10th Meeting of the Western States Section of The Combustion Institute
Organized by the University of Wyoming
October 2–3, 2017
Laramie, Wyoming

pyMARS: an Open Software Package for Reducing Chemical Kinetics Models

Phillip O. Mestas[1], *Parker Clayton*[1], *and Kyle E. Niemeyer*[1,*]

[1]*Mechanical, Industrial and Manufacturing Engineering, Oregon State University, 104 Rogers Hall, Corvallis OR, United States*

[*]*Corresponding author:* kyle.niemeyer@oregonstate.edu

Abstract: Chemical kinetic models play a crucial role in reacting-flow simulations. Detailed models can contain hundreds to thousands of independent species, and thousands to tens of thousands of reactions. Performing high-fidelity, multidimensional simulations using such models poses prohibitive computational costs, so kinetic model reduction plays an active role in enabling future research. This study introduces pyMARS (Python-based Model Automated Reduction Software), an open-source software package that performs automatic model reduction using a variety of methods, based on the Cantera suite for handling chemical kinetics. These reduction method include the directed relation graph (DRG), DRG with error propagation (DRGEP), DRG-aided sensitivity analysis (DRGASA), and DRGEP with sensitivity analysis (DRGEPSA) methods. pyMARS (currently) uses autoignition simulations to generate thermochemical data for the reduction process and evaluate the error of potential skeletal models. Future version of pyMARS will include additional skeletal reduction methods, and serve both as a tool for reduction of large kinetic models and a testbed for determining optimal reduction strategies.

Keywords: ***Chemical kinetic models, Model reduction, Reaction kinetics***

1. Introduction

It is critical to have accurate and descriptive chemical kinetic models when performing reaction-flow simulations. Detailed models that describe systems such as gasoline and jet fuel surrogates, like the ones developed by Battin-Leclerc [1], tend to be very large. Due to the complexities of these systems, the size of a detailed model used for simulation can contain hundreds to thousands of independent species and thousands to tens of thousands of individual reactions. Westbrook et al. [2] and Herbinet et al. [3] have both developed reasonable models that are of this magnitude. However, the size of these models comes with a heavy computational cost. Running simulations becomes unreasonable when the chemical kinetic models are this large; thus, it is imperative to have an alternate method for simulating models that alleviates the problem [4, 5].

The solution is chemical kinetic model reduction. Reduced mechanisms relieve the burden of computational costs while still maintaining accuracy if developed correctly [6]. This study introduces pyMARS (Python-based Model Automated Reduction Software), an open-source software package that supports various model reduction techniques documented in literature by making use of the Cantera [7] suite for handling chemical kinetics. pyMARS is written in Python and released openly under the MIT license. The source code can be found at `github.com/Niemeyer-Research-Group/pyMARS`. These automatic reduction techniques can be performed

on models, taking out as many details as possible while still maintaining a specified amount of accuracy.

pyMARS currently supports the DRG (Directed Relation Graph) and DRGEP (Directed Relation Graph with Error Propagation) methods for model reduction [8, 9]. Both techniques use ideas from graph theory to determine which species should be eliminated from the model. They both represent the relationships between the species in the model graphically with the edge weights representing the dependency of each species on another. The DRG method only accepts edge weights that fall above a certain threshold and determines what should stay in the model based on what is connected to a few target species on the graph. The DRGEP method accepts all connections and looks at the greatest path from the target species to determine if a species should stay in the model or not. pyMARS iterates through different threshold values until it determines which one provides the desired level of error. The program then produces a new model without including any species that were cut out, or any reactions that involve them. The error is calculated by running an autoignition simulation with the original and reduced models under specified conditions, and comparing the difference between the autoignition times [4].

2. Methods

pyMARS currently has implementations of the previously developed DRGEP and DRG methods for model reduction [8, 9]. Both of the currently supported methods are iterative processes that repeatedly cut out species, and then compare the autoignition delay with that of the original mechanism to determine if the new model is still within a specified accuracy range. Although they have many similarities, the methods differ most greatly by how they use the graph to determine what species should be removed from the model. Before running the program, the user will have to provide the file that holds the detailed model and a file that contains the initial conditions of the system. The user must also provide a list of target species as well as a desired error level.

Once the program has started with the DRG method selected, an autoignition simulation is run to determine thermochemical data used to calculate direct interaction coefficients. These coefficients represent the dependency of one species on another species. The direct interaction coefficient of species A on species B is defined by Lu and Law [8] as r_{AB} where:

$$r_{AB} \equiv \frac{\sum_{i=1,I} |\nu_{A,i}\omega_i\delta_{Bi}|}{\sum_{i=1,I} |\nu_{A,i}\omega_i|} \tag{1}$$

$$\delta_{Bi} = \begin{cases} 1, & \text{if the ith elementary reaction involves species B,} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

I represents the number of reactions in the system, $\nu_{A,i}$ the stoichiometric coefficient of species A in the ith reaction, and ω_i the overall reaction rate of the ith reaction. The direct interaction coefficients are used as the weights on a directed graph of the species in the system. One graph is created for each set of initial conditions presented using the tools available in the NetworkX library [10]. An edge is only included on the graph if its weight is greater than a specified threshold value. Once the graph is created, a depth first search is done from each of the target species. If any species is not reached through the search, it is cut out of the mechanism for that threshold value. The program uses an iterative process to determine the maximum amount of species that can be removed before

reaching the allowed error level. Iteration occurs by altering the threshold value. The program starts with an arbitrary threshold value that is repeatedly divided by 10 until the error is calculated to be zero. The error percentage is calculated based off of autoignition delay by creating a reduced mechanism and running the autoignition simulation with that mechanism. The autoignition delay from that simulation is compared to the autoignition delay from the original mechanism to determine the error percentage. When a starting threshold value has been determined, the threshold value will be slightly increased and the error will be calculated again. Once the error has gone beyond the allowed level, the iteration will stop. All species not connected to the target species on the final graph are removed from the model, and all reactions involving any of those species are removed as well. Lu and Law [11] showed this method to be efficient, especially on large models. Target species selection for the DRG method is important, but not vital because all of the species that should be maintained will be connected. Since the method uses a depth first search to determine what will be used in the reduced model, all of the species that should be kept will be as long as one of the target species is in that group.

In addition, pyMARS also supports the DRGEP method of model reduction. Although its implementation is similar to that of DRG, the two methods vary in the way the graph is created and searched. Instead of making a new graph to search for each threshold level, the DRGEP method requires only one graph per initial condition before iteration and uses them to make a dictionary to determine what should be cut out. Direct interaction coefficients are calculated similarly to DRG. Details of the calculations were recorded by Niemeyer and Sung [12] as well as Pepiot-Desjardins and Pitsch [9] as:

$$r_{AB} = \frac{|\sum_{i=1}^{n_R} \nu_{A,i}\omega_i\delta_{Bi}|}{\max(P_A,C_A)} \tag{3}$$

where

$$\delta_{Bi} = \begin{cases} 1, & \text{if the ith elementary reaction involves species B,} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

$$P_A = \sum_{i=1}^{n_R} \max(0,\nu_{A,i}\omega_i) \tag{5}$$

$$C_A = \sum_{i=1}^{n_R} \max(0,-\nu_{A,i}\omega_i) \tag{6}$$

n_R represents the number of reactions in the system. To determine the overall interaction coefficient (OIC) for each species, a search is preformed from each target species on each graph to determine the greatest possible path to each other species. This search is done using a modified version of Dijkstra's algorithm [13] that finds the greatest possible path to each node when taking the products of the edge weights. Because all direct interaction coefficients are less than one, this algorithm can be used. Wagner [14] showed many of the optimizations that can be made to Dijkstra's algorithm and Niemeyer and Sung [12] established that this implementation of Dijkstra's algorithm is the best graph searching algorithm for DRGEP specifically. The NetworkX implementation of Dijkstra's algorithm was modified to fit the algorithm needed for pyMARS. The greatest path to each species from any target species on any of the graphs is stored in a dictionary. These values represent each species OIC. Species that are cut out of the mechanism are those with an OIC below the threshold value. The threshold value is iterated in the same way as it is in the DRG method. Each species

OIC is dependent on what target species are used. Because of this, target species selection is vital for the DRGEP method. The distance a species has from the target species will impact its OIC.

The DRG and DRGEP methods for model reduction are both currently supported by pyMARS. The methods use the same iterative process and error calculations. Although they both use ideas in graph theory to determine what should be cut from the mechanism, their implementations vary. DRG limits the connections that are placed on the graph, and only keeps species that are connected to the target species. DRGEP connects all species, and uses distance from the target species to determine what should be cut from the model. Both implementations return an accurate and smaller version of a detailed model without the user being involved in the details of the reduction.

3. Results

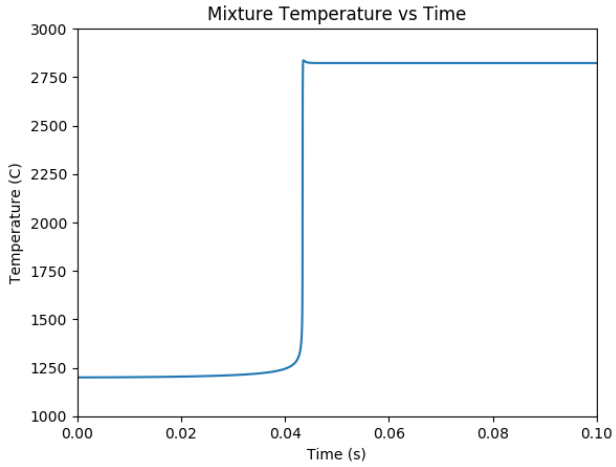

Figure 1: Data from an autoignition simulation done with the original GRI 3.0 mechanism. The initial conditions are a pressure of 1 atm, a temperature of 1200 K, and an equivalence ratio of 1.0.

So far, we have tested pyMARS on smaller sample mechanisms. One of these models is GRI-Mech 3.0 [15]. This model is relatively small, containing only 56 species and 325 reactions, but pyMARS was still able to reduce its size significantly. Figure 1 shows the temperature vs. time for a simulation using the original GRI mechanism. The conditions for this simulation are an initial temperature of 1200 K, a pressure of 1 atm, and an equivalence ratio of 1.0. The graph serves as a good visualization of autoignition delay.

When GRI-Mech 3.0 was used in pyMARS, both the DRG and DRGEP methods were able to reduce the model to nearly half of its original size while still maintaining less than five percent error in autoignition delay. Figure 2 shows a simulation ran on the reduced mechanism output by the DRGEP method with the same initial conditions. It is nearly identical to the plot produced from the original mechanism, with an autoignition delay that is almost the same as it was initially. pyMARS was able to produce a mechanism that is nearly half the size of its original version, but still can be used to accurately simulate the system. This result is significant because it will allow increased performance on necessary simulations with heavy computational costs. The methods supported by pyMARS can now be implemented on models quickly without much work from the user. This has potential to significantly speed up research for those using larger models.

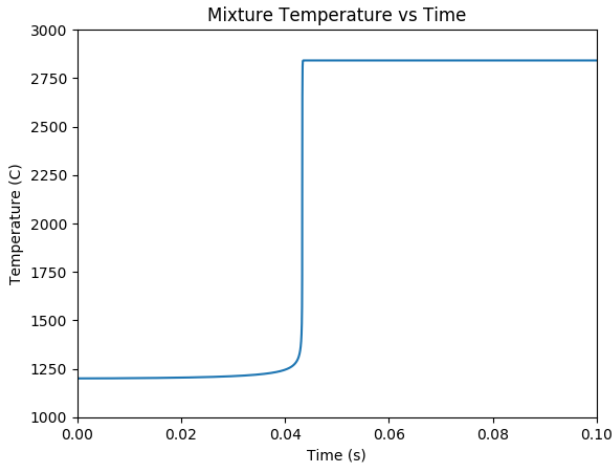

Figure 2: Data from an autoignition simulation done with the reduced GRI 3.0 mechanism with the same initial conditions as before. It is very close to the plot of the data from the original simulation. This shows that the reduced model can accurately describe the system.

4. Conclusions and future work

pyMARS is a valuable toolkit that is able to greatly reduce chemical kinetic models without sacrificing accuracy. It currently includes implementations for two different methods of model reduction. Both methods have so far been effective in reducing model size. Having a smaller model that maintains the accuracy of the original reduces the computational cost associated with detailed models. This result will have a positive impact on future research because it will greatly speed up long simulation times.

Although pyMARS is able to reduce the size of a model efficiently, there is room for improvement by adding additional methods that may prove more effective in some situations. One of these methods is sensitivity analysis as described by Rabitz et al. [16]. This method involves ordering the species based on their overall interaction coefficients. Once the species are ordered, they are removed one by one from the model and the error is calculated. This method suffers from long execution times because each species must be removed one by one. Therefore, it is efficient to use in conjunction with the DRGEP method as shown by Niemeyer et al. [17]. There are also plans to include an isomer lumping reduction method similar to the method described by Huang et al. [18]. With multiple methods for model reduction implemented, pyMARS will be more useful in diverse situations where some methods may be preferred over others. There are also plans to include automated testing for crucial functions to ensure that future versions don't harm the already existing code. More features will be added to pyMARS in the future such as the ability to produce plots from simulations, or write csv files that contain data about the reduction. With increased capabilities, pyMARS will be a more robust toolkit that will allow for models to be reduced even further.

Acknowledgements

This material is based upon work supported by the National Science Foundation under grant ACI-1535065, and the URSA Engage program at Oregon State University.

References

[1] F. Battin-Leclerc, Detailed chemical kinetic models for the low temperature combustion of hydrocarbons with application to gasoline and diesel fuel surrogates, (2008), DOI: 10.1016/j.pecs.2007.10.002.

[2] C. K. Westbrook, W. J. Pitz, O. Herbinet, H. J. Curran, and E. J. Silke, A comprehensive detailed chemical kinetic reaction mechanism for combustion of n-alkane hydrocarbons from n-octane to n-hexadecane, Combustion and Flame (2009), DOI: 10.1016/j.combustflame.2008.07.014.

[3] O. Herbinet, W. J. Pitz, and C. K. Westbrook, Detailed chemical kinetic oxidation mechanism for a biodiesel surrogate, Combustion and Flame (2008), DOI: 10.1016/j.combustflame.2008.03.003.

[4] K. E. Niemeyer, C. J. Sung, and M. P. Raju, Skeletal mechanism generation for surrogate fuels using directed relation graph with error propagation and sensitivity analysis, Combustion and Flame (2010), DOI: 10.1016/j.combustflame.2009.12.022, arXiv: arXiv:1607.05079v1.

[5] T. Lu and C. K. Law, Toward accommodating realistic fuel chemistry in large-scale computations, (2009), DOI: 10.1016/j.pecs.2008.10.002.

[6] Y. Ra and R. D. Reitz, A reduced chemical kinetic model for IC engine combustion simulations with primary reference fuels, Combustion and Flame (2008), DOI: 10.1016/j.combustflame.2008.05.002.

[7] D. G. Goodwin, H. K. Moffat, and R. L. Speth, *Cantera: An Object-oriented Software Toolkit for Chemical Kinetics, Thermodynamics, and Transport Processes*, http://www.cantera.org, Version 2.3.0, 2017.

[8] T. Lu and C. K. Law, A directed relation graph method for mechanism reduction, Proceedings of the Combustion Institute (2005), DOI: 10.1016/j.proci.2004.08.145.

[9] P. Pepiot-Desjardins and H. Pitsch, An efficient error-propagation-based reduction method for large chemical kinetic mechanisms, Combustion and Flame (2008), DOI: 10.1016/j.combustflame.2007.10.020.

[10] A. A. Hagberg, D. A. Schult, and P. J. Swart, Exploring Network Structure, Dynamics, and Function using NetworkX, Proceedings of the 7th Python in Science Conference Pasadena, CA USA, (2008), pp. 11–15.

[11] T. Lu and C. K. Law, Linear time reduction of large kinetic mechanisms with directed relation graph: N-Heptane and iso-octane, Combustion and Flame (2006), DOI: 10.1016/j.combustflame.2005.02.015.

[12] K. E. Niemeyer and C.-J. Sung, On the importance of graph search algorithms for DRGEP-based mechanism reduction methods, (2011), DOI: 10.1016/j.combustflame.2010.12.010.

[13] E. Dijkstra, A Note on Two Problems in Connexion with Graphs, Numerische Mathematik (1959).

[14] D. Wagner and T. Willhalm, LNCS 4393 - Speed-Up Techniques for Shortest-Path Computations, (2007), URL: http://i11www.informatik.uni-karlsruhe.de/.

[15] G. P. Smith, D. M. Golden, M. Frenklach, N. W. Moriarty, B. Eiteneer, M. Goldenberg, C. T. Bowman, R. K. Hanson, S. Song, W. C. Gardiner, V. V. Lissianski, and Z. Qin, GRI-Mech 3.0, http://www.me.berkeley.edu/gri_mech/.

[16] H. Rabitz, M. Kramer, and D. Dacol, Sensitivity Analysis in Chemical Kinetics, Ann. Rev. Phys. Chern 34 (1983) 4–9.

[17] K. E. Niemeyer, C. J. Sung, and M. P. Raju, Skeletal mechanism generation for surrogate fuels using directed relation graph with error propagation and sensitivity analysis, Combustion and Flame (2010), DOI: 10.1016/j.combustflame.2009.12.022, arXiv: arXiv:1607.05079v1.

[18] H. Huang, M. Fairweather, J. F. Griffiths, A. S. Tomlin, and R. B. Brad, A systematic lumping approach for the reduction of comprehensive kinetic models, Proceedings of the Combustion Institute (2005), DOI: 10.1016/j.proci.2004.08.001.