High-dimensional, unsupervised cell clustering for computationally efficient engine simulations with detailed combustion chemistry

Federico Perini[1,a]

[a]*Dipartimento di Ingegneria Meccanica e Civile, Università di Modena e Reggio Emilia, I-41125 Modena, Italy*

Abstract

A novel approach for computationally efficient clustering of chemically reacting environments with similar reactive conditions is presented, and applied to internal combustion engine simulations. The methodology relies on a high-dimensional representation of the chemical state space, where the independent variables (i.e. temperature and species mass fractions) are normalized over the whole dataset space. An efficient bounding-box-constrained k-means algorithm has been developed and used for obtaining optimal clustering of the dataset points in the high-dimensional domain box with maximum computational accuracy, and with no need to iterate the algorithm in order to identify the desired number of clusters. The procedure has been applied to diesel engine simulations carried out with a custom version the KIVA4 code, provided with detailed chemistry capability. Here, the cells of the computational grid are clustered at each time step, in order to reduce the computational time needed by the integration of the chemistry ODE system. After the integration, the changes in species mass fractions of the clusters are redistributed to the cells accordingly. The numerical results, tested over a variety of engine conditions featuring both single- and multiple-pulse injection operation with fuel being injected at 50 degrees BTDC allowed significant computational time savings of the order of 3 to 4 times, showing the accuracy of the high-dimensional clustering approach in catching the variety of reactive conditions within the combustion chamber.

Keywords:
cell clustering, detailed chemistry, high-dimensional space, k-means, internal combustion engines

1. Introduction

The advancements in computational resources have allowed, in recent years, combustion research to achieve quantitative predictive capabilities thanks to

*federico.perini@unimore.it

the adoption of chemical kinetics models in conjunction with multidimensional simulations [1]; the study of the interactions of physical and chemical processes, especially at the smallest scales [2], as well as the development of novel combustion concepts, which are able to exploit the variety of reactive conditions a fuel-oxidizer system can account for [3–6], is however urging the need for using comprehensive, detailed reaction models [7], which can be made of thousands species and more than ten thousands reactions [8–13]. In order to cope with such mechanism dimensions, a number of approaches have been developed with the aim of either achieving suitable computational time scaling with the mechanism dimension, or of avoiding unnecessary computations. For example, a fast and practical approach is to provide a reduced mechanism subset, by identifying the active species as the ones among which strong reaction exchanges occur [14–18]; other approaches simplify the computational effort by dividing reactions into fast and slow groups [19–22], or implement storage-retrieval techniques for reducing the number of chemistry ODE system integrations by adopting suitable approximations to high-dimensional functions [23–26]. Research is also active in identifying proper ODE integration techniques computationally suitable for stiff chemistry problems: as acknowledged [7], the dominating computational cost when integrating large reaction mechanism is due to factorization of the Jacobian matrix, if implicit or semi-implicit integration methods are adopted; for this reason most approaches aim at either adopting matrix-free integration methods or at reducing the Jacobian-related cost by either expressing it in a sparse format, or by introducing simplified, approximate Jacobian formulations [27–34].

The approach developed in the present work belongs to a number of studies which aim at reducing the overall computational cost due to detailed chemistry computations in multidimensional computational fluid dynamics (CFD) simulations of practical combustion systems – such as internal combustion engines –, by actually reducing the number of reacting environments the detailed chemistry ODE system needs to be integrated into. As Babajimopoulos et al. firstly showed [35], significant computational time savings can be achieved when the instantaneous chemical composition within a multidimensional domain fulfills a pattern-like structure, for instance a fuel-oxidizer charge stratification, such as that occurring in typical homogeneous-charge compression ignition (HCCI) engine combustion. The multidimensional domain can thus be represented as a multi-zone environment, where each zone owns defined temperature and mixture equivalence ratio, so that only one detailed chemistry ODE system needs to be solved per zone in each global advancement time-step, and then the results of the integration can be backward remapped to the original cells, proportionally to their initial compositions. This concept has been generalised by Liang et al. [36], where the recognition of the homogeneous zones has been set up as an evolutionary clustering problem, where the independent variables are cell's temperature and local mixture equivalence ratio. Barths et al. [37] applied a similar approach as a two-way coupling between the actual CFD simulation and a zero-dimensional multi-zone environment made up of a limited number of homogeneous, chemically reacting zones. Shi et al. [38] have shown that

significant computational time savings can be achieved also when the clustering procedure is done according to a cell proximity criterion. Another approach has been chosen by Goldin et al. [39], where the whole species and energy composition space has been adopted for describing each computational cell, and applied to 1D and 2D laminar flame computations.
The main idea underlying the present work is that the robustness and the potential of the partitioning approach mainly rely on the smartness of the cell clustering algorithm, intended as its ability to:

- catch the variety of reactive conditions in the multidimensional domain, that cannot be simplified into a unique parameter, and that are usually ruled by species associated to fast timescales;
- automatically identify, at each timestep, the optimal number of clusters;
- minimize the inner inhomogeneity of the agglomerates by covering the whole domain of compositions in a sparse way;
- do not introduce significant computational overhead and thus be suitable for large-scale parallel computations.

The paper thus describes the study of a complete, unsupervised high-dimensional clustering approach, validated and applied for internal combustion engine simulations with detailed chemistry; its presentation is structured as follows. In section 2, all the aspects of the approach developed for chemistry-based cell clustering are presented: first of all, the clustering problem is defined by introducing the dataset representation, the relationships between chemically-reacting CFD cells and their images in the clustering space, and suitable distance metrics; then, an unsupervised initialisation procedure is reported, which sets the initial cluster partition up as a grid-like structure. A full description and validation of a novel crisp clustering algorithm of the k-means class is then presented; the algorithm, named 'bounding-box-constrained' (BBC) k-means, is tailored for clustering chemistry-based datasets which typically model thermodynamic systems, whose behaviour is strongly nonlinear with respect to the variables' values. Finally, in section 3 the implementation of the proposed procedure into a customized version of the KIVA-4 code [40], provided with detailed chemistry capability, is presented and discussed focusing on the accuracy and on the computational time savings allowed by adoption of this procedure with respect to a standard solution where a chemistry ODE system is integrated in each cell of the domain. The analysis shows that the procedure proved to be robust on a variety of diesel engine cases involving different combustion modes, and that overall speed-ups of at least three times have been achieved for all the tested cases, with almost negligible overhead introduced by the clustering and remapping procedure.

2. Unsupervised high-dimensional clustering (UHDC)

As acknowledged [36, 39], the issue of reducing the overall computational demands due to solving chemical kinetics in multidimensional simulations can be addressed as a three-step procedure: (1) optimal clustering of the chemically reactive cell in the computational domain into a number of chemically homogeneous environments; (2) solution of the chemistry ODE system in each cluster, yielding internal energy and species mass fractions source terms at the cluster level; (3) mass-conservating redistribution of the cluster-level time-integrated quantities to the each single cell.
As far as the clustering problem is concerned, two main approaches are possible: crisp clustering algorithms such as the k-means [41] consider cluster centers as the average values among their own points; this approach is particularly suitable for grouping CFD domain cells on a chemistry basis thanks to its limited computational demands, where the most time-consuming task is that required by evaluation of point-to-center distances; its intrinsic averaging however tends to deteriorate the diversity of the initial integration conditions of the cells, and needs to be restored by a specific backward remapping procedures. On the other hand, in fuzzy clustering algorithms such as the common fuzzy c-means [42] each point belongs to each cluster center to a certain degree of membership; thus, cluster centers usually follow a more disperse distribution in the domain, and every single point can be viewed as the weighted average of all the cluster centers. From the chemical kinetics point of view, this latest aspect could be beneficial, as each cell in the computational grid may be represented as the weighted average of a smaller number of sparse and far-away-from-each-other reactive conditions. However, the strong non-linearity and anisotropic behaviour of chemical kinetics in combustion systems make this approach less suitable: the membership exponent approach used to quantify membership values to cluster centers would also require that species mass fraction ranges would be properly scaled so that each problem variable would have the same degree of importance in contributing to the membership function. Furthermore, fuzzy clustering algorithms typically require significantly higher computational efforts, that would render their adoption useful only in presence of huge computational domains.

2.1. High-dimensional clustering problem formulation

The chemical kinetics initial value problems treated in this work are used in order to compute species mass fractions and internal energy source terms as part of the operator-splitting context adopted in the KIVA family of codes [40, 43]. The dimensions of the chemistry integration problem are given by the number of the independent variables: $n_{eq} = n_s + 1$, n_s being the number of gas-phase species, plus the cell's internal energy term. The variety of combustion conditions typically occurring in a multidimensional domain suggests the need to consider an enough representative subset of these variables, in order to have each cell cluster represent a definite combustion regime. This approach has for example been adopted by Goldin et al. [39] and tested in laminar flame calculations. Other approaches, such as the one by Liang et al. [36], introduce

representative scalars of the combustion conditions such as the mixture's global equivalence ratio; in this paper we adopt a high-dimensional cluster representation for two reasons: (1) to preserve maximum computational accuracy at the broadest variety of combustion regimes, eventually sacrifying some amount of reduction in computational demands; (2) as acknowledged [44], high-dimensional clusters are usually sparse, and it is of absolute importance for computational combustion simulations to preserve the correct spatial stratification in mixture compositions.
Hence, we first of all define the chemistry state space representation for each cell j in the computational domain:

$$\boldsymbol{y}_j = [T_j, Y_{1,j}, Y_{2,j}, \dots, Y_{n_s,j}]^T, \tag{1}$$

being made up by cell temperature T $[K]$ and gas-phase species mass fractions, $Y_i, i = 1, \dots, n_s$. The clustering problem formulation thus features a number of tuples (points) equal to the number of active cells in the domain, with each point corresponding to the subset representation of the j-th cell's chemical state:

$$\boldsymbol{x}_j = [x_{1,j}, x_{2,j}, \dots, x_{d,j}]^T, j = 1, \dots, p, \tag{2}$$

where d is the total number of dimensions in the high-dimensional representation. The elements in the array $\boldsymbol{x}_j$ model the cell's temperature and the species mass fractions of the subset $\mathbb{S}$ of selected species for clustering:

$$\begin{aligned} x_{1,j} &= T_j, \\ x_{2:d,j} &= Y_{k,j}, \forall k \in \mathbb{S}. \end{aligned} \tag{3}$$

The clustering problem requires that the set of points $\boldsymbol{x}_j$, $j = 1, \dots, p$ is partitioned into an optimal number k of (chemically-)homogeneous clusters. Each cluster center, according to the crisp clustering choice, is a point itself in the reduced d-dimensional space, whose components are the mathematical averages of the points values that belong to that partition:

$$\boldsymbol{c}_i = [c_{1,i}, c_{2,i}, \dots, c_{d,i}]^T = \frac{1}{n_i} \sum_{j=1}^{n_i} \boldsymbol{x}_j, \qquad i = 1, \dots, k. \tag{4}$$

This property is particularly useful, and can be applied especially when returning to the high-dimensional chemical state space representation, where each cluster center can be fully modeled as the physical average of its owned cells; each i-th cluster center, containing n_i computational cells, thus owns proper mass m, pressure p, temperature T, density ρ, internal energy U and gas-phase composition:

$$
\begin{aligned}
m_i &= \sum_{c=1}^{n_i} m_c; \\
V_i &= \sum_{c=1}^{n_i} V_c; \\
Y_{j,i} &= \frac{\sum_{c=1}^{n_i} m_c\, Y_{j,c}}{m_i}, \qquad j = 1, \ldots, n_s; \\
T_i &= \frac{\sum_{c=1}^{n_i} m_c\, c_{v,c}\, T_c}{\sum_{c=1}^{n_i} m_c\, c_{v,c}}; \\
p_i &= \rho_i\, R\, T_i \sum_{j=1}^{n_s} \frac{Y_{j,i}}{W_j};
\end{aligned}
\tag{5}
$$

(6)

where R represents the universal gas constant in molar units, $W_j, j = 1, \ldots, n_s$ the species molar weights, and $c_{v,c} = \partial U_c / \partial T$ the constant volume specific heat value.

The hereby defined clustering problem adopts physical properties of the cells-points as independent problem variables; their direct adoption is however of difficult implementation: as a matter of fact, clustering algorithms rely on arbitrary distance metrics in order to evaluate the modeled distance between points, and every problem dimension is treated consistently by distance metrics by definition. For this reason, the adoption of variables that have different scales is not feasible for practical clustering problems in combustion: for instance, distances between mass fractions, which vary in the interval $Y \in [0, 1]$, would always be neutralized by temperature distances, where, at a certain time in a typical multidimensional domain, the temperature range can be wider than $1000K$.

For this reason, in this approach a normalization function has been implemented for each variable range, so that during the clustering problem, each variable is bound in the $[0, 1]$ range, and the whole cloud of points $\boldsymbol{x}_j$, $j = 1, \ldots, p$ is transferred into a unity d-dimensional hyperbox. In particular, each point in the domain is assigned to a corresponding image $\boldsymbol{\xi}_j$ in the unity hyperbox, and the shape function that pursues the variables' normalization is expressed as:

$$
\xi_{j,i} = \frac{x_{j,i} - \min_i (x_{j,i})}{\max_i (x_{j,i}) - \min_i (x_{j,i})}; \qquad i = 1, \ldots, p; \qquad j = 1, \ldots, d; \tag{7}
$$

here, computation of the inverse function is not needed because of the bijective relationship between points in the physical space and their images in the normalized hyperbox, so that cluster centers in the physical space can be directly built by averaging of the physical points values. Figure 1 presents a schematic view of the normalization process, in the simple case where only two dimensions are considered.

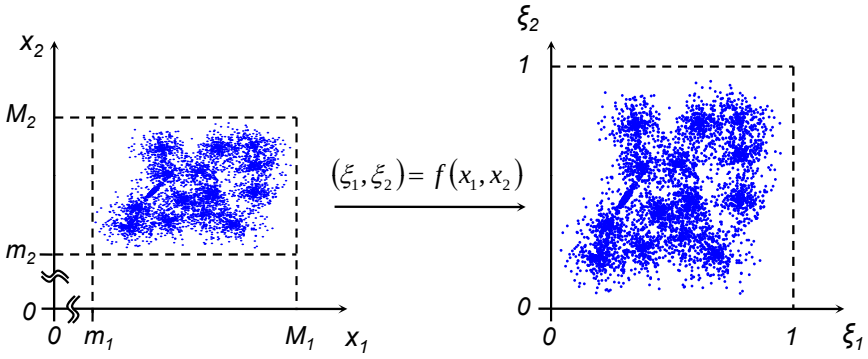

Figure 1: Two-dimensional representation of the variable normalization process into a d-dimensional unity hyperbox.

Distance metrics constitute the last parameter for completing the clustering problem definition, and also its choice is of absolute importance onto the effectiveness of the clustering algorithm. In this work, the general distance metric in Minkowski's form has been considered for the implementation:

$$d\left(\boldsymbol{x}_p, \boldsymbol{x}_q\right) = \left(\sum_{j=1}^{d} \left| x_{j,p} - x_{j,q} \right|^{\alpha} \right)^{1/\alpha} , \tag{8}$$

with α a real positive exponent. All of the possible metrics expressed by this formula suite the distance metrix axioms, including symmetry, identity, and triangle inequality; this formulation also reduces to the Euclidean distance metrics in case $\alpha = 2$. Some recent clustering-related literature points out that distances with high exponent value, e.g. $\alpha \geq 2$, are less suitable in crisp, high-dimensional clustering algorithms, and that values of the exponent in the range $0.1 < \alpha \leq 1$ are recommended. In [45], for example, it has been proven that low-exponent distance metrics, such as the Manhattan or 'taxicab' distance at $\alpha = 1$, or metrics with an even lower value of α, better identify cluster proximity along the same direction, and this feature is particularly useful for high-dimensional clusters where data are usually very sparse. Even if distance metrics with exponent values lower than unity have shown the best performances in clustering, in this work the $\alpha = 1$ Manhattan metrics have been adopted, as their evaluation requires the lowest computational demand in comparison with any other of the Minkowski distance metrics, which would require two real number powers per distance value.

2.2. Grid-like usupervised optimal cluster initialisation

In order to cope with one of the major problems of data clustering, i.e. the choice of the optimal number of clusters, k, a novel and tailored approach is here introduced, that is devoted to both providing a robust initialisation to the

cluster centers for the iterative clustering algorithm, and to implicitly defining the optimal number of clusters for the current partition. In this way we eliminate the outer iteration loop which characterizes typical unsupervised optimal clustering procedures, where the clustering algorithm is repeated at increasing number of partition clusters, and that is stopped after that the desired partition quality has been reached. This on one side provides significant computational time savings for the clustering process, and on the other hand it guarantees that the cluster centers in the partition do not collapse one onto each other. This latest feature is shown in the following paragraph.

We first of all define a clustering dimensional sparsity value in the unity hyperbox, $\varepsilon_j \in (0,1], j = 1, \ldots, d$ that estimates the desired maximum extension of a cluster's distribution of images along each dimension, and that is directly linked to its dimensional counterpart in the points space, $E_j, j = 1, \ldots, d$. This quantity is used to define a dimensional span, i.e. an estimated number of equally-spaced grid-like subdivisions along each dimension:

$$s_j = 2 + \text{int}\left(\varepsilon_j^{-1}\right), \qquad j = 1, \ldots, d, \tag{9}$$

where the *int* function pursues rounding to integer towards zero. Using the span value of each dimension, a d-dimensional grid is built, which discretizes the unity hyperbox $[0,1]^d$ where the normalized point images lay. Each vertex in the grid represents a potential cluster center initialisation: as a matter of fact, only the vertices of the active cells, which contain at least one point image, can become cluster centers; this idea is the seed for the bounding-box clustering procedure described in the following. Figure 2 represents an example of the grid-like cluster centers initialisation in a two-dimensional space.

It is worth to point out that this initialisation procedure yields an optimal number of cluster centers at the desired clustering accuracy, and a well-spaced cluster center initialisation, that can speed up the clustering algorithm convergence. The only input needed by the user is the choice of the desired maximum cluster sparsity values E_j, that can be made on the knowledge of the physical model on which clustering is applied; in the case of combustion chemistry, only two distinct values, one for temperature and one for species mass fractions are needed. Their detailed analysis is anyway discussed in the Results section.

Some other important properties of the grid-like cluster center initialisation, which will be recalled in the following, are:

- Each (active or inactive) cluster center is assigned a unique index value, defined as follows:

$$n_i = \sum_{j=1}^{d} \left(\prod_{k=1}^{j-1} s_k \right) \cdot \text{int}\left(\frac{\xi_{j,i}}{\varepsilon_j} \right), \qquad i = 1, \ldots, k; \tag{10}$$

- Each point image is surrounded by a 'bounding box' whose vertexes are all active cluster centers at the initialisation;

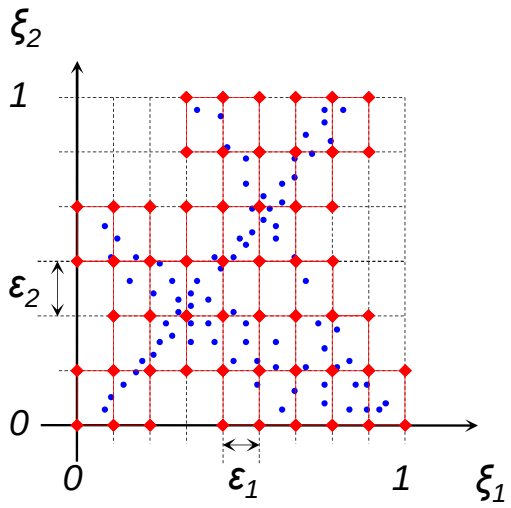

Figure 2: Cluster initialisation in two dimensions for a sample points set (blue dots); Dimensional span $s_1 = 10$, $s_2 = 7$. Potential number of cluster centers: 70; initialised cluster centers (red diamonds): $k = 54$.

- The number of bounding cluster centers b to each point image only depends on the grid dimensionality:
$$b = 2^d; \tag{11}$$
- Each active bounding box contains at least one point image;
- The final number of clusters after completion of the clustering algorithm can be lower than the number of initialised centers, as some of them may result to be be empty.

2.3. Bounding-box-constrained k-means

The two major computational inefficiencies related to crisp clustering algorithms, such as the k-means, in high-dimensional spaces, are the need to perform, at each iteration, all the possible point-to-cluster distance evaluations, of the order $O(kp)$, and the computational effort of the distance evaluation itself, that is of the order of the points dimensionality, $O(d)$. There have been efforts in the literature targeting to the reduction of the number of point-to-cluster evaluations [46, 47]; these approaches basically exploit distance metrics properties such as the triangle inequality to limit the number of distance evaluations from each point, only to the restricted set of clusters that are near enough to it; these approaches however require additional memory, whose storage and retrieval can be computationally less efficient than the standard full approach at low- and mid-size problem dimensions, e.g. $d \leq 15$, due to scattered access to memory areas.

Starting from the grid-like cluster center initialisation idea, a bounding-box constrained variant to the k-means algorithm is thus proposed, that has been tailored for the clustering problem in multidimensional simulations with detailed chemistry solution, but that is of general usage, and can profitably be tested also for different clustering problems. In particular, the algorithm exploits the idea that, if the cluster centers have been uniformly initialised across the zones of the space unity hyperbox that are covered with point images, each of them will lay in the surroundings of its initialisation value even after the end of the iterative clustering process. The core feature that makes the developed 'bounding-box-constrained' procedure differ from the standard k-means algorithm is thus that each image can be assigned to its surrounding cluster centers only, i.e. those which define the image's grid-like bounding box vertexes at the initialisation. Seen from the cluster center perspective, this assumption means that not all the images can belong to each center, but only those which lay in the grid boxes surrounding the cluster center at the initialisation.

The modified, bounding-box-constrained k-means algorithm developed is executed according to the main steps summarized in Algorithm 1; further optimization issues have been addressed in the Fortran implementation concerning the code's execution and the need to manage formation of empty clusters, but they are beyond the scope of the present paper. The main computational advantage allowed by the current algorithm is its reduced computational need, which is of the order $O(2^d p)$, where the number of bounding cluster centers per point, 2^d, is significantly lower than the overall number of active clusters k in practical computations.

In order to stress the validity of the developed algorithm, its performance upon a standard, two-dimensional test case retrieved from [48] is presented. In particular, the testcase adopted consists of 5,000 two-dimensional points belonging to 15 Gaussian clusters scattered at a medium degree of overlapping, and two different initialisation scenarios have been considered: a first grid-like initialisation made up of 20 total cluster centers, with spans $s_1 = 5$ and $s_2 = 4$; and a second one made up of 100 total cluster centers, where $s_1 = s_2 = 10$. The implementation of Algorithm 1 has been compared with a reference k-means implementation, and the same required number of clusters; the results are reported in Figure 3 and 4. More in detail, the first example, which considers a desired number of clusters of the same order of the real cluster layout, shows that pretty different partitions are obtained using the two algorithms: the standard k-means yields a higher number of cluster centers in the central region of the domain, while in the external region it happens that two clearly distinct clusters are assigned to a unique cluster in the partition. On the other hand, the clustered partition obtained through the bouding-box-constrained (BBC) k-means shows a slightly more even distribution, thanks to the fact that the cluster centers positions are constrained to remain approximately in the same region as the one they were initialised into. As a possible drawback of this procedure, the limited mobility of the cluster centers to their surrounding zone has created a split cluster in the lower right part of the domain. The overall algorithm performance appears however to guarantee a much more balanced

Algorithm 1 Bounding-box-constrained k-means

Require: $\boldsymbol{c}_j$, $j = 1, \ldots, k$ {initialised cluster centers}
Require: cluster(ip) $ip = 1, \ldots, p$ {partition at initialisation}
  {cluster(ip) = cluster index to which point ip belongs}
  {Carry on main k-means iteration}
  $it \leftarrow 0$
  repeat
    $it \leftarrow it + 1$ {Iteration counter}
    $swap \leftarrow 0$ {Points changing cluster at current iteration}
    for $ip = 1$ **to** p **do**
      $ci \leftarrow$ cluster(ip)
      {n_{ci} = cluster population of cluster ci}
      if $n_{ci} > 1$ **then**
        for $ib = 1$ **to** b **do**
          {bbox(i, j) = indexes of bounding cluster centers to point j; $i = 1, \ldots, b$}
          $cj \leftarrow$ bbox(ib, ip)
          if $cj = ci$ **then**
            $dist(ib, ip) \leftarrow \frac{n_{cj}}{n_{cj}-1}\, d(\boldsymbol{x}_{ip}, \boldsymbol{c}_{cj})$
          else
            $dist(ib, ip) \leftarrow \frac{n_{cj}}{n_{cj}+1}\, d(\boldsymbol{x}_{ip}, \boldsymbol{c}_{cj})$
          end if
        end for
        $cj \leftarrow$ bbox$(\arg\min(dist(:, ip)), ip)$
        {swap point between centers ci and cj}
        {and update cluster centers}
        if $ci \neq cj$ **then**
          $\boldsymbol{c}_{ci} \leftarrow \frac{n_{ci}}{n_{ci}-1}\boldsymbol{c}_{ci} - \frac{1}{n_{ci}-1}\boldsymbol{x}_{ip}$
          $\boldsymbol{c}_{cj} \leftarrow \frac{n_{cj}}{n_{cj}+1}\boldsymbol{c}_{cj} - \frac{1}{n_{cj}+1}\boldsymbol{x}_{ip}$
          $n_{ci} \leftarrow n_{ci} - 1$
          $n_{cj} \leftarrow n_{cj} + 1$
          $swap \leftarrow swap + 1$
          cluster$(ip) \leftarrow cj$
        end if
      end if
    end for
  until $swap = 0$ **or** $it = it_{\max}$

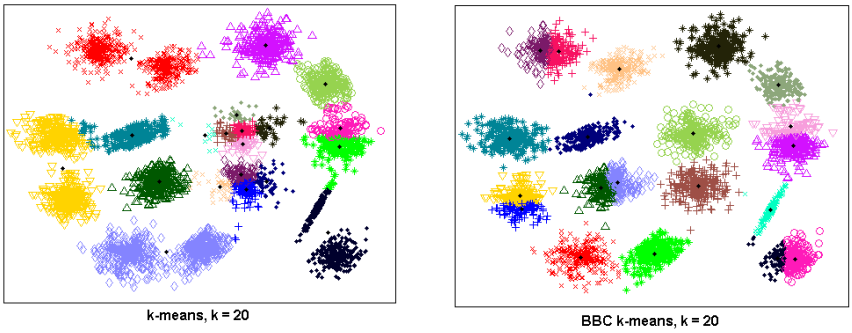

Figure 3: Comparison between standard and bounding-box-constrained (BBC) k-means at $k = 20$. Cluster centers: black dots; points: various symbols. Dataset from [48].

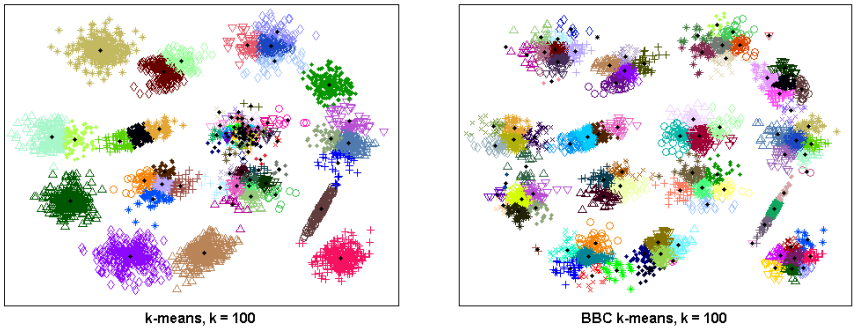

Figure 4: Comparison between standard and bounding-box-constrained (BBC) k-means at $k = 100$. Cluster centers: black dots; points: various symbols. Dataset from [48].

cluster distribution. This latest aspect is of particular importance in clustering chemically reacting cells in CFD, where the average cluster dimension can be as low as 3-4 cells (i.e., a 3 to 4 times reduction in the number of chemistry ODE systems to be integrated at each timestep), and where the strongly non-linear behaviour of the reaction mechanism requires maximum cluster homogeneity. Figure 4 points out a consistent behaviour of the two algorithms with respect to the previous observations; in this case, where the desired number of clusters is much higher than the optimum value, it can be observed that the BBC k-means subdivides the 15 real cluster regions into 4 to 7 clusters each; the standard k-means algorithm, instead, although catching correctly all of the real clusters, shows a deep concentration of cluster centers in the central region of the domain, thus leading to a pretty unbalanced allocation of the cluster partitions.

2.4. Cluster data remapping

The last stage of the procedure is remapping of the integrated variations in species mass fractions at the cluster level, back to the original fluid cells. As pointed out in the contributions by Babajimopoulos et al. [35], and by Liang et al. [36], a simple, weighted remapping of the species mass fractions changes from each cluster to its owned cells would gradually deteriorate the solution, through species mixing. The backward remapping procedure by Liang et al. [36] has thus been adopted, which has been shown to well suit crisp, chemistry-based cell clustering in multidimensional domains, and that complies with needs for species non-negativity and mass conservation.
Considered the species mass fraction variation at the cluster level, due to the integration of the chemistry ODE system for the time interval Δt, we have, for cluster c:

$$\Delta Y_{j,c} = Y_{j,c}(t + \Delta t) - Y_{j,c}(t), \qquad j = 1, \ldots, n_s; \tag{12}$$

then, for each cell i belonging to that cluster, the variation in species densities $\rho_{j,i}$ at the cell level after the integration of the chemistry ODE system at the cluster level is computed as:

$$\rho_{j,i}\,(t + \Delta t) = \rho_{j,i}\,(t) + \begin{cases} \Delta Y_{j,c}\,\rho_i, & \text{if } \Delta Y_{j,c} \geq 0; \\ \Delta Y_{j,c}\,\rho_c\,\rho_{j,i}/\rho_{j,c}, & \text{if } \Delta Y_{j,c} < 0. \end{cases} \qquad j = 1, \ldots, n_s \tag{13}$$

Finally, the cells' specific internal energies can instead be updated directly on the basis of the updated species mass fractions, exploiting the species' heats of formation values $h_j^0, j = 1, \ldots, n_s$:

$$U_i\,(t + \Delta t) = U_i\,(t) + \sum_{j=1}^{n_s} h_j^0\,[Y_j\,(t + \Delta t) - Y_j\,(t)]. \tag{14}$$

3. Results

3.1. Validation of the computational model

A modified version of the KIVA-4 code [40] has been used for this study. In particular, species thermodynamic properties for the fluid flow and detailed fuel combustion kinetics are modelled through a vectorized chemistry code presented in [49], and recently updated in order to provide fast simulation capabilities for large reaction mechanisms [34]. The code considers detailed gas-phase thermochemistry including reactions in Arrhenius form, third-body reactions with enhanced molecularities and pressure-dependent fall-off reactions in Lindemann's and Troe's forms. As far as fuel spray is concerned, the physical and thermal properties of diesel #2 in the KIVA database are used to model diesel fuel, while a dynamic model adopting the empirical correlation by Reitz and

Engine	FIAT Multijet 1.3L
Engine type	high-speed direct injection diesel
Valves per cylinder	4
Bore × stroke [mm]	69.6 × 82.0
Connecting rod length [mm]	131.3
Squish height [mm]	0.59
Compression Ratio	17.6:1
Cylinder displacement [l]	0.624
Intake valve closure	147 °BTDC
Exhaust valve opening	112 °ATDC

Table 1: Engine specifications for FIAT 1.3l high speed, direct injection engine.

Bracco [50], as applied in [51] for high pressure diesel sprays, has been implemented to model the effects of instantaneous operating conditions and injector nozzle geometry on the initial spray angle. The skeletal reaction mechanism for n-heptane fuel chemistry, made of 29 species and 56 reactions [52], and developed for multidimensional homogeneous-charge compression ignition engine simulations has been chosen, as the relative computational savings allowed by the proposed method are mechanism-independent. The baseline computational model solves the system of ordinary differential equations for combustion chemistry in each cell of the computational grid, where the local temperature value is greater than $600K$; the global timestep advancement is ruled by the KIVA-4 stability and convergence constraints [40], where the chemistry limiting value is ruled by the maximum mass-specific increase in the cell's internal energy. An overall maximum timestep $\Delta t_{\max} = 1.0e-05s$ has been set; the choice of such a pretty high value has been made in order to avoid the performance of the clustering algorithm from being altered, by artificially increasing the total number of simulation timesteps where chemistry has to be solved.

A high speed, direct injected diesel engine of current production has been modelled, whose specifications are reported in Table 1, featuring a common-rail injection system operating at $1600bar$ maximum pressure and capable of multiple injections (cfr. Table 2). Table 3 reports instead the details of the three engine operating conditions considered, which involve different combustion behaviours: at high engine speed and maximum load, featuring a unique injection pulse, down to maximum torque engine speed and full load, featuring three injection pulses starting from 50 crank angle degrees before TDC. Finally, the engine geometry has been modelled as a 60-degrees sector mesh, made of 24780 cells at BDC, as represented in Figure 5. The grid has an average spatial resolution of about $1.1mm$, and includes modelling of the crevice volume between piston and cylinder liner; the KIVA grid movement routine has been modified in order to guarantee a minimum of three cell layers at the walls, for better fitting of the thermal and velocity boundary layers.

Injection system	common-rail
Injector type	electronically controlled
Max. Injection pressure [MPa]	160
Number of nozzle holes	6
Nozzle hole diameter [mm]	0.121
Injection angle	15°

Table 2: Fuel injection system specifications.

	Case 1	Case 2	Case 3
Rotating speed [rpm]	1500	3000	4000
Number of injection pulses	3	2	1
start of main injection [°BTDC]	0.5	7.0	21.5
start of pre- injection [°BTDC]	50.0	–	–
start of pilot injection [°BTDC]	5.0	21.8	–
injected fuel mass [mg]	37.1	36.4	35.1
injection pressure [MPa]	800	1200	1600

Table 3: Engine validated operating parameters.

Validation of the engine model is presented in Figures 6, 7 and 8 in comparison with experimental data, in terms of average in-cylinder pressure and temperature traces [53], and where instantaneous apparent heat release rates have been reconstructed using Rassweiler and Withrow's method [54]. The plots show a very good agreement of the predicted traces with the experimental measurements, and in particular the detailed chemistry capability proves to yield correct predictions of the low temperature combustion region due to the early injection stage in case 1, and of the premixed combustion heat release peak in case 2, at maximum engine speed, as well as ignition delay timings in both validation cases.

3.2. Clustering procedure validation and setup

As mentioned in the previous paragraphs, the bounding-box-constrained algorithm developed for chemistry clustering proceeds unsupervised, by automatically determining the optimal number of clusters, and the initial cluster partition on the basis of a grid-like subdivision of the normalized images domain. The only parameters that rule over the process, and that need to be set prior to executing the algorithm, are the desired sparsity values $E_j, j = 1, \ldots, d$ which are needed to define the overall span over each dimension, as in Equation 9; and the choice of the subset $\mathbb{S}$ containing the selected species for clustering. Since the independent variables in the chemistry clustering problem are represented by the reactor temperature and species mass fractions of Eq. 3, only two sparsity constraints can however be defined, namely E_T and E_Y; as far as the species subset is concerned, in the following analysis a subset made of 9 species has been considered as a reference, as reported in Table 4. In particular, the species that

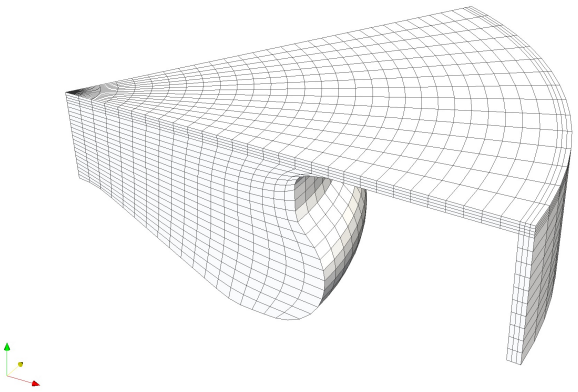

Figure 5: 60-degree sector computational grid for the 1.3l engine at top dead centre.

Subset	Species
$\mathbb{S}_9$	$C_7H_{16}, O_2, OH, O, HO_2, N_2, CO, CO_2, H_2O$
$\mathbb{S}_7$	$C_7H_{16}, O_2, OH, HO_2, CO, CO_2, H_2O$
$\mathbb{S}_5$	$C_7H_{16}, O_2, HO_2, CO_2, H_2O$
$\mathbb{S}_4$	$C_7H_{16}, O_2, CO_2, H_2O$

Table 4: Species subsets chosen for the clustering algorithm performance analysis.

rule over combustion timing and heat release have been chosen for the reference subset $\mathbb{S}_9$, and some more reduced sets have been derived from that by removing the species which typically show smaller concentrations during combustion. Fuel, oxidizer and main combustion products are instead always present.
Finally, in Table 5 some problem-independent parameters have been reported, which prevent the algorithm from exceeding reasonable memory requirements, or by stalling in infinite loops – for example, in order to avoid that one last image steps back and forth between two adjacent cluster centers, the algorithm is stopped after a maximum number of iterations or if the number of swapping images is lower than a specified threshold, measured as the ratio between the swapped images at the last iteration and the total number of active images.

Parameter	Value
Max number of iterations	50
Minimum mass fraction for species activation	1.0e-03
Maximum span along species dims.	8
Maximum span along Temperature dim.	500
Minimum clusters-to-cells ratio	1.0e-03

Table 5: Problem-independent parameters ruling over the bounding-box-constrained algorithm runtime.

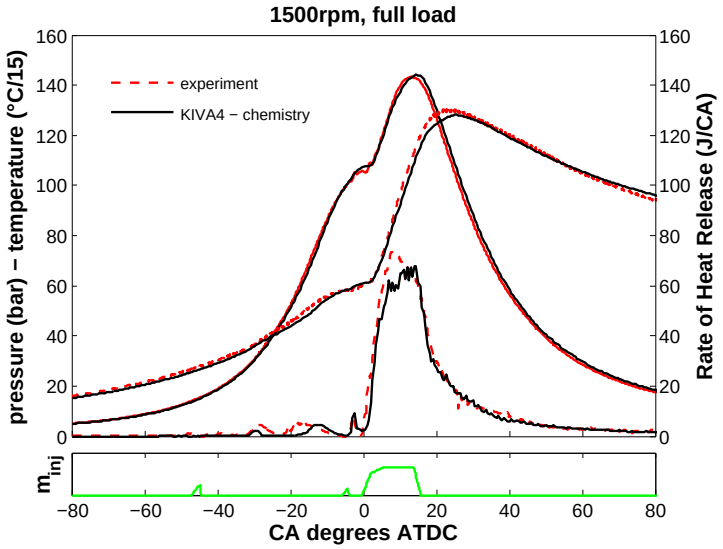

Figure 6: Model validation for KIVA4-chemistry code with ERC reaction mechanism [52], operating case 1.

3.2.1. Performance of the BBC k-means algorithm

As a first point which shows the performance of the proposed clustering procedure, Figures 9, 10 and 11 compare the performances of the BBC k-means algorithm with the standard k-means implementation, for a reference clustering setup which considers subset $\mathbb{S}_5$, and where ε_T fits a temperature sparsity of $E_T = 20K$ and ε_Y a species sparsity equal to $E_Y = 0.005$. In each of the three cases, both clustering methods have shown an excellent agreement in term of predicted average in-cylinder properties in comparison with the baseline case which performs full chemistry computation; the most significant point with respect to the overall CPU times is that the BBC k-means introduced almost negligible computational overhead in comparison to standard k-means, where instead the CPU time needed for clustering significantly exceeds that needed by the integration of the chemistry ODE systems of the CFD cells. This leads to the observation that the standard k-means approach appears to be unsuitable for high-dimensional clustering in CFD. Also, the CPU time due to chemistry when adopting the standard k-means is higher compared to the BBC k-means approach, due to the possibility of the cluster centers to freely redistribute across the image domain. The bounding-box approach instead constrains the cluster centers to their original initialisation region, and the clusters initialised at the borders of the images 'cloud' of Figure 2 are more likely to remain empty.

A more accurate look at the local, in-cylinder distribution of scalars has been reported in Figure 12, where the local values of temperature and mass fractions

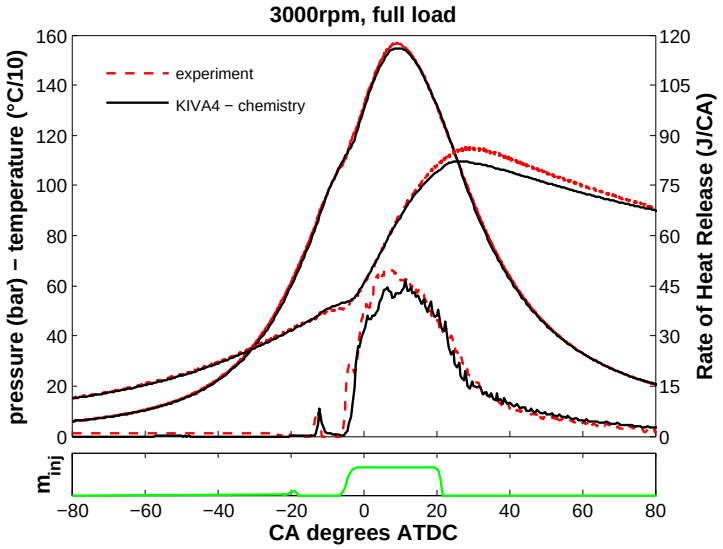

Figure 7: Model validation for KIVA4-chemistry code with ERC reaction mechanism [52], operating case 2.

of fuel, CO_2 and HO_2 are plotted for case 1 at a vertical cross-section plane intersecting the injection axis. The shots have been taken at CA = 8.0 degrees after TDC, at the peak of heat release and average cycle in-cylinder pressure, and compare the full chemistry solution with that deriving from the BBC k-means algorithm of Figure 9. A generally very good agreement between the predicted local values of the computation with cell clustering is observed for all the quantities, ant it appears to be independent on the combustion chamber zones, including the squish region and the piston bowl walls where massive wall impingement occurs, as well as on the possible presence of liquid fuel as in the inner spray jet core. The trend that shows good local agreement between the full-chemistry and the clustered solutions is also confirmed by predicted NO_x emissions in Figure 13. Here, the NO_x formation sub-mechanism was modeled by 5 species and 12 reactions, extracted from GRI-mech 3.0 [55]; none of the NO_x species was included in the clustering algorithm's species subset. The results confirm that the very good match of the clustered simulation versus full chemistry is present on a local basis, not only in terms of main thermodynamic properties, but also for individual species concentrations.

Finally, Figure 14 shows the detailed computational time requirements of the BBC k-means algorithm steps in the validation case 1. From the plot, it's clear that the most demanding tasks of the clustering procedure are the execution of the BBC k-means and the integration of the clusters' chemistry ODE systems; the cell grouping and integration remapping phases appear to be always less demanding than any other phase by almost two orders of magnitude. The plot

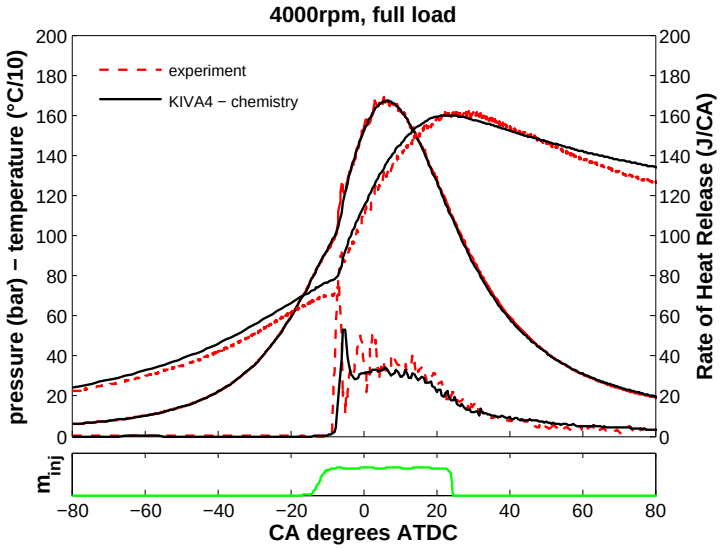

Figure 8: Model validation for KIVA4-chemistry code with ERC reaction mechanism [52], operating case 3.

also points out that, approximately from top dead center, and almost until 20 degrees ATDC, i.e. at the peak of global heat release, higher temperature and species stratification within the cylinder leads to a higher number of clusters. Here, chemistry takes more than 90% of the total procedure requirements.

3.2.2. Sensitivity to the method parameters

An analysis of the proposed clustering procedure has been run in order to assess the algorithm sensitivity to parameter choice, and to determine an optimal configuration for engine combustion chemistry applications, intended as the better tradeoff between computational efficiency, while still maintaining full accuracy of the simulation results. In particular, the influence of temperature grid discretization has been studied, considering values ranging from $E_T = 5K$ to $E_T = 100K$; species dimensional spans s_Y have been tested, ranging from $s_Y = 3$ up to $s_Y = 6$; finally, the adoption of the four proposed species subsets of Table 4 has been analysed. For all of the analyses, the measure of the simulation accuracy has been computed with an error function, that has been developed in order to quantify the deviations from the full chemistry case by computing the numerical average relative squared discrepancies in terms of average in-cylinder pressure and temperature, at a $\Delta\theta = 1°CA$ pace:

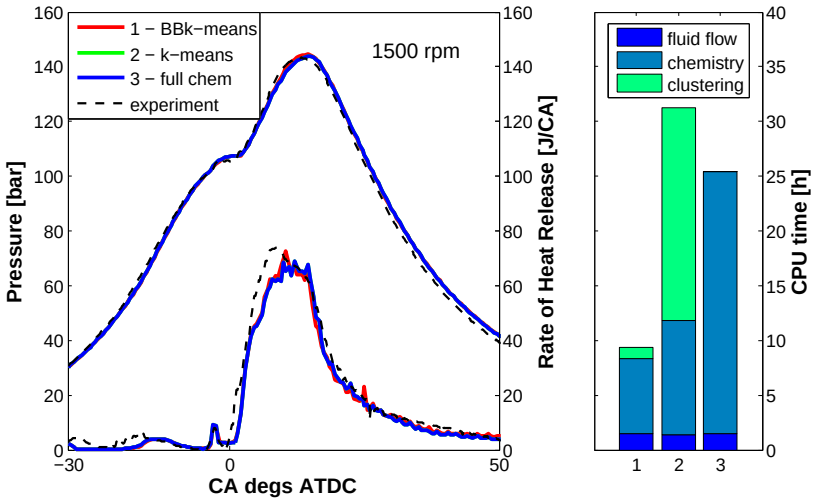

Figure 9: Comparison between standard and bounding-box-constrained k-means cell clustering, operating case 1. Left: in-cylinder pressure and heat release traces; right: cumulative CPU times for solving fluid flow, chemistry ODE system, and cell clustering.

$$e = \frac{1}{\theta_{IVC} - \theta_{EVO}} \left[\int_{\theta_{IVC}}^{\theta_{EVO}} \frac{\left| p^{full}(\theta) - p^{clu}(\theta) \right|}{p^{full}(\theta)} d\theta + \right. \\ \left. + \int_{\theta_{IVC}}^{\theta_{EVO}} \frac{\left| T^{full}(\theta) - T^{clu}(\theta) \right|}{T^{full}(\theta)} d\theta \right]; \tag{15}$$

in the formulation, the $full$ superscript indicates the reference run with full chemistry solution in each computational cell, and clu indicates the engine simulation with cell-clustered chemistry. The results of the analysis are summarized in Figure 15, in terms of error function value and overall CPU time of the simulations. As expected, an increase the dimensions of the chosen species subset leads to a general increase in overall accuracy, at a higher computational cost that can increase by two or three times, and with a slightly less than logarithmic trend with the dimensions of the subset; at the simplest subset choices, i.e. $\mathbb{S}_4$ and $\mathbb{S}_5$, however, no significant differences can be observed, mainly due to the uttermost influence of the major species. As far as the dependency on temperature accuracy is concerned, the plot shows that reducing the temperature grid width significantly increases the overall simulation time only at values of $E_T < 20K$, while at the highest values of E_T the CPU time saving does not appear to be worth the corresponding increase in simulation error. A different behaviour of the clustering procedure has instead been observed with respect to a change in species span, s_Y: all of the simulations were completed in similar amounts

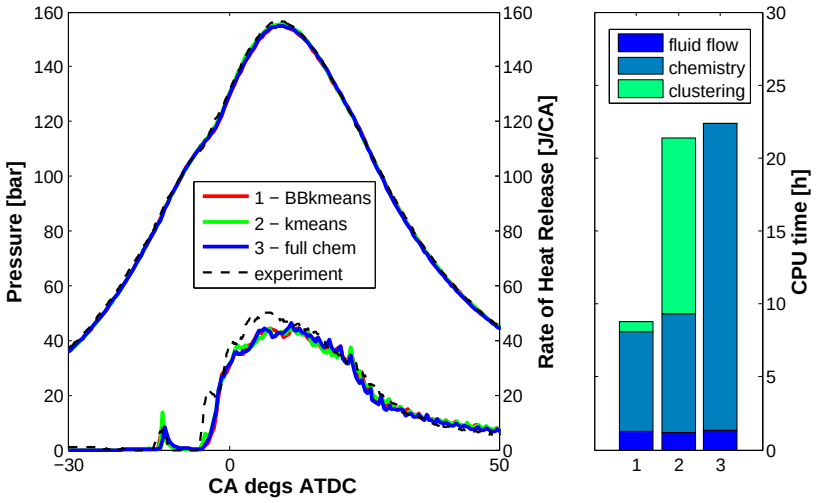

Figure 10: Comparison between standard and bounding-box-constrained k-means cell clustering, operating case 2. Left: in-cylinder pressure and heat release traces; right: cumulative CPU times for solving fluid flow, chemistry ODE system, and cell clustering.

of time, with small or even negligible improvements in terms of accuracy at increasing values of the grid span along the species dimensions. This phenomenon appears to confirm that the high-dimensional representation implies very sparse data arrangement within the image space, where the temperature dimension accuracy rules over the whole clustering process, while initialising more cluster centers along the species dimensions often has them be in empty zones of the domain.

Finally, in Figure 16 the impact of different clustering parameters on the simulations of operating case 1 has been plotted in terms of in-cylinder species mass fractions of some important species; in all of the plots, it appears that the most important deviations from the reference full chemistry simulation occur towards the end of the simulation and at the more reactive species, such as OH and HO_2; the general trend is however that a very good agreement with the reference solution is respected by all of the cases with cell clustering. Overall, the optimal configuration for the clustering algorithm has been chosen as the one at the 'elbow' of the CPU times curve, i.e. at the values where a further increase in clustering accuracy leads to a significant increase in computational needs that is not justified by a similar increase in simulation accuracy: $E_T = 20K$, $s_Y = 4$, $\mathbb{S} = \mathbb{S}_5$.

4. Concluding remarks

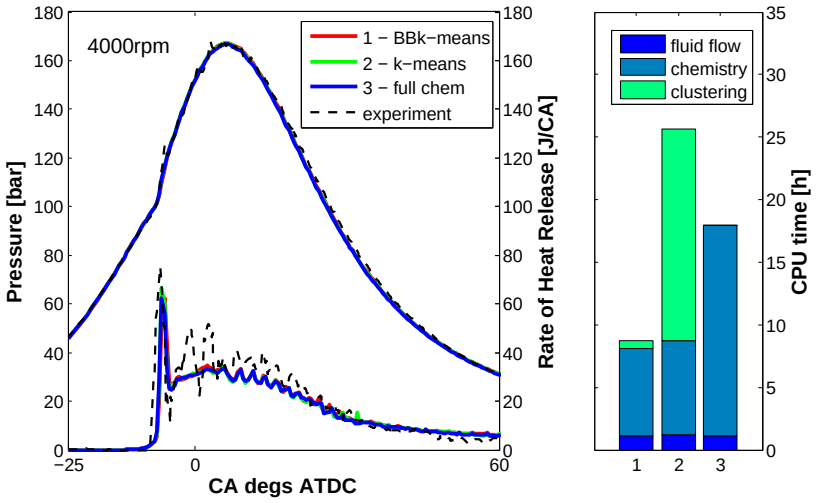

Figure 11: Comparison between standard and bounding-box-constrained k-means cell clustering, operating case 3. Left: in-cylinder pressure and heat release traces; right: cumulative CPU times for solving fluid flow, chemistry ODE system, and cell clustering.

A novel approach for clustering chemically reacting cells in multidimensional CFD simulations has been presented. The approach relies on a high-dimensional representation of the chemical state space. An unsupervised initial cluster partitioning procedure initialises the distribution of cluster centers in the variables space; then, a bounding-box-constrained (BBC) k-means algorithm pursues data clustering, with minimum computational effort even at high dimensionality, and preserving cluster center sparsity. The ODE system describing fuel chemistry is integrated in every cell-averaged cluster; the results of the integration are then mapped back to each cell according to the methodoloty proposed by Babajimopoulos et al. [35] and by Liang et al. [36]. The procedure has then been validated by modelling a small, high speed direct injected diesel engine, at three operating conditions which operate different combustion modes. Sensitivity analyses have also been carried out in order to assess the algorithm validity at different desired clustering parameters, including optimal temperature and species sparsity, and selected species subset. Overall, the following conclusions are drawn:

- a bounding-box-constrained k-means algorithm proved to be suitable to cluster high-dimensional chemistry datasets, whose behaviour is strongly non-linear, as the final cluster centers maintain a desirable sparsity pattern within the variables' space, avoiding excessive mixing;
- the unsupervised grid-like cluster center initialisation contributes to convergence of the BBC k-means algorithm, which execution required significantly lower computational times than the standard k-means approach;

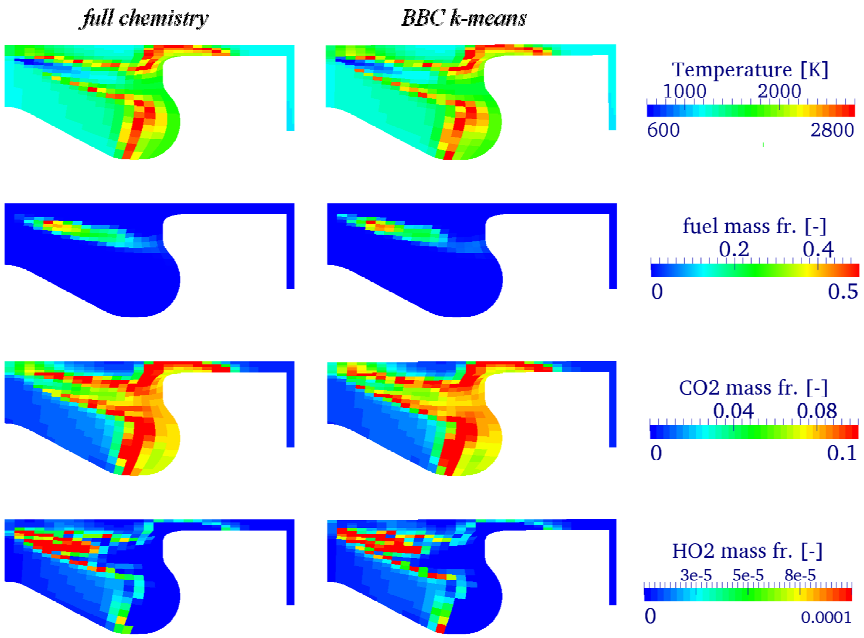

Figure 12: Comparison between predicted in-cylinder local temperature values [K] and some important species mass fractions at operating case 1, engine speed $1500rpm$, $CA = 8.0$ degrees ATDC. Full chemistry computation (left) vs. BBC k-means-based clustering (right).

- the optimal clustering configuration was ruled by temperature requirements, at grid values around $E_T = 15 - 20K$;
- a subset of the most important combustion reactants and products, $\mathbb{S} = \{C_7H_{16}, O_2, HO_2, CO_2, H_2O\}$, with an initial span of $s_Y = 4$ centers along each dimension was found to be the optimal trade-off between accuracy and computing time;
- simulated NO_x emissions showed a very good agreement of the clustered simulation with both the full-chemistry solution and the experimental datum, even if none of the NO_x species was included in the clustering algorithm's species subset. This suggests that the agreement is good also cell-by-cell on a local basis, which is fundamental for correctly predicting overall engine-out emissions;
- the limited computational requirements of the proposed procedure make it suitable for large scale computations on distributed memory systems, where it can be run on each node to speed up the solution of combustion chemistry, when a significant number of cells has to be solved.

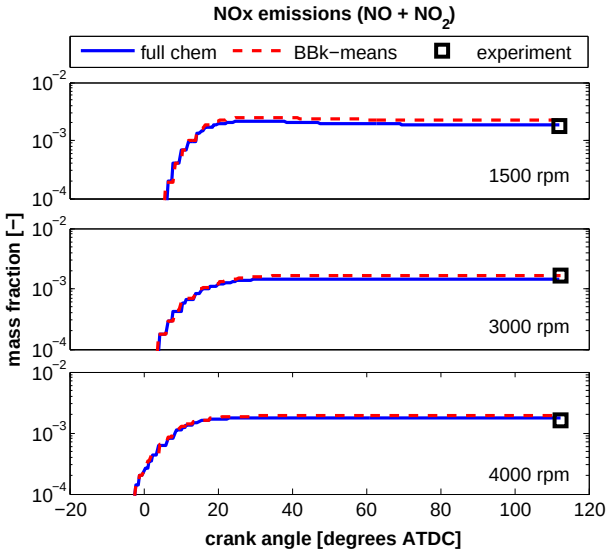

Figure 13: Comparison between predicted and experimental NO_x ($NO + NO_2$) in-cylinder mass fractions using either full chemistry or BBC k-means clustering, for the three operating cases considered.

References

[1] C. K. Law, Combustion at a crossroads: Status and prospects, Proceedings of the Combustion Institute 31 (2007) 1 – 29.

[2] R. Hilbert, F. Tap, H. El-Rabii, D. Thvenin, Impact of detailed chemistry and transport models on turbulent combustion simulations, Progress in Energy and Combustion Science 30 (2004) 61 – 117.

[3] S.-C. Kong, R. D. Reitz, Application of detailed chemistry and cfd for predicting direct injection hcci engine combustion and emissions, Proceedings of the Combustion Institute 29 (2002) 663 – 669.

[4] S.-C. Kong, R. D. Reitz, Use of detailed chemical kinetics to study hcci engine combustion with consideration of turbulent mixing effects, Journal of Engineering for Gas Turbines and Power 124 (2002) 702–707.

[5] S. L. Kokjohn, R. M. Hanson, D. A. Splitter, R. D. Reitz, Fuel reactivity controlled compression ignition (rcci): a pathway to controlled high-efficiency clean combustion, International Journal of Engine Research 12 (2011) 209–226.

[6] S. Kokjohn, R. Hanson, D. Splitter, J. Kaddatz, R. Reitz, Fuel reactivity controlled compression ignition (rcci) combustion in light- and heavy-duty engines, SAE International Journal of Engines 4 (2011) 360–374.

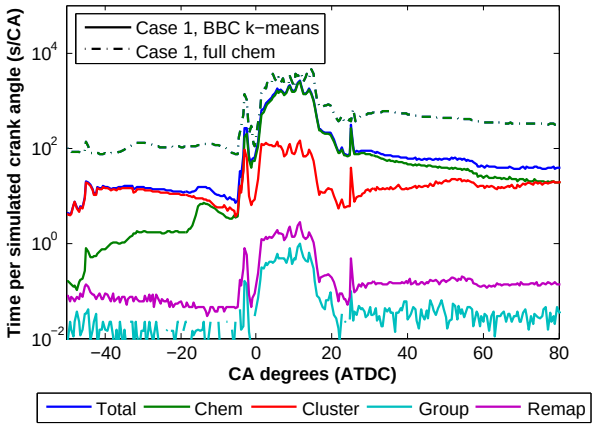

Figure 14: Details of clustering-related computational times per simulated crank angle, case 1. Total times and chemistry times coincide in the simulation with full chemistry.

[7] T. Lu, C. K. Law, Toward accommodating realistic fuel chemistry in large-scale computations, Progress in Energy and Combustion Science 35 (2009) 192 – 215.

[8] O. Herbinet, W. J. Pitz, C. K. Westbrook, Detailed chemical kinetic oxidation mechanism for a biodiesel surrogate, Combustion and Flame 154 (2008) 507 – 528.

[9] O. Herbinet, W. J. Pitz, C. K. Westbrook, Detailed chemical kinetic mechanism for the oxidation of biodiesel fuels blend surrogate, Combustion and Flame 157 (2010) 893 – 908.

[10] H. Curran, P. Gaffuri, W. Pitz, C. Westbrook, A comprehensive modeling study of n-heptane oxidation, Combustion and Flame 114 (1998) 149 – 177.

[11] H. Curran, P. Gaffuri, W. Pitz, C. Westbrook, A comprehensive modeling study of iso-octane oxidation, Combustion and Flame 129 (2002) 253 – 280.

[12] C. K. Westbrook, W. J. Pitz, O. Herbinet, H. J. Curran, 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 156 (2009) 181 – 199.

[13] C. Westbrook, W. Pitz, P. Westmoreland, F. Dryer, M. Chaos, P. Osswald, K. Kohse-Hinghaus, T. Cool, J. Wang, B. Yang, N. Hansen, T. Kasper, A detailed chemical kinetic reaction mechanism for oxidation of four small alkyl esters in laminar premixed flames, Proceedings of the Combustion Institute 32 (2009) 221 – 228.

[14] T. Lu, C. K. Law, A directed relation graph method for mechanism reduction, Proceedings of the Combustion Institute 30 (2005) 1333 – 1341.

[15] T. Lu, C. K. Law, On the applicability of directed relation graphs to the reduction of reaction mechanisms, Combustion and Flame 146 (2006) 472 – 483.

[16] T. Lu, C. K. Law, Strategies for mechanism reduction for large hydrocarbons: n-heptane, Combustion and Flame 154 (2008) 153 – 163.

[17] K. E. Niemeyer, C.-J. Sung, On the importance of graph search algorithms for drgep-based mechanism reduction methods, Combustion and Flame 158 (2011) 1439 – 1443.

[18] F. Perini, J. L. Brakora, R. D. Reitz, G. Cantore, Development of reduced and optimized reaction mechanisms based on genetic algorithms and element flux analysis, Combustion and Flame 159 (2012) 103 – 119.

[19] S. Lam, D. Coussis, Understanding complex chemical kinetics with computational singular perturbation, Symposium (International) on Combustion 22 (1989) 931 – 941.

[20] A. Zagaris, H. Kaper, T. Kaper, Analysis of the computational singular perturbation reduction method for chemical kinetics, Journal of Nonlinear Science 14 (2004) 59–91. 10.1007/s00332-003-0582-9.

[21] U. Maas, S. Pope, Simplifying chemical kinetics: Intrinsic low-dimensional manifolds in composition space, Combustion and Flame 88 (1992) 239 – 264.

[22] U. Maas, S. Pope, Implementation of simplified chemical kinetics based on intrinsic low-dimensional manifolds, Symposium (International) on Combustion 24 (1992) 103 – 112.

[23] S. Pope, Computationally efficient implementation of combustion chemistry using in situ adaptive tabulation, Combustion Theory and Modelling 1 (1997) 41–63.

[24] B. Yang, S. Pope, Treating chemistry in combustion with detailed mechanismsin situ adaptive tabulation in principal directionspremixed combustion, Combustion and Flame 112 (1998) 85 – 112.

[25] Q. Tang, S. B. Pope, Implementation of combustion chemistry by in situ adaptive tabulation of rate-controlled constrained equilibrium manifolds, Proceedings of the Combustion Institute 29 (2002) 1411 – 1417.

[26] L. Lu, S. B. Pope, An improved algorithm for in situ adaptive tabulation, Journal of Computational Physics 228 (2009) 361 – 386.

[27] C. J. Aro, Chemsode: a stiff ode solver for the equations of chemical kinetics, Computer Physics Communications 97 (1996) 304 – 314.

[28] S.-L. Kim, J.-Y. Choi, I.-S. Jeung, Y.-H. Park, Application of approximate chemical jacobians for constant volume reaction and shock-induced combustion, Applied Numerical Mathematics 39 (2001) 87 – 104.

[29] D. A. Schwer, J. E. Tolsma, W. H. Green, P. I. Barton, On upgrading the numerics in combustion chemistry codes, Combustion and Flame 128 (2002) 270 – 291.

[30] V. Damian, A. Sandu, M. Damian, F. Potra, G. R. Carmichael, The kinetic preprocessor kpp-a software environment for solving chemical kinetics, Computers & Chemical Engineering 26 (2002) 1567 – 1579.

[31] A. Sandu, R. Sander, Simulating chemical systems in fortran90 and matlab with the kinetic preprocessor kpp-2.1, Atmospheric Chemistry and Physics 6 (2006) 187–195.

[32] F. Bisetti, Integration of large chemical kinetic mechanisms via exponential methods with krylov approximations to jacobian matrix functions, Combustion Theory and Modelling (in press) 1–32.

[33] F. Perini, G. Cantore, R. Reitz, An analysis on time scale separation for engine simulations with detailed chemistry, SAE Technical Paper 2011-24-0028.

[34] F. Perini, E. Galligani, R. Reitz, An analytical jacobian approach to sparse reaction kinetics for computationally efficient combustion modelling with large reaction mechanisms, Energy & Fuels 26 (2012) 4804–4822.

[35] A. Babajimopoulos, D. N. Assanis, D. L. Flowers, S. M. Aceves, R. P. Hessel, A fully coupled computational fluid dynamics and multi-zone model with detailed chemical kinetics for the simulation of premixed charge compression ignition engines, International Journal of Engine Research 6 (2005) 497–512.

[36] L. Liang, J. G. Stevens, J. T. Farrell, A dynamic multi-zone partitioning scheme for solving detailed chemical kinetics in reactive flow computations, Combustion Science and Technology 181 (2009) 1345–1371.

[37] H. Barths, C. Felsch, N. Peters, Mixing models for the two-way-coupling of cfd codes and zero-dimensional multi-zone codes to model hcci combustion, Combustion and Flame 156 (2009) 130 – 139.

[38] Y. Shi, R. P. Hessel, R. D. Reitz, An adaptive multi-grid chemistry (amc) model for efficient simulation of hcci and di engine combustion, Combustion Theory and Modelling 13 (2009) 83–104.

[39] G. M. Goldin, Z. Ren, S. Zahirovic, A cell agglomeration algorithm for accelerating detailed chemistry in cfd, Combustion Theory and Modelling 13 (2009) 721–739.

[40] D. J. Torres, M. F. Trujillo, Kiva-4: An unstructured ale code for compressible gas flow with sprays, Journal of Computational Physics 219 (2006) 943 – 975.

[41] J. A. Hartigan, M. A. Wong, Algorithm as 136: A k-means clustering algorithm, Journal of the Royal Statistical Society. Series C (Applied Statistics) 28 (1979) 100–108.

[42] J. C. Bezdek, R. Ehrlich, W. Full, Fcm: The fuzzy c-means clustering algorithm, Computers & Geosciences 10 (1984) 191 – 203.

[43] A. A. Amsden, P. O'Rourke, T. Butler, Kiva-2: A computer program for chemically reactive flows with sprays, NASA STI/Recon Technical Report N 89 (1989) 27975.

[44] A. Hinneburg, C. Aggarwal, D. A. Keim, What is the nearest neighbor in high dimensional spaces?, Morgan Kaufmann, 2000, pp. 506–515.

[45] C. Aggarwal, A. Hinneburg, D. Keim, On the surprising behavior of distance metrics in high dimensional space, in: J. Van den Bussche, V. Vianu (Eds.), Database Theory ICDT 2001, volume 1973 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 2001, pp. 420–434. 10.1007/3-540-44503-X-27.

[46] C. Elkan, Using the triangle inequality to accelerate -means (2007).

[47] G. Hamerly, Making k-means even faster, in: SIAM International Conference on Data Mining (SDM).

[48] University of eastern finland speech and image processing unit clustering datasets, 2011.

[49] F. Perini, Optimally reduced reaction mechanisms for Internal Combustion Engines running on biofuels, Ph.D. thesis, University of Modena and Reggio Emilia, 2011.

[50] R. Reitz, F. Bracco, On the dependence of spray angle and other spray parameters on nozzle design and operating conditions, SAE Technical paper 790494 (1979).

[51] F. Perini, E. Mattarelli, Development and calibration of an enhanced quasi-dimensional combustion model for hsdi diesel engines, International Journal of Engine Research 12 (2011) 311–335.

[52] A. Patel, S. Kong, R. Reitz, Development and validation of a reduced reaction mechanism for hcci engine simulations, SAE Technical Paper 2004-01-0558.

[53] V. I. Golovitchev, L. Montorsi, C. A. Rinaldini, A. Rosetti, Cfd combustion and emission formation modeling for a hsdi diesel engine using detailed chemistry, ASME Conference Proceedings 2006 (2006) 349–358.

[54] G. M. Rassweiler, L. Withrow, Motion pictures of engine flames correlated with pressure cards, SAE Technical paper 380139 (1938).

[55] G. P. Smith, D. M. Golden, M. Frenklach, N. W. Moriarty, B. Eiteneer, M. Goldenberg, C. T. Bowman, R. K. Hanson, S. Song, J. William C. Gardiner, V. V. Lissianski, Z. Qin, Gri-mech 3.0, accessed september 2012.

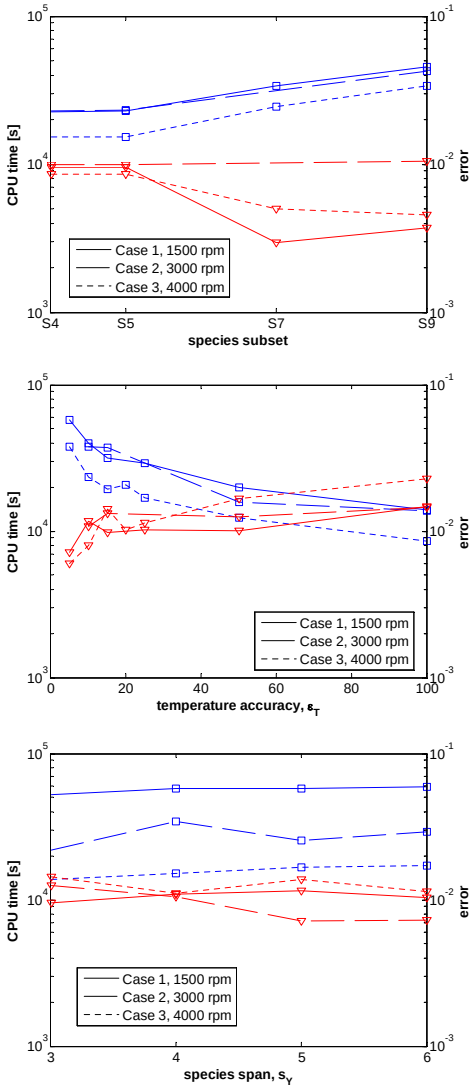

Figure 15: Sensitivity analysis of the clustering-remapping approach at the three operating cases of Table 3: species subset (top), temperature accuracy (center), species span (bottom). Red lines with triangle marks plot error values, Blue lines with square marks represent overall simulation times.

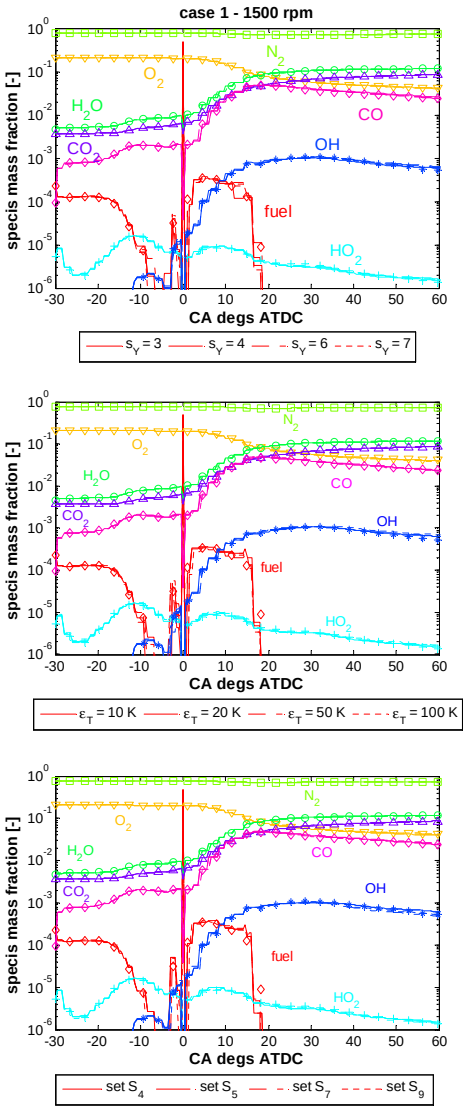

Figure 16: Predicted in-cylinder mass fractions of some important species at varying clustering parameters, for operating case 1: species span (top), temperature span (center), species subset (bottom). Full chemistry solution (symbols) vs. clustered-chemistry cases (lines).