Transport in Porous Media manuscript No.
(will be inserted by the editor)

Morten R. Kristensen · Margot Gerritsen · Per G. Thomsen · Michael L. Michelsen · Erling H. Stenby

Efficient Reaction Integration for In-Situ Combustion Simulation

May 30, 2006

Abstract In numerical simulations of the in-situ combustion enhanced oil recovery process, a major task is the integration of the stiff systems of differential-algebraic equations describing chemical reactions and phase equilibrium. It is therefore of great importance to identify suitable integration methods and to design efficient and robust solvers that are tailored to the specific application. Using a time-stepping methodology based on operator splitting we propose the use of implicit one-step methods of the ESDIRK class for integration of reactions. To facilitate the algorithmic development we construct a 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. Through bench-

M.R. Kristensen
Department of Chemical Engineering, Technical University of Denmark, Building 229, DK-2800 Kgs. Lyngby, Denmark
E-mail: mrk@kt.dtu.dk

M. Gerritsen
Department of Petroleum Engineering, Stanford University, Green Earth Sciences Building, Stanford, CA, 94305-2220, U.S.A.
E-mail: margot.gerritsen@stanford.edu

P.G. Thomsen
Informatics and Mathematical Modelling, Technical University of Denmark, Building 305, DK-2800 Kgs. Lyngby, Denmark
E-mail: pgt@imm.dtu.dk

M.L. Michelsen
Department of Chemical Engineering, Technical University of Denmark, Building 229, DK-2800 Kgs. Lyngby, Denmark
E-mail: mlm@kt.dtu.dk

E.H. Stenby
Department of Chemical Engineering, Technical University of Denmark, Building 229, DK-2800 Kgs. Lyngby, Denmark
E-mail: ehs@kt.dtu.dk

mark studies the new ESDIRK solvers are shown to improve computational speed when compared to off-the-shelf stiff ODE solvers.

Fluid phase changes are known to cause convergence problems for the integration method. We propose an algorithm for detecting and locating phase changes based on discrete event system theory. Experiments show that the algorithm improves the robustness of the integration process when near phase boundaries by significantly reducing the number convergence and error test failures.

Keywords Enhanced oil recovery · in-situ combustion · multi-scale methods · reservoir simulation · operator splitting · differential-algebraic equations · ESDIRK methods · discrete event systems

1 Introduction

The world continues to rely substantially on petroleum fossil fuels as a primary energy source. No economical substitute for crude oil is yet available or is likely to become available in the next couple of decades. While the number of new discoveries of petroleum reservoirs decreases, the need to produce the known reservoirs more effectively increases. So far, only the easiest to produce petroleum accumulations have been utilized. Maintaining the supply of oil to support economic growth in industrial and developing countries requires new and innovative methods in order to unlock the remaining crude oil reserves. A large part of the remaining reserves (numbers?) exist in the form of so-called heavy crudes (10 to 20 °API). The development of such reserves by traditional methods (pressure depletion, water flooding) is often inefficient due to the high viscosity of the oil. Thermal recovery processes, which rely on viscosity reduction of the oil through heat injected (steam or hot water) or in-situ generated, are well suited to unlock heavy oils in an environmentally sound manner. The thermal recovery process known as *in-situ combustion* has been a source of interest and effort for several decades. In-situ combustion is the process of injecting air (or air enriched with oxygen) into oil reservoirs to oxidize a portion of the crude oil and enhance recovery through the heat and pressure produced. Contrary to other thermal recovery processes the main part of the energy required to displace the oil in in-situ combustion is generated inside the reservoir from the heat released by chemical reactions between oxygen and fractions of the crude oil. In-situ combustion is technically and economically an attractive process, particularly since the portion of the crude burned is likely to be the heaviest and least valuable. Whereas it is generally classified as a technique that is applicable for heavy oils because of the dramatic reduction in oil viscosity with temperature, in-situ combustion also promotes production through thermal expansion and gas drive caused by combustion gases.

The purpose of our research is the development of accurate, efficient and reliable mathematical models and computational tools for performance evaluation of in-situ combustion processes. In this paper we focus on models and algorithms for integration of chemical reactions in in-situ combustion simulations.

1.1 In-Situ Combustion

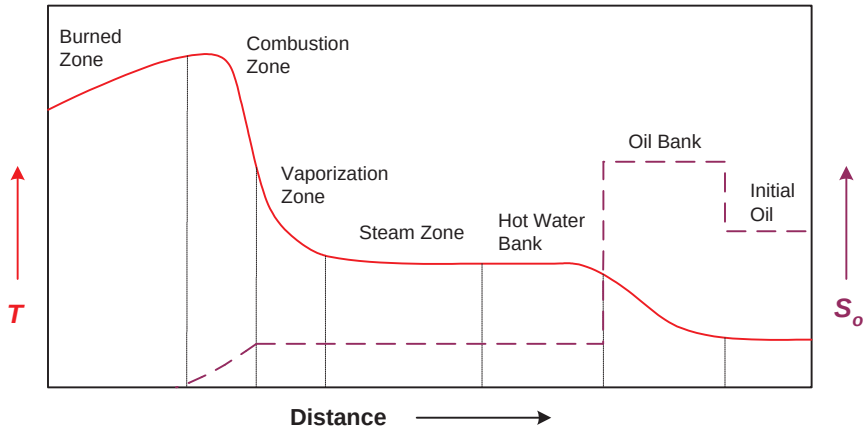


Fig. 1 Schematic representation of characteristic temperature and saturation distributions in forward in-situ combustion (adapted from Prats (1986), not to scale).

In the forward in-situ combustion process oil is ignited at the injection well and the combustion front is propagated towards the production well by continuous injection of air. As the combustion front progresses into the reservoir, several zones can be found between the injector and the producer. A common misconception of the combustion front propagation is that of an underground fire. Instead, the front is more of a glow passing slowly through the reservoir. Figure 1 gives a schematic representation of characteristic temperature and saturation zones in in-situ combustion. Starting from the injection well the burned zone is the volume already swept by the combustion zone. The burned zone contains the injected air and possibly a residue of burned fuel. The combustion zone has the highest temperature and this is where most of the energy is generated. Injected oxygen reacts with residual hydrocarbons generating carbon oxides and water. Hydrocarbons contacted by the leading edge of the high temperature zone undergo thermal cracking and vaporization. Mobilized light components are transported downstream where they mix with the original crude. The heavy residue, which is normally referred to as coke, is deposited on the core matrix and is the main fuel source for the combustion process. Downstream of the vaporization zone is the steam plateau which is formed from water of combustion and vaporization of formation water. Further downstream the steam condenses into a hot water bank when the temperature drops below the steam saturation temperature. The leading edge of the hot water bank is the primary area of oil mobilization where the oil is banked by the hot water. A more detailed description may be found in Prats (1986) and Castanier and Brigham (2004).

The actual mechanisms responsible for oil displacement in in-situ combustion vary with the type of oil. For heavy oils the increase in oil mobility

with elevated temperatures is the primary mechanism assisted by gas flood effects and hot and cold water drive. In the case of lighter oils, the flue gas mixture resulting from combustion provides the primary mobilizing force for the oil downstream of the combustion zone (Moore et al., 2002*a*). The gas-oil mixture may be immiscible, or partly or completely miscible.

1.2 A Complex Multi-Scale Process

In-situ combustion is one of the most physically complex enhanced oil recovery processes currently in use. Driven by complex chemical reactions the oil mobility is increased with the elevated temperatures and the fluids are displaced by a combination of steam, water and gas drive. The spatial as well as temporal scales in in-situ combustion vary over many orders of magnitude. The bulk of the chemical reactions take place in the narrow reaction zone that may be less than a meter in thickness compared to reservoir scales of hundreds or thousands of meters. Moreover, combustion reactions often occur in fractions of a second, whereas the temporal scales associated with convective transport may be running to days or years. Accurate prediction of field performance in such a multiscale process is an immense challenge requiring a hierarchical approach, in which both spatial and temporal resolution is adapted in order to capture the crucial input from all levels of activity.

The overall performance of an in-situ combustion process in governed in a complex way by reservoir heterogeneity, well configurations, injection rates and composition, initial oil saturation and distribution and both thermodynamic and chemical properties of the rock and fluids. Reliable prediction of field performance requires a fully integrated approach in which the important contributions from all levels are taken into account. In-situ combustion is indeed a multiphysics process bringing together multiphase porous media flow, chemical kinetics and phase equilibria.

The spatial scales affecting in-situ combustion span from large geological features such as faults of the size of the reservoir to the very small scale at which chemical reactions happen in the combustion zone. Faults, fractures and the placement of wells determine global flow patterns, but local displacement efficiency is governed by small scale heterogeneity in porosity and permeability of the reservoir and by the chemical and thermodynamic behavior of the fluids. Permeability fields are often obtained from high-resolution geocellular models having gridblock sizes on the order of a meter. Reservoir simulations, however, are carried out using gridblocks that are 1–2 orders of magnitude larger due to computational constraints. Upscaling of the permeability or transmissibility field, in which local flow behavior is taken into account, is routinely done. As mentioned above, the bulk of the chemical reactions happen in a narrow combustion zone being less than a meter in thickness compared to standard gridblock sizes of, say, 50 meters. Thus, the spatial scale for chemical reactions is smaller than the smallest scale normally resolved in reservoir simulations. Consequently, the temperature profile on the simulation grid will be too smooth, and important phenomena such as ignition/extinction or quenching may not be predicted correctly.

Relating to temporal scale, a number of different processes may be identified in in-situ combustion, each having its own characteristic scale. Most of the existing in-situ combustion models include convective mass transfer, convective and conductive heat transfer, kinetically controlled chemical reactions and fluid phases in thermodynamic equilibrium (Crookston et al., 1979; Grabowski et al., 1979; Coats, 1980). The phase equilibrium assumption implicitly states that the timescales for the interphase mass transfer processes occurring when phases come to an equilibrium state, are much faster than all other timescales. Of the remaining processes, the chemical reactions are likely to occur on timescales that are again much faster than the scales for mass and heat transport.

Although being multiscale in nature, the question remains whether all the processes in in-situ combustion represent essential physics that needs to be resolved in a simulation. The goal of in-situ combustion simulation is to provide reliable predictions of performance, typically in terms of production, for a given in-situ combustion project. The production certainly depends on large scale features such as well placement, but the small scale behavior, spatial as well as temporal, in the combustion zone may be equally important. In-situ combustion processes are driven by chemical reactions. Chemical kinetics depends strongly on temperature, thus failing to capture temperature peaks and, in general, smoothing out temperature profiles on too coarse a grid will lead to inaccurate prediction of reaction, which in turn will affect the amount of heat released and combustion gases evolved, ultimately resulting in wrong predictions of oil displacement. Ahead of the combustion front (see Figure 1) the oil is mobilized by a combination of steam, water and gas drive. Lighter oil components will vaporize easily and be transported downstream. The compositional behavior in this region will determine the amount and composition of the oil left behind as fuel for the combustion. Therefore, accurate prediction of phase behavior as well as flow is likely to impact overall performance. Errors at this small scale will feed into overall production calculations, thereby rendering the results unreliable. Hence, the important processes in in-situ combustion are indeed multiscale with strong nonlinear interactions between different scales and efficient computational methods must be developed that handle this multi-scale nature.

1.3 A Time-Stepping Methodology

The multiscale challenge may be approached computationally by either attempting to resolve all relevant scales or by making use of appropriate subgrid scale models to represent the small scale processes. Subgrid scale modeling is not addressed in this paper, but instead we concentrate on how to resolve different scales. In particular we focus on the temporal scales of chemical reactions. For discussions on spatial resolution in ISC simulation, see for example Gerritsen et al. (2004) and Nilsson et al. (2005) where Adaptive Mesh Refinement (AMR) techniques are employed.

In order to capture the essential process dynamics in a numerical simulation, the reaction kinetics must be integrated using timesteps that are much smaller than those necessary for capturing the effects of convection

and conduction. Applying the same integration method to all processes is often very inefficient. An intuitive way of approaching the problem numerically is, instead, by using so-called splitting methods where convective and conductive terms in the equations are separated from reaction terms. Each global timestep then consists of a series of substeps, in which the individual physical processes (convection, conduction, reaction, etc.) are integrated separately, the advantage being that efficient, tailored integration methods can be applied to each sub-process. For example, stiff chemical kinetics is best treated by implicit methods, whereas convective transport can often be integrated explicitly. The separation of scales in the problem is thereby exploited numerically, the cost being a numerical error introduced by the splitting, since in each substep only one process is taken into account and the interaction with the other processes neglected.

Using an operator splitting framework, the main challenge becomes the design of specialized integration methods for the individual substeps. For example, in a reaction substep each gridblock is then effectively treated as a small kinetic cell with homogeneous pressure and temperature and well mixed fluids. Depending on the number of gridblocks and the number of global timesteps, several million solves of the reaction substep may easily be needed during a full scale in-situ combustion simulation. Hence, highly efficient numerical algorithms are needed for this substep.

Recently, Younis and Gerritsen (2006) presented a novel operator splitting method for thermally reactive, compositional reservoir simulation. We will review the method in Section 4.1 and present the tailored methods for reaction integration within this framework.

1.4 Main Objectives and Key Contributions

This paper addresses the issues in in-situ combustion simulation related to chemical kinetics and phase behavior and the interaction of the two. Needless to say, being at the very core of an in-situ combustion simulator, efficient computation of kinetics and phase behavior is crucial to overall performance. The goal of this effort is therefore to develop a model that allows the isolated study of in-situ combustion kinetics and phase behavior. We will use the model as a tool for designing efficient numerical methods for integration of kinetics and phase behavior in ISC simulations. Moreover, we will address the important question of how to handle phase changes during simulation.

We propose a kinetic cell model in this paper. In addition to being an important substep when solving the in-situ combustion equations using an operator splitting approach, the kinetic cell model provides in its own right a testbed for studying the interaction of kinetics and phase behavior and the implications for the numerical solver.

The long-term goal of this research effort is the development of efficient, physically accurate and reliable simulation tools for performance prediction of in-situ combustion projects. The kinetic cell model and the numerical methods proposed in this work represent an important step in that direction.

The key contributions of this paper are:

1. A kinetic cell model that (i) facilitates construction of efficient, tailored integration methods for in-situ combustion kinetics, and (ii) enables isolated sensitivity studies of the kinetics / phase behavior interaction.
2. Efficient numerical integrators based on specialized Runge-Kutta methods tailored for operator splitting integration of in-situ combustion kinetics and phase behavior.
3. An algorithm for robust detection and location of phase changes based on discrete event system theory.

1.5 Paper Outline

The paper is organized as follows. Section 2 provides a review of physical models for in-situ combustion reactions and serves to motivate the development of the kinetic cell model. Section 3 introduces the kinetic cell model equations. The integration methods for the kinetic cell are presented in Section 4 in the framework of an operator splitting method and important implementational aspects are discussed including an algorithm for robust detection and location of phase changes. Finally, Section 5 presents the results from performance comparisons between the new solver and state-of-the-art stiff ODE solvers.

2 Physical Modeling

Before developing the equations for the kinetic cell model we take a brief look at the underlying physical assumptions and the current state of reaction models for ISC reactions.

Apart from chemical reactions, ISC models typically include convective mass transport and convective and conductive heat transport. A condition generally assumed to prevail in ISC, as well as in all other reservoir processes, is that the fluids are in thermodynamic equilibrium at every point within the reservoir (Prats, 1986). This equilibrium assumption implicitly states that the timescales for the interphase mass transfer processes occurring when phases come to an equilibrium state, are much faster than all other timescales. A simple order of magnitude estimate shows that, for non-reacting systems, this assumption is reasonable. The validity of the assumption for reactive systems is, however, unclear, but in lack of better alternatives we shall assume that components transfer between phases under equilibrium conditions. In addition to thermodynamic equilibrium it is commonly assumed that the fluids and reservoir rock are in thermal equilibrium implying that the resistance to heat transfer between fluids and rock is negligible.

2.1 Chemical Reactions

A typical crude oil is a complex mixture of hundreds of different chemical species. Hence, the chemical reactions that happen in the presence of oxygen

are equally complex and numerous. In the field of computational combustion increasingly large reaction models are being used often involving several hundred different intermediary components and thousands of elementary reactions (Westbrook et al., 2005). Reaction pathways and rate coefficients can be derived theoretically from quantum chemical calculations. However, the fuels being studied are pure light hydrocarbon components or very simple mixtures, and even for such simple fuels the reaction pathways become enormously complex. Deriving detailed reaction pathways for crude oil oxidation in porous media is far beyond current capabilities. Determining reactions and reaction rates in in-situ combustion is therefore approached in a much more empirical way. The current state of reaction models is reviewed in this section.

Among the first researchers to group the in-situ combustion reactions into different classes according to the temperature range, in which they occur, were Bousaid and Ramey Jr. (1968) and Burger and Sahuquet (1972). They observed distinct variations in the oxygen uptake rate with temperature when air was cycled through an oil sample. Today it is generally accepted that three classes of reactions dominate in in-situ combustion: (1) low temperature oxidation reactions (LTO), (2) medium temperature reactions and (3) high temperature oxidation reactions (HTO) (Castanier and Brigham, 2004). The temperature ranges associated with each group of reactions are roughly 150 to 300°C for LTO, 300 to 450°C for medium temperature reactions and above 450°C for HTO.

The nature of the reactions taking place in each regime depends on the type of oil. For heavy oils the LTO reactions are oxygen addition reactions producing partially oxygenated compounds (alcohols, ketones, aldehydes, etc.) and only few carbon oxides. Lighter oils undergo full H/C bond breaking combustion reactions in the LTO regime but may undergo oxygen addition reactions at lower temperatures. Oxygen addition increases oil viscosity, and thus operating in this regime is inefficient. In a forward combustion process LTO reactions may take place when oxygen is available downstream of the combustion front due to either channeling around the front or insufficient consumption in the front (Fassihi et al., 1984*a*). The effects of LTO on in-situ combustion performance have been studied by several researchers (Dabbous and Fulton, 1974; Adegbesan et al., 1987; Freitag and Verkoczy, 2005).

At intermediate temperatures after the LTO reactions a series of cracking or pyrolysis reactions take place. These reactions are the primary source of coke formation for the HTO reactions. Coke formation dictates the quantity of fuel available for combustion. Excessive fuel deposition retards the rate of advance of the combustion front whereas insufficient fuel deposition may not provide enough heat for self-sustained combustion. Finally, in the HTO reactions the coke is burned generating carbon oxides and water.

Both LTO and HTO reactions are believed to be heterogeneous gas/liquid or gas/solid reactions (Fassihi et al., 1984*b*; Castanier and Brigham, 2004), in which oxygen from the gas stream diffuses to the surface of the oil/coke, adsorps on the surface, reacts with a hydrocarbon component, and finally combustion products desorp and diffuse back into the gas stream. For surface reactions in general each of these processes can be rate limiting. Only few

researchers comment on this, the majority assuming that the overall process is kinetically controlled, which seems widely accepted (Islam et al., 1989). In disagreement with the assumption of heterogeneous reactions Adegbesan et al. (1987) claim that LTO reactions occur with oxygen dissolved in the oil phase, and hence argue for the importance of accurate description of oxygen solubility.

Most researchers seem to agree that the overall oxidation mechanism of crude oil in porous media is an overlap of three consecutive reactions: LTO, cracking/pyrolysis and HTO. Both light and heavy oils undergo these reactions but in different temperature ranges. LTO reactions are by far the least understood. Although generally recognized as being important to prediction of in-situ combustion performance, no reliable LTO reaction models exist today. Recently Freitag and Verkoczy (2005) observed a shift in reaction order for oxygen partial pressure with temperature and concluded that at least two different reaction mechanisms govern LTO. The negative effects of LTO are mainly an increase in oil viscosity and enhanced trapping of oil due to gas phase shrinkage caused by the oxygen removed from the gas phase in oxygen addition reactions (Moore et al., 2002*b*). The role of LTO in controlling fuel availability is, however, still a topic for further research.

Only little attention has been given to the development of reaction models based on the compositional analysis of the oil. The use of SARA fractions (saturates, aromatics, resins and asphaltenes) represents one such effort (Freitag and Verkoczy, 2005; Kok and Karacan, 1997). In the SARA approach both chemical reactivity and phase behavior is taken into account when grouping the components. The SARA fractions are chromatographically separated according to polarity of the compounds. Saturates are saturated hydrocarbons with straight or branched chains, little ring-structure, and only little nitrogen, sulfer and oxygen content. Aromatics contain one or more aromatic rings. Resins are the second heaviest fraction having high polarity due to considerable nitrogen, sulfer and oxygen content. Finally, asphaltenes are normally defined as the fraction of the crude insoluble in n-heptane. The presence of different functional groups in each fraction will determine overall chemical reactivity. Thus, the SARA approach groups according to reactivity. Each fraction may still have a large boiling range, and a further subdivision based on boiling point may be used to obtain a better phase behavior representation. The SARA based approach seems to be the best and most promising method currently available for in-situ combustion oil characterization.

Summarizing this section the essential components of physical reaction models for in-situ combustion are: (1) a separation in reaction regimes between LTO, cracking/pyrolysis and HTO, (3) formation of a solid phase acting as the main fuel in the combustion zone, (3) heterogeneous gas/liquid and gas/solid reactions with the surface reactions being the rate controlling step over adsorption/desorption and diffusion to/from the surface, and (4) a possible shift from LTO to HTO depending on oxygen flowrate and other critical parameters.

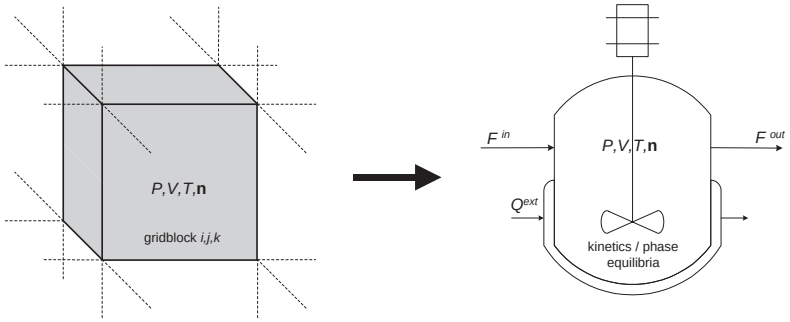


Fig. 2 Using an operator splitting approach for the ISC equations each gridblock is effectively treated as a small chemical reactor.

3 Mathematical Modeling

The governing equations for ISC, considering the physical processes discussed above, can be found in e.g. Younis and Gerritsen (2006). In this section we present the kinetic cell model.

3.1 A Virtual Kinetic Cell (VKC)

As illustrated conceptually in Figure 2 each gridblock in a reaction substep is treated as a small chemical reactor or kinetic cell in which only chemical kinetics and phase behavior are taken into account. The governing equations for this substep are obtained by ignoring convective and conductive terms in the overall ISC equations. The kinetic cell model is proposed here in a slightly more generalized form including in-/outflow terms for mass and energy along with external heating/cooling to allow a more flexible framework for studying the kinetics/phase behavior interactions.

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 uniform in the cell. For convenience, we introduce the molar concentration of a component in a phase, c_{ij}, and the overall molar concentration of a component in the system, C_i:

$$c_{ij} = \varphi_f x_{ij} \rho_j S_j \tag{1}$$

$$C_i = \sum_{j \in \{o,w,g\}} c_{ij} \tag{2}$$

in which S_j, $j \in \{o, w, g\}$, is the phase saturation, ρ_j the molar phase density 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:

$$\varphi_f = \varphi_v - \frac{C_s}{\rho_s} \tag{3}$$

in which C_s and ρ_s are the density and concentration of solid component, respectively. Solid concentration is defined with respect to total volume. The mass conservation of each chemical component is written as:

$$\frac{dC_i}{dt} = \sum_{k=1}^{n_r} A_{ik} r_k + \frac{F_i^{in} - F_i^{out}}{V}, \qquad i = 1, \ldots, n_c \tag{4}$$

in which F_i^{in} and F_i^{out} are the molar flowrates of component i in and out of the cell, respectively. V is the cell volume. r_k is 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 (negative for reactants and positive for products).

The energy conservation equation is:

$$\frac{dU}{dt} = \sum_{k=1}^{n_r} (-\Delta H_k) r_k + \frac{H^{in} - H^{out} + Q^{ext}}{V} \tag{5}$$

in which 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. Finally, U is the total internal energy of the system per unit volume expressed as:

$$U = (1 - \varphi_v)\rho_r U_r + (\varphi_v - \varphi_f)\rho_s U_s + \varphi_f \sum_{j \in \{o,w,g\}} U_j \rho_j S_j \tag{6}$$

The energy source/sink is modelled as:

$$Q^{ext} = UA(T_r - T) \tag{7}$$

in which T_r is the heating/cooling temperature and UA is an overall heat transfer coefficient. 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 flowrates in (4) may be expressed as:

$$F_i^{out} = Q^{out} \rho_g x_{ig} \tag{8}$$

where ρ_g is the gas density and Q^{out} is the volumetric flowrate, which is taken proportional to the difference between the cell pressure and the external pressure:

$$Q^{out} = k_v \left(P - P^{ext}\right) \tag{9}$$

in which 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:

$$V_v = V_f + V_s \tag{10}$$

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

$$k_k = \alpha_k \exp\left(-\frac{E_k}{RT}\right) \tag{11}$$

in which α_k and E_k are the pre-exponential factor and activation energy for reaction k, respectively, and R is the universal gas constant. Three types of reactions occur (LTO, cracking/pyrolysis and HTO). We will assume that the reactions are first order in all reactants. Oxidation reactions for oil phase components are modelled as:

$$r_k = k_k \cdot Px_g^{O_2} \cdot C_i \tag{12a}$$

in which index i refers to an oil component. For solid components the rate expression is:

$$r_k = k_k \cdot Px_g^{O_2} \cdot C_s \tag{12b}$$

Finally, for cracking/pyrolysis reactions:

$$r_k = k_k \cdot C_i \tag{12c}$$

In the treatment of phase equilibrium we assume that components partition into at most two phases. Components in the solid phase exist exclusively in this phase. Moreover, we assume that the water phase consists of water only and that the equilibrium between hydrocarbon components in the gas and oil phases can be described by a simple equilibrium constant correlation:

$$K_i = \frac{x_{ig}}{x_{io}} \tag{13}$$

in which the equilibrium factors are assumed to vary only with pressure and temperature. Using this simplified phase equilibrium description the flash calculation reduces to solving the Rachford-Rice equation for the molar gas phase fraction, β:

$$\sum_{i\in\{hc\}} (x_{ig} - x_{io}) = \sum_{i\in\{hc\}} \frac{C_i(K_i - 1)}{1 - \beta + \beta K_i} \tag{14}$$

We choose as primary variables temperature, pressure and the n_c overall component concentrations. The VKC equations, (4), (5) and (10), comprise a set of differential-algebraic equations (DAEs). Including the total internal energy as a variable and Eq. (6) as an extra constraint allows us to write the DAE in semi-explicit form where temperature is aligned with the energy constraint (6) and pressure is aligned with the volume constraint (10). In multiphase regions Eq. (14) is appended to the equation system and aligned with the gas phase fraction. Detection of phase changes is discussed later in Section 4.4.

Apart from being a useful tool for developing tailored integration methods for ISC reactions, the VKC provides a testbed for studying the kinetics / phase equilibrium interplay in ISC. Interaction with the reservoir can be mimicked through the boundary conditions, thus allowing us to study, for example, the sensitivity to oxygen availability by varying oxygen inflow rate or the effects of heat losses due to external cooling. Kinetic cell experiments are routinely carried out in the laboratory to determine burning characteristics of a particular oil. Using data from these experiments the VKC could be applied in inverse problems to determine reaction kinetic parameters – a topic for future work.

3.2 Test Cases

Two ISC reaction models are used in this paper as test examples and for benchmarking the numerical solvers developed in the next section.

3.2.1 The Minimal Model

This model includes a minimal realistic set of components and reactions to represent ISC behavior. The components are: light oil (LO), heavy oil (HO), coke (C), oxygen (O_2), water (W) and inert gas (IG). The reactions are:

$$\begin{aligned} \mathrm{LO} + \mathrm{O_2} &\rightarrow \mathrm{IG} + \mathrm{W} \\ \mathrm{HO} + \mathrm{O_2} &\rightarrow \mathrm{IG} + \mathrm{W} \\ \mathrm{HO} &\rightarrow \mathrm{LO} + \mathrm{C} + \mathrm{IG} \\ \mathrm{C} + \mathrm{O_2} &\rightarrow \mathrm{IG} + \mathrm{W} \end{aligned}$$

The reactions account for cracking of heavy oil and complete oxidation of coke and the two oil components.

3.2.2 The SARA Based Model

This model has 14 (pseudo) components and 14 reactions. It is based on SARA grouping of the oil. The reactions are listed below and the components involved are listed in Table 1. The reactions include pyrolysis, LTO and HTO of the SARA fractions along with HTO of pyrolysis coke and the partially oxidized LTO residue. The LTO residues of resins/aromatics and saturates are non-volatile oil components whereas the LTO residue of asphaltenes is solid. The model and associated parameters were generously made available to us by Norman Freitag, Saskatchewan Research Council, Canada. Details can be found in the references: Freitag and Verkoczy (2005), Freitag and Exelby (2006) and Ren et al. (2005).

crac./pyr.

$$\begin{cases} \mathrm{Asph} \rightarrow \mathrm{H_2O} + \mathrm{Arom} + \mathrm{Sat} + \mathrm{Lites} + \mathrm{PyrCoke} \\ \mathrm{Resins} \rightarrow \mathrm{H_2O} + \mathrm{Asph} + \mathrm{Arom} + \mathrm{Sat} + \mathrm{Lites} \\ \mathrm{Arom} \rightarrow \mathrm{Asph} + \mathrm{Lites} \end{cases}$$

LTO

$$\begin{cases} \mathrm{Asph} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} + \mathrm{OxdAsph} \\ \mathrm{Resins} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} + \mathrm{OxdResAr} \\ \mathrm{Arom} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} + \mathrm{OxdResAr} \\ \mathrm{Sat} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} + \mathrm{OxdSat} \end{cases}$$

HTO

$$\begin{cases} \mathrm{PyrCoke} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \\ \mathrm{OxdAsph} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \\ \mathrm{OxdResAr} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \\ \mathrm{OxdSat} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \\ \mathrm{Asph} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \\ \mathrm{Resins} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \\ \mathrm{Arom} + \mathrm{O_2} \rightarrow \mathrm{H_2O} + \mathrm{CO_2} \end{cases}$$

Component Name	Abbreviation	Phase(s)
Water	H_2O	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	CO_2	oil/gas
Nitrogen	N_2	gas
Oxygen	O_2	gas
Oxidized asphaltenes	OxdAsph	solid
Pyrolysis coke	PyrCoke	solid

Table 1 List of (pseudo) components for the SARA based model. The components can exist in the phase(s) listed in the last column.

4 Numerical Modeling

4.1 Operator Splitting

As mentioned in the introduction, numerical time-integration of the multi-scale ISC process can be accomplished by use of operator splitting methods. These methods break down the problem into independent sub-problems, typically by regarding each physical process (convection, conduction, reaction) as a separate operation. The main advantage of splitting methods is that tailored integration methods can be applied to each sub-problem. For example, the stiff reaction kinetics can be integrated with methods designed for stiff ODEs whereas convective transport can be integrated with high-resolution explicit methods. The solutions to the sub-problems are subsequently tied back together to form an approximation to the full equation. This type of splitting is often referred to as *additive splitting* (ADS). Each sub-problem only involves one physical process or a subset of processes and the solutions to the sub-problems do not represent consistent solutions to the full problem. For IMEX-type splittings, on the other hand, the equations are not decoupled and each intermediate stage of an IMEX scheme honors the full equation. Instead, for IMEX schemes, a single integration method is chosen for the full set of equations and the individual variables are then treated either explicitly or implicitly. A well-known example of an IMEX scheme in petroleum engineering is the IMPES (IMplicit Pressure Explicit Saturation) scheme in which pressure is treated implicitly and saturation explicitly. ADS methods can be applied to additively separable operators, like convection and reaction in the ISC problem. They are desirable because they allow sub-problem

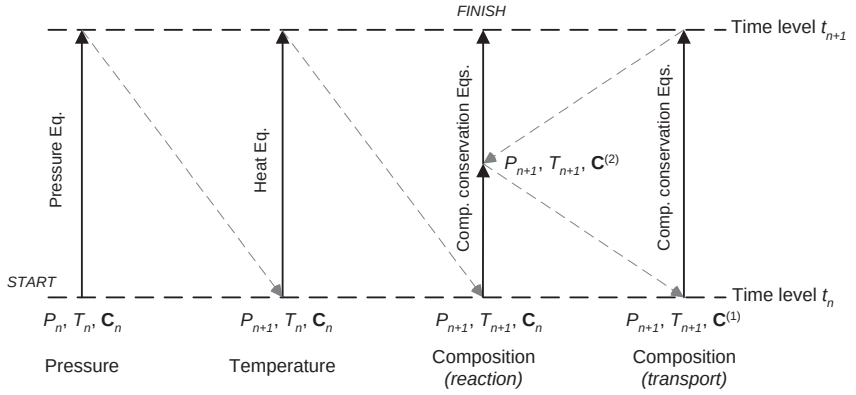


Fig. 3 Illustration of the IAFS scheme for thermally reactive, compositional reservoir simulation (Younis and Gerritsen, 2006). Pressure and temperature are solved using a cascaded IMEX approach and aligned with the pressure and heat equations, respectively. The reaction and transport parts are separated using a symmetric ADS (Strang splitting). Initial conditions for each substep are indicated in the figure.

integration by tailored methods. However, some operators are not additively separable, an example being the flow-transport coupling in multiphase porous media flow problems in which the saturation and pressure dependent parts of convection are non-separable.

The mixed nature of the ISC problem with both additively separable and non-separable operators inspired Younis and Gerritsen (2006) to propose a hybrid IMEX-ADS method – the IMEX-ADS Fractional-Step (IAFS) method. The IAFS method combines the traditional IMPEC (IMplicit Pressure Explicit Composition) approach with an explicit temperature treatment with a symmetric ADS for the convection and reaction operators.

To outline the computational steps in the IAFS scheme we cast the overall ISC equations in simplified form:

$$\frac{dA}{dt} = -F_T^m + R_T^m \tag{15a}$$

$$\frac{dU}{dt} = -F_T^e + R_T^e \tag{15b}$$

$$\frac{d\mathbf{C}}{dt} = -\mathbf{F}^m + \mathbf{R}^m \tag{15c}$$

in which Eq. (15a) is the pressure equation (sum of component conservation equations), Eq. (15b) is the heat equation and Eq. (15c) is the set of component conservation equations. F denotes discretized advection and R denotes the reaction term. Heat conduction is neglected in this formulation allowing explicit treatment of temperature. The sequence of steps in the IAFS scheme is illustrated in Figure 3. First, pressure is solved aligned with the pressure equation using explicit temperature and concentrations. Second, temperature is solved using the new pressure and explicit concentrations. Finally, concentrations are updated using a symmetric ADS (in this case Strang splitting)

to separate the transport and reaction operators:

$$\frac{d\mathbf{C}^{(1)}}{dt} = \mathbf{R}^m(\mathbf{C}^{(1)}), \qquad \mathbf{C}^{(1)}(t_n) = \mathbf{C}_n \quad \text{on} \quad [t_n, t_n + \Delta t/2]$$

$$\frac{d\mathbf{C}^{(2)}}{dt} = -\mathbf{F}^m(\mathbf{C}^{(2)}), \qquad \mathbf{C}^{(2)}(t_n) = \mathbf{C}^{(1)}(t_n + \Delta t/2) \quad \text{on} \quad [t_n, t_{n+1}]$$

$$\frac{d\mathbf{C}^{(3)}}{dt} = \mathbf{R}^m(\mathbf{C}^{(3)}), \qquad \mathbf{C}^{(3)}(t_n + \Delta t/2) = \mathbf{C}^{(2)}(t_{n+1}) \quad \text{on} \quad [t_n + \Delta t/2, t_{n+1}]$$

In the next section we will develop tailored integration methods for the reaction substeps in the Strang splitting scheme.

4.2 Choice of Integration Method for Reactions

The kinetic cell equations consist of coupled differential and algebraic equations. We will consider the following general form of the DAE system:

$$\mathbf{M}\frac{d\mathbf{u}}{dt} = \mathbf{f}(t, \mathbf{u}), \qquad \mathbf{u}(t_0) = \mathbf{u}_0 \tag{16}$$

in which $\mathbf{u} \in \mathbb{R}^n$ is a vector of state variables depending on t, and $\mathbf{f}$ is a vector function mapping $\mathbb{R} \times \mathbb{R}^n$ into $\mathbb{R}^n$. For the kinetic cell model the mass matrix, $\mathbf{M} \in \mathbb{R}^n \times \mathbb{R}^n$, is simply a diagonal matrix with ones in the diagonal indicating a differential equations and zeroes indicating an algebraic equations. The initial conditions are assumed consistent with the algebraic constraints.

The characteristics of the kinetic cell model and of the application in a splitting method are:

- Coupled system of ODEs and AEs
- Stiff chemical kinetics
- Discontinuities in right-hand-side functions arising due to phase changes
- Short integration intervals due to splitting
- Typical dimension of the state vector: $10 - 20$

The kinetic cell equations must be integrated in small intervals governed by the global time step of the splitting scheme (c.f. Figure 3). Thus, the solver must be capable of solving efficiently the DAE with frequent interruptions due to exchange of information with other levels of the scheme. Moreover, the solver must be able to handle discontinuities due to phase changes. The relative accuracy range of interest, considering the accuracy of inputs and the accuracy to which other steps in the scheme are computed, is $10^{-2} - 10^{-4}$, i.e. approximately 2–4 significant correct digits in the results.

The short integration intervals 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, $t_{n-j},\ j = 0, 1, \ldots, s$. Since multi-step methods rely on several

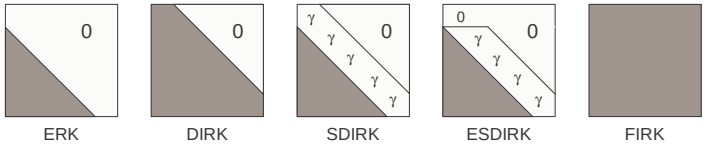


Fig. 4 Structure of the **A** matrix in the Butcher tableau for different classes of RK methods.

previous integration steps, they are often implemented in a variable order formulation, in which they start out at low order and slowly build up higher order as more information becomes available. In the initial phase of integration, when working at low order, multi-step methods therefore need short time steps in order to meet a certain tolerance. At a later stage, when higher order information becomes available, multi-step methods will often outperform one-step methods. The different nature of one-step and multi-step methods suggest that multi-step methods will be advantageous for problems with long and smooth integration intervals, whereas one-step methods will benefit 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, only the family of Runge-Kutta (RK) methods (Runge, 1895; Kutta, 1901) will be considered. 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 (16) may be expressed as:

$$\mathbf{M}\mathbf{U}_i = \mathbf{M}\mathbf{u}_n + \Delta t_n \sum_{j=1}^{s} a_{ij}\mathbf{f}(t_n + c_j\Delta t_n, \mathbf{U}_j) \tag{17a}$$

$$\mathbf{M}\mathbf{u}_{n+1} = \mathbf{M}\mathbf{u}_n + \Delta t_n \sum_{i=1}^{s} b_i\mathbf{f}(t_n + c_i\Delta t_n, \mathbf{U}_i) \tag{17b}$$

in which $\mathbf{U}_i$ denotes the solution at the ith, $(i = 1, \ldots, s)$, internal stage of integration step n and Δt_n denotes the time step length. The coefficients of RK schemes are often represented in a Butcher tableau:

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \ldots & a_{1s} \\ c_2 & a_{21} & a_{22} & \ldots & a_{2s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_s & a_{s1} & a_{s2} & \ldots & a_{ss} \\ \hline \mathbf{u}_{n+1} & b_1 & b_2 & \ldots & b_s \end{array} \quad = \quad \begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array} \tag{18}$$

Runge-Kutta methods are classified according to the structure of their Butcher tableau as illustrated in Figure 4. For explicit methods (ERK), matrix **A** is strictly lower triangular, implying that all the internal stages (17a) can be calculated explicitly, making these methods computationally fast and straightforward to implement. However, in general, ERK methods have poor stability properties, which make them unsuited for stiff problems (Hairer

et al., 1996). The four remaining subclasses of Runge-Kutta methods in Figure 4 are all implicit. The values of the internal stages can no longer be calculated explicitly from the values of the previous stages. Each integration step of an implicit method requires the solution of a system of ns nonlinear equations. Normally, an iterative method, such as Newton's method, is applied. For diagonally implicit methods (DIRK) the stage values can be calculated sequentially, thereby lowering the computational cost compared to fully implicit methods (FIRK), for which all ns equations must be solved simultaneously. If all diagonal elements of $\mathbf{A}$ are identical and upper diagonal elements are zero, the method is said to be singly diagonally implicit (SDIRK). Finally, if the first stage of an SDIRK method is explicit, the method is said to be explicit singly diagonally implicit (ESDIRK).

We have chosen the class of ESDIRK methods for the VKC solver. The diagonal structure of these methods allows sequential evaluation of the internal stages, the cost of each stage being relatively low. Since the diagonal elements are equal, the iteration matrix for solving the nonlinear stage equations (17a) need only be evaluated and factorized once per integration step. The methods are both A- and L-stable as well as stiffly accurate (Hairer and Wanner, 1996). Stiffly accurate methods avoid the order reduction phenomenon observed by Prothero and Robinson (1974) when applied to stiff ODEs. Finally, 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 (Hairer and Wanner, 1996). 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.

ESDIRK methods satisfying the stiffly accurate condition have the following form:

$$\begin{array}{c|cccccc} 0 & 0 & 0 & & & & \\ c_2 & a_{21} & \gamma & 0 & & & \\ c_3 & a_{31} & a_{32} & \gamma & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \\ c_{s-1} & a_{s-1,1} & a_{s-1,2} & a_{s-1,3} & \cdots & \gamma & 0 \\ 1 & b_1 & b_2 & b_3 & \cdots & b_{s-1} & \gamma \\ \hline \mathbf{u}_{n+1} & b_1 & b_2 & b_3 & \cdots & b_{s-1} & \gamma \end{array} \tag{19}$$

Only recently has the focus shifted to ESDIRK methods, but the DIRK and SDIRK methods have been around for more than 30 years. According to Kværnø et al. (1996) the first mention of diagonally implicit RK methods was by Butcher (1964) who noted that these methods would be well suited for practical use. However, dedicated research into the properties of the methods did not start until the early seventies. In the mid seventies several researchers published papers and theses devoted to the subject. Most notably are the works of Nørsett (1974) and Crouzeix (1975). Butcher and Chen (2000) were among the first to recognize the advantage of including an explicit stage in SDIRK methods. They proposed ESDIRK-like extensions to several classes of implicit RK methods. Alexander (2003) proposed a four stage ESDIRK

method of order 3 with an embedded error estimator of order 4 and further showed that the method was uniquely determined by the order and stability conditions. Thus, for a four stage method the optimal embedded ESDIRK pair has orders 3 and 4 if the A- and L-stability conditions are imposed on the lowest order method. 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, Kværnø (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. ESDIRK methods have been applied as time integrators in fully implicit fluid dynamics simulations (Bijl et al., 2002) as well as in operator splitting methods for integrating the reaction part of convection-diffusion-reaction equations (Kennedy and Carpenter, 2003).

4.3 Efficient Implementation

Five ESDIRK methods of orders 2-5 have been implemented. 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 which we will denote ESDIRK12. In general, we will use ESDIRK$p-1/p$ to denote an embedded pair of orders $p-1$ and p. The major task in an ESDIRK method is the solution of the nonlinear algebraic equations arising in each internal stage. Efficient control of the iterative scheme applied to these equations along with control of discretization error by step size adjustments are the two most important aspects of implementation.

4.3.1 Error and Convergence Control

Modern ODE solvers adapt the step size during integration to meet a specified accuracy requirement for the solution. The step size is adapted based on an estimate of the local error. This estimate is obtained, as indicated above, using embedded formulas in which two solutions of different orders, p and $p-1$, are subtracted to generate an error estimate for the lowest order method. The error estimate is essentially free, since it involves no additional function evaluations or solutions of implicit systems.

Adjusting the step size to meet an accuracy requirement is basically a control problem. We have implemented a predictive controller as suggested by Gustafsson (1992) leading to the following control law for selecting the step size:

$$\Delta t_n = \frac{\Delta t_{n-1}}{\Delta t_{n-2}} \left(\frac{\varepsilon}{r_n}\right)^{k_2/(p+1)} \left(\frac{r_{n-1}}{r_n}\right)^{k_1/(p+1)} \Delta t_{n-1} \tag{20}$$

k_1 and k_2 are the gain parameters for the controller, while ε is the desired tolerance (including a safety factor). Gustafsson (1992) suggests using $k_1 = k_2 = 1$. The aggressiveness of the controller can be tuned by tuning these

parameters. r_n is the norm of the estimated local error:

$$r = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{err_i}{atol_i + rtol_i \cdot |u_i|}\right)^2} \tag{21}$$

in which *atol* and *rtol* are (componentwise) absolute and relative error tolerances specified by the user. Experiments have shown that the controller (20) 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:

$$\Delta t_n = \left(\frac{\varepsilon}{r_n}\right)^{1/(p+1)} \Delta t_{n-1} \tag{22}$$

as implemented in many ODE solvers.

4.3.2 Stage Value Predictions

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

$$\left[\mathbf{M} - \Delta t\gamma\frac{\partial \mathbf{f}}{\partial \mathbf{u}}\right]\Delta \mathbf{U}_i = \mathbf{M}\mathbf{u}_n + h\sum_{j=1}^{i-1} a_{ij}\mathbf{f}(t_n + c_j\Delta t, \mathbf{U}_j^{(k)}) - \mathbf{M}\mathbf{U}_i^{(k)} \tag{23a}$$

$$\mathbf{U}_i^{(k+1)} = \mathbf{U}_i^{(k)} + \Delta \mathbf{U}_i \tag{23b}$$

In order to optimize this process, we need efficient strategies for

- evaluating the Jacobian and factorizing the iteration matrix, $\mathbf{M} - h\gamma\partial\mathbf{f}/\partial\mathbf{u}$.
- generating initial guesses for the iterations.
- terminating the iterations.

The iteration matrix is evaluated and factorized once per integration step. Implementations tailored for large-scale applications will often benefit from a strategy in which the Jacobian and its factorization are reused over several consecutive steps. For the kinetic cell, the typical state dimension is 10–20 and Jacobian evaluations, function evaluations and LU-factorizations are all comparable with respect to computational cost. Hence, we evaluate the iteration matrix and perform a factorization in each step.

The convergence rates in the Newton iterations may be substantially improved and the convergence failures minimized if good starting values are chosen. The most primitive approach to obtaining a guess for the next stage is to use the most recent stage values. A better approach is to construct a predictor (a polynomial) based on information from the previous integration step and use it to extrapolate the stage values in the current step. The predictor can be constructed in several ways. From the function values and the derivatives at t_{n-1} and t_n we can construct a Hermite polynomial and use it to extrapolate stage values between t_n and t_{n+1}. We exploit instead the *continuous extension* of the method (Enright et al., 1986):

$$\mathbf{u}(t_n + \theta\Delta t_n) = \mathbf{u}_n + \Delta t_n \sum_{i=1}^{s} b_i^*(\theta)\mathbf{f}(t_n + c_i\Delta t_n, \mathbf{U}_i) \tag{24}$$

in which the quadrature weights, b_i^*, depend in θ:

$$b_i^*(\theta) = b_{i,1}^*\theta + b_{i,2}^*\theta^2 + \ldots + b_{i,p}^*\theta^p \tag{25}$$

The coefficients for the continuous extension can be determined by requiring the same set of order conditions satisfied that was used to determine the coefficients of the actual method. A prediction of the stage value at the ith internal stage is then obtained by substituting $\theta_i = 1 + c_i h_{n+1}/h_n$ in (24).

The strategy for the non-linear solutions is inherently coupled to the outer control of integration error and selection of step sizes. The termination accuracy for the Newton iterations affects the accuracy of the overall solution. The termination criterion for the iterations should be related to the accuracy required in the solution. Houbak et al. (1985) showed that for stiff problems the termination criterion should be based on the residual, $\mathbf{R}$, of the nonlinear equations, rather than the displacement, $\Delta\mathbf{U}$. The iterations are terminated when:

$$\|\mathbf{R}^{(k)}\| \leq \kappa \cdot rtol \tag{26}$$

in which $\|\cdot\|$ denotes a 2-norm and $\mathbf{R}$ is the (scaled) vector of residuals. $rtol$ is the relative error tolerance for the local discretization error. κ is normally chosen between 0.5 and 0.01. For reasons of robustness we chose $\kappa = 0.01$. During iterations the convergence rate is estimated as:

$$\alpha_k = \frac{\|\mathbf{R}^{(k)}\|}{\|\mathbf{R}^{(k-1)}\|} \tag{27}$$

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. A target minimum convergence rate of 0.3 is used as suggested by Gustafsson (1992).

4.4 Discontinuities Due to Phase Changes

In the kinetic cell the reaction rates depend on component concentrations in a specific phase, the most common type of reaction being oxygen in the gas phase reacting with a hydrocarbon component in the oil phase. Pure gas phase reactions, in which oxygen reacts with a hydrocarbon gas component, are often neglected. The phase dependence of the reactions introduces certain difficulties in the solution of the kinetic cell equations. Phase changes in the cell cause discontinuities in the right-hand-side functions of the model. Straightforward integration across these discontinuities may lead to poor convergence and repeated step failures.

We propose in this section 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 mark the occurrence of a "discrete event", e.g. a change from a single phase region to a two-phase region. The time of the phase change is not known in advance, but depends on the model solution. Thus, the solution must be advanced until the condition

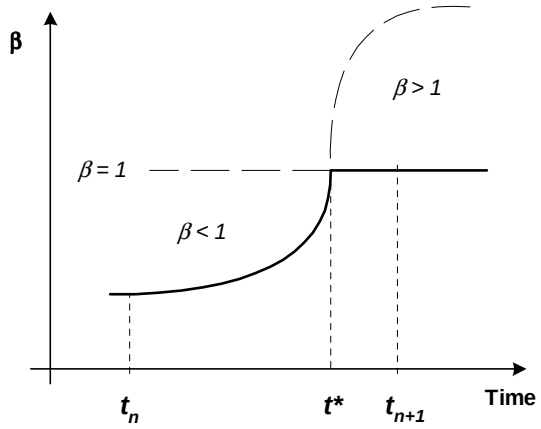


Fig. 5 Possible variation of the gas phase fraction over one integration step. At $t = t^*$, $\beta = 1$ and the oil phase vanishes.

that triggers a phase change becomes satisfied. The detection of the phase change and subsequent location of the exact time when the change occurred are the main components of the proposed discrete event algorithm.

Consider the situation depicted in Figure 5. β denotes the gas phase fraction. In general, the phase change will occur somewhere between the natural steps of the solver. In this case, the condition for a change to the single phase region ($\beta = 1$) is satisfied about three quarters into the integration step. The exact time for the change is labelled t^*. The simplest approach for dealing with this situation would be to accept t_{n+1} as the time of change. However, accepting a solution well beyond the phase boundary results in a negative phase amount for the vanishing phase.

4.4.1 Discrete Event Systems

We will represent the discrete event DAE system in the following form:

$$\frac{d\mathbf{u}}{dt} = \mathbf{f}(t, \mathbf{u}, \mathbf{v}), \qquad \mathbf{u}(t_0) = \mathbf{u}_0 \tag{28a}$$

$$0 = \mathbf{g}(t, \mathbf{u}, \mathbf{v}), \qquad \mathbf{v}(t_0) = \mathbf{v}_0 \tag{28b}$$

$$0 < q_j(t, \mathbf{u}, \mathbf{v}), \qquad j = 1, \ldots, n_{ev} \tag{28c}$$

where the DAE system has been reformulated in semi-explicit form. $\mathbf{u} \in \mathbb{R}^{n_d}$ is the vector of 'differential' variables and $\mathbf{v} \in \mathbb{R}^{n_a}$ is the vector of 'algebraic' variables. $\mathbf{f}$ and $\mathbf{g}$ are vector functions mapping $\mathbb{R} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_a}$ into $\mathbb{R}^{n_d}$ and $\mathbb{R}^{n_a}$, 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 $m \in \mathcal{I}_{state}$ then each state is represented in the form (28) with

$\{n_d, n_a, n_{ev}\}_m$. Thus, the number of equations and event functions may differ between individual states. If one of the event functions becomes satisfied, a change occurs to the system state associated with this event function.

To illustrate the concepts we consider the discrete event formulation of the VKC. We neglect changes associated with the water phase and, hence, 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. When in the two-phase region the 'active' equations for the VKC are Eq. (4), (5), (6), (10) and (14):

$$\frac{dC_i}{dt} = \sum_{k=1}^{n_r} A_{ik} r_k + \frac{F_i^{in} - F_i^{out}}{V}, \qquad i = 1, \ldots, n_c \tag{29a}$$

$$\frac{dU}{dt} = \sum_{k=1}^{n_r} (-\Delta H_k) r_k + \frac{H^{in} - H^{out} + Q^{ext}}{V} \tag{29b}$$

$$0 = U - (1 - \varphi_v)\rho_r U_r + (\varphi_v - \varphi_f)\rho_s U_s + \varphi_f \sum_j U_j \rho_j S_j \tag{29c}$$

$$0 = V_v - V_f - V_s \tag{29d}$$

$$0 = \sum_{i \in \{hc\}} \frac{C_i(K_i - 1)}{1 - \beta + \beta K_i} \tag{29e}$$

For this system state we have $n_d = n_c + 1$, $n_a = 3$ and $n_{ev} = 2$. The two event functions associated with the two-phase region are:

$$0 < \sum_{i \in \{hc\}} C_i(1 - K_i) \tag{29f}$$

$$0 < \sum_{i \in \{hc\}} C_i \left(1 - \frac{1}{K_i}\right) \tag{29g}$$

corresponding to disappearance of either the gas phase or the oil phase, respectively. When a change to a single phase region occurs, Eq. (29e) is omitted from the formulation. Each single phase state then has one associated event function defining the criterion for a switch back to the two-phase region.

4.4.2 Discrete Event Algorithm

The requirements for the event detection algorithm are to (i) detect all phase changes by monitoring sign changes in the event functions, and (ii) to locate the exact time for the changes that occur. Several general purpose event detection algorithms have been suggested in the literature, a good reference being the paper by Park and Barton (1996). The event detection algorithm is based on the principle of *discontinuity locking* (Park and 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 then located. The approach is based on the assumption that the system of equations is mathematically well behaved in a small neighborhood of the change. The solution may be unphysical (negative phase fractions), but we will exploit the fact that the solution trajectories vary smoothly in the neighborhood of the change (c.f. the smooth continuation of the phase fraction curve after t^* in Figure 5).

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 (24) provided by the ESDIRK methods. The algebraic variables could, in principle, also be interpolated by a suitable polynomial. However, to avoid *discontinuity sticking* problems (Park and Barton, 1996), we locate the phase changes by solving the following system of equations:

$$\mathbf{g}(t^*, \mathbf{u}^p(t^*), \mathbf{v}(t^*)) = 0 \tag{30a}$$

$$q^*(t^*, \mathbf{u}^p(t^*), \mathbf{v}(t^*)) = 0 \tag{30b}$$

whenever a change has been detected. The unknowns are the algebraic variables, $\mathbf{v}$, and the time of change, t^*. $\mathbf{u}^p(t)$ denotes the interpolating polynomial for the differential variables and q^* is the event function that became satisfied between t_n and t_{n+1}. The structure of the ESDIRK integration algorithm with discrete event detection is outlined below:

Require: $t_{initial}$, t_{final}, $\mathbf{u}(t_{initial})$, initial system state
  while $t \leq t_{final}$ **do**
    Advance solution from t_n to t_{n+1} using the ESDIRK scheme
    Compute error estimate
    if *integration_error_acceptable* **then**
      Evaluate event functions
      if *sign_change_occurred* **then**
        Locate time of change by solving (30)
        Set time equal to t^*
        Initialize in new system state
      end if
    else
      Recompute time step with reduced Δt
    end if
  end while

5 Results

5.1 VKC Simulations

As a first example we illustrate the VKC by simulating a ramped temperature experiment using the minimal reaction model presented in Section 3.2.1. Ramped temperature experiments are often carried out in the laboratory to determine burning characteristics for different oils.

The simulated setup consists of an oil sample placed in the kinetic cell which is heated externally from 100°C to 600°C over a period of 10 hours

Initial conditions	
Water	0.
Light oil	0.
Heavy oil	0.55moles
Oxygen	0.
Inert gas	0.45moles
Coke	0.
Temperature	373K
Pressure	137atm
Total cell volume	1.4L
Air feed rate	10L/hr
Rock porosity	0.4
Heat transfer coefficient	$60\text{kJ}/(\text{mole} \cdot \text{K})$
Valve coefficient	$10\text{L}/(\text{hr} \cdot \text{atm})$

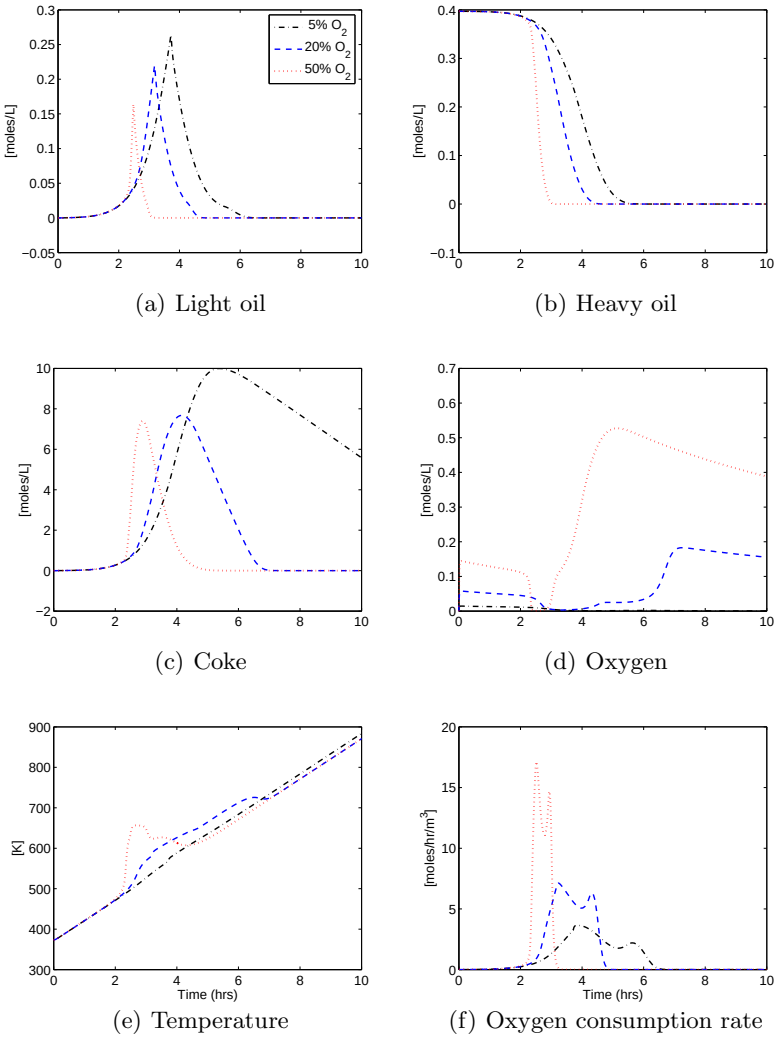
Table 2 Initial and operational conditions for simulating the ramped temperature experiment using the VKC.

with a constant feed of air of 10L/hr. Initial and operational conditions are summarized in Table 2. The fluid phases are assumed ideal and the equilibrium K-factors are assumed to depend only on pressure and temperature. Figure 6 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 favours the direct oxidation which leads to a significant increase in temperature above the heating temperature.

5.2 Phase Changes

To illustrate the importance of proper handling of phase changes we consider again the ramped temperature experiment. For the case with 20% oxygen in the feed, a change from single phase oil to two-phase oil-gas occurs at $t = 3.15$hrs. Attempting to integrate directly across the phase change results in repeated step failures in the solver as illustrated in Figure 7. The figure shows the step size sequence as selected by the controller (20) for a section of the integration interval near the phase change. Each time the solver attempts a step across the phase boundary, a convergence or error test failure results forcing the the solver to reduce the step size. The solver fails 8 times and reduces the step size by two orders of magnitude before finally stepping across successfully. Using the phase change detection algorithm, the change is detected and consistently located without failed steps. Initializing in the



(a) Light oil (b) Heavy oil

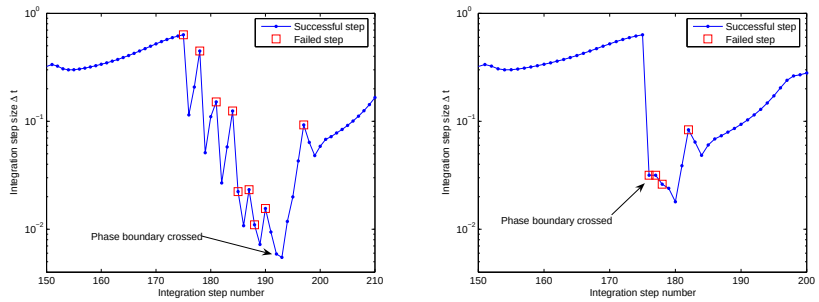
(c) Coke (d) Oxygen

(e) Temperature (f) Oxygen consumption rate

Fig. 6 Simulation results from a ramped temperature experiment using the minimal reaction model. The temperature is raised from 100°C to 600°C over a period of 10 hours. Cell temperature, oxygen consumption rate and component concentrations are shown for 3 different oxygen feed concentrations.

new two-phase region, however, results in 4 step failures before settling at the new level. Completely avoiding step failures is difficult since a change into the two-phase region changes the dynamics of the problem.

In general, we have observed that the algorithm for phase change detection and location improves the robustness near phase changes by significantly lowering the number of integration step failures. Our observations also indicate that a solver not equipped with a phase change algorithm will, in most



(a) Without phase change detection. (b) With phase change detection.

Fig. 7 Step size sequences for the ESDIRK23 solver near a phase change when simulating the ramped temperature experiment. The phase change algorithm improves the integration robustness when crossing phase boundaries.

cases, successfully integrate across the change after repeated step size reductions caused by convergence and error test failures. Thus, the phase change algorithm is mainly needed to avoid unnecessary step failures when crossing phase boundaries. 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 we emphasize the improved robustness near phase boundaries.

5.3 Performance Comparison

To evaluate the performance of the new, tailored ESDIRK implementations we have compared them to two widely used off-the-shelf, general purpose, stiff ODE solvers: DASSL (Petzold, 1982) and LSODE (Hindmarsh, 1983). Both DASSL and and LSODE are variable order, variable step size implementations of the backward differentiation formulas (BDF). The codes are fully documented elsewhere and will not be discussed in further detail here. Among the ESDIRK methods implemented we show results for the following methods:

ESDIRK12: Two stage, second order trapezoidal rule with implicit Euler as error estimator.
ESDIRK23: Three stage method by Kværnø (2004)
ESDIRK34: Four stage method by Alexander (2003).
ESDIRK45: Seven stage method by Kværnø (2004).

Coefficients for the methods can be found in the respective papers.

5.3.1 Setup of Experiments

To mimic the application within the IAFS splitting environment as closely as possible, we modify the VKC equations slightly assuming fixed temperature

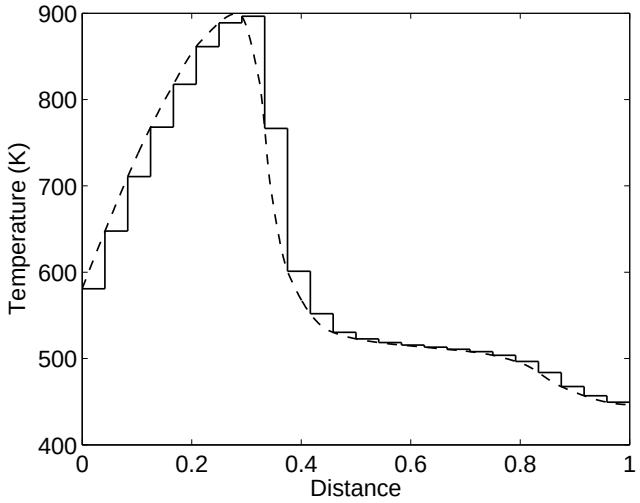


Fig. 8 The temperatures for the integration subintervals in the performance comparison are obtained by sampling the characteristic ISC temperature profile. The continuous temperature profile is shown along with a zero-order parametrization corresponding to $N = 25$ equidistant subintervals.

and pressure in each splitting interval and solve only for component concentrations. The VKC equations thus reduce to the set of ODEs represented by Eq. (4) along with the phase equilibrium constraint (14). We split the total integration interval into N subintervals. For each subinterval we then input a new temperature and pressure and restart the solvers. For DASSL and LSODE, which are not equipped with functionality for handling discontinuities due to phase changes, we solve the implicit equation (14) whenever the ODE right-hand-side functions are evaluated. The temperature input for each subinterval is obtained by sampling the characteristic ISC temperature profile as shown in Figure 8. In this way, the temperature variation corresponds, roughly, to the variation that a single gridblock in an ISC simulation would "experience". Pressures are assumed constant during all subintervals.

As a benchmark example we use the SARA based reaction model presented in Section 3.2.2. In terms of number of components and reactions we think that this model represents best the requirements in realistic ISC simulations. With respect to component and mixture properties 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 20atm 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

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.

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:

$$\mathrm{SCD} := -\log_{10}\left[\max_i \left|\frac{u_i(t_{end}) - u_i^{ref}(t_{end})}{u_i^{ref}(t_{end})}\right|\right] \tag{31}$$

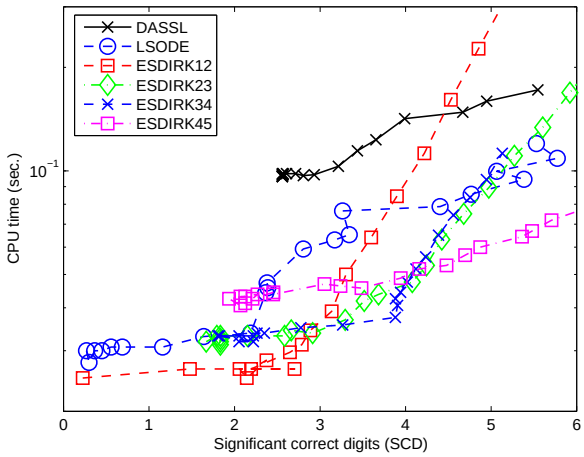
It was verified that all concentrations are greater than zero at t_{end}. The results of the solver comparisons are presented as work-precision diagrams where efficiency is measured by CPU time. Thus, to produce the diagrams the problem is solved using a range of input tolerances for each of the solvers tested. In addition, the following conditions apply to the experiments:

- All codes are compiled using the Compaq Visual Fortran compiler.
- The runs are made on a Pentium 4, 3GHz PC with 512MB RAM.
- The CPU time is taken as the mean over 25 runs. The CPU time is, of course, machine dependent, but the relative magnitudes shown here should be applicable to scalar computers in general.
- 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[1]. 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 approximated internally by finite differences.
- All solvers use dense linear algebra techniques.

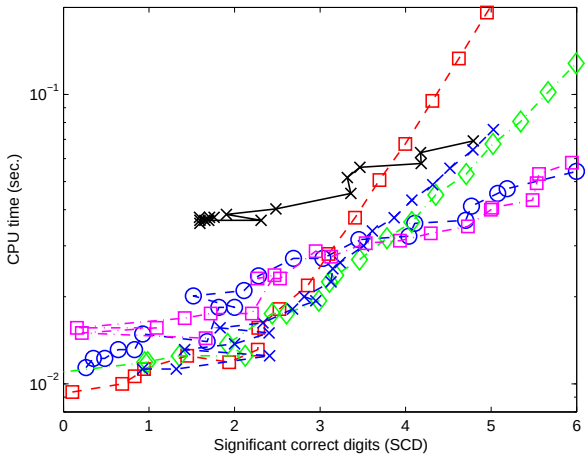
5.3.2 Work–Precision Diagrams

Figure 9 shows the work-precision diagrams when solving the SARA based reaction model using 25 and 100 subintervals. First of all we note that the

[1] For this purpose, the ESDIRK solvers use the algorithm suggested by Hairer and Wanner (1996).



(a) 100 subintervals.



(b) 25 subintervals.

Fig. 9 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

resulting accuracies in the solution are, of course, different from the local tolerances provided as input to the solvers. The tolerances merely govern the local error and step size control. Several general observations are made. In the high accuracy range (4–6 significant correct digits) the methods perform according to their order, and asymptotically the high order methods (DASSL, LSODE and ESDIRK45) will outperform the lower order methods

(e.g. ESDIRK12 and ESDIRK23). However, we are mainly interested in the accuracy range corresponding to 2–4 significant correct digits. For this range, the low and intermediate order methods perform best. 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 all seem to settle at a constant level when working at very loose tolerances instead of continuing the linear trend from the high accuracy end of the interval. Moreover, the behavior in the low accuracy end seems more erratic for all solvers. This can be partly attributed to the error and convergence controller. Consider for example ESDIRK45. An input tolerance level of $atol = rtol = 10^{-1}$ still produces 2 SCD. Inspection shows that the step size is limited by convergence in the nonlinear solver and not by teh local error. Thus, the large step sizes allowed by the error controller when working at loose tolerances lead to convergence failures in the nonlinear solver which then forces a step size reduction. Hence, the resulting accuracy is higher than requested, but at the cost of increased computation time. The change of slope in the ESDIRK curves is therefore due to a change in the mechanism governing step size selection. ODE error control devices perform best when operating at tight tolerance levels.

Comparing $N = 25$ and $N = 100$ shows that all solvers need more time when the integration is interrupted frequently. The overhead associated with restarts is, however, larger for the BDF methods. When the BDF methods are restarted they revert to first order and slowly build up higher order information. Thus, for the initial phase after a restart they need small time steps whereas the one-step ESDIRK methods recommence at high order and, hence, do not suffer as much from the frequent restarts.

In line with our comments in the previous section regarding the phase change algorithm, we note that both DASSL and LSODE manage to integrate across the phase changes. Inspection of the process shows, however, that both methods experience 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 in Figure 9.

Overall, the differences in performance are 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.

6 Conclusion

The work presented in this paper addresses time integration of the multi-scale in-situ combustion process. In particular, we have focused on developing specialized solvers for integration of the stiff chemical kinetics subject to phase equilibrium constraints. To facilitate the algorithmic development we constructed a kinetic cell model that allows us to study the kinetics and phase behavior part of in-situ combustion in an isolated setting. Based on analysis of in-situ combustion kinetics characteristics we chose as stiff integrator the

class of ESDIRK methods. We extended the methods with algorithms for handling the discontinuities that arise due to phase changes. In addition, we outlined important aspects for efficient implementation of the ESDIRK methods.

Performance comparisons between the ESDIRK solvers and off-the-shelf, dedicated stiff ODE solvers showed that the new solvers are comparable or better in terms of computational speed, especially over short integration intervals as required when implemented in an operator splitting environment. The proposed method for handling phase changes during simulation proved robust in detecting and locating changes by significantly reducing the number of convergence and error test failures when crossing phase boundaries.

6.1 Future Research

The results of this paper represent one step towards a full time-stepping methodology that uses specialized time integrators for the various physical sub-processes. Coupling of the methods for the reaction substep with other steps in the splitting scheme and establishing and testing the full framework is forthcoming.

For the reaction and phase equilibrium part of in-situ combustion alone, the directions for future research include:

- Extension to full equation-of-state based phase equilibrium including extension of the algorithm for detecting phase changes. This may be non-trivial for several reasons. For example, 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 fit into the framework of event functions as introduced in Section 4.4.
- Test of other classes of integration methods. Our results indicate that implicit one-step methods are well suited for integrating in-situ combustion kinetics. Other classes, besides the ESDIRK methods, could be of interest. For example, results from operator splitting integration of kinetics in large atmospheric chemistry models show that the Rosenbrock methods are good candidates as well (Sandu, Verwer, Loon, Carmichael, Potra, Dabdub and Seinfeld, 1997; Sandu, Verwer, Bloom, Spee, Carmichael and Potra, 1997).
- Use of efficient proxies for kinetics. Many repeated integrations are needed in the operator splitting approach. Often the conditions change only little from gridblock to gridblock and from time-step to time-step which suggests using a tabulation approach instead of repeating the integrations. The ISAT (In-Situ Adaptive Tabulation) method (Pope, 1997) has been used in computational combustion problems with significant savings in computation time compared to direct integration.
- Use of the VKC model for parametric sensitivity studies and for inverse problems in connection with laboratory experiments to determine reaction kinetic parameters. The VKC could be used both as a post-processing tool when analyzing experimental data and estimating parameters and in

the experimental design phase for determining optimal experimental conditions.

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

Adegbesan, K. O., Donnelly, J. K., Moore, R. G. and Bennion, D. W. (1987), Low-temperature-oxidation kinetic parameters for in-situ combustion: Numerical simulation, Society of Petroleum Engineers. SPE 12004: Presented at the 58th Annual Technical Conference and Exhibition, San Francisco, CA, October 5-8.

Alexander, R. (2003), 'Design and implementation of DIRK integrators for stiff systems', *Applied Numerical Mathematics* **46**, 1–17.

Bijl, H., Carpenter, M. H., Vatsa, V. N. and Kennedy, C. A. (2002), 'Implicit time integration schemes for the unsteady compressible navier-stokes equations: Laminar flow', *Journal of Computational Physics* **179**, 313–329.

Bousaid, I. S. and Ramey Jr., H. J. (1968), 'Oxidation of crude oil in porous media', *Society of Petroleum Engineers Journal* **234**, 137–148. SPE 1937.

Burger, J. G. and Sahuquet, B. C. (1972), 'Chemical aspects of in-situ combustion - heat of combustion and kinetics', *Society of Petroleum Engineers Journal* **253**, 410–422. SPE 3599.

Butcher, J. C. (1964), 'Implicit runge-kutta processes', *Mathematics of Computation* **18**(85), 50–64.

Butcher, J. C. and Chen, D. J. L. (2000), 'A new type of singly-implicit runge-kutta methods', *Applied Numerical Mathematics* **34**, 179–188.

Castanier, L. M. and Brigham, W. E. (2004), In-situ combustion. Society of Petroleum Engineers Handbook.

Coats, K. H. (1980), 'In-situ combustion model', *Society of Petroleum Engineers Journal* **269**, 533–554.

Crookston, R. B., Culham, W. E. and Chen, W. H. (1979), 'A numerical simulation model for thermal recovery processes', *Society of Petroleum Engineers Journal* pp. 37–58.

Crouzeix, M. (1975), Sur L'approximation Des Équations Différentielles Opérentielles Lineéaires Par Des Méthodes de Runge-Kutta, PhD thesis, Université Paris.

Dabbous, M. K. and Fulton, P. F. (1974), 'Low-temperature-oxidation reaction kinetics and effects on the in-situ combustion process', *Society of Petroleum Engineers Journal* **255**, 253–262. SPE 4143.

Enright, W. H., Jackson, K. R., Nørsett, S. P. and Thomsen, P. G. (1986), 'Interpolants for runge-kutta formulas', *ACM Transactions on Mathematical Software* **12**(3), 193–218.

Fassihi, M. R., Brigham, W. E. and Ramey Jr., H. J. (1984*a*), 'Reaction kinetics of in-situ combustion: Part 1 - observations', *Society of Petroleum Engineers Journal* pp. 399–407.

Fassihi, M. R., Brigham, W. E. and Ramey Jr., H. J. (1984*b*), 'Reaction kinetics of in-situ combustion: Part 2 - modelling', *Society of Petroleum Engineers Journal* pp. 408–416.

Freitag, N. P. and Exelby, D. R. (2006), 'A SARA-based model for simulating the pyrolysis reactions that occur in high-temperature EOR processes', *Journal of Canadian Petroleum Technology* **45**(3), 38–44.

Freitag, N. P. and Verkoczy, B. (2005), 'Low-temperature oxidation of oils in terms of SARA fractions: Why simple reaction models don't work', *Journal of Canadian Petroleum Technology* **44**(2), 54–61.

Gerritsen, M., Kovscek, A., Castanier, L., Nilsson, J., Younis, R. and He, B. (2004), Experimental investigation and high resolution simulator of in-situ combustion processes: 1. simulator design and improved combustion with metallic additives, Society of Petroleum Engineers. SPE 86962: Presented at the SPE International Thermal Operations and Heavy Oil Symposium, Bakersfield, CA, March 16-18.

Grabowski, J. W., Vinsome, P. K., Lin, R. C., Behie, A. and Rubin, B. (1979), A fully implicit general purpose finite-difference thermal model for in-situ combustion and steam, Society of Petroleum Engineers. SPE 8396: Presented at the 54th Annual Fall Technical Conference and Exhibition of the SPE, Las Vegas, Nevada, September 23-26.

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

Hairer, E., Nørsett, S. and Wanner, G. (1996), *Solving Ordinary Differential Equations I*, second revised edn, Springer.

Hairer, E. and Wanner, G. (1996), *Solving Ordinary Differential Equations II*, second revised edn, Springer.

Hindmarsh, A. C. (1983), ODEPACK, a generalized collection of ODE solvers, *in* R. S. Stepleman, ed., 'Scientific Computing (IMACS Transactions on Scientific Computing, Vol. 1)', North-Holland, Amsterdam, pp. 55–64.

Houbak, N., Nørsett, S. P. and Thomsen, P. G. (1985), 'Displacement or residual test in the application of implicit methods for stiff problems', *IMA Journal of Numerical Analysis* **5**(3), 297–305.

Islam, M. R., Chakma, A. and Farouq Ali, S. M. (1989), State-of-the-art of in-situ combustion modelling and operations, Society of Petroleum Engineers. SPE 18755: Presented at the SPE California Regional Meeting, Bakersfield, CA., April 5-7.

Kennedy, C. A. and Carpenter, M. H. (2003), 'Additive runge-kutta schemes for convection-diffusion-reaction equations', *Applied Numerical Mathematics* **44**, 139–181.

Kok, M. V. and Karacan, C. O. (1997), 'Behavior and effect of SARA fractions of oil during combustion'. SPE 37559: Presented at the International Thermal Operations and Heavy Oil Symposium, Bakersfield, CA, February 10-12.

Kutta, W. (1901), 'Beitrag zur näherungsweisen integration totaler differentialgleichungen', *Zeitschrift für Mathematik und Physik* **46**, 435–453.

Kværnø, A. (2004), 'Singly diagonally implicit runge-kutta methods with an explicit first stage', *BIT Numerical Mathematics* **44**, 489–502.

Kværnø, A., Nørsett, S. P. and Owren, B. (1996), 'Runge-kutta research in trondheim', *Applied Numerical Mathematics* **22**, 263–277.

Moore, G., Mehta, R. and Ursenbach, M. (2002*a*), 'Air injection for oil recovery', *Journal of Canadian Petroleum Technology* **41**(8), 16–19.

Moore, R. G., Mehta, S. A. and Ursenbach, M. G. (2002*b*), A guide to high pressure air injection (HPAI) based oil recovery, Society of Petroleum Engineers. SPE 75207: Presented at the SPE/DOE Improved Oil Recovery Symposium, Tulsa, Oklahoma, April 13-17.

Nilsson, J., Gerritsen, M. and Younis, R. (2005), A novel adaptive anisotropic grid framework for efficient reservoir simulation, Society of Petroleum Engineers. SPE 93243: Presented at 2005 SPE Reservoir Simulation Symposium, Houston, Texas, January 31 - February 2.

Nørsett, S. P. (1974), Semi-explicit runge-kutta methods, Technical Report 6, Department of Mathematics, University of Trondheim. ISBN-82-7151-009-6.

Park, T. and Barton, P. I. (1996), 'State event location in differential-algebraic models', *ACM Transactions on Modeling and Computer Simulation* **6**(2), 137–165.

Petzold, L. R. (1982), 'DASSL: A differential/algebraic system solver'. 10th IMACS World Congress on System Simulation and Scientific Computation.

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

Prats, M. (1986), *Thermal Recovery*, Vol. 7 of *SPE Monograph Series*, Society of Petroleum Engineers.

Prothero, A. and Robinson, A. (1974), 'On the stability and accuracy of one-step methods for sloving stiff systems of ordinary differential equations', *Mathematics of Computation* **28**(125), 145–162.

Ren, Y., Freitag, N. P. and Mahinpey, N. (2005), 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.

Runge, C. (1895), 'über die numerische auflösung von differentialgleichungen', *Mathematische Annalen* **46**, 167–178.

Sandu, A., Verwer, J. G., Bloom, J. G., Spee, E. J., Carmichael, G. R. and Potra, F. A. (1997), 'Benchmarking stiff ODE solvers for atmospheric chemistry problems: II. rosenbrock solvers', *Atmospheric Environment* **31**(20), 3459–3472.

Sandu, A., Verwer, J. G., Loon, M. V., Carmichael, G. R., Potra, F. A., Dabdub, D. and Seinfeld, J. H. (1997), 'Benchmarking stiff ODE solvers for atmospheric chemistry problems: I. implicit vs. explicit', *Atmospheric Environment* **31**(19), 3151–3166.

Westbrook, C. K., Mizobuchi, Y., Poinsot, T. J., Smith, P. J. and Warnatz, J. (2005), 'Computational combustion', *Proceedings of the Combustion Institute* **30**, 125–157.

Williams, R., Burrage, K., Cameron, I. and Kerr, M. (2002), 'A four-stage index 2 diagonal implicit runge-kutta method', *Applied Numerical Analysis* **40**, 415–432.

Younis, R. and Gerritsen, M. (2006), Multiscale process coupling by adaptive fractional stepping: An in-situ combustion model, Society of Petroleum Engineers. SPE 93458: Presented at the 2006 SPE/DOE Symposium on Improved Oil Recovery, Tulsa, Oklahoma, April 22-26.