Sikalo, Nejra; Hasemann, Olaf; Schulz, Christof; Kempf, Andreas; Wlokas, Irenäus:

A Genetic Algorithm-Based Method for the Automatic Reduction of Reaction Mechanisms

DOI: http://dx.doi.org/10.1002/kin.20826

URN: urn:nbn:de:hbz:464-20170504-114028-3

Link: https://duepublico.uni-duisburg-essen.de:443/servlets/DocumentServlet?id=42266

Source: International Journal of Chemical Kinetics, 46: 41–59 (2014); First published: 21 November 2013

A Genetic Algorithm-Based Method for the Automatic Reduction of Reaction Mechanisms

N. Sikalo*[, a], O. Hasemann[a], C. Schulz[b,c], A. Kempf[a,c,d] , I. Wlokas[a]

[a]IVG, Institute for Combustion and Gas Dynamics – Fluid Dynamics
[b]IVG, Institute for Combustion and Gas Dynamics – Reactive Fluids
[c]CENIDE, Center for Nanointegration Duisburg-Essen
[d]CCSS, Center for Computational Sciences and Simulation
University of Duisburg-Essen, 47048 Duisburg, Germany

*Corresponding author:
Irenäus Wlokas**,** University of Duisburg-Essen**,** 47048 Duisburg, Germany,
Phone: +49 203 379 8104**,** fax: +49 203 379 1953**,** email: irenaeus.wlokas@uni-due.de

Note: This document is an authors' version of the manuscript that was accepted for publication in International Journal of Chemical Kinetics. Changes made during the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Please cite the original publication as:

Sikalo, N., Hasemann, O., Schulz, C., Kempf, A. and Wlokas, I. (2014), *A Genetic Algorithm-Based Method for the Automatic Reduction of Reaction Mechanisms*. Int. J. Chem. Kinet., 46: 41–59. doi:10.1002/kin.20826

Abstract

An automatic method for the reduction of chemical kinetics mechanisms under specific physical or thermodynamic conditions is presented. The method relies on the genetic algorithm search logic to gradually reduce the number of reactions from the detailed mechanism while still preserving its ability to describe the overall chemistry at an acceptable error. Accuracy of the reduced mechanism is determined by comparing its solution to the solution obtained with the full mechanism under the same initial and/or physical conditions. However, not only the chemical accuracy and the size of the mechanism are considered, but also the time for its solution which helps to avoid stiff and slow converging mechanisms, thus preferring the fast solutions. The reduction method is demonstrated for a detailed mechanism for methane combustion, GRI-Mech

3.0, which was reduced from 325 reactions and 53 species to 58 reactions and 26 species, and for an iron pentacarbonyl mechanism by Wlokas *et al.* (2013), [1] originally consisting of 144 reactions and 34 species, which was reduced to 37 reactions and 24 species. The performance of the reduced mechanisms is shown for homogeneous constant-pressure reactors and for burner-stabilized flames. The results show a good agreement between reduced and full mechanisms for both the reactor and flame cases. The presented method is flexible and can be easily adjusted to either yield more accurate (but bigger) or smaller (but less accurate) reduced mechanisms, depending on the user's preference.

Keywords: Reaction Mechanisms, Genetic Algorithms, Reduction

1. Introduction

The simulation of combustion processes involves solving complex finite rate chemistry models consisting of large numbers of chemical reactions and species, often coupled with turbulence and complex geometry. Although modern computers and specialized software packages are capable to solve for large numbers of chemical reactions, even relatively simple practical problems are often too complex for an accurate numerical simulation. Detailed chemical kinetics descriptions of hydrocarbon oxidation may require the tracking of hundreds of chemical species and thousands of reaction steps [2, 3]. Although such mechanisms describe the overall chemical process very accurately, a large number of equations must be resolved. Under a specific range of physical conditions, reaction mechanisms can be reduced and still provide a reasonable accuracy defined by user-specified criteria.

There are several attractive approaches to the reduction of chemical reaction mechanisms; a comprehensive review on these methods is given by Tomlin *et al.* [4]. Some approaches are based on a time-scale analysis, such as the so-called quasi-steady-state assumption (QSSA) considered by Chen [5] and Montgomery *et al*. [6] which requires identifying species that react on a large time scale relative to other species. The QSS assumption consists of assuming that the net rate of production of a species is zero and is valid under combustion conditions for a large number of minor and intermediate species. The QSSA however requires some "chemist's intuition" although tools have been developed to aid with the use of this method. The intrinsic low-dimensional manifold method [7, 8] is based on a reaction trajectory quickly converging to a low-dimensional manifold that can be represented by a smaller number of species forming a reduced mechanism. The method of computational singular perturbation (CSP) is described by

Lam [9], Lu *et al*. [10], Valorani *et al.* [11] and it analyses the Jacobian matrix to distinguish between slow and fast reactions.

In order to identify and extract species that are kinetically important, methods as path flux analysis [12, 13] and sensitivity analysis [14], which evaluates the change in the species concentration due to small changes in the rate constants, are commonly used. A lumping method, which groups concentrations of species into a reduced species set, was proposed by Kuo and Wei [15]. More recently, an integer programming approach was proposed by Petzold and Zhu [16], Androulakis [17] and Bhattacharjee *et al.* [18] and applied to large-scale combustion models. The integer programming solution is defined as the smallest possible reduced mechanism consistent with the user-set tolerances, which retains the original set of species but contains a smaller set of reactions. However, single reduced mechanisms might not be accurate enough over the whole region of the combustion system and the local chemistry might not be well described. To address this problem, the adaptive chemistry approach was proposed by Schwer *et al*. [19] and Banerjee and Ierapetritou [20], which replaces the detailed mechanism with a number (library) of locally accurate reduced mechanisms adjusted for changing conditions within the flow.

Edwards *et al*. [21] used genetic algorithms to reduce reaction mechanisms taking into account both the difference between the species concentration profiles in a burner-stabilized premixed flame, for the detailed and the reduced mechanisms, and the size of the reduced mechanism. This methodology worked well for a small mechanism (18 reactions and 10 species) but the behavior of the reduction method was not investigated for larger mechanisms. Montgomery *et al*. [22] used a genetic algorithm to determine which species of a detailed mechanism fit into the QSSA assumption. This approach yielded plausible results for a methane combustion mechanism and

outperformed simple trial and error. A more recent work by Elliott *et al.* [23] used genetic algorithm in two steps; first to eliminate unimportant species from the full mechanism and then to optimize the reduced mechanism against experimental data. Calculations of full and reduced mechanisms were performed on a premixed burner-stabilized flame. The size of the desired subset of species to be extracted from the full mechanism was fixed and specified from the start. A methane combustion mechanism (GRI-Mech 3.0, [24]) was reduced and the rate coefficients were optimized against experimental results from Bernstein et al. [25] in a subsequent step. Later on, Elliott *et al*. [26] employed standard and hybrid genetic algorithms to reduce and optimize a reaction mechanism for the combustion of aviation fuel/air mixtures containing 338 reactions and 67 species. The reduced mechanism consists of 215 reactions and 50 species and was optimized using a hybrid real coded genetic algorithm that was improved with the Rosenbrock method [27]. Hernández *et al*. [28] used a genetic algorithm as a single method to reduce the number of reactions in the full mechanism; the mechanisms were evaluated with constant volume reactor simulations for a number of different sets of initial conditions. These conditions were chosen such that they have a strong effect on the autoignition process of a homogeneous charge compression ignition engine (HCCI). The key reaction paths that occur during a hydrocarbon oxidation were first identified by analyzing reaction rates, and then kept in each chromosome of the population. This approach was demonstrated on a methane oxidation mechanism that was reduced from 53 species and 350 reactions to 47 species and 57 reactions using a genetic algorithm as a single reduction method. Because of its large size, the Diesel fuel surrogate (n-heptane/toluene) oxidation mechanism consisting of 664 species and 2796 reactions has been reduced with a four-step methodology (species lumping, species sensitivity analysis,

reaction sensitivity analysis and at the end, the genetic algorithm method previously described) to 116 species and 252 reactions.

Genetic algorithms in kinetics studies, however, are mostly used for the optimization of rate coefficients in reduced mechanisms (obtained from another method), and not for the reduction itself [29, 30, 31, 32].

The relevance of the CPU time of the reduced mechanism has been considered in the work from Nagy and Turányi [33], which describes the error minimization-based reduction methods used for reducing very large mechanisms. This work describes the elimination of both the unimportant species using a Simulation Error Minimization Connectivity Method (SEM-CM) and the unimportant reactions using the Principal Component Analysis of Matrix F with Simulation Error Minimization (SEM-PCAF). The SEM-PCAF method yields several different reduced mechanisms which all have an error very close to the minimal error. As the accuracy criteria are equally satisfied by all these reduced mechanisms, the final choice is made based on their computational times. The mechanism with the smallest computational time is accepted as the best found reduced mechanism.

The present study focuses on the use of genetic algorithms for reduction purposes only, aiming at providing a simple, robust, automatic method that does not require a detailed understanding of chemistry from the user, while ensuring that resulting mechanism is not only small but also easy to solve.

The method uses an objective function that considers both the accuracy of the mechanism and the computational cost for its solution, leading to a compromise that can be matched to very specific requirements. The cost for the solution is considered through the number of reactions of the reduced mechanism and the time required for the solution, helping to avoid large or stiff

reaction mechanisms. This evolution towards mechanisms that can be solved at low cost has the added advantage that the mechanisms from later generations are typically easier to solve, so that the optimization algorithm itself will run faster.

2. Structure of the Genetic Algorithm

The concept of the genetic algorithm (GA) is derived from the mechanism of evolution and natural genetics [34], which are represented through computational operations. Many modifications of genetic algorithm operators exist and can enhance the performance for a given application. Genetic algorithms work particularly well for optimization problems that involve not only the accuracy of the solution but also finding it efficiently, which makes GAs very suitable for the reduction of reaction mechanisms. The genetic algorithm finds a near optimal solution to a problem, where "optimal" is defined by an evaluation function (also known as objective- or fitness function). Before starting the search, one must represent a potential solution in a way that can easily be processed by the genetic algorithm. In analogy to the natural genetics, such an encoded solution is called a chromosome. For the evaluation function calculation, the chromosomes created by the GA must be 'translated' into the shape that corresponds to the real problem that we are aiming to solve. Such representation is called phenome. In this case, the phenome is a reaction mechanism with its full kinetics information suitable for simulations of the reactive cases. Mechanisms only vary in presence or absence of reactions defined in the detailed mechanism. The chromosome only contains the information whether a given reaction is included or not. This is explained in greater detail in the following sections. Evaluation is performed on the phenome. All other genetic algorithm operations, such as initialization, selection, crossover and mutation, are performed on the chromosome.

The genetic algorithm search for the solution consists of the steps of:

(1) *Initialization*, where a set (population) of initial chromosomes is generated

(2) *Evaluation*, where the performance of each chromosome from the population is being evaluated and corresponding fitness is assigned to the each chromosome

(3) *Selection*, where the chromosomes (according to their fitness) are chosen from the current population as the basis for the next generation of chromosomes (children)

(4) *Crossover*, where the selected chromosomes exchange information (genes) to create children for the next generation

(5) *Mutation*, where a small number of the children's genes is randomly changed in order to increase diversity of the population

This new population undergoes the process of evaluation, selection, crossover and mutation until termination criteria are satisfied or a given number of generations has passed. However, when creating a new population using crossover and mutation, there is a possibility that the best chromosomes might be lost. Therefore, the highest scoring chromosome(s) from each generation is (are) copied unchanged to the next generation (elitist selection). The elitist selection preserves the best found solution from the current total history of generations. As soon as a better one is found, the old elitist solution is replaced by the newly found one. That way, the algorithm preserves the best solution found so far, but it continuously searches for a better one. This procedure is illustrated in Figure 1.

3. Implementation Details

This section explains the use of genetic algorithms for the elimination of reactions from the detailed reaction mechanism. We aim to reduce the size of a reaction mechanism, ensuring a good accuracy for a certain application and preventing the formation of excessive stiffness in the

resulting mechanism. A reaction mechanism is a list of all reactions that (can) occur during the combustion of a specific gas mixture (with detailed sets of data needed for evaluation of their thermodynamic and transport properties), and the reactions which occur among these species with all the coefficients required for the evaluation of the corresponding reaction rates.

The conversion rate from reactants to products in the elementary reactions is controlled by chemical reaction rates where the forward rate constant is assumed to depend on temperature according to the Arrhenius equation (Eq. (1)), where A_i is the pre-exponential factor, β_i is the temperature exponent and $E_{a,i}$ is the activation energy.

$$k_{\mathrm{fwd},i}(T) = A_i T^{\beta_i} \exp\left(\frac{-E_{a,i}}{RT}\right) \tag{1}$$

Large mechanisms can contain several hundreds of reacting species and thousands of reactions. In a detailed mechanism of N_R reactions, a smallest possible subset of N_r reactions exists that can reproduce the predictions of the detailed mechanism with a given user-defined accuracy. The genetic algorithm aims at finding this smallest mechanism (or one that is only slightly bigger) by satisfying accuracy and size criteria which are defined by an overall fitness function. The genetic algorithm seeks the minimum of this function, while gradually eliminating reactions from the mechanism. Elimination is triggered not only by the fitness criteria, but also through mutation operations adjusted to this problem. In the following sections, the reduction methodology is explained in greater detail.

3.1 Encoding and Initialization

A reaction mechanism is mapped to a chromosome by representing each reaction as a binary gene. The presence of a reaction is represented by 1, while 0 denotes that the reaction is absent. The length of each chromosome corresponds to the number of reactions in the full mechanism,

N_R, as formulated by Eq. (2), where α_i indicates whether the i^{th} reaction is present in the reduced mechanism.

$$\alpha = [\alpha_1, \ldots, \alpha_{N_R}] \in \{0,1\}^{N_R} \tag{2}$$

The first generation of chromosomes (initial population) is generated in a way that all the chromosomes in the initial population are equal and represent the original mechanism (all α_i = 1). This approach has been chosen due to a concern that, if the initial population was created randomly, key-reactions could be lost in the first step. Although there is no initial diversity among the chromosomes and the evolution effectively starts with small changes due to mutation in the second generation, the full-mechanism initialization is proven to be the safest starting point of the search for a reaction subset smaller than the full mechanism. Alternatively, the population can be initialized in a way that every initial chromosome just deactivates a single, randomly chosen reaction, such that

$$\sum_{i=1}^{N_R} \alpha_i = N_R - 1 \tag{3}$$

where the index i, for which $\alpha_i = 0$, is chosen randomly with a uniformly distributed probability (Eq. 4).

$$P[\alpha_i = 0] = \frac{1}{N_R} \tag{4}$$

This initialization introduces a small diversity among individuals, which avoids that a complete set of identical chromosomes is solved in the first step, whereas maintaining (almost) original chromosomes ensures that none of the original reactions gets lost. (The population diversity increases afterwards with the crossover and the mutation, which are discussed later.) Number of chromosomes in the population (population size) is user-defined. As an example, initial sets of

four chromosomes representing a eight-reaction mechanism, as according to the two ways of initialization described above are shown in Fig. 2.

After the population is created, the chromosomes are used to generate the corresponding phenomes (sub-mechanisms). These are incorporated into the calculation of the reactive case (e.g., a perfectly stirred reactor or a laminar flame) and solved. The solution is evaluated as described in the next section to provide information about how well the reduced mechanism performs compared to the full mechanism.

3.2 Evaluation

The evaluation function must consider the requirements one has for a reduced mechanism; it must therefore provide good tradeoff between the size of the reduced mechanism, its accuracy and its stiffness to minimize the computational effort associated with the resulting reduced mechanism. The evaluation function (or objective or fitness function) is the key part of the optimization technique and is usually the "trickiest" part to define. The evaluation function determines the difference between the results obtained from the original mechanism (taken as accurate) and the reduced mechanism. Because the size of the reduced mechanism and its stiffness is also relevant, we also consider the remaining number of reactions and the CPU time required for solving the equation system (Eq. 5). The evaluation can be based on any aspect of the simulation, e.g., for a homogeneous reactor or for a laminar diffusion flame.

In the present work, a homogeneous constant-pressure reactor model is used because its moderate computational requirements permit to conduct detailed parameter studies. In the present work, ignition delay time τ_i, steady-state temperature T_s, number of remaining reactions and CPU time were tracked, but the concentrations of certain species could also be considered by

simple inclusion in the evaluation function. The genetic algorithm must then find a minimum of the function *f* that considers the combined advantages of the reduced mechanism.

$$f = f_{\tau_i} + f_{T_s} + f_{N_R} + f_{CPU} \quad (5)$$

The ignition delay time error function $f_{\tau i}$ $(\tau_{i(0)}, \tau_{i(r)})$ depends on the difference in simulated ignition delay times of the reduced $\tau_{i(r)}$ and the full mechanism $\tau_{i(0)}$. For the steady state temperature error function $f_{Ts}(T_{in}, T_{s(0)}, T_{s(r)})$ we consider the initial temperature of the reactor T_{in} and the steady state temperature predicted by the reduced $T_{s(r)}$ and the full mechanism $T_{s(0)}$. The function that "controls" the number of reactions in a reduced mechanism, $f_{NR}(N_R, N_r)$ depends on the number of reactions in the full mechanism N_R and the number of reactions in the reduced mechanism N_r. The runtime for the simulation of each reduced mechanism is tracked using a function $f_{CPU}(t_{CPU(0)}, t_{CPU(r)})$ that depends on the time required for reaching a solution for the simulation with the reduced $t_{CPU(r)}$ and the full mechanism $t_{CPU(0)}$. Unlike the method from [33], our approach considers the computational time as a part of the objective function, its importance relative to the other criteria can be freely suggested by the user to match a certain need.

Each of the terms from the Eq. (5) is discussed in further detail in Section 4.

3.3 Selection, Crossover, and Mutation

The probability of selecting a chromosome for the next generation depends on its fitness value so that the fittest chromosomes are usually preferred, whereas some less fit chromosomes can still contribute valuable diversity. In this work a tournament selection is applied [35]. The general concept of tournament selection relies on a random selection of two or more chromosomes from the population, and choosing the fittest one among them. This is repeated *n* times, where *n* is the

number of chromosomes in the population (population size). Tournament selection has many advantages discussed by Goldberg and Deb [36], Blickle and Thiele [37], and Zhong *et al*. [38].

At the crossover stage, all of the chromosomes (except those retained by the elitist selection) are paired up and their genes are exchanged with a previously defined probability. There are several ways to perform a crossover. In our case the crossover of two chromosomes $\alpha, \beta \in \{0,1\}^{N_R}$ is accomplished by randomly choosing a single point at index $i \in \{1,\ldots,N_R\}$ from which the genes of the two chromosomes are swapped, yielding children

$$\gamma_1 = [\alpha_1, \alpha_2, \ldots, \alpha_i, \beta_{i+1}, \beta_{i+2}, \ldots, \beta_{N_R}] \text{ and } \gamma_2 = [\beta_1, \beta_2, \ldots, \beta_i, \alpha_{i+1}, \alpha_{i+2}, \ldots, \alpha_{N_R}]$$

This operation is known as a single-point crossover and is illustrated on Fig. 3.

After the formation of the children, a mutation operation takes place altering only a small number of randomly chosen genes in the chromosomes with a user-defined probability p_m (Eq. 6).

$$\left(\forall i \in \{1,2,\ldots,N_R\}\right)\ P[\alpha_i = 0] = p_m \tag{6}$$

This operation is important as it introduces diversity and helps to avoid stagnation at local optima. However, the mutation probability should be fairly low, otherwise it might randomize the entire search and destroy a good solution. In this work, the aim is to drive the solution towards smaller mechanisms, therefore, the mutation is done in one direction only, i.e., genes with a value 1 are converted to 0 while a value of 0 remains unchanged. This means that reactions are being gradually eliminated during the evolution beyond the elimination resulting from the evolution pressure applied by the fitness function. It must be stressed that for this particular goal, the mutation rate must be adjusted to the mechanism size (chromosome length) as the aggressive elimination of reactions might irreversibly eliminate important reaction paths and push the solution beyond an optimum. As the size of the mechanism is already a part of the

evaluation function (Section 4), the mutation should only provide a little help with the faster convergence towards the smaller mechanisms. (The effect of using a one-directional instead of a two-directional mutation operation is presented in Fig. A5 and discussed in Appendix A.) With the mutators defined in this way, the genetic algorithm is strongly directed towards reduction. A general one-directional mutation scheme is illustrated on Figure 4.

Having completed the operations of crossover and mutation, a new population is created and evaluated once again for the next cycle of selection and reproduction.

3.4 Evaluation Function

Achieving a significantly reduced mechanism with the preserved overall chemistry implies searching for a minimal deviation in terms of chemical properties between the reduced and the detailed mechanism. In other words, the acceptable deviation must be specified and incorporated into the evaluation function. In the work from Elliott *et al*. [23] the evaluation function incorporated species concentration profiles (generated using PREMIX code, [39]) for each genome and the aim of the genetic algorithm was to maximize this function, Eq. 7.

$$f_{\text{PREMIX}} = \left\{ \varepsilon + \sum_{j=1}^{N_{\text{conditions}}} \left(\sum_{k=1}^{n_s^{\text{reduced}}} W_k \frac{\left\| Y_{jk}^{\text{calc}} - Y_{jk}^{\text{orig}} \right\|_{L^2}}{\left\| Y_{jk}^{\text{orig}} \right\|_{L^2}} \right) \right\}^{-1} \tag{7}$$

The function takes into account $N_{\text{conditions}}$ boundary conditions, the mole concentrations of the k^{th} species in the j^{th} set of boundary conditions for the reduced and the detailed mechanism (Y_{jk}^{calc} and Y_{jk}^{orig} respectively), the normalized weights for each species which denotes their importance (W_k) and the coefficient ε was introduced to avoid numerical overflow. In later work, [26], the mechanism evaluation was based on a combination of functions evaluating mechanisms for

perfectly stirred reactors (PSR) and laminar premixed flame simulations, which improved the applicability of the reduced mechanism. The mathematical form of the PSR function was of a similar form as the Eq. 7, and the overall evaluation function maximized by the genetic algorithm was created by multiplying the PSR and the PREMIX function.

Another example, Perini *et al*., [32] used the genetic algorithm to optimize the reduced mechanism, extending the function from [26] to take into account the temperature and the species time evolution. These functions use the accuracy of the reduced mechanism as the reduction criteria, and the single values were normalized against their corresponding original values using the L^2 norm (Eq. 8).

$$\|f\|_{L^2} = \left(\int f^2(x)dx\right)^{1/2} \tag{8}$$

A work from Hernandez et al. [28] defines the objective function that evaluates the difference between the ignition delay times (τ) and the combustion temperature (T^{comb}) of the reduced and the detailed mechanism under N different sets of initial conditions for the constant volume simulations. The function uses linear normalization of the results from the reduced mechanism, against the corresponding values from the detailed mechanism (Eq. 9).

$$error_i = \frac{\sum_N \frac{1}{2}\left(\frac{\left|\tau_{\text{reduced}} - \tau_{\text{detailed}}\right|}{\tau_{\text{detailed}}} + 4 \cdot \frac{\left|T^{\text{comb}}_{\text{reduced}} - T^{\text{comb}}_{\text{detailed}}\right|}{T^{\text{comb}}_{\text{detailed}}}\right)}{N} \tag{9}$$

In this paper only few definitions of the function were described in order to give a brief overview on some existing approaches to definition of the evaluation function.

As previously stated, for a homogeneous reactor the properties of interest are the ignition delay time and a steady-state temperature. In addition, the concentration of species can be tracked by the evaluation function. This function can be defined by simple linear expressions (Eq. 10 and

11) of the relative difference between the results from the reduced and the detailed mechanism, featuring weighting factors w_1 and w_2 respectively.

$$f_{\tau_i} = w_1 \left| \frac{\tau_{i(r)} - \tau_{i(0)}}{\tau_{i(0)}} \right| \tag{10}$$

$$f_{T_s} = w_2 \left| \frac{T_{s(r)} - T_{s(0)}}{T_{s(0)} - T_{in}} \right| \tag{11}$$

In addition to the accuracy criteria, the evaluation of the cost of solving the mechanism was introduced to improve the reduction, to avoid the stiffness and to reduce the runtime in the later iterations of the genetic algorithm. This reduced runtime is particularly important for the parallel solution of different mechanism (chromosomes), as it helps to avoid that the processors have to wait for the solution of exceptionally slowly- or non-converging mechanisms, enabling the evaluation of more generations in a shorter time. For the cost criteria, a different type of functions was used as the aim of reducing computational cost is different than reducing the error. This is due to the fact that from the deviation point of view, the solution must approach an optimal value, while for the cost, the lowest possible value should be reached. Simple expressions for the cost evaluation were introduced in linear form, where the runtimes and number of reactions for the potential reduced mechanism was normalized against the runtime and the number of reaction of the detailed mechanism (Eq. 12 and 13) and multiplied with weighting factors w_3 and w_4 respectively.

$$f_{N_R} = w_3 \frac{N_r}{N_R} \tag{12}$$

$$f_{CPU} = w_4 \frac{t_{CPU(r)}}{t_{CPU(0)}} \tag{13}$$

Although relatively simple and obvious, these expressions were not satisfactory and therefore replaced by more complex expressions that were found to give a good trade-off between

different terms of the evaluation functions and to improve the overall convergence of the algorithm. For the ignition delay time error expression (Eq. 10), we have finally chosen a logarithmic dependence on the relative difference between the ignition delay times of the reduced and the full mechanism (Eq. 14). This logarithmic expression has the advantage of a steep convergence towards the optimal point for the accuracy ($\tau_{i(0)}$). If $\tau_{i(r)}$ is not predicted well enough, but the performance is still promising, the chromosome has a chance to survive and 'fix' the $\tau_{i(r)}$ in the next generation.

$$f_{\tau_i} = \ln\left(1 + \sigma \cdot \left|\frac{\tau_{i(r)} - \tau_{i(0)}}{\tau_{i(0)}}\right|\right) \tag{14}$$

The sharpening factor σ is used to adjust the steepness of the function, thus directing the solution to the reference point with more or less evolution pressure. The influence of this factor on the corresponding function term is illustrated in Fig. 5a. The same type of the function is adopted for the steady-state temperature error (Eq. 15).

$$f_{T_s} = \ln\left(1 + \sigma \cdot \left|\frac{T_{s(r)} - T_{s(0)}}{T_{s(0)} - T_{in}}\right|\right) \tag{15}$$

Changing its sharpness factor will generally put more weight on the accuracy, hence leading to chemical mechanisms that are more accurate in terms of ignition delay time and reaching a steady-state temperature. A number of numerical experiments have shown that the logarithmic form of the accuracy function does not outweigh other terms of the overall objective function (due to the low-steepness in the area of large deviations). To avoid outweighing the other parts of the function, the cost terms had to be modified as well, leading to a sigmoid form for the number of reactions (Eq. 16) and the runtime (Eq. 17).

$$f_{N_R} = \left[1 + \exp\left(\sigma \cdot \left(1 - \frac{N_r}{\lambda \cdot N_R}\right)\right)\right]^{-1} \quad (16)$$

$$f_{CPU} = \left[1 + \exp\left(\sigma \cdot \left(1 - \frac{t_{CPU(r)}}{\lambda \cdot t_{CPU(0)}}\right)\right)\right]^{-1} \quad (17)$$

The asymptotic behavior of the sigmoid functions (Fig. 5c and 5d) prevents the cost criteria to outweigh the other part of the overall function, so that a sensible solution can be expected in the steep part of the curve. The sharpening coefficient σ and shifting coefficient λ are required for tuning the functions in such a way to favor chemically reasonable solutions. The effect of the different expressions for the cost evaluation function terms on the overall algorithm behavior is examined in Appendix A.

The overall evaluation function composed of the terms described above is found to provide a fast convergence towards the plausible solution. These expressions are shown to preserve the important features observed during the evolution ensuring a good tradeoff between the individual criteria and if necessary, weights for each part of the overall objective function can be introduced and controlled easily.

4. Results

The reduction technique was tested for the GRI3.0 mechanism for natural gas combustion consisting of 325 reaction steps and 53 species, and for an iron-oxide formation mechanism [1] that consists of 144 reactions and 34 species. The reduction was done for a homogeneous constant-pressure reactor model only, but the performance of the resulting reduced mechanisms was also analyzed for a laminar flame model. The reduction ran for 1,000 generations with a

population size of 48 on HPC server providing 24 AMD Opteron 2.6 GHz cores within 192 and 72 CPUh for methane and iron pentacarbonyl respectively.

4.1 Methane Combustion Mechanism

The test case for reducing the GRI-Mech 3.0 was constructed to correspond to the test case by [23] for the combustion of a stoichiometric mixture of 12.5 % methane, 25 % oxygen and 62.5 % argon (mole fractions) in a homogeneous constant-pressure reactor model. The reduced mechanism was then applied for a burner-stabilized flame model with conditions taken from [23]. The reactor has been simulated at a pressure of 2670 Pa and an initial temperature of 1800 K. For the genetic algorithm search process, a population of 48 chromosomes was set, corresponding to the number of CPU cores available on a single server node. The coefficients of the evaluation functions (Eqs. 14 – 17) are presented in Table 1.

The tournament selection, a single-point crossover with a crossover rate of 0.4 and a one-directional mutation operation with the mutation rate of 0.003 provided a plausible convergence within 1000 generations (Table 2). The mutation was applied to the entire chromosome, so that a small mutation probability had to be chosen for each gene. As a result, the best performing mechanism was found in generation 339 with an overall score of 0.01898 (Fig. 6), consisting of 58 reactions (Fig. 7). The evolution diagram (Fig. 6) shows the best scores within a generation and the statistical distribution of the scores for all the individuals, expressed as quantiles (Q).

To see what this raw score means in terms of physical properties of the combustion process, temperature and species profiles have been calculated and compared to the ones from the detailed mechanism. A comparison of temperature and major species profiles from the detailed

and reduced mechanisms calculated for a homogeneous constant-pressure reactor is shown in Figures 8a and 8b.

To test the suitability of the reduced mechanism for a flame, a burner-stabilized premixed flame [23] was used to calculate important species profiles and temperature profiles along the flame axis, for the full and the reduced mechanisms. The mixture composition and pressure were set as in the reactor case, with the unburned gas temperature of 373.7 K. Temperature and species profiles along the flame axis calculated from the reduced and detailed mechanism are compared in Figs. 8a and 8b. It is interesting that even though the mechanism was not reduced with the flame calculations within the evaluation function, the mechanism was still able to produce similar flame results as the full mechanism. This implies that a reduced mechanism for a flame can be obtained very efficiently by first reducing the mechanism for an (easy to solve) homogeneous reactor and only then continuing the optimization for the (more expensive to solve) flames.

4.2 Iron Pentacarbonyl Mechanism

An iron pentacarbonyl mechanism that describes the gas-phase synthesis of iron and iron-oxide nanoparticles is considered, which was found to have a significant catalytic effects on the kinetics of hydrocarbon flames. An extensive mechanism for iron pentacarbonyl doped hydrogen or methane flames was first introduced by Rumminger et al. [40], enhanced by Staude et al. [41] and extended for iron oxide formation by Wlokas et al. [1]. For the hydrogen combustion branch, the mechanism by Li et al. [42] is used. The mechanism is reduced using genetic algorithms. The results are validated against the detailed mechanism. The reacting mixture is composed of the

following mole fractions: $X_{H_2} = 0.2857$, $X_{O_2} = 0.2857$, $X_{FeC_5O_5} = 0.0003$, $X_{Ar} = 0.4248$ and reacts at a pressure of 3000 Pa at an initial temperature of 1100 K. For the genetic algorithm search, a population size of 48 chromosomes, a tournament selection and a single-point crossover (crossover rate 0.6) were used. The one-directional mutation rate for this case is 0.002 (Table 3). The evaluation function terms from Eqs. (14 –17) were adopted, with an additional term that is used to preserve the concentration of a fictitious gaseous species Fe_2O_3 [1], (Eq. 18).

$$f_{Fe_2O_3} = \ln\left(1 + \sigma \cdot \left| \frac{X_{Fe_2O_3(r)} - X_{Fe_2O_3(0)}}{X_{Fe_2O_3(0)}} \right| \right) \tag{18}$$

The coefficients for tuning the evaluation function (Eqs. 15–19) used for this case are given in Table 4.

Evolution diagrams for the overall evaluation function and the number of active reactions are shown in Figures 9 and 10.

The reduced iron pentacarbonyl mechanism was used for the simulation of reactions in a freely-propagating flame calculation and the results were compared to those obtained with the detailed mechanism. The flame was simulated with the same mixture composition and pressure as for the reactor case; with an unburned gas temperature of 373 K. Figures 11 and 12 show a good agreement between the results for the detailed and the reduced mechanism.

5. Conclusions

A method based on a genetic algorithm was used to reduce the size of reaction mechanisms by eliminating reactions that are “unimportant” for a specific set of conditions. The method gradually eliminates reactions from the full mechanism according to the user’s defined accuracy and cost criteria, and maintains the most important properties of the full mechanism (namely

describing temperature and species profiles and ignition delay time). The parameters of the genetic algorithm and the specific definition of the evaluation function can be set to allow a flexible adjustment of the reduction criteria to the needs of the user. The tracking of the runtime for each calculation has shown to enhance the convergence of the reduction process towards a fast solution by avoiding stiffness and long waiting times on the other processor cores during the parallel computing. The new methodology was applied to reduce a methane combustion mechanism (GRI-Mech 3.0) from 325 reactions and 53 species to 58 reactions and 26 species; and to reduce a combined iron pentacarbonyl and hydrogen combustion mechanism from 144 reactions and 34 species to 37 reactions and 24 species. The mechanisms were evaluated for both homogeneous reactor and laminar flame cases. The reduction method was found to be easily adjusted for generating reduced mechanisms with a user-specified tradeoff between size and accuracy.

Bibliography

1. Wlokas, I., Faccinetto, A., Tribalet, B., Schulz, C., Kempf, A. Int J Chem Kinet 2013, 45(8), 487-498.
2. Curran, H.J.; Gaffuri, P.; Pitz, W.J.; Westbrook, C.K. Combust Flame 1998, 114 (1–2), 149-177.
3. Mehl, M.; Pitz, W.J.; Westbrook, C.K.; Curran, H.J. Proc Comb Inst 2011, 33, 193-200.
4. Tomlin, A.S.; Turanyi, T.; Pilling, M.J.; in: M.J. Pilling (Ed.), Low-Temperature Combustion and Auto Ignition, in: Comprehensive Chemical Kinetics 1997, 35, 293-437.
5. Chen, J.-Y. Workshop on "Numerical Aspects of Reduction in Chemical Kinetics" CERMICS-ENPC Cite Descartes - Champus sur Marne, France, (September 2nd, 1997).
6. Montgomery, Ch. J.; Marc, A.; Cremer, M.A.; Chen, J.-Y; Westbrook, Ch. K.; Maurice, L.Q. 35th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit 20-24 June 1999, Los Angeles, California.
7. Maas, U.; Pope, S.B. Proc Comb Inst 1994, 25, 1349-1356.
8. Maas, U. Comput Visual Sci 1998, 1, 69 - 81.
9. Lam, S. H. Combust Sci Technol 1993, 89, 5-6, 375-404.
10. Lu T. F.; Ju Y.; Law C.K. Combust Flame 2001, 126 (1-2), 1445-1455.
11. Valorani, M.; Creta, F.; Goussis, D. A.; Najm, H. N.; Lee, J. C. Combust Flame 2006, 146, 29-51.
12. Sun, W.; Chen, Z.; Gou, X.; Ju, Y. Combust Flame 2010, 157, 1298–1307.
13. Androulakis, I.P.; Grenda, J.M.; Bozzelli, J.W. AIChE Journal 2004, 50 (11), 2956–2970.
14. Turanyi, T. J Math Chem 1990, 5, 203–248.
15. Kuo, J.; Wei, J. Ind Eng Chem Fundam 1969, 8 (1), 114–123.
16. Petzold, L.; Zhu, W. J. AIChE Journal 1999, 45 (4), 869–886.
17. Androulakis, I.P. AIChE Journal 2000, 46 (2), 361-371.
18. Bhattacharjee, B.; Schwer, D.A.; Barton, P.I.; Green, W.H. Combust Flame 2003, 135, 191–208.
19. Schwer, D. A.; Lu, P.; Green, W. H. Jr. Combust Flame 2003, 133, 451–465.
20. Banerjee, I. and Ierapetritou, M. G. Chem Eng Sci 2003, 58, 4537 – 4555.
21. Edwards K.; Edgar T.F.; Manousiouthakis V. I. Comput Chem Eng. 1998, 22, 239–246.
22. Montgomery Ch. J.; Yang C.; Parkinson A. R.; Chen J.-Y. Combust Flame 2006, 144, 37–52.
23. Elliott, L.; Ingham, D.B.; Kyne, A.G.; Mera, N.S.; Pourkashanian, M.; Wilson, Ch.W. Ind Eng Chem Res 2005, 44, 658-667.
24. Smith, G. P.; Golden, D. M.; Frenklach, M.; Moriarty, N. W.; Eiteneer, B.; Goldenberg, M.; Bowman, T. C.; Hanson, R. K.; Song, S.; Gardiner, W. C. Jr.; Lissianski, V. V.; Qin, Z.; http://www.me.berkeley.edu/gri_mech/ (accessed 20.12.2012)
25. Bernstein, J. S.; Fein, A.; Choi, J. B.; Cool, T. A.; Sausa, R. C.; Howard, S. L.; Locke, R. J.; Miziolek, A. W. Combust Flame 1993, 92, 85.

26. Elliott, L.; Ingham, D.B.; Kyne, A.G.; Mera, N.S.; Pourkashanian, M.; Whittaker, S. Comput Chem Eng 2006, 30, 889–900.
27. Rosenbrock, H. H. Computer Journal 1960, 3, 175 - 184.
28. Hernández, J. J.; Ballesteros, R.; Sanz-Argent, J.; Math Comput Model 2010, 52, 1185-1193.
29. Polifke, W.; Geng, W.; Dobbeling, K.; Combust Flame 1998, 113, 119–35.
30. Elliott, L.; Ingham, D.B.; Kyne, A.G.; Mera, N.S.; Pourkashanian, M.; Wilson, C.W.; Genetic and Evolutionary Computation Conference, Late Breaking Papers, GECCO, New York, 2002, 138–44.
31. Elliott, L.; Ingham, D.B.; Kyne, A.G.; Mera, N.S.; Pourkashanian, M.; Wilson, C.W. Prog Energy Combust Sci 2004, 30, 297-328.
32. Perini, F.; Brakora, L. J.; Reitz, D. R.; Cantore, G. Combust Flame 2012, 159, 103-119.
33. Nagy, T.; Turányi, T.; Combust Flame 2009, 156, 417-428.
34. Goldberg. D. E. Addison-Wesley, Reading, MA, 1989.
35. Miller, B. L.; Goldberg, D. E. Complex Syst 1995, 9, 193–212.
36. Goldberg, D. E.; Deb, K. in: G.J.E. Rawlins (Ed.), Foundations of Genetic Algorithms, Morgan Kaufmann, Los Altos, 1991, 69–93.
37. Blickle, T. and Thiele, L. TIK-Report, Zurich (1995)
38. Zhong, J., Hu, X., Gu, M., Zhang, J., Proceeding of the International Conference on Computational Intelligence for Modelling, Control and automation, and International Conference of Intelligent Agents, Web Technologies and Internet Commerce 2005, 1115 – 1121.
39. Kee, R. J.; Miller, J. A.; Jefferson, T. H.; Sandia Report SAND80-8003; Sandia National Laboratories, Sandia, NM, 1980.
40. Rumminger, M. D.; Reinelt, D.; Babushok, V.; Linteris, G. T. Combust Flame 1999, 116, 207-219.
41. Staude, S., Hecht, C., Wlokas, I., Schulz, C., Atakan, B., Zeitschrift für Physikalische Chemie 2009, 223(4-5), 639-649.
42. Li, J.; Zhao, Z.; Kazakov, A.; Chaos, M.; Dryer, F. L.; Scire, J. J. Int J Chem Kinet 2007, 39, 109-136.
43. Connaire, M. O.; Curran, H. J.; Simmie, J. M.; Pitz, W. J.; Westbrook, Ch. K. Int J Chem Kinet 2004, 36, 603-622.
44. Grefenstette, J. J., IEEE Transactions On systems, Man and Cybernetics 1986, Vol. SMC-16 (1), 122-128.

Appendix A.

Effect of initialization

The effect of different initializations was studied, comparing an initialization with the full mechanism to an initialization where one reaction was deactivated in every chromosome. Figure A1 shows the evolution of the total score over the generations, indicating that the type of initialization only affects the very first generations. For this case, the setup starting from the full mechanism has surpassed the incomplete initialization after approximately fifty generations, but the latter has converged faster initially.

Effect of one-way vs. two-way mutation

A mutator was chosen to create additional evolution pressure towards smaller mechanisms by only permitting mutations to remove reactions, but not to add them again. Where this approach involves the risk that a key reaction might be "lost" forever, we have found this approach to be generally beneficial, as can be seen by the faster convergence in Fig. A2 as compared to Fig. A3. The overall error and the remaining reactions over 500 generations for the sigmoid normalization of the runtime and the number of reactions are shown for the two-directional mutation (Fig. A3) and the one-directional mutation (Fig. A4). The two-directional mutation results in somewhat slower convergence and many previous tests have shown that it may lead to a slightly bigger mechanism, as would be expected, since this mutator shifts the evolutionary pressure towards small mechanism. But the overall effect is small. Nevertheless, the mutation rate for both mutators must be carefully chosen to prevent the solution to fall beyond the optimum.

Effect of the choice of evaluation functions

In order to examine the effect of the major parameters of the genetic algorithm on its overall convergence, a mechanism containing 104 reactions [43], combined with the mechanism from [40], was taken as a test case. The study considers various evaluation functions for the cost parameters (runtime and number of reactions), the initialization, and different mutators. The terms for the evaluation of the accuracy criteria (Eqs. 14 and 15) are defined in a way that directs the potential solution towards the correct solution calculated from the detailed mechanism (reference point). On the other hand, cost evaluation terms (Eqs. 16 and 17) aim to direct the potential solution towards the smallest possible value. In order to achieve a satisfactory trade-off between the two usually 'opposed' parts (accuracy and cost) of the overall evaluation function, the cost criteria must be designed carefully to avoid outweighing the accuracy criteria. Numerous experiments were done with different shapes and weighting factors for the functions. In the present paper, only three types of functions were presented, namely linear, square and sigmoid as shown in Fig. A4.

For testing, these functions were weighted to correspond to the best-performing sigmoid function adopted as the optimum for the number of reactions (Eq. 16) and the runtime (Eq. 17). The linear normalization function for the runtime is $f_{CPU} = 0.125\left(t_{CPU(r)}/t_{CPU(0)}\right)$ while the linear function for the number of reactions is $f_{N_R} = 0.6667\left(N_r/N_R\right)$. Square normalization functions are, for the runtime $f_{CPU} = 0.03125\left(t_{CPU(r)}/t_{CPU(0)}\right)^2$ and for the number of reactions $f_{N_R} = 0.8889\left(N_r/N_R\right)^2$. The accuracy terms use Eq. (14) for the ignition delay time, Eq. (15) for the temperature, and Eq. (18) for the iron-oxide mole fractions. A homogeneous reactor case is used for the evaluation. A single-point crossover with a probability of 0.4 was applied, combined with a tournament selection for a population size of 48. The full-mechanism initialization and one-directional

mutation were used. The effects of linear, square and sigmoid normalization on the overall error and the reduced mechanism's size behavior are shown in Figs. A5–7.

Square normalization functions give better convergence behavior of the algorithm in comparison to the linear functions during 500 generations as shown on Fig. A6. The algorithm using linear functions (Fig. A5) did not reach the stagnation point in 500 generations, while square (Fig. A6) and sigmoid functions (Fig. A7) did. Not only was the convergence influenced by the choice of the normalization cost functions, but the accuracy as well. In our cases, the sigmoid form of the cost functions gave the best accuracy-cost balance in comparison to the other two (Fig. A7).

Effect of the crossover rate

For a constant population size of 48 individuals, as used in the analysis above, the crossover rate was varied to examine its influence on the overall convergence of the algorithm within 500 generations. Figure A8 shows the behavior of the minimal value of the overall error for runs with different crossover rates. The crossover rate has a small impact on the convergence of the algorithm within 500 generations, but a good compromise between the convergence speed and the minimal error value is achieved for a crossover rate of 0.4 which is consistent with the "rule of thumb" for different genetic algorithm applications [44].

Effect of the population size

The effect of population size on the rate of convergence was investigated at a constant crossover rate of 0.4. Figure A9 shows the expected behavior of faster convergence with a bigger population size, but it must be noted that the effort for the solution of a single generation increases with the size of the population, as well as the variance of the population. This is

illustrated on Figure A10, where the overall error distribution in terms of its minimum and the median (Q50) is shown along the entire sampling space. The number of samples is the number of the genomes in the population multiplied with the number of generations.

Tables

Table 1 Parameter values for the evaluation function terms used to reduce the GRI-Mech 3.0

Term	**σ**	**λ**
Eq. (14)	1	-
Eq. (15)	1	-
Eq. (16)	6	0.75
Eq. (17)	6	4

Table 2 Genetic algorithm parameters used for the reduction of GRI-Mech 3.0

Parameter	**Value**
Population size	48
Selection	Tournament selection
Crossover	Single-point
Crossover rate	0.4
Mutation	One directional
Mutation rate	0.003
Number of generations	1000

Table 3 Parameter values for the evaluation function terms used to reduce the iron pentacarbonyl mechanism.

Term	**σ**	**λ**
Eq. (14)	4	-
Eq. (15)	4	-
Eq. (16)	6	0.75
Eq. (17)	6	4
Eq. (18)	4	-

Table 4 Genetic algorithm parameters used for the reduction of the iron pentacarbonyl mechanism

Parameter	**Value**
Population size	48
Selection	Tournament selection
Crossover	Single-point
Crossover rate	0.6
Mutation	One directional
Mutation rate	0.002
Number of generations	1000

Figures

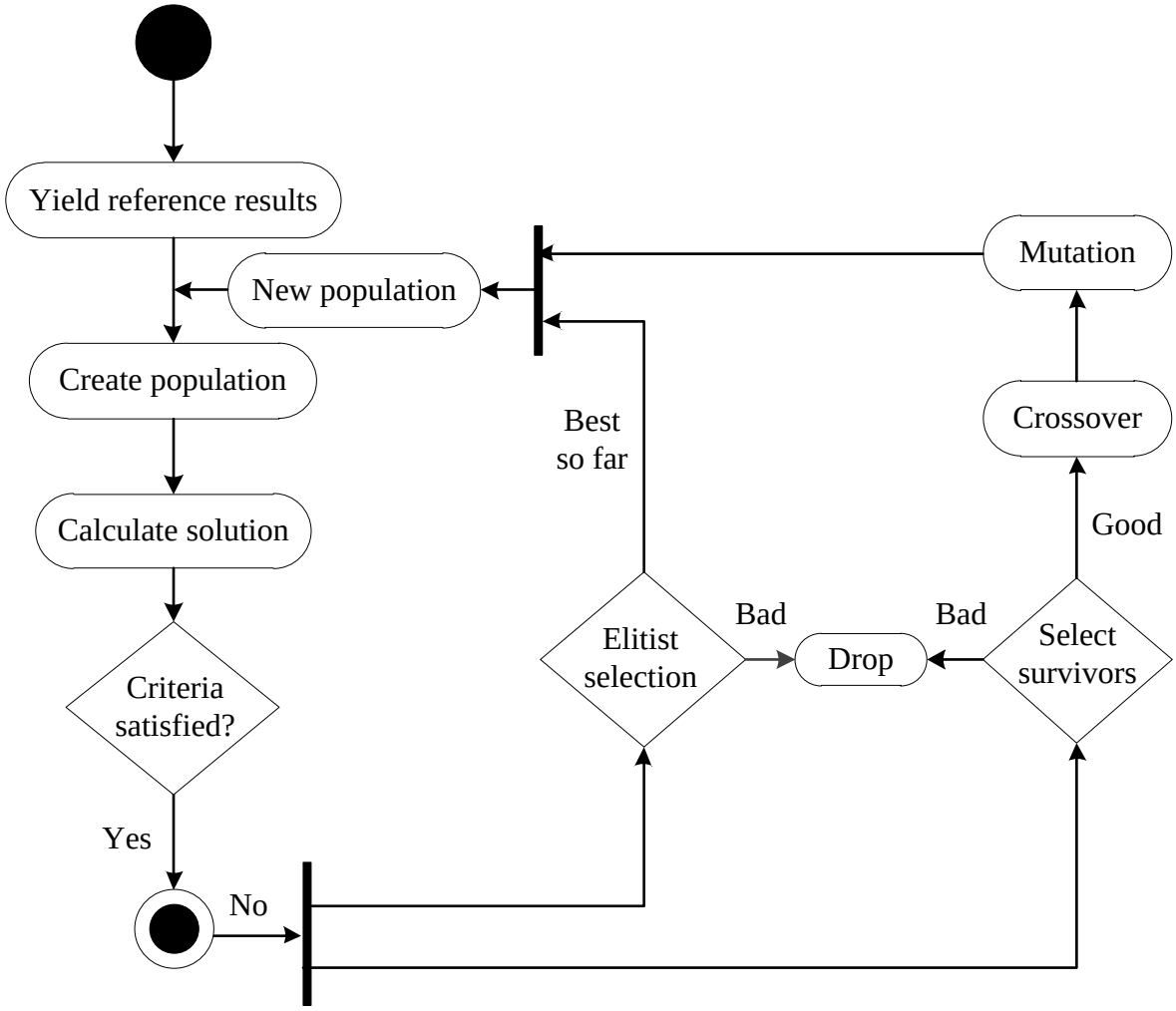


Figure 1 Activity chart of the genetic algorithm

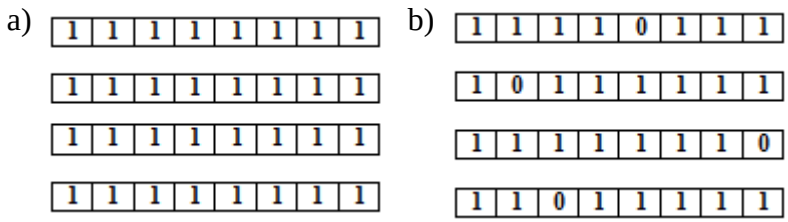


Figure 2 Initial populations of four mechanism representations (chromosomes) for a) a full-mechanism initialization and b) mechanisms with only one reaction missing

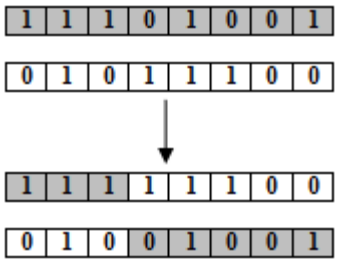


Figure 3 A single-point crossover between two parents

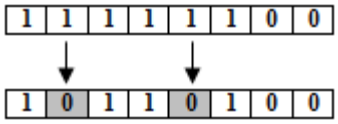
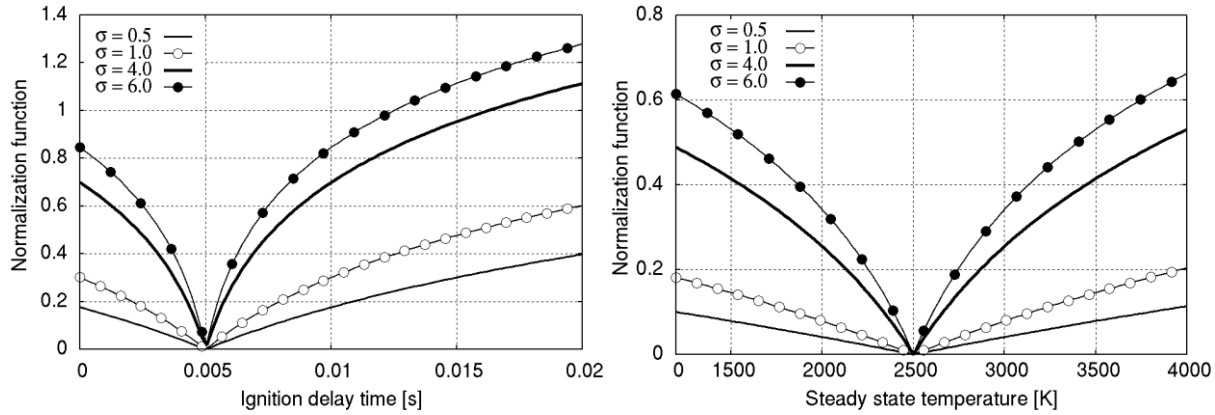


Figure 4 One-directional mutation

a) b)



c) d)

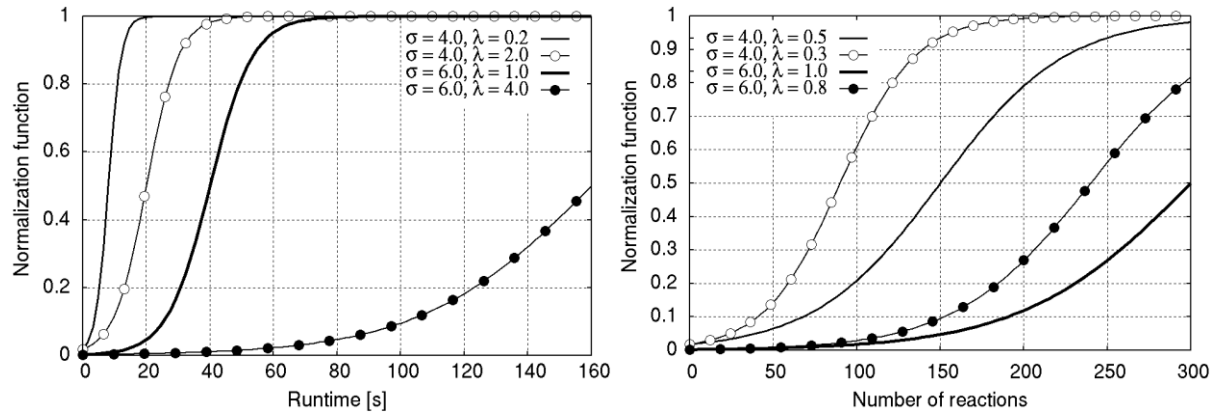


Figure 5 Evaluation function terms for a) ignition delay time, b) steady-state temperature, c) number of reactions and d) computational time.

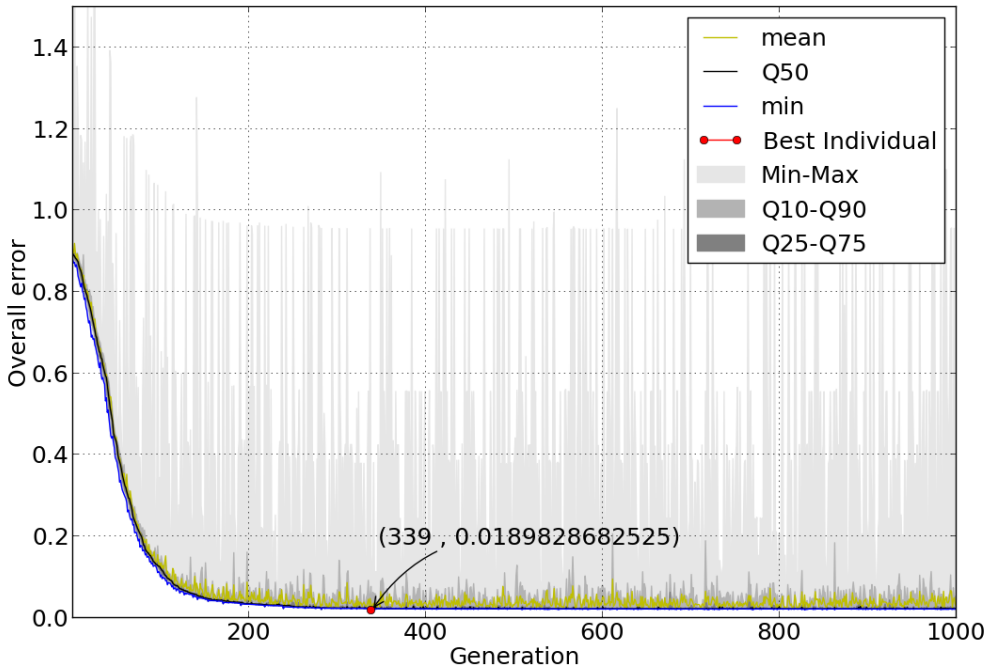


Figure 6 Overall score for reduced mechanisms during evolution

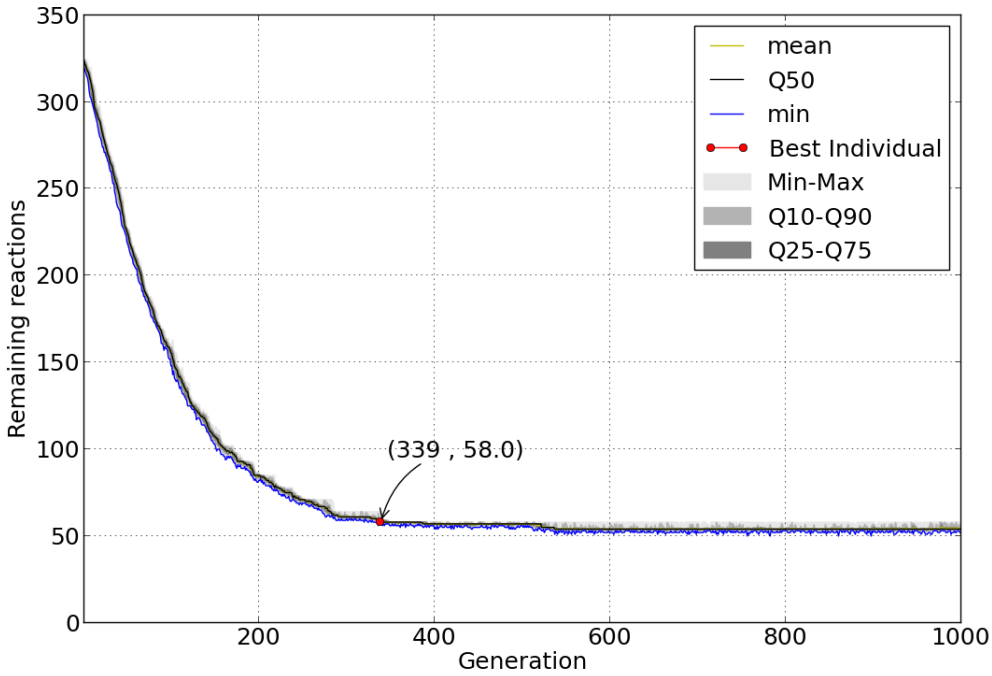


Figure 7 Number of reactions in the reduced mechanisms (chromosomes)

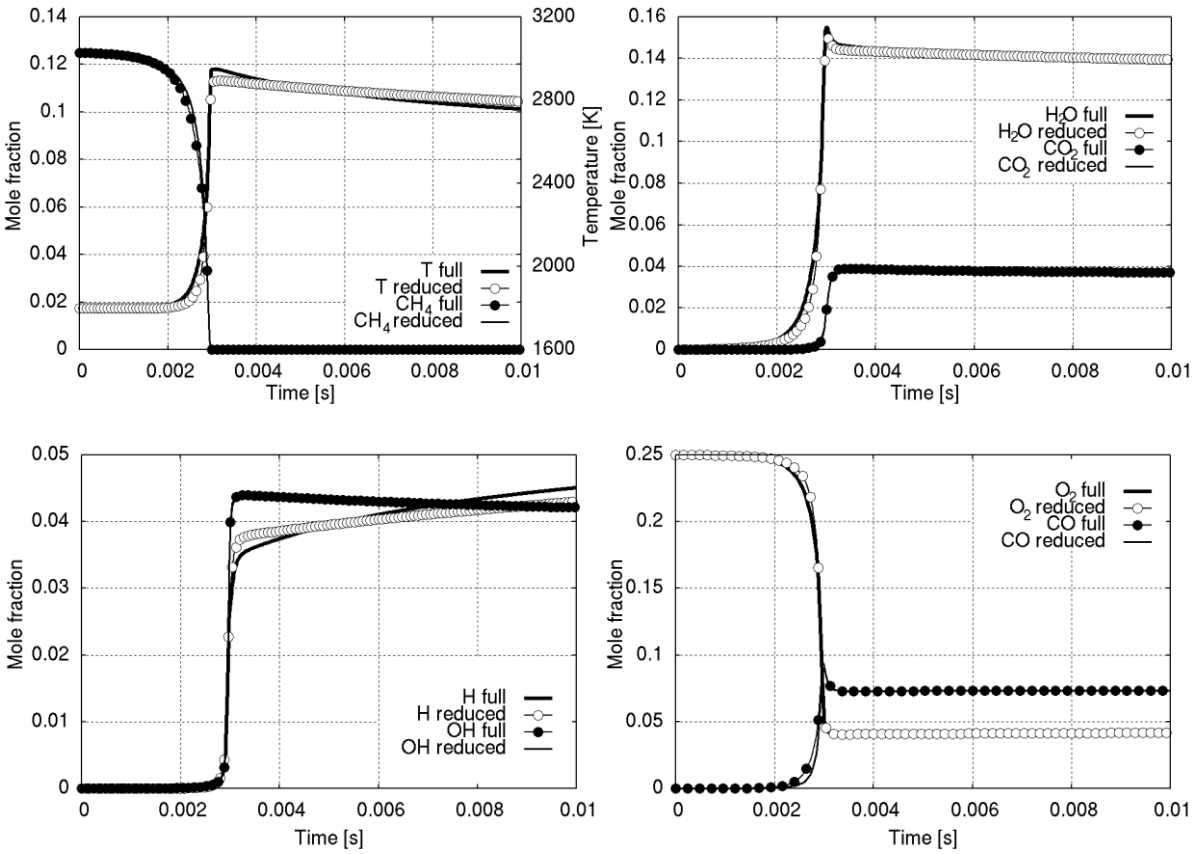


Figure 8a Comparison of the temperature and the species profiles for the full and the reduced GRI-Mech 3.0 for the reactor.

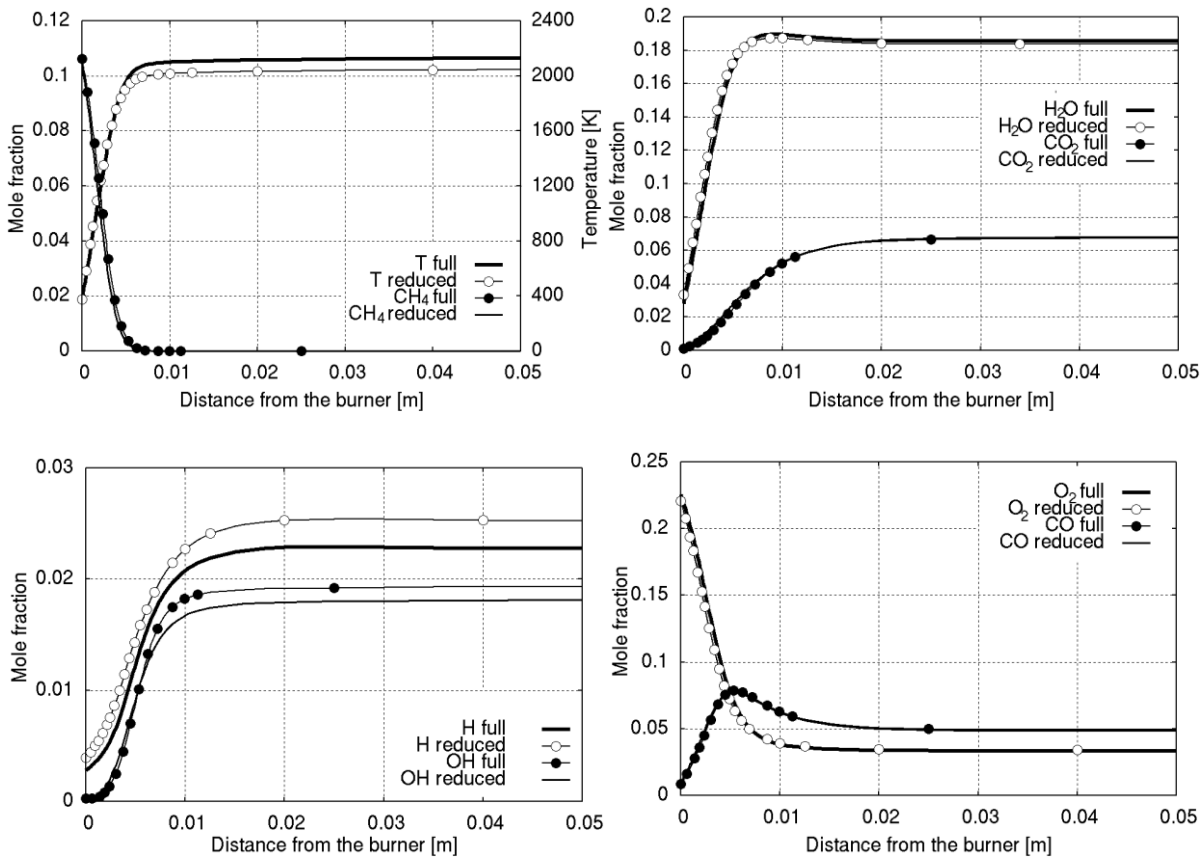


Figure 8b Comparison of the temperature and the species profiles for the full and the reduced GRI-Mech 3.0 for the flame.

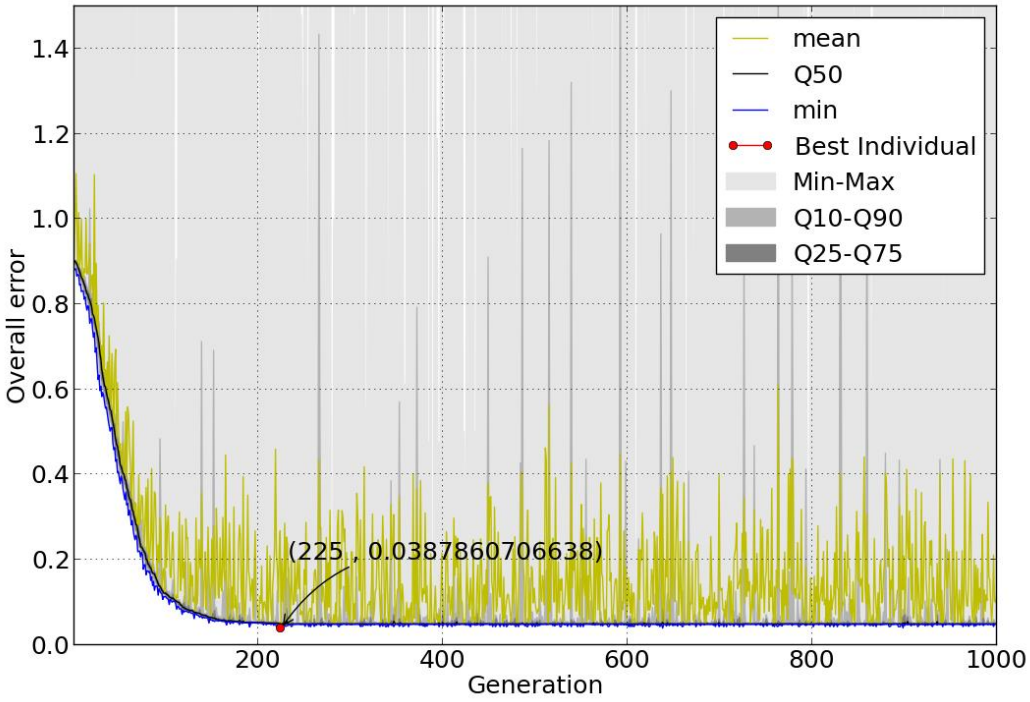


Figure 9 Evolution of the reduced mechanisms in 1000 generations.

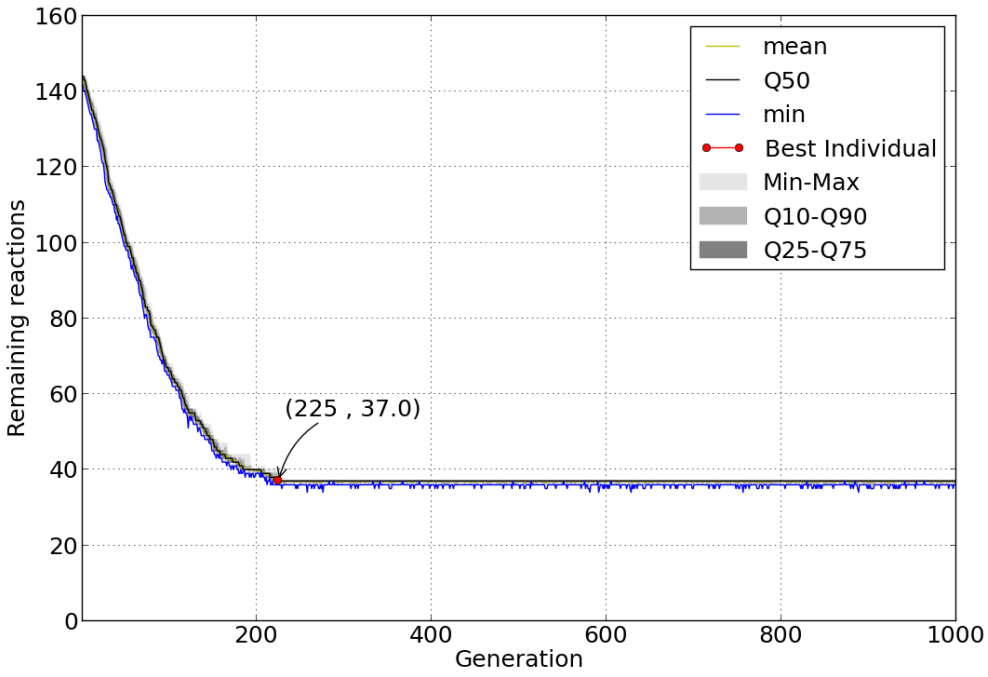


Figure 10 Evolution of the overall score of the reduced mechanisms in 1000 generations.

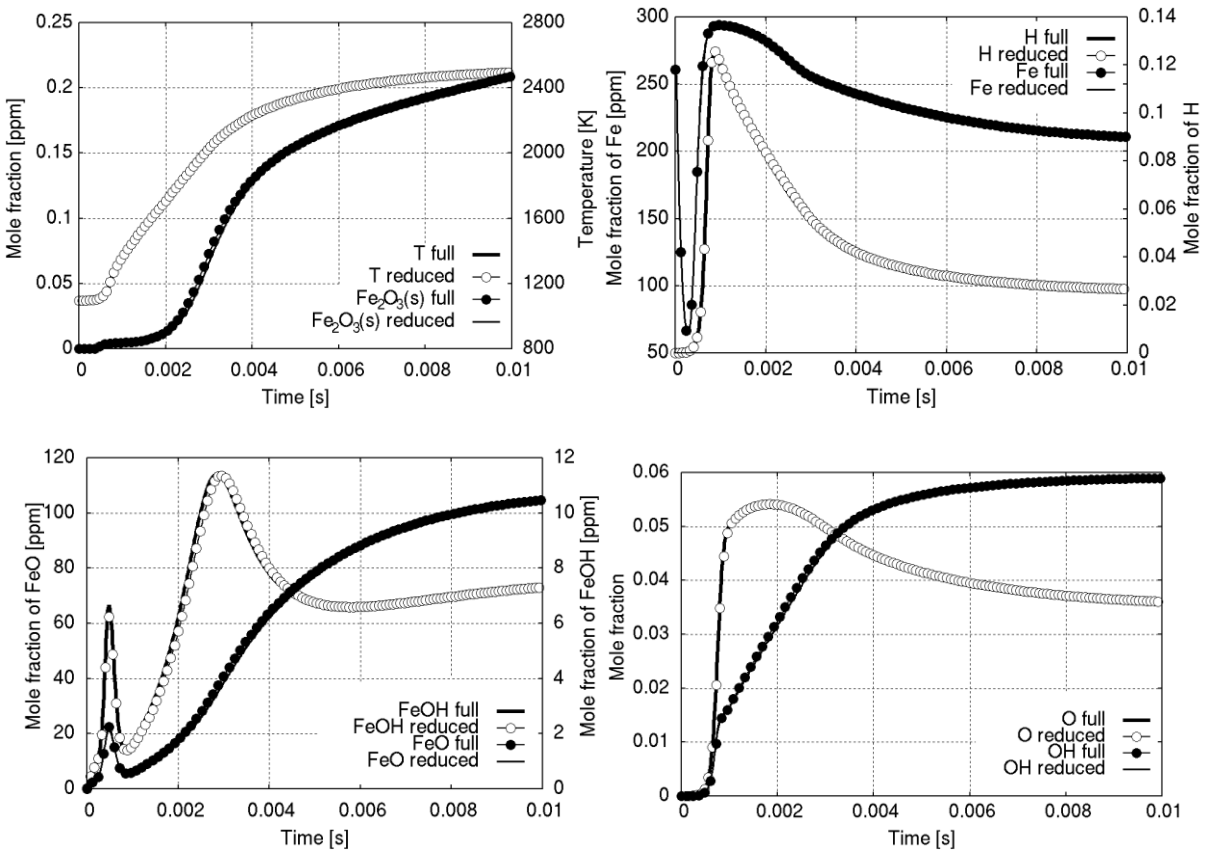


Figure 11 Simulated temporal variation of temperature and important iron and hydrogen species concentrations for a homogeneous constant-pressure reactor based on the full and the reduced mechanism.

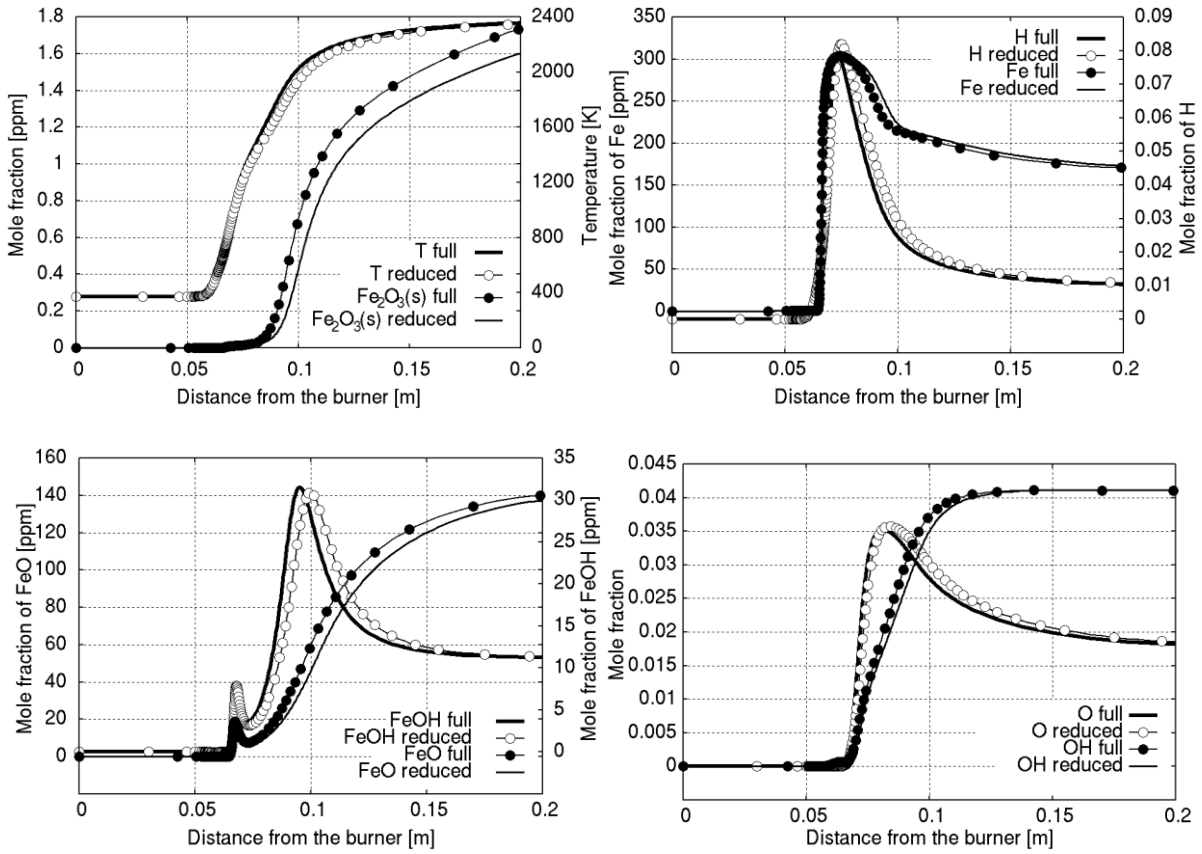


Figure 12 Simulated spatial variation of temperature and important iron and hydrogen species concentrations for a freely-propagating flame based on the full and the reduced mechanism.

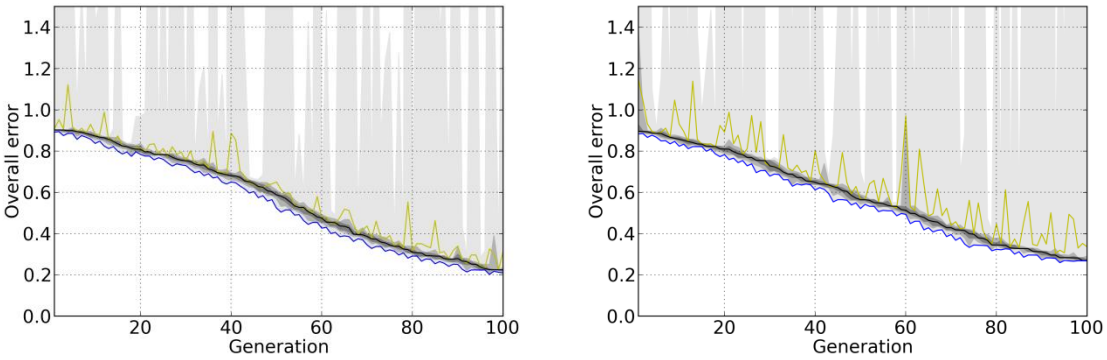


Figure A1 Overall error convergence over 100 generations, for the case of the full-mechanism initialization (left) and "one reaction missing" initialization (right).

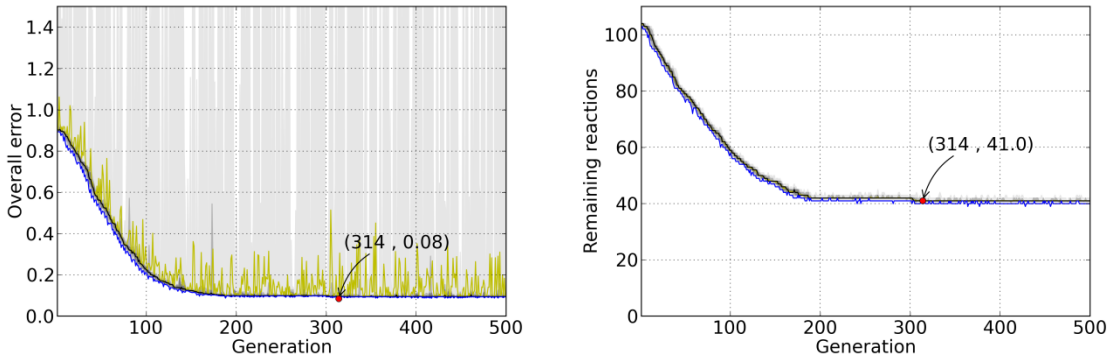


Figure A2 Overall convergence with the two-directional mutation (sigmoid normalization of the runtime and the number of reactions).

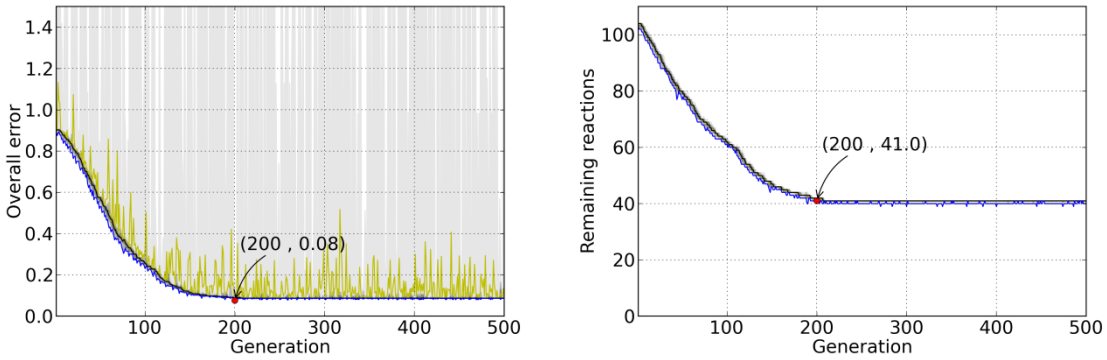


Figure A3 Overall convergence with the one-directional mutation (sigmoid normalization of the runtime and the number of reactions).

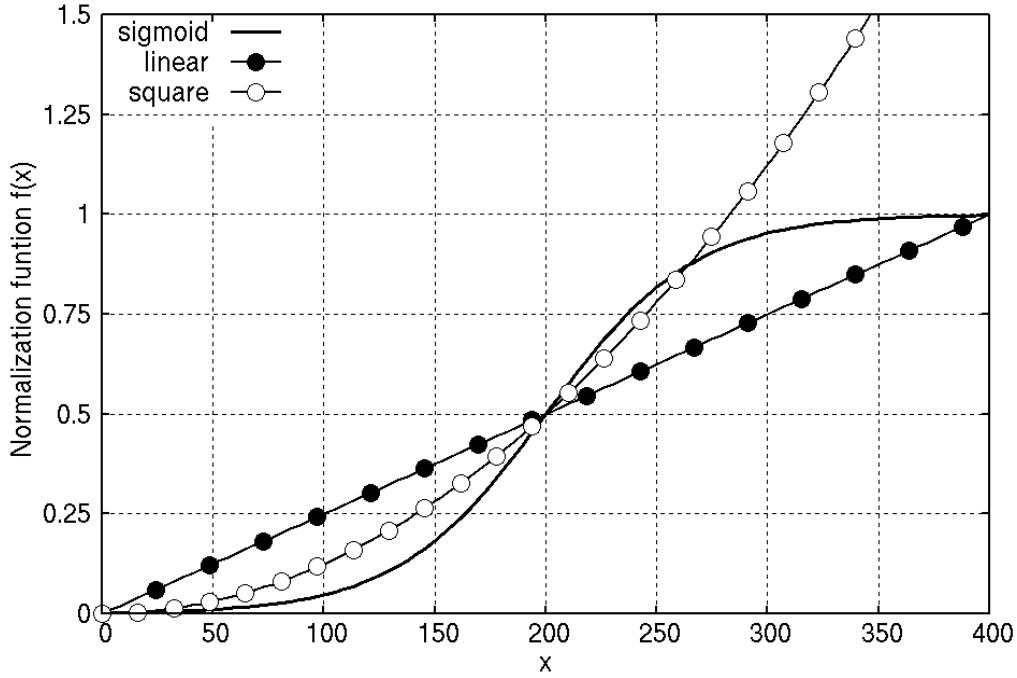


Figure A4 Linear, square and sigmoid normalization function *f*(*x*), where *x* stands for the cost parameter.

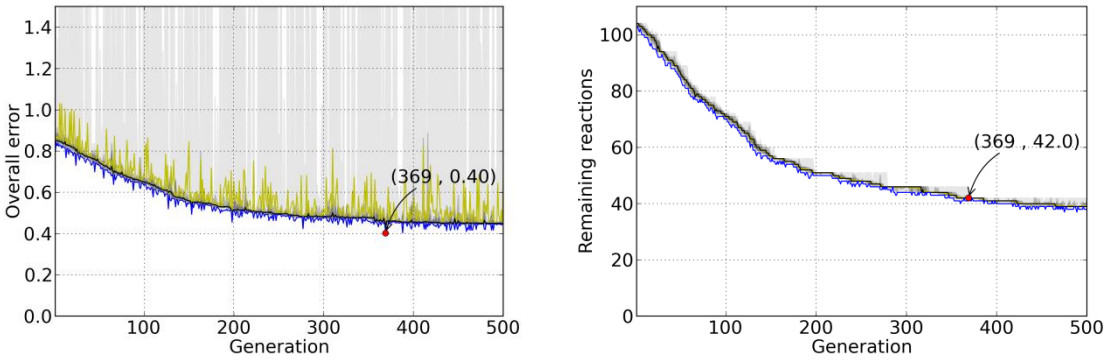


Figure A5 The overall error and the remaining reactions over 500 generations for the linear normalization of the runtime and the number of reactions.

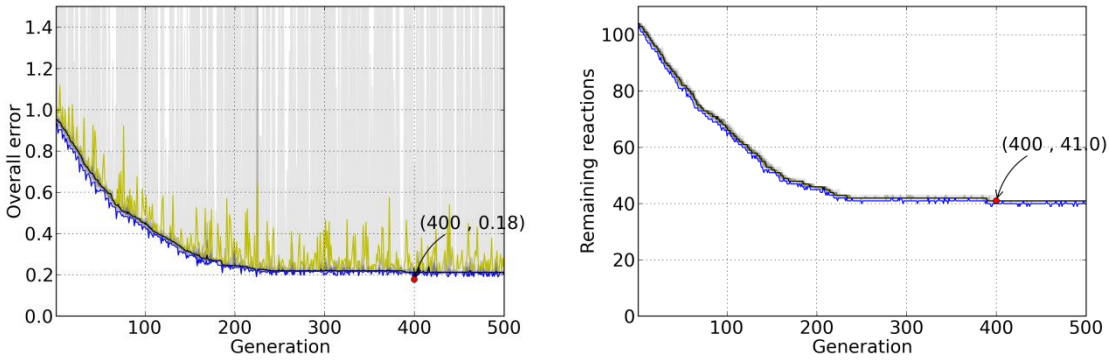


Figure A6 The overall error and the remaining reactions over 500 generations for the square normalization of the runtime and the number of reactions.

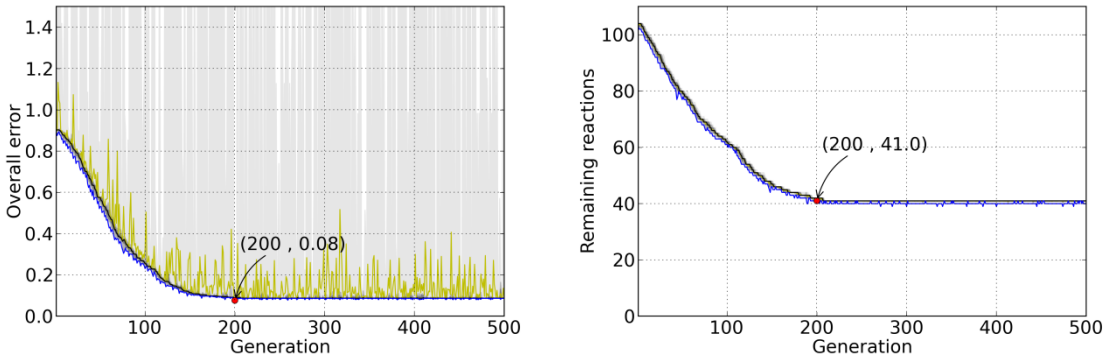


Figure A7 The overall error and the remaining reactions over 500 generations for the sigmoid normalization of the runtime and the number of reactions.

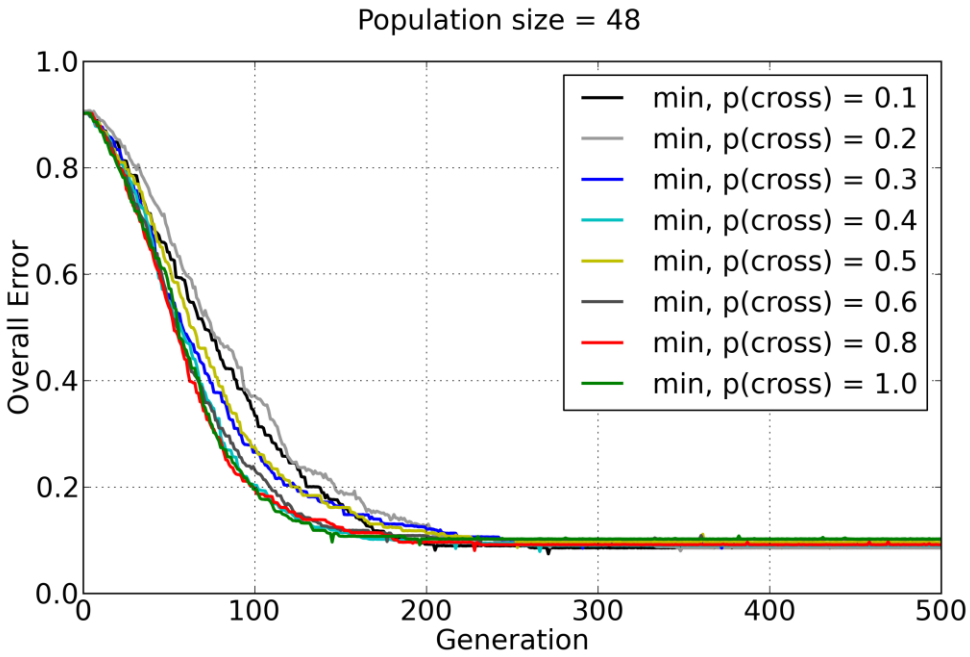


Figure A8 The overall error over 500 generations for runs with the constant population size of 48 individuals and different crossover rates, p(cross).

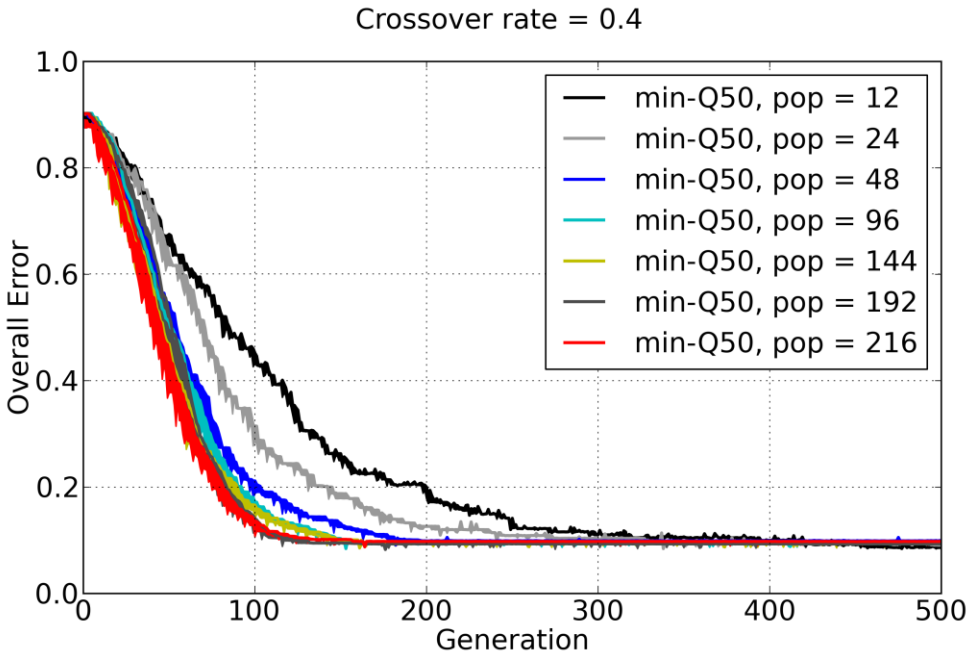


Figure A9 The overall error minimum and the median distribution over 500 generations for runs with the constant population crossover rate of 0.4 and different population size.

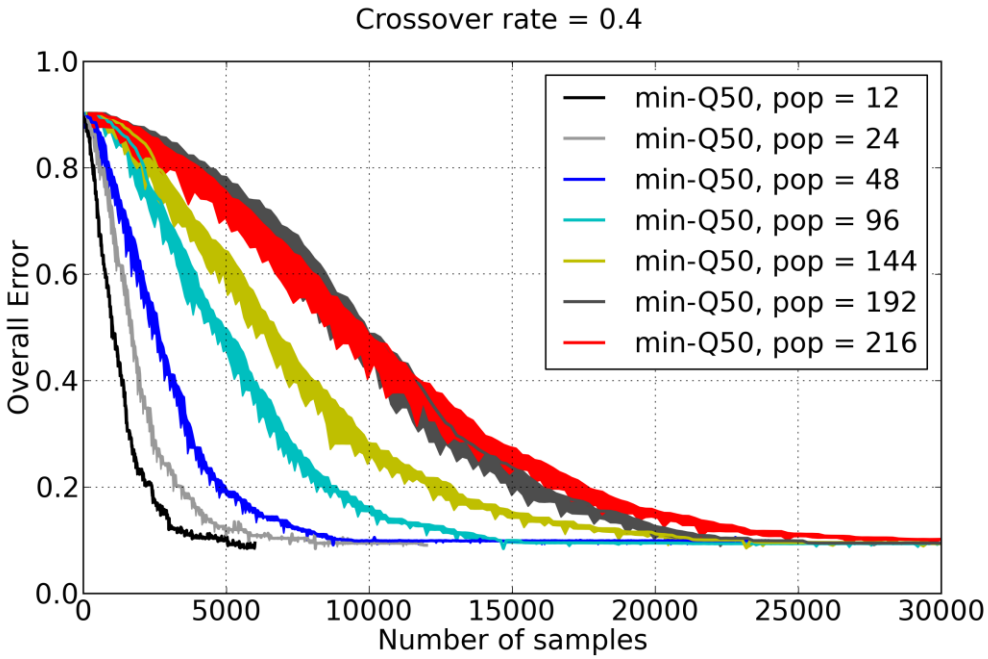


Figure A10 The overall error minimum and the median distribution over the overall number of investigated samples within 500 generations for runs with the constant population crossover rate of 0.4 and different population size.

Figure Captions

Figure 1 Activity chart of the genetic algorithm

Figure 2 Initial populations of four mechanism representations (chromosomes) for a) a full-mechanism initialization and b) mechanisms with only one reaction missing

Figure 3 A single-point crossover between two parents

Figure 4 One-directional mutation

Figure 5 Evaluation function terms for a) ignition delay time, b) steady-state temperature, c) number of reactions and d) computational time.

Figure 6 Overall score for reduced mechanisms during evolution

Figure 7 Number of reactions in the reduced mechanisms (chromosomes)

Figure 8a Comparison of the temperature and the species profiles for the full and the reduced GRI-Mech 3.0 for the reactor.

Figure 8b Comparison of the temperature and the species profiles for the full and the reduced GRI-Mech 3.0 for the flame.

Figure 9 Evolution of the reduced mechanisms in 1000 generations.

Figure 10 Evolution of the overall score of the reduced mechanisms in 1000 generations.

Figure 11 Simulated temporal variation of temperature and important iron and hydrogen species concentrations for a homogeneous constant-pressure reactor based on the full and the reduced mechanism.

Figure 12 Simulated spatial variation of temperature and important iron and hydrogen species concentrations for a freely-propagating flame based on the full and the reduced mechanism.

Figure A1 Overall error convergence over 100 generations, for the case of the full-mechanism initialization (left) and "one reaction missing" initialization (right).

Figure A2 Overall convergence with the two-directional mutation (sigmoid normalization of the runtime and the number of reactions).

Figure A3 Overall convergence with the one-directional mutation (sigmoid normalization of the runtime and the number of reactions).

Figure A4 Linear, square and sigmoid normalization function $f(x)$, where x stands for the cost parameter.

Figure A5 The overall error and the remaining reactions over 500 generations for the linear normalization of the runtime and the number of reactions.

Figure A6 The overall error and the remaining reactions over 500 generations for the square normalization of the runtime and the number of reactions.

Figure A7 The overall error and the remaining reactions over 500 generations for the sigmoid normalization of the runtime and the number of reactions.

Figure A8 The overall error over 500 generations for runs with the constant population size of 48 individuals and different crossover rates, p(cross).

Figure A9 The overall error minimum and the median distribution over 500 generations for runs with the constant population crossover rate of 0.4 and different population size.

Figure A10 The overall error minimum and the median distribution over the overall number of investigated samples within 500 generations for runs with the constant population crossover rate of 0.4 and different population size.