

Improved Chemical Kinetics Numerics for the Efficient Simulation of Advanced Combustion Strategies

Federico Perini, Bishwadipa Das Adhikary, Jae Hyung Lim, Xingyuan Su, Youngchul Ra, Hu Wang, and Rolf Reitz
University of Wisconsin

ABSTRACT

The incorporation of detailed chemistry models in internal combustion engine simulations is becoming mandatory as local, globally lean, low-temperature combustion strategies are setting the path towards a more efficient and environmentally sustainable use of energy resources in transportation. In this paper, we assessed the computational efficiency of a recently developed sparse analytical Jacobian chemistry solver, namely 'SpeedCHEM', that features both direct and Krylov-subspace solution methods for maximum efficiency for both small and large mechanism sizes. The code was coupled with a high-dimensional clustering algorithm for grouping homogeneous reactors into clusters with similar states and reactivities, to speed-up the chemical kinetics solution in multi-dimensional combustion simulations. The methodology was validated within the KIVA-ERC code, and the computational efficiency of both methods was evaluated for different, challenging engine combustion modeling cases, including dual fuel, dual direct-injection and low-load, multiple-injection RCCI, direct injection gasoline compression ignition (GDICI), and HCCI engine operation using semi-detailed chemistry representations. Reaction mechanisms of practical applicability in internal combustion engine CFD simulations were used, ranging from about 50 up to about 200 species. Computational performance for both methods was observed to reduce the computational time for the chemistry solution by up to more than one order of magnitude in comparison to a traditional, dense solution approach, even when employing the same high-efficiency internal sparse algebra and analytical formulations. This confirms that consideration of detailed chemistry is not a bottleneck anymore, allowing use of larger and more refined meshes. Further research that focused on algorithms for fast and efficient advection with a large number of species is suggested.

CITATION: Perini, F., Das Adhikary, B., Lim, J., Su, X. et al., "Improved Chemical Kinetics Numerics for the Efficient Simulation of Advanced Combustion Strategies," *SAE Int. J. Engines* 7(1):2014, doi:10.4271/2014-01-1113.

INTRODUCTION

Research for advanced, cleaner and more efficient combustion strategies in internal combustion engines has been guided in the last decade by the advancements in computer modeling. This has enabled understanding of the interactions between fuel-air mixture preparation, ignition kinetics and pollutant formation in both spark-ignited and compression ignition combustion modes, through more comprehensive and reliable computer models [1]. Advancements in chemical kinetics modeling have allowed the development of detailed reaction mechanisms for diesel and gasoline fuel surrogates [2-3], that incorporate thousands of species and elementary reaction steps, providing insight into the interactions between hydrocarbon compounds on ignition, and at the same time yielding quantitatively accurate predictions of the major pollutant species - carbon monoxide (CO), unburned hydrocarbons (UHC), nitrogen oxides (NO_x), as well as soot formation precursors (such as polycyclic aromatic hydrocarbons, PAH) [4].

While these models have successfully been applied to simple, zero- or one-dimensional reactor or flame simulations, the computational burden for solving detailed chemical kinetics in multi-dimensional computational fluid dynamics (CFD) simulations is still too demanding when adopting fully comprehensive models. This is due to the stiffness of the system of ODEs describing chemical kinetics, and the expense of advecting a huge number of species over the discretized domain. Thus, current practical solution approaches typically make use of splitting the chemical kinetic system from the CFD solver, and achieve detailed fuel kinetics modeling through:

- Skeletal mechanisms for multiple fuels or multi-component fuel surrogates, with up to about 200 species [5-6]; or local reduction of more detailed reaction mechanisms through on-the-fly reduction methods, often referred to as "dynamic adaptive chemistry" (DAC) methods [7, 8, 9, 10]; or direct integration of the detailed mechanisms using advanced, sparse solution numerics [11, 12, 13];
- Strategies to reduce the number of chemistry integrations

from a full-chemistry approach, that requires solution of a chemical kinetics system on every cell of the CFD domain at every time-step of the flow solver, to a limited number of representative reactor configurations, through storage-retrieval techniques [14, 15, 16, 17], multi-zone approaches [18], cell clustering algorithms [19, 20, 21].

In this work, we have developed and applied an accurate and computationally efficient methodology to incorporate detailed chemical kinetics in practical internal combustion engine simulations, featuring validated reaction mechanisms for multiple and multi-component fuels. The approach features a sparse analytical Jacobian chemistry solver, with optimal-degree interpolation of expensive thermo-chemistry functions ("SpeedCHEM") [11], to accomplish the integration of the chemical kinetic ODE system, and a high-dimensional clustering algorithm (HDC) [22] to reduce the span of the discretized domain by grouping cells with similar reactivity within a hyper-dimensional state space representation.

The approach was coupled with the KIVA-ERC code, and both the SpeedCHEM solver [23,24] and the HDC clustering algorithm [25] were validated for standard reference cases. The approach was applied and its accuracy and scalability were verified by simulating a variety of engine combustion cases, including dual fuel, dual direct-injection and low-load, multiple-injection RCCI, direct injection gasoline compression ignition with multiple injections (GDICI), direct injection, and HCCI engine operation using a semi-detailed reaction mechanism.

METHODOLOGY

Solution of the Chemical Kinetic ODE System

We developed a computational methodology that features full integration of detailed reaction mechanisms with no mechanism reduction, to provide a numerically exact solution. The SpeedCHEM chemistry program [11] solves the system of ordinary differential equations of chemical kinetics in the form needed in multi-dimensional CFD codes, and other cases, including constant-volume ignition simulations: given an arbitrary reaction mechanism, containing a set of n_r reactions among n_s species,

$$\sum_{i=1}^{n_s} \nu'_{k,i} M'_i \leftrightarrow \sum_{i=1}^{n_s} \nu''_{k,i} M''_i, \quad k = 1, \dots, n_r, \quad (1)$$

where ν' and ν'' represent stoichiometric reaction coefficients for reactants and products, respectively, and M the species names, the code computes the laws that determine the temporal evolution of the thermodynamic state of the described homogeneous, reacting gas-phase mixture with species mass conservation, described by:

$$\frac{dY_i}{dt} = \frac{W_i}{\rho} \sum_{k=1}^{n_r} (\nu''_{k,i} - \nu'_{k,i}) q_k, \quad i = 1, \dots, n_s, \quad (2)$$

where the q_k terms represent the reaction rates progress variable, W the species molecular weights, and ρ the mixture density. The equation yields the rates of change of species mass fractions Y_i , that overall verify the mass conservation constraint: $\sum Y_i = 1$, or $\sum dY_i/dt = 0$. Energy conservation, for ignition in a constant-volume adiabatic reactor, is given by:

$$\frac{dT}{dt} = -\frac{1}{\bar{c}_v} \sum_{i=1}^{n_s} \left(\frac{U_i}{W_i} \frac{dY_i}{dt} \right), \quad (3)$$

where U represents the species' molar internal energies, $\bar{c}_v$ the mixture average specific heat at constant volume in mass units, and the ODE yields the rate of change of reactor temperature, dT/dt .

Libraries for the evaluation of species and mixture-averaged thermodynamic properties and of the kinetic laws for different reaction types (Arrhenius, three-body, etc.) are included in the code. All of the equations are calculated in a matrix-based representation that makes use of an internal sparse matrix algebra library, specifically developed for this application. As Jacobian matrix evaluation and solution are the most demanding tasks during the integration of the ODE system [11,26] due to its stiff and sparse nature. Solution of the chemical kinetics problem is accomplished by integrating the system of equations with a number of computationally efficient stiff-ODE solvers, including VODE, LSODE, DASSL, RADAU5, RODAS, and MEBDF. Where not available, the internal ODE's sparse algebra for each of these solvers was extended with the capability for solving linear systems using sparse matrices and direct sparse factorization through LU decomposition. Furthermore, an optimal-degree interpolation approach for computationally expensive temperature-dependent thermodynamic functions and reaction rate parameters allowed their evaluation to be sped up by up to two orders of magnitude in comparison with their evaluation using intrinsic compiler functions [25].

The major advantage of this computational setup is that a numerically accurate solution can be provided in an overall CPU time that scales linearly with the number of species, as reported in Figure 1 for constant-volume ignition cases. The overall computational performance was tested for an ignition delay study on a range of reaction mechanisms for single and multi-component hydrocarbon fuels. The tests featured 18 ignition delay calculations at stoichiometric conditions, initial pressure of 20 bar, initial temperatures between 700K and 1400K, and integration tolerances of $\epsilon_R = 10^{-8}$, $\epsilon_A = 10^{-20}$, on an Intel Core i5-2400 personal computer running at 3100 MHz and with 8GB DDR3 memory with 1333 MHz frequency.

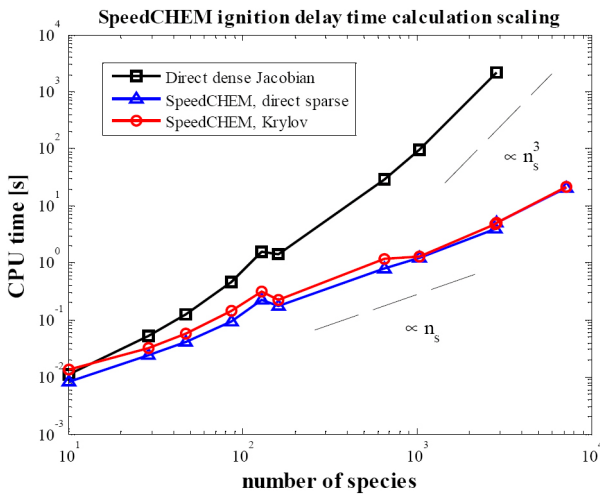


Figure 1. Linear scalability of the SpeedCHEM solver for ignition delay calculations. (black) dense, (blue) sparse, (red) Krylov linear system solver comparison.

It was observed that, for the ignition delay calculations, the SpeedCHEM solver ran from about 2 up to about 7 times faster than what has been reported for other sparse chemistry solvers [12] for the range of mechanism sizes of interest to CFD simulations, i.e., $50 < n_s < 700$.

When large reaction mechanisms are used, the direct solution of Jacobian-associated linear systems through sparse LU decomposition as in the aforementioned solvers may be expensive. Thus, preconditioning capabilities were implemented in the solver to enable coupling with iterative Krylov subspace solvers, such as the GMRES algorithm implemented in LSODKR, VODPK, DASKR. A full study of the derivation and calibration of a suitable preconditioner is reported in [26]; however, it was shown, as also summarized in Figure 1, that using the direct solver was the optimum approach for the range of mechanism sizes applicable to CFD simulations. Significant CPU time savings were enabled by the Krylov solver only for extremely large reaction mechanisms ($n_s > 7000$).

High-Dimensional Cell Clustering (HDC)

Even with the computational speed-ups allowed by the SpeedCHEM package, the full chemistry solution in engine CFD simulations is still inefficient due to its inability to identify regions in the domain with similar reactivities, and thus to reduce the number of reactor calculations at every time-step of the fluid solver. Turbulence-chemistry interaction models have been developed to account for the detailed chemistry at the cell level. However, the neglect of sub-grid scale effects, and the use of direct chemistry integration has been shown to provide good results for both conventional diesel and low-temperature combustion [27], even on coarse grids. This enables the possibility of adopting a clustering approach that groups cells based on their thermo-chemical state.

The approach adopted in this study is reported in Figure 2: 1) First, a suitable cell clustering algorithm is used to identify cells in the discretized domain with similar thermodynamic states and reactivities; 2) then, the cells are 'grouped', or 'clustered', into larger homogeneous adiabatic reactors based on a mass weighting approach; 3) the chemistry ODE system is integrated only for this smaller set of 'cell clusters', providing rates of change of species mass fractions for each in the current CFD solver time step; 4) finally, the rates of change in species mass fractions are re-mapped back to the individual cells, using a mass-conservation based approach.

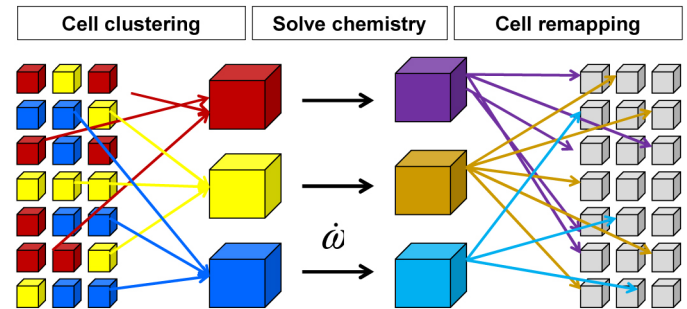


Figure 2. Schematic of the CFD chemistry solution approach with cell clustering.

In the present study, SpeedCHEM was used as the chemistry solver, and the procedure developed by Babajimopoulos et al. [18] and improved by Liang et al. [28] was adopted for remapping. A high-dimensional clustering algorithm [22] was developed and used to accomplish for cell clustering.

The developed algorithm considers the chemical states of each cell in the domain, viz., temperature and selected species mass fractions, to define a high-dimensional representation of the initial reactor conditions, $\mathbf{x}_j$, $j = 1, \dots, n_{\text{cells}}$, where the total number of dimensions d is given by temperature plus a selected subset of species $\mathbf{S}$:

$$\begin{aligned} x_{1,j} &= T_j, \\ x_{2:d,j} &= Y_{k,j}, \forall k \in \mathbf{S}. \end{aligned} \quad (4)$$

In order to reduce the variable dependency of the results, the algorithm operates in a purely geometrical hyper-space, where each cell's variable is normalized within its current global, in-cylinder range. Thus, the HDC algorithm needs to accomplish the task of clustering a cloud of points in a high-dimensional hyperspace.

To provide the algorithm with the required accuracy for combustion calculations, the following strategies were adopted:

- The distance measure should not mitigate distances among each single dimension, as every variable in the high-dimensional representation has the same physical meaning; thus, the 'Manhattan' distance metrics were

adopted, as represented in Figure 3, which express the distance function d between two arbitrary points $\mathbf{x}$ and $\mathbf{y}$ in the high-dimensional space as:

$$d_M(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d |x_i - y_i|. \quad (5)$$

- The user needs to provide temperature and mass fraction accuracies that should not be exceeded by the algorithm. Thus, it was chosen to initialize the clustering procedure with an initial set of cluster centers, placed in a grid-like fashion (also extremely efficient as it is possible to assign each cluster center a fast unique index as in block-structured grids), as reported in Figure 4. Some of the clusters may be deleted when they are not boxing any points; the initial distances among the cluster centers being defined by the accuracy constraints in the physical (dimensional) space;
- The cluster centers should not move further than the requested accuracy distances during the execution of the clustering algorithm, otherwise the accuracy settings would not be complied with. Thus, a “bounding-box” constraint was defined, i.e., that every cluster center could only have as member points, any of the ones initially placed in the neighboring ‘boxes’.

These considerations led to the development of a modified version of the widely adopted k -means algorithm [29] into a ‘bounding-box-constrained k -means’, or BBC- k -means [22]. The algorithm proceeds similarly as the k -means algorithm, i.e., at every iteration, the clustered partition is improved by updating the cluster center positions as the geometrical averages of their member points' positions, and then re-evaluating all the point-to-cluster distances to assign every point to the closest cluster center (whose position has changed from the previous iteration). The grid-like discretization and boxing constraints improve the general k -means approach by 1) establishing a physically correct cluster center initialization; 2) complying with the desired accuracy constraints in the physical domain; 3) reduce the algorithm's computational cost from $O(n_p n_c)$, where n_p is the number of points and n_c the number of clusters, to $O(n_p 2^d)$, where d is the current number of dimensions, as every point is compared only to its boxing cluster centers, and not to all of them anymore. This dramatically affects the algorithm's performance in practical cases, where point-to-cluster distance calculations represent the bottleneck to the algorithm's scaling. Another important feature of the adopted approach is that the number of cluster centers, and thus the overall speed-up of the chemistry integration procedure, is automatically determined by the number of dimensions and the accuracy constraints only, and not on the number of points to be grouped.

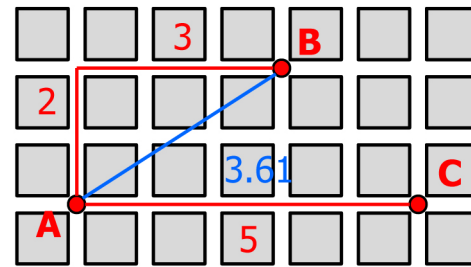


Figure 3. Comparison between Euclidean (blue) and ‘Manhattan’ (red) distance measures, in a two-dimensional space.

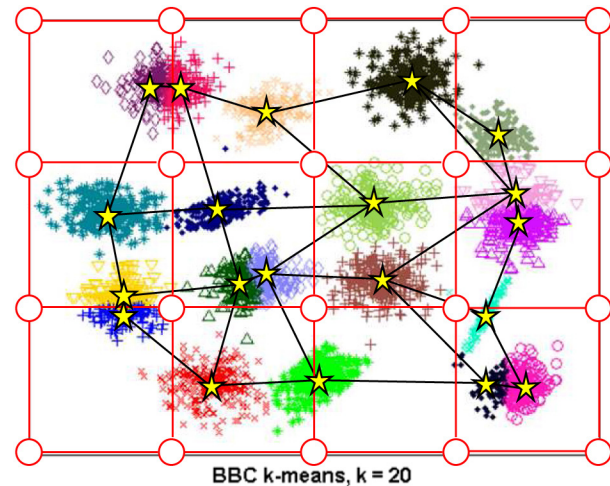


Figure 4. Example of operation of the HDC algorithm over a 2D dataset. The benchmark points set is colored according to final cluster membership. Red circles indicate initial cluster center positions, yellow stars with black lines the final cluster centers positions, and conserved grid-like structure.

An overview of the mechanics of the present approach is reported in Figure 4 for a sample 2D benchmark featuring 15 well-defined point clouds [30], and where the clustering algorithm is initialized to 20 initial cluster centers. The initial cluster grid-like discretization has red circles and lines, while the final cluster centers are yellow stars and black lines.

ENGINE SIMULATION RESULTS

Computational Setup

The developed computational approach was tested for a range of engine simulation problems, featuring very different combustion regimes and engine operation modes. All of the simulations were run using the KIVA-ERC code [31], a customized version of the KIVA-3v code with improved sub-models for spray dynamics and vaporization, turbulence and wall heat transfer treatments. The original detailed chemistry capability was replaced by the current SpeedCHEM solver and the high-dimensional clustering algorithm implementation. Details of the sub-models are reported in Table 1.