{w}^p(t^*)) = \delta,
$$

where $\mathbf{v}^p(t)$ and $\mathbf{w}^p(t)$ are interpolating polynomials for the differential and algebraic variables, respectively. δ is a small tolerance employed to ensure $q^* > 0$ at t^*. However, a consistent initialization at t^* may result in values of the algebraic variables that are different from $\mathbf{w}^p(t^*)$ if t^* does not coincide with the mesh points or the internal quadrature nodes of the ESDIRK scheme

$$\mathbf{w}^p(t^*) \neq \mathbf{w}(t^*), \qquad \text{if} \qquad t^* \neq t_n + c_i h, \quad i = 1, \ldots, s.$$

Consequently, the value of the event function may have changed

$$q^*(t^*, \mathbf{v}^p(t^*), \mathbf{w}^p(t^*)) \neq q^*(t^*, \mathbf{v}^p(t^*), \mathbf{w}(t^*)),$$

and the event function may no longer be satisfied ($q^*(t^*, \mathbf{v}^p(t^*), \mathbf{w}(t^*)) < 0$). If this is the case, the same event is detected again immediately in the next integration step.
To avoid discontinuity sticking problems, we locate the phase changes by solving the system of equations

$$\begin{aligned} \mathbf{g}(t^*, \mathbf{v}^p(t^*), \mathbf{w}(t^*)) &= 0, \\ q^*(t^*, \mathbf{v}^p(t^*), \mathbf{w}(t^*)) &= \delta, \end{aligned} \tag{7}$$

where the unknowns are the algebraic variables, $\mathbf{w}$, and the event time, t^*. Only few Newton iterations are required to solve the system (7), since good starting guesses are available.

3. Experimental

No experimental work is reported this quarter.

4. Results and discussion

4.1 Ramped temperature experiment

As a first example, we illustrate the VKC by simulating a ramped temperature experiment using the minimal reaction model. Ramped temperature experiments are often carried out in the laboratory to determine burning characteristics for different oils.

Initial conditions (moles)		**Operational conditions**	
Water	0	Total cell volume	0.0014 m^3
Light oil	0	Air feed rate	$2.78.10^{-6}$ m^3/s
Heavy oil	0.55	Rock porosity	0.4
Oxygen	0	Heat transfer coefficient	60 kJ/(mole K)
Inert gas	0.45	Valve coefficient	$2.74 \cdot 10^{-11} m^3/(s \cdot Pa)$
Coke	0	Temperature	373 K
		Pressure	13.88 MPa

Table 2. Initial and operational conditions for simulating the ramped temperature experiment using the VKC.

The simulated setup consists of an oil sample placed in the kinetic cell which is heated externally from 373K to 873K over a period of 10 hours with a constant feed of air of $2.78 \cdot 10^{-6}\,\text{m}^3/\text{s}$. Initial and operational conditions are summarized in Table 2. Figure 2 shows the cell temperature, oxygen consumption rate and concentrations of light oil, heavy oil, coke and oxygen at 3 different concentrations of oxygen in the feed.

The minimal reaction model used in this experiment does not include LTO reactions in the traditional sense of oxygen addition reactions, but we still observe two peaks in oxygen consumption rate originating from direct oxidation of the oil based components and oxidation of coke which occurs at a higher temperature. The variation with oxygen feed concentration shows that, as expected, a low oxygen concentration promotes cracking of heavy oil whereas a high oxygen concentration favors the direct oxidation which leads to a significant increase in temperature.

In the second example we study the influence of the activation energy. We again simulate the ramped temperature experiment with an oxygen feed concentration of 21%. We perform a base simulation using the original reaction parameters and two perturbed simulations, in which we first increase and then decrease activation energies for all reactions by a factor of 2. We choose the pre-exponential factors such that the nominal and perturbed reaction rate constants coincide at a specified temperature, so that

$$\alpha_{perturbed} = \alpha_{nominal} \cdot \exp\left(\frac{1}{RT_{coin.}} \left(E_{perturbed} - E_{nominal} \right) \right).$$

Here, $E_{nominal}$ and $E_{perturbed}$ refer to the nominal and perturbed activation energies, respectively, and $T_{coin.}$ is the temperature at which the two rate expressions coincide. We choose $T_{coin.} = 623\text{K}$, which corresponds to the mean temperature during the simulation.

Figure 3 shows the cell temperature, oxygen consumption rate and concentrations of light oil, heavy oil, coke and oxygen at the three different activation energies. Again, we see two peaks in oxygen consumption rate originating from direct oxidation of the oil components and oxidation of coke. Increasing the activation energies results in an initial delay of the reaction onset time, but a more rapid oxidation of oil and coke after onset, which is also reflected in the increase in oxygen consumption rate.

The phase change algorithm improves the integration robustness when crossing phase boundaries. For the case with 20% oxygen in the feed, a change from single phase oil to two-phase oil-gas occurs at $t = 3.15\text{hrs}$. Attempting to integrate directly across the phase change results in repeated step failures in the solver. Each time the solver attempts a step across the phase boundary, a convergence or error test failure results forcing the solver to reduce the step size. The solver fails 8 times and reduces the step size by a factor of 100 before stepping across successfully. Using the phase change detection algorithm, the change is detected and consistently located without failed steps. Initialization in the new two-phase region, however, results in 4 step failures and a step size reduction by a factor of approximately 30. Completely avoiding step failures and step size reductions is difficult since a change into the two-phase region changes the dynamics of the problem.

4.2 Performance comparison

To evaluate the performance of the new, tailored ESDIRK implementations we have compared them to two stiff ODE solvers that are commonly used: DASSL (Petzold, 1982) and LSODE (Hindmarsh, 1983). Both solvers are based on backward differentiation formulas (BDF), and are variable order. We show results for four ESDIRK12, 23, 34 and 45.

We split the total integration interval into N subintervals. We assume the pressure is constant throughout the entire integration. The temperature is kept fixed in each subinterval. Its value is obtained by sampling the characteristic ISC temperature profile as shown in Figure 4. Thus, we mimic implementation of the solvers within an operator splitting environment for the full ISC equations.

Component	**Mole fraction**
Water	0
Inert oil	0.0005
Oxidized resins/aromatics	0
Oxidized saturates	0
Asphaltenes	0.0229
Resins	0.0914
Aromatics	0.2680
Saturates	0.5625
Light oil	0.0547
Carbon dioxide	0
Nitrogen	0
Oxygen	0
Oxidized asphaltenes	0
Pyrolysis coke	0

Table 3. Initial overall oil composition for the SARA based model.

We use the SARA based reaction model. In terms of number of components and reactions we think that this model best represents the requirements in realistic ISC simulations. We assume ideal fluid phases and use a pressure and temperature correlation for the equilibrium K-factors. Initial compositions are listed in Table 3. A pressure of 2.023MPa is used in all subintervals . Air is cycled through the cell at a constant rate. The total simulation time is 100hrs and two experiments are carried out using $N = 25$ and $N = 100$ subintervals, respectively.

The numerical results are compared to a very accurate reference solution computed by ESDIRK34 using $atol = rtol = 10^{-14}$. The measure of accuracy is based on the max-norm of the relative error at the end of the integration interval. The accuracy is represented as the minimum number of significant correct digits, SCD, in the solution, defined as

$$\text{SCD} := -\log_{10}\left[\max_i \left|\frac{u_i(t_{end}) - u_i^{ref}(t_{end})}{u_i^{ref}(t_{end})}\right|\right].$$

Solver comparisons are presented as work-precision diagrams where efficiency is measured using CPU time. Thus, to produce the diagrams the problem is solved using a range of input tolerances for each of the solvers tested. All codes are compiled using the Compaq Visual Fortran compiler with the same optimization settings. The CPU time for each case is averaged over 25 runs on a Pentium 4, 3GHz PC with 512MB RAM. The input tolerances are chosen as $atol = rtol = 10^{-(1+j/3)}$, $j = 0,\ldots,21$. The initial integration step size is computed internally by the solvers. For this purpose, the ESDIRK solvers use the algorithm suggested by Hairer & Wanner (1996). Between subintervals the last step of the current interval is passed as a guess for the first step in the next interval. All solvers require the Jacobian of the ODE right-hand-side functions. These are computed analytically using automatic differentiation.

Figure 5 shows the work-precision diagrams when solving the SARA based reaction model using 25 and 100 subintervals. The resulting accuracies in the solution are, of course, different from the local tolerances provided as input to the solvers that govern the local error and step size control. In the high accuracy range (4-6 significant correct digits) the methods perform according to their order: The high order methods (DASSL, LSODE and ESDIRK45 (fourth order)) outperform the lower order methods (e.g. ESDIRK12 (first order) and ESDIRK23 (second order)). However, we are mainly interested in the accuracy range corresponding to 2-4 significant correct digits, since we expect the accuracies obtained in other parts of the equation to fall in this range. The low and intermediate order methods perform best in this range. For example, with 100 subintervals and 3 significant correct digits the ESDIRK23 solver is two times faster than LSODE and 3-4 times faster than DASSL.

The work-precision curves for the ESDIRK solvers settle at a constant level in the low accuracy range instead of continuing the linear trend found for high accuracies. Moreover, the behavior in this range is more erratic. This can be partly attributed to the error and convergence controller. Consider, for example, ESDIRK45. An input tolerance level of $atol = rtol = 10^{-1}$ gives better than expected results (SCD = 2), but this comes at the cost of increased computation time. Inspection shows that the step size is limited by convergence in the nonlinear solver and not by the local error. The relatively large step sizes allowed by the error controller lead to convergence failures in the nonlinear solver, which then forces a step size reduction. This shows that ODE error control devices perform best when operating at low tolerance levels.

Comparing the results for $N = 25$ and $N = 100$ shows that all solvers need more time when the integration is interrupted frequently. The BDF methods have the largest restart overhead because, when restarted, they revert to first order requiring small time steps, and slowly build up higher order information. Both DASSL and LSODE manage to integrate across the phase changes. Inspection of the process shows, however, that both methods experience a high number of step failures, and lower their order before successfully stepping across. The loss in efficiency associated with these step failures alone is difficult to isolate but is reflected in the results shown in Figure 5. The differences in performance are relatively small when using only 25 subintervals. The advantage of one-step methods over multi-step methods is, however, clearly observed when the number of subintervals is increased, the difference between DASSL and the ESDIRK solvers being the most notable.

Figure 5 also shows that none of the solvers are very sensitive to the required accuracy. Although with some irregularities in the low accuracy range, the computational cost increases linearly with the required accuracy in the solution.

4.3 Discussion

The work presented in this paper addresses time integration of reactive, compositional and thermal porous media processes, and focusses in particular on the design of specialized solvers for integration of stiff chemical kinetics subject to phase equilibrium constraints.

To facilitate the algorithmic development we constructed a virtual kinetic cell. The VKC allows us to study the kinetics and phase behavior of reactive transport processes in an isolated setting. Through user prescribed inflow and outflow boundary conditions, interactions with the reservoir can be studied. Two ISC reaction models with 6 and 14 components, respectively, were implemented and tested. The VKC was demonstrated with a ramped temperature experiment using the 6-component model. Ramped temperature experiments are routinely carried out in the laboratory. In future work we will validate the simulation results against experimental data. Although not explored in this work, the VKC can also be applied to help determine optimal experimental conditions in the design phase.

Because of the stiff chemical kinetics, the low to medium accuracy required in the solution, and the short integration intervals experienced in an operator splitting environment, we selected the class of ESDIRK methods for the temporal integration of the chemical reactions. To increase computational efficiency, we select the time step size with a predictive controller for the local integration error, and generate starting guesses for the modified Newton iterations using continuous extensions of the ESDIRK methods. Performance comparisons between the ESDIRK solvers and the popular stiff ODE solvers DASSL and LSODE, show that the higher order ESDIRK solvers are at least comparable if not superior in terms of computational speed, especially over the relatively short integration intervals that are generally required in an operator splitting environment. If we aim for two to four correct digits in the solution, which we expect to be of practical interest in ISC simulations, the low and medium order ESDIRK methods (ESDIRK12, ESDIRK23 and ESDIRK34) are two times faster than LSODE and three or four times faster than DASSL. Apart from the ESDIRK methods, other classes of integration methods could be of interest, such as the Rosenbrock methods Sandu *et al.* (1997), which we will consider in future work.

We extended the ESDIRK methods to handle the discontinuities that arise due to phase changes. Phase changes are detected by monitoring sign changes of special event functions. The proposed algorithm proved robust in detecting and locating phase changes, and lowered the number of convergence and error test failures by more than 50%. The DASSL and LSODE solvers that are not equipped with a phase change algorithm will, in most cases, successfully integrate across the change after repeated step size reductions caused by convergence and error test failures. Overall efficiency gains from using the phase change algorithm when measured over a long integration interval with only one or two phase changes occurring are modest, but the improved robustness near phase boundaries is certainly valuable.

The VKC model uses a simplified constant K-value phase equilibrium description. We are currently working on the extension to full equation-of-state based phase equilibrium. In a rigorous flash algorithm the phase state depends on the outcome of a stability analysis which cannot be expressed in closed form and, hence, does not directly fit into the framework of event functions. For transitions from a two-phase region to a single phase region, we can still detect changes by monitoring phase fractions. In the reverse situation the problem is now more complicated. In principle, this would require a stability analysis to be performed each time conditions change to check if the new single phase state is unstable. A possible solution is to exploit shadow regions as

in Rasmussen *et al*. (2006) to speed up flash calculations, which will reduce the number of stability analyses required.

5. Conclusions

The virtual kinetic cell (VKC) is a novel and useful tool for analyzing kinetics and phase behavior in reactive, compositional and thermal processes. It can provide simulation support for laboratory kinetic cell experiments, and be used for studying interactions between kinetics and phase behavior as well as interactions with the reservoir through specialized boundary conditions.

ESDIRK methods are well-suited for integration of the stiff kinetics due to their strong stability properties. In particular, when implemented in an operator splitting environment, the ESDIRK methods outperform stiff multi-step methods. Experiments show that the methods lead to 50-75% reductions in computational costs compared to multi-step methods.

Phase change detection using a discrete event system approach is an attractive tool which significantly reduces integration step failures and therefore computational time.

Acknowledgements

The authors would like to thank Norman Freitag, Saskatchewan Research Council, Canada, for generously providing detailed reaction kinetic data for the SARA based reaction model.

References

R. Alexander, Design and Implementation of DIRK Integrators for Stiff Systems, Applied Numerical Mathematics, 46, 2003.

W. H. Enright, K. R. Jackson, S. P. Norsett and P. G. Thomsen, Interpolants for Runge-Kutta Formulas, ACM Transactions on Mathematical Software, 12(3), 1986.

N. P. Freitag and B. Verkoczy, Low-Temperature Oxidation of Oils in Terms of {SARA} Fractions: Why Simple Reaction Models Don't Work, Journal of Canadian Petroleum Technology, 44 (2), 2005.

N. P. Freitag and D. R. Exelby, A SARA-Based Model for Simulating the Pyrolysis Reactions That Occur in High-Temperature EOR Processes, Journal of Canadian Petroleum Technology, 45(3), 2006.

K. Gustafsson, Control of Error and Convergence in ODE Solvers, PhD thesis, Department of Automatic Control, Lund Institute of Technology, 1992.

E. Hairer, S.P. Norsett and G. Wanner, Solving Ordinary Differential Equations I, 2nd Edition, Springer, 1992.

E. Hairer and G. Wanner, Solving Ordinary Differential Equations II, 2nd Edition, Springer, 1996.

A. C. Hindmarsh, ODEPACK, A Generalized Collection of ODE Solvers, in Scientific Computing IMACS Transactions on Scientific Computing, Vol. 1, North-Holland, Amsterdam, 1983.

A. Kvaerno, Singly Diagonally Implicit Runge-Kutta Methods With an Explicit First Stage, BIT Numerical Mathematics, 44, 2004.

T. Park and P. I. Barton, State Event Location in Differential-Algebraic Models, ACM Transactions on Modeling and Computer Simulation, 6(2), 1996.

L. R. Petzold, DASSL: A Differential/Algebraic System Solver, 10th IMACS World Congress on System Simulation and Scientific Computation, 1982.

A. Prothero and A. Robinson, On the Stability and Accuracy of One-Step Methods for Solving Stiff Systems of Ordinary Differential Equations}, Mathematics of Computation, 28(125), 1974.

C. P. Rasmussen, K. Krejbjerg, M. L. Michelsen and K. E. Bjurstrom, Increasing the Computational Speed of Flash Calculations With Applications for Compositional, Transient Simulations, SPE Reservoir Evaluation & Engineering, 2006.

Y. Ren, N. P. Freitag and N. Mahinpey, A Simple Kinetic Model for Coke Combustion During an In Situ Combustion ISC Process, Presented at the 6th Canadian International Petroleum Conference, Calgary, Alberta, June 7-9, 2005.

H. H. Rosenbrock, Some General Implicit Processes for Numerical Solution of Differential Equations, Computer Journal, 6(4), 1963.

A. Sandu, J. G. Verwer, M. Van Loon, G. R. Carmichael, F. A. Potra, D. Dabdub and J. H. Seinfeld, Benchmarking Stiff ODE Solvers for Atmospheric Chemistry Problems: I. Implicit Vs. Explicit, Atmospheric Environment, 31(19), 1997.

A. Sandu, J. G. Verwer, J. G. Bloom, E. J. Spee, G. R. Carmichael and F. A. Potra, Benchmarking Stiff ODE Solvers for Atmospheric Chemistry Problems: II. Rosenbrock Solvers, Atmospheric Environment, 31(20), 1997.

STARS, Advanced Process and Thermal Reservoir Simulator, Computer Modelling Group Ltd., 2004

R. Younis and M. Gerritsen, Multiscale Process Coupling by Adaptive Fractional Stepping: An In-Situ Combustion Model, SPE 93458: Presented at the 2006 SPE/DOE Symposium on Improved Oil Recovery, Tulsa, Oklahoma, April 22-26, 2006.

R. Williams, K. Burrage, I. Cameron and M. Kerr, A Four-Stage Index 2 Diagonal Implicit Runge-Kutta Method, Applied Numerical Analysis, 40, 2002.

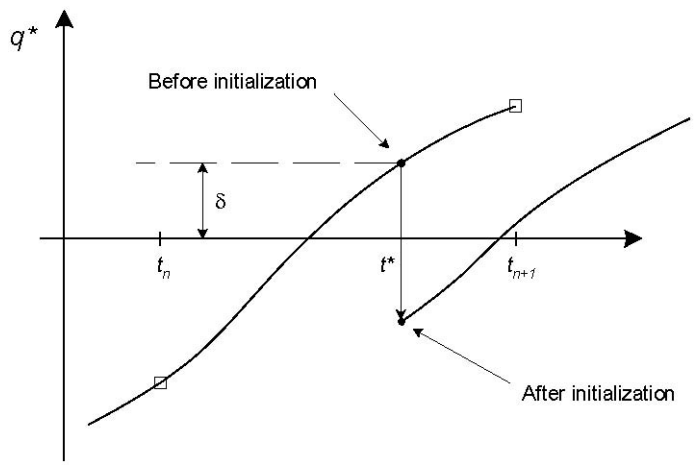

Figure 1. Illustration of discontinuity sticking resulting from inconsistent algebraic variables.

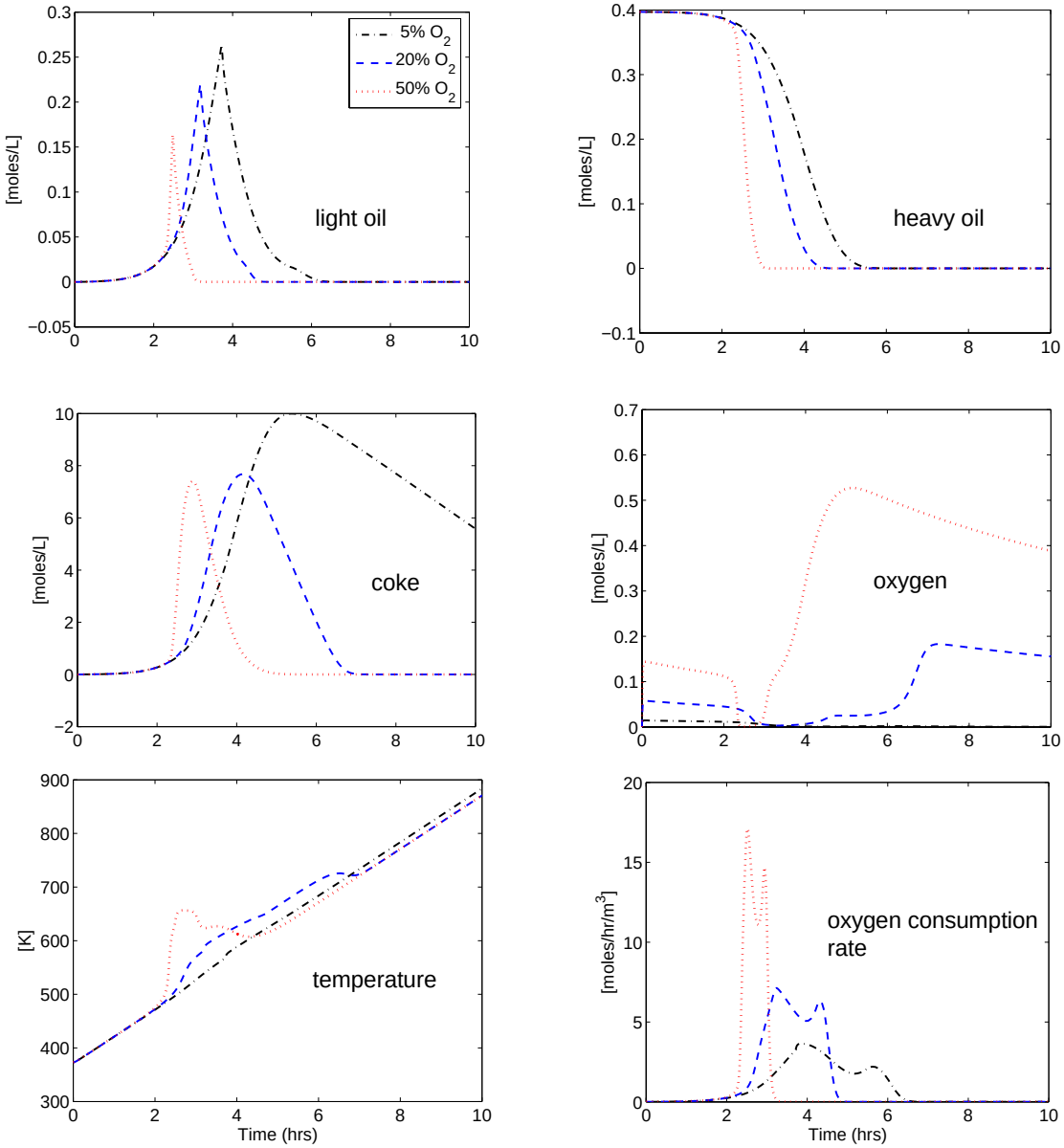

Figure 2. Simulation results from a ramped temperature experiment using the minimal reaction model. The temperature is raised from 373K to 873K over a period of 10 hours. Cell temperature, oxygen consumption rate and component concentrations are shown for 3 different oxygen feed concentrations.

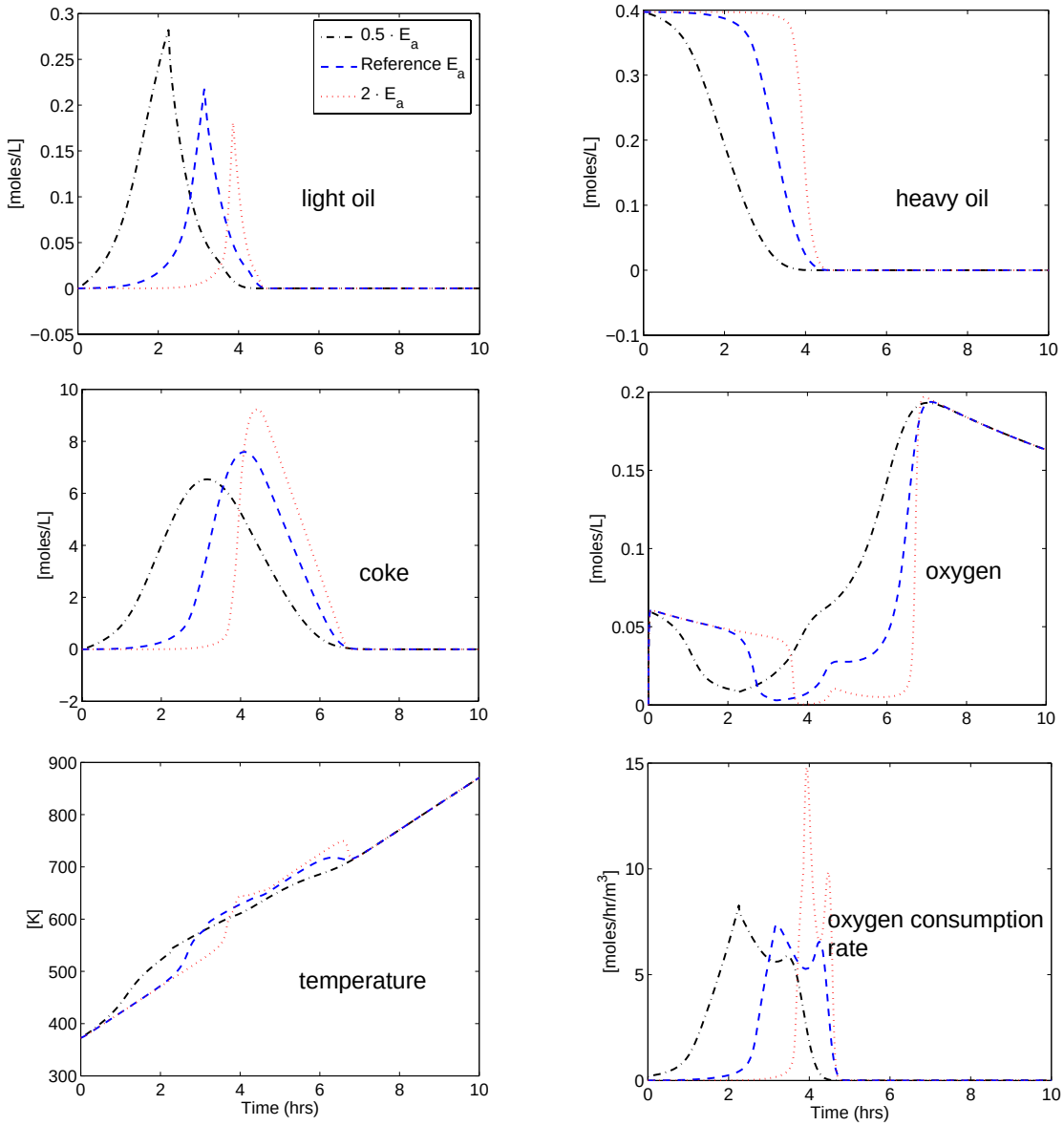

Figure 3. Simulation results from a ramped temperature experiment using the minimal reaction model. The temperature is raised from 373K to 873K over a period of 10 hours. Cell temperature, oxygen consumption rate and component concentrations are shown for at 3 different reaction activation energies.

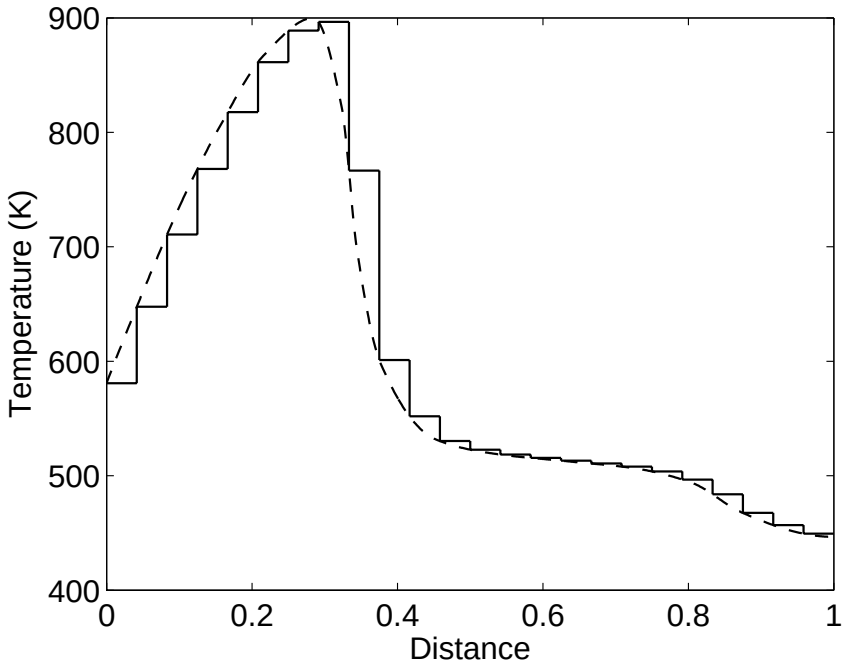

Figure 4. The temperatures for the integration subintervals in the performance comparison are obtained by sampling the characteristic ISC temperature profile. The continuous temperature profile is shown along with a zero-order parametrization corresponding to $N = 25$ equidistant subintervals.

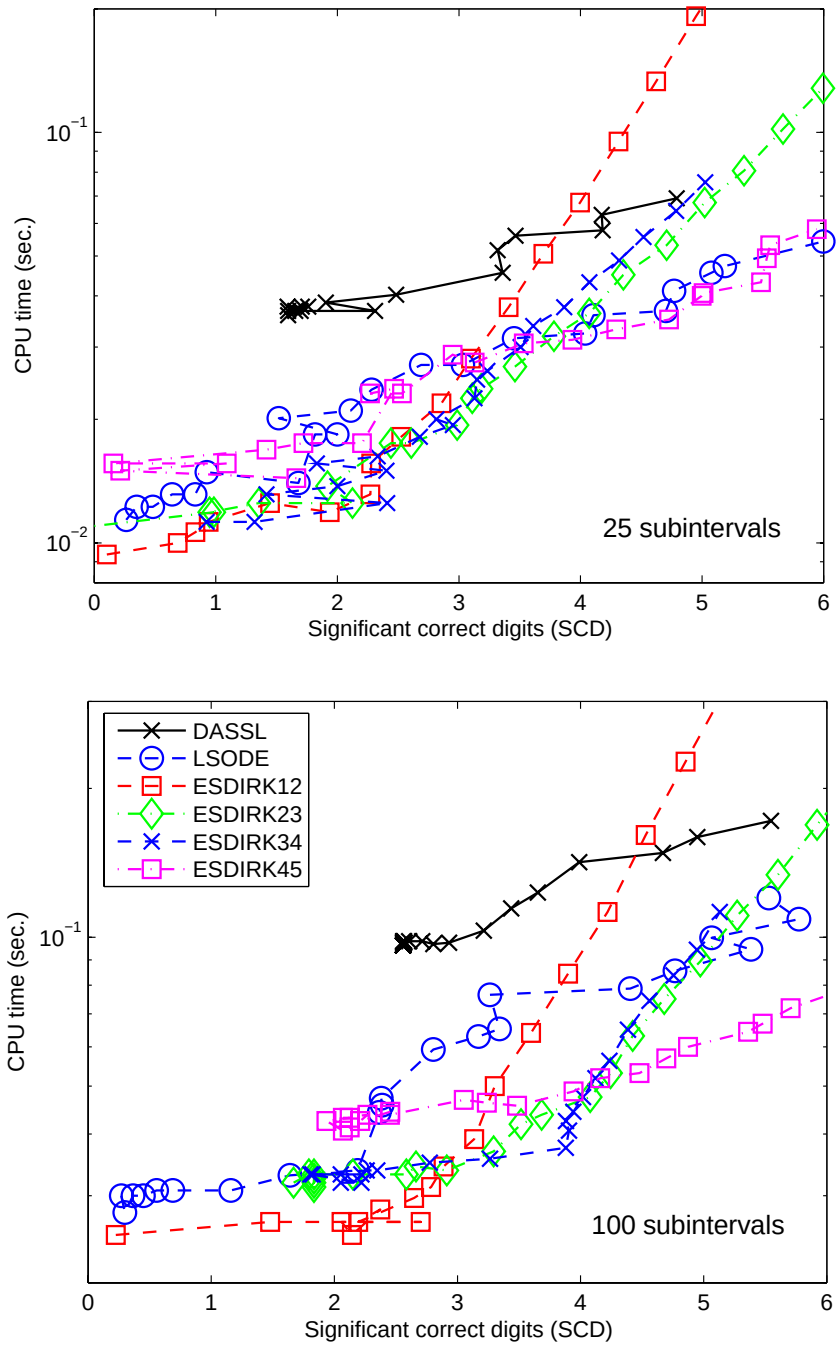

Figure 5. Work-precision diagrams for the SARA based ISC reaction model. The tailored ESDIRK implementations are compared to the widely used DASSL and LSODE codes. Comparisons are made for 25 and 100 subintervals.