

An Automatic Sparse Matrix Solver for Reduced Reaction Kinetics

A.Arvidsson¹, T.Løvås², F.Mauss³

¹ Department of Combustion Physics, Lund Institute of Technology
P.O. Box 118, S-22100 Lund, Sweden

² Department of Engineering, Queen Mary University of London
London E1 4NS, UK

³ Department of Environmental Process Engineering
BTU Cottbus, Germany

1 Introduction

The treatment of reliable chemistry in complex flow codes for engine and furnace simulations are of great importance for the correct predictions of emission levels and control parameters such as ignition time and flame temperatures. However, the inclusion of complex chemistry in CFD codes is demanding in terms of both computational time and memory requirements. The chemical reaction system is governed by a set of differential equations determining the time evolution of the chemical species based on consumption and production through chemical reactions. The chemical reactions occur on time scales ranging over several orders of magnitude, and the numerical solution of such non-linear systems are therefore very stiff and hence very time consuming to solve. The problem is the subject of intensive research concerning the development of methods to reduce the set of differential equations without loss of important chemical information [1-5]. They often rely on the Quasi Steady State Assumption (QSSA) for species reacting on the shortest chemical time scales, replacing differential equations for these species with simpler algebraic equations.

There are a series of methodologies that are proposed in literature which aim to identify the steady state species, and to incorporate their evolution in the remaining set of equations for the non-steady state species. In this paper the steady state species are identified by means of a level of importance parameter, LOI, which is a combined sensitivity and chemical time scale identity [6]. Thereafter automatically written subroutines are produced containing the exact expressions for the remaining non- steady state species relations, in addition to an inner iteration loop for the algebraic steady state relations. However, the inner iteration can contain a set of strongly coupled algebraic equations, which in it self might be time consuming to solve. Hence the need for algorithms that speed up the both the solution for the differential equations and the algebraic equations are needed. In this paper we present such a methodology based on a simple procedure that automatically optimizes the sparseness of the matrix. In the next section we outline the numerical and computational aspects of the method. Then we investigate the performance of the reduced mechanisms compared to that of the full mechanism both in terms of accuracy and computational savings for homogenous autoignition of n-heptane. Finally we present our concluding remarks.

2 The Sparse solver for Reduced Chemical Mechanism

The equations that need to be solved for each species are the standard species conservation equations. The differential equations are typically solved with a Newton solver (in the following called; outer solver), and for each time step an entire Newton iteration must be performed for the set of equations. Furthermore, for each evaluation of the set of partial differential equations, an inner loop for the algebraic equations, representing the steady state species, is called, i.e. the solver (in the following called inner solver) for the algebraic equations is called many times. Therefore, high demands are put on the new inner solver in terms of computational speed and accuracy, to guarantee good convergence of the outer solver.

The new algorithm uses a modified Newton solver for the algebraic equations. A Newton solver was chosen as a starting point because of the high accuracy demands. Using a Newton solver as a starting point implies that a Jacobian matrix must be calculated followed by a Gauss elimination. It also means that the linear solver and the update of the source-vector must be called at each iteration step of the Newton solver for the algebraic equations. The Newton solver can therefore be speeded up considerably by optimizing the building of the Jacobian, the Gauss elimination, the linear solver and the update of the source-vector.

The CPU time that can be gained strongly depends on the sparsity pattern of the Jacobian. The subroutines in the Newton solver can be automatically written as program source code (preprocessed), since the sparsity pattern of the Jacobian matrix is constant in time. The reason for this is that the species involved in the chemical reactions in the mechanism are fixed, that is the non-zero elements in the Jacobian matrix will always be located at the same place. Since the locations of the non-zero elements are known *a priori*, the operations which are to be made in the Gauss elimination of the Jacobian matrix are already known. Hence the entire Gauss elimination can be preprocessed in the automatically written subroutines for the reduced chemistry. The locations of the non-zero elements in the upper triangular matrix, which are left after the Gauss elimination, are also known since the entire procedure is a function of the sparsity pattern of the Jacobian matrix. This means that the linear-solver also can be preprocessed in the written subroutines. The time consuming subroutines in the inner solver are the Gauss elimination, the linear solver and the update of the source-vector. The speed of these subroutines is proportional to the number of operations involved in them, which in turn depends on the sparsity pattern of the Jacobian. Hence, the sum of the non-zero elements in the upper triangular matrix and the number of operations in the Gauss elimination must be minimized in order for the solver to be as fast as possible. The sparsity pattern depends on the order of the steady state species, which means that the speed of the algebraic solver strongly depends on the order of the steady state species. Which sparsity pattern of the Jacobian, that is which order of steady state species, that minimizes the sum of operations in the two subroutines is non-trivial to find analytically. Therefore an optimization algorithm is used, which is based on a simulated annealing procedure. A Metropolis algorithm is used to accept or decline the new steady state species vector, which is generated by a random change of the steady state species order. This algorithm will not guarantee that the global minimum is found, but in most cases a “deep enough” local minimum is found. The result is that the CPU time is gained for three reasons: the first being that the Gauss elimination, the update of the source-vector and the linear solver is pre-coded, so expensive IF-statements and DO-loops over index lists are excluded in the subroutines. Both procedures limit the optimization of the program. The second reason is that only the operations involving non-zero elements are written to the subroutines and the third is that number of operations is minimized by optimizing the sparseness pattern of the Jacobian.

3 Selection of QSSA species

Candidates for selection as steady state species are short lived species. The steady state species were selected based on their level of importance, LOI, a combined chemical lifetime and sensitivity measure, where the species' sensitivity was taken towards the prediction of OH [6]. Species with a level of importance lower than a user defined cut-off limit were selected for steady state, and their equations solved in accordance with the procedure outlined in the previous section. The choice of cut-off limit is arbitrary and depends on the desired level of reduction. It is found that close to the cut-off limit the LOI values are very similar; hence these species

are almost equally good steady state candidates. Around this limit the steady state candidates can be automatically tested one by one. It is often found that *groups* of species should be set to steady state rather than single species.

4 Results

A detailed mechanism for n-heptane consisting of 95 species is reduced down to 52 species when the entire initial temperature range, 667-1250 K, is considered. When only one initial temperature, 900 K, is considered the detailed mechanism is reduced to 30 species. In all cases the pressure and equivalence ratio were kept constant at $P=40$ bar and $\phi=1.0$. The CPU time decreases as the number of steady state species increases, while the accuracy of the ignition delays times are within 15% of the detailed mechanism and the species concentration profiles have similar accuracy compared to the detailed case. This is illustrated in Figures 1 and 2. Figure 1 shows the normalized CPU time versus number steady state species for $T=900$ K. The CPU time is 60 % of the detailed mechanism when 65 out of 95 species are in steady state. Figure 2 shows the accuracy of the reduced reaction mechanisms in terms of the normalized ignition delay time versus number steady state for $T=900$ K. The ignition delay time is 90 % of the detailed mechanism when 65/ 95 species are in steady state. The CPU time is found to fluctuate somewhat close to the final reduction level. This is a result of what was discussed in the previous section, which was that a better solution is sometime obtained when pairs or groups of species are set to steady state.

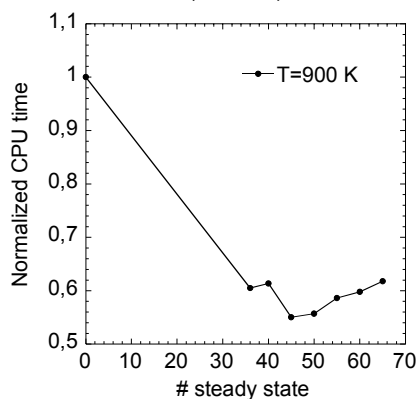


Figure 1: Normalised CPU time vs. number of steady state species for n-heptane (95 species in total) for $T=900$ K, $P=40$ bar and $\phi=1.0$

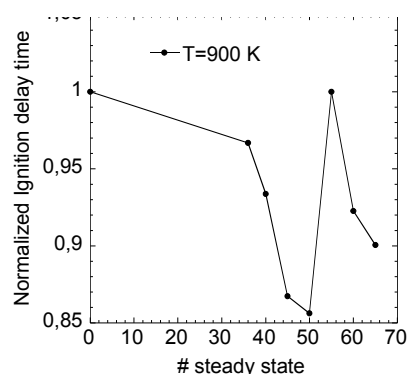


Figure 2: Normalised ignition delay time vs. number of steady state species for n-heptane (95 species in total) for $T=900$ K, $P=40$ bar and $\phi=1.0$

It is desirable that the reduced reaction mechanisms are valid for a wide range of physical conditions. Also, the mechanisms should require less CPU time for increasing reduction for these changing conditions. It is generally found that when the reduced mechanisms are to be valid for a wider range of physical conditions, the level of reduction is not as high as when a very narrow range chosen. It was found that when the temperature range was extended to $T=667-1250$ K, the final number of steady state species was 43 species out of 95. Figure 3 shows the normalized CPU time versus number steady state species for $T=667$, 900 and 1250 K, i.e. from low-temperature regime through out the transition to high-temperature regime. The CPU time is 60 % of the detailed mechanism when 43 out of 95 species are in steady state. This represents close to a linear reduction in CPU time to reduction of number of species. Theoretically a stronger reduction is predicted both according to the number of elements eliminated from the out iteration (n^2), and the number of elements reduced by the optimisation of the inner iteration loop. However, this does not account of communication between the two iteration loops which in itself requires CPU time that is somewhat difficult to quantify. An overall linear reduction of CPU in accordance to the reduction level is experienced also for other simulations the authors have performed. Figure 4 shows the ignition delay time versus $1000/T$ [K] for the range $T=667-1250$ K, for different level of reduction. It is clear

from this comparison that the reduced reaction mechanisms perform very well in the whole temperature range with a deviation of much less than 10% from the predicted ignition time employing the full reaction mechanism

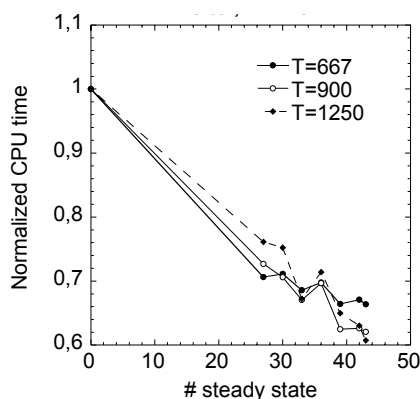


Figure 1: Normalised CPU time vs. number of steady state species for n-heptane (95 species in total) for T=667, 900 and 1250 K, P=40 bar and $\phi=1.0$

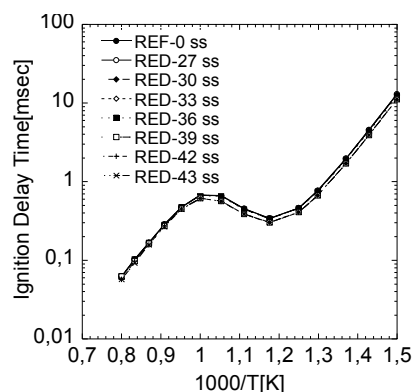


Figure 2: Ignition delay time vs. $1000/T$ for n-heptane (95 species in total) resulting from reduced mechanisms with increasing degree of reduction. T=667-1250 K, P=40 bar and $\phi=1.0$

5 Conclusions

A reduced reaction mechanism for n-heptane applied to ignition scenarios has been presented together with an optimising procedure applied to the Newton solver used for the set of steady state species. A reduction of almost 50% was possible for a mechanism that is valid for a wide range of physical conditions. When a narrower range is applicable, a significantly stronger reduction is possible, here up to 70% of the original mechanism. This results in a significant decrease in CPU time as the number of steady state species increases, while the accurate results in species profiles and ignition delay times are still maintained. By means of a Gauss-elimination procedure that only takes the non-zero elements into account, a significant speed up of the overall computation is achieved when this is accounted for in the automatically written routines. We observe a close to linear reduction in CPU time according to fraction of reduced species.

References

- [1] Turányi, T., (1990) Reduction of large reaction mechanisms, *New J. Chem.*, 14, 795–803.
- [2] Tomlin, A.S. and Turnyi, T. and Pilling, M.J. (1997), Mathematical tools for construction, investigation and reduction of combustion mechanisms, in : M.J. Pilling (Ed.), *Low-Temperature Combustion and Autoignition*, Comprehensive Chemical Kinetics, Vol. 35, Elsevier.
- [3] Mauss, F. and Peters, N. (1993), Reduced Kinetic Mechanisms for Premixed Methane-Air Flames, in: Peters, N. and Rogg, B. (Eds.), *Reduced Kinetic Mechanisms for Application in Combustion Systems*, Lecture Notes in Physics, New Series, m 15, Springer Verlag, pp. 58-75.
- [4] Maas, U. and S.B. Pope, (1994), Laminar Flame Calculations using Simplified Chemical Kinetics Bases on Intrinsic Low-Dimensional Manifolds, *Proc. Comb. Inst.*, Vol. 25, 1349-1356.
- [5] Lam, S. H. and D.A. Goussis, (1988), Understanding Complex Chemical Kinetics with Computational Singular Perturbation, *Proc. Comb. Inst.*, Vol. 22, 931-941.
- [6] Løvås, T., Nilsson, D. and Mauss, F. (2000), Automatic Reduction Procedure for Chemical Mechanisms Applied to Premixed Methane-Air Flames, *Proc. Comb. Inst.*, Vol. 28, pp. 1809-1817.