Optimization algorithms for the conditions of complex catalytic reactions

K F Koledina\textsuperscript{1,2} and S N Koledin\textsuperscript{3}

\textsuperscript{1}\textit{Institute of Petrochemistry and Catalysis, Russian Academy of Sciences, prospect Oktyabrya 141, Ufa, Bashkortostan, Russia, 450075}
\textsuperscript{2}\textit{Ufa State Petroleum Technological University, Pervomayskaya str. 14, Ufa, Bashkortostan, Russia, 450062}

\textsuperscript{3}\textit{E-mail: koledinakamila@mail.ru}

Abstract. A procedure was developed for the investigation of complex catalytic reactions using optimization algorithms. The first stage is a solution of the inverse problem of chemical kinetics involving the recovery of unknown parameters by single-objective optimization algorithms. In the second stage, based on the kinetic model, we solved the problem of determining the optimal conditions for the reaction resorting to multi-objective optimization algorithms. Parallelization models of computing for solving the multi-objective optimization problem are considered.

1. Introduction
Studies of complex catalytic reactions are faced with problems of optimization. For example, to develop the mathematical model of a chemical reaction, it is necessary to derive kinetic parameters proceeding from the correspondence between the calculated reactant concentrations and experimental data. The developed mathematical model is used to solve the problem of optimization of reaction conditions by the specified objective functions.

2. Study of complex catalytic reactions by mathematical methods
Figure 1 shows the scheme of studying complex catalytic reactions by mathematical methods. First, it is necessary to develop the mathematical model for a catalytic reaction. The homogeneous catalytic reactions and reactions proceeding at moderate pressures are assumed to follow the Arrhenius kinetics \textsuperscript{[1]}. When closed systems are considered, the state in the stationary points obeys the law of mass action \textsuperscript{[2]}. Hence, system components (reactants) obey equations (1).

\[
\frac{dx_j}{dt} = \sum_{i=1}^{J} v_{ij} w_j (k_j, x_i), \quad i = 1, ..., I
\]

starting conditions: \( x_i(t = 0) = x^0_i \). Where \( t \) is a time, min; \( v_{ij} \) are stoichiometric coefficients; \( J \) is a number of steps; \( x_i \) is a concentration of a reactant, mol/l; \( I \) is a number of compounds; \( w_j \) is a rate of j-
The $k_j$ is a rate constant of steps (reduced), 1/min; $E_j$ is an activation energy of reactions, kcal/mol; $T$ is a temperature, K; $k_j^0$ are pre-exponential factors, 1/min.

System (1) represents the Cauchy problem [3]. By forwarding, the problem is meant determination of the unknown concentrations of substances as functions of time, with the rate constants being specified. The inverse problem implies the derivation of rate constants such that the solution of system (1) with these rate constants would give $x_i$ values as close as possible to experimental results. Every inverse problem is a set of forwarding problems with specified rate constants [4-6].

The parameters of kinetic equations, that is, rate constants and activation energies, should be found from the minimization conditions of functional (2) [7].

$$\min, \sum_{p=1}^{P} \sum_{i=1}^{I} \gamma_i (x_i^p - x_i^{0}) \rightarrow \text{min},$$

where $x_i^p$ and $x_i^{0}$ are the experimental and calculated values of the concentrations of the components, $\gamma_i$ is weighting factor, $I$ is a number of substances, $P$ is a number of measuring points.

The developed mathematical model can be used to determine the optimal reaction conditions. Depending on the specified objective functions of optimization, one of the optimization problems is solved (Fig. 1):

a) If one objective function has been defined, a single-objective optimization problem is solved.

b) If there is more than one mutually independent objective function of optimization, a multi-objective optimization (MOO) problem is solved.

c) If it is possible to control the reaction of interest with time and influence the considered criteria, the optimal control problem is solved.

The solution to the above optimization problems represents the optimal reaction conditions.

3. Single-objective optimization algorithms

The optimization problems are solved using the following single-objective optimization algorithms: global search, Hooke-Jeeves method, genetic algorithm, simulated annealing, and others [8]. These
algorithms may be both local and global. The global search algorithm is based on gradient calculation methods. The genetic algorithm mimics the principles of biological evolution. The simulated annealing algorithm mimics the physical annealing process, aiming at improving the current minimum, with the search volume being gradually reduced.

In solving the optimization problem in chemistry, the key difficulty is that all theoretical optimization studies have separately addressed a particular criterion (product yield). Then the next criterion was optimized (such as the consumed amount of reactants) and so on. Two different conditions of a chemical reaction were selected (or more conditions, if there were more than two optimized parameters). These states provided an answer to the optimization issue [9, 10]. It is noteworthy that reaction conditions were different in these states; in a sense, they were extremal, which indicated the impossibility of the chemical reaction in industry. Therefore, subsequently, in the design of the reactor and the production process as a whole, these conditions were neglected. Some average characteristic was chosen as a compromise for the quality criteria. The situation was even more complicated during practical implementation where the experimental integrity condition is virtually unattainable in a laboratory. Thus, the benefit of optimization nearly vanished for laboratory conditions. Therefore, it is relevant to formulate and solve the multi-objective optimization problem.

4. Multi-objective optimization algorithms

The mathematical optimization methods in chemical engineering have been developed by quite some Russian and foreign researchers such as R. Aris [11], V. V. Kafarov, A. I. Boyarinov [9], V. I. Bykov [12], M. G. Slinko [13], and others. These researchers consider and utilize various optimization approaches: 1) investigation methods of the functions of classical analysis; 2) methods based on the use of Lagrange undetermined multipliers; 3) variational calculus; 4) dynamic programming; 5) maximum principle; 6) linear programming; 7) non-linear programming; and 8) geometric programming. The optimization methods in chemical engineering are mainly devoted to single-objective optimization against the product yield criterion. Meanwhile, multi-objective optimization is faced with the need to attain a compromise between the optimization criteria. These approaches are addressed by both Russian and foreign authors. The authors of [14] consider a method for solving the Pareto approximation problem using a genetic algorithm (NSGA-II), which provides a plausible accuracy for complicated problems. Modifications of the predator-prey algorithm were considered [14]. Several known test and practically significant multi-objective optimization problems were used to compare the efficacy of Pareto approximations using a genetic algorithm and particle swarm, ant colony, and bee colony optimizations in combination with various fitness function generation methods.

The multi-objective optimization problem implies the solution domain located among the Pareto solutions [15]. The decision-maker is interested in a compromise solution, but it should be the best one according to mathematical calculations. The points that belong to the solution should be unimprovable, in other words, non-dominated [16].

The essence of solving the MOO problem using probing algorithms (e.g., the network algorithm) is to split the domain of definition into small cells, find the solution of objective functions for each cell, and identify the non-dominated points (Pareto front) and the related Pareto set [17]. A drawback of the algorithm is that it requires high computing power and long computing time. As the number of parameters increases, the network algorithm is hardly applicable, as the expenses considerably increase. This difficulty can be resolved by using evolutionary multi-objective optimization algorithms.

Vector estimation is one of the early evolutionary algorithms for solving multi-objective optimization problems [16]. Here the use of vector fitness function is proposed to choose the solution points (individuals). The step of selection of individuals is implemented as a cycle in which the appropriate part (a fraction of population or subpopulations) is selected every time considering each of the q criteria. Then the whole population is completely mixed and crossover and mutation operators are applied. The vector method shown in Fig. 2 ensures the survival of the best individuals for each criterion and simultaneously increases the probability of the multiple selections of the best individuals.
The evolutionary algorithms were developed towards the procedure of ranking of individuals [18-20]. The non-dominated sorting genetic algorithm is considered to be the most efficient ranking-based method of the late 20th century (NSGA) (Srinivas и Deb, 1994 г.) [21, 22]. The first non-dominated points (front) are considered and are matched by some value of fitness function. Then these points are decomposed over these values and are excluded from further analogous procedures. After each stage, every current non-dominated point is associated with the minimum decomposed value of the fitness function (Fig. 3).

The a priori algorithms reduce the multi-objective optimization problem to the single-objective optimization problem by specifying weighting coefficients for partial objective functions (3)

$$F(X) = \sum_{i=1}^{\ell} \eta_i f_i(X),$$

where $\eta_i$ is a the specified weighting coefficient, $f_i$ are partial objective functions, $F$ is a vector of objective functions, $X$ is a vector of control parameters.

The further development of multi-objective optimization algorithms was focused on the upgrading of the previously developed algorithm. The algorithms SPEA2 [23], PESA-II [24], and Non-dominated Sorting Genetic Algorithm II NSGA-II [25, 26] were developed. The attention was concentrated on increasing the diversity of individuals and applying elitism for the preservation of the best solutions. As compared with other methods, NSGA-II proved to be most accurate for the determination of the Pareto dominant points. This algorithm also has drawbacks in the case of an increasing number of criteria.

At the beginning of the algorithm, each individual is assigned a rank according to ranking principles. The non-dominated points have the first rank, the points that are dominated only by first-rank points have the second rank, and so on. The crowding of the obtained individuals is evaluated;
the greater the distance between them, the higher the population diversity. In every iteration, the offsprings are selected considering the rank and crowding (proximity) of individuals. Subsequently, in the iteration, the best points are chosen based on crossover and mutation, thus ensuring the diversity of the next population. The parents and offsprings are combined into one population corresponding to the best solutions, and so on.

The efficacy of Pareto approximation is evaluated considering the quality of the obtained Pareto approximation and the time expenditure [16, 27-29]. The quality of Pareto approximation can be assessed using the following characteristics:

1) Uniformity of distribution of solutions (the average distance between the solutions). This is defined as the Euclidean distance of all pairs of solutions from the obtained crowds.

$$\bar{d}(A) = \frac{1}{|C_1| \times |C_2|} \sum_{i \in C_1, j \in C_2} d(i, j),$$ (4)

where \(i \in C_1, j \in C_2\) pairs of solutions are aggregations, agglomerations, \(C_1, C_2\) - clusters and \(d(i, j)\) is the Euclidean distance between solutions \(i\) and \(j\).

2) Average spread as a measure of evenness of the distribution of solutions.

$$I_s(A) = \sqrt{\frac{1}{|A|-1} \sum_{j=1}^{|A|} \left( d - \bar{d} \right)^2},$$ (5)

where \(|A| = |C_1| + |C_2|\) is the size of the solution set.

3) Size of the found solution set (number of the set elements). The size of the resulting set of solutions should be maximized.

$$|A| \rightarrow \max.$$ (6)

4) Length of the solution. This criterion defines the length of the Pareto approximation in the space of objective functions over the whole set of optimization criteria.

$$I_{de}(A) = \left\{ \sum_{k=1}^{j} \max_{j \in [k,A]} \left| f_j(X^A_k) - f_j(X^A_i) \right| \right\} \rightarrow \max$$ (7)

where \(f\) are objective functions; \(F\) is the set of objective functions.

These characteristics are applicable for evaluation of the Pareto set (genotype) – optimal values in the set of variable parameters and Pareto front (phenotype) – optimized values in the set of objective functions.

5. Parallelization of the computational process for multi-objective optimization

During the last decades, numerous effective evolutionary algorithms have been proposed for multi-objective optimization, taking into account all scientific advances in the Pareto dominance and genetic algorithms; furthermore, the computing resources have considerably enhanced, which allows conduction of high-performance computations in acceptable periods [30-32].

The parallelization of algorithms for multi-objective optimization is based on decomposition and structuring of the population (the set of possible solutions). That is, splitting of the initial population into several subsets (subpopulations). Decomposition can be accomplished by various procedures. The parallelization procedures depend on the parallelization model. The most popular parallelization models are as follows: the island model, global client/server type model, and cellular model.

The island model is most popular for parallelization of the multi-objective optimization computing (Fig. 4).

$$|S| = \bigcup_{i=1}^{\left| P \right|} |S_i|,$$ (8)

$$S = \bigcup_{i=1}^{\left| P \right|} S_i,$$ (9)
where $S$ is a multipopulation, $S_i$ are subpopulations (islands), $|P|$ is the number of processors used.

![Figure 4. Island model of parallelization.](image)

The multi-population of the variable parameters is split into subpopulations based on the number of available processors according to (8). (9). According to Fig. 4, the individuals are distributed over several isolated islands. These subpopulations will develop independently, with the subsequent synchronization of the results.

The global model is presented according to the client/server type (Fig. 5). The server implements the optimization algorithm, while clients solve the direct problem and estimate the minimization functional, which requires a lot of computing resources.

![Figure 5. Global model of parallelization.](image)

In the cellular model, subpopulations consist of single individuals. In each time instant, a subpopulation can interact only with the neighboring individuals. The neighborhood is defined as a regular network, which is shown in Fig. 6.

![Figure 6. Cellular model of parallelization.](image)

### 6. Conclusion

Thus, studies of complex catalytic reactions by mathematical methods are faced with optimization problems, which require the use of appropriate algorithms. An optimization problem is involved both in a solution of the inverse problem of chemical kinetics (recovery of unknown parameters) and in the
elucidation of the optimal reaction conditions based on the kinetic model. The application of optimization algorithms is also relevant to the problems of chemical kinetics. The optimization is especially valuable for the analysis of new chemical reactions.

7. References
[1] Dimitrov V 1982 Simple kinetics (Novosibirsk: Science and Life) p 380
[2] Gubaydullin I, Koledina K and Sayfullina L 2014 Eng. J 18(1) 13-24
[3] Nurislamova L, Gubaydullin I, Koledina K and Safin R 2016 React Kinet Mech Cat 117(1) 1-14
[4] Koledina K and Gubaidullin I 2016 Rus J Phys Chem A 90(5) 914-921
[5] Zainullin R, Koledina K, Akhmetov A and Gubaidullin I 2017 Kinetics and Catalysis 58(3) 279-289
[6] Koledina K, Koledin S, Shchadneva N and Gubaidullin I 2017 Rus J Phys Chem A 91(3) 444-449
[7] Koledina K, Koledin S, Schadneva N, Mayakova Y and Gubaydullin I 2017 React Kinet Mech Cat 121(2) 425-428
[8] Krotov V 1990 Fundamentals theory of optimal control (M.: Higher school) p 429
[9] Boyarinov A and Kafarov V 1975 Methods of Optimization in Chemical Technology (Moscow: Chemistry) p 576
[10] Kramers H and Westerterp K 1967 Chemical reactors. Calculation and management (M.: Chemistry) p 263
[11] Aris R 1963 Optimum design of chemical reactors (M.: Foreign lit.) p 238
[12] Bykova V and Juravlev V 2002 Modeling and optimization of chemical-technological processes (Krasnoyarsk: CPI KSTU) p 298
[13] Slinko M 2007 Theoretical Foundations of Chemical Technology 41(1) 16-34
[14] Weiyang T, Souma Ch and Achille M 2013 Structural and Multidisciplinary Optimization 47(3) 367-388
[15] Goldstein A 2015 Optimization in the environment of MATLAB: study guide (Perm: Publishing Perm.Nat. researches Polytechnic University) p 192
[16] Karpenko A 2016 Information and Mathematical Technologies in Science and Management 2 8-17
[17] Koledina K, Koledin S and Gubaydullin I 2017 CEUR Workshop Proceedings 1966 5-9
[18] Schaffer D 1985 In Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms (Hillsdale: NJ) 93-100
[19] Sobol I and Statnikov R 2006 The Choice of Optimal Parameters in Problems With Many Criteria: Textbook. Manual for Universities (Moscow: Drofa) p 175
[20] Abraham A, Jain L and Goldberg D 2005 Evolutionary multiobjective optimization: theoretical advances and applications (New York: Springer Science) p 302
[21] Horn J, Nafpliotis N and Goldberg D 1994 Evolutionary Computation (CEC) IEEE Congress 82-87
[22] Srinivas N and Deb K 1994 Evolutionary computation 2(3) 221-248
[23] Zitzler E, Laumanns M and Thiele L 2002 Evolutionary Methods for Design Optimisation and Control with Application to Industrial Problems EUROGEN 3242(103) 95-100
[24] Corne D, Jerram N, Knowles J and Oates M 2001 GECCO 2001: Proceedings of the Genetic and Evolutionary Computation Conference 283-290
[25] Deb K, Mohan M and Mishra S 2003 Evolutionary Multi-Criterion Optimization 2632 222-236
[26] Baynazarova N, Koledina K and Pichugina D 2016 CEUR Workshop Proceedings 1576 425-431
[27] Bischl B, Mersmann O, Trautmann H and Preuß M 2012 Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference (New York, USA) 313-320
[28] Kerschke P 2014 EVOLVE A Bridge between Probability, Set Oriented Numerics and Evolutionary Computation 115 131
[29] Munoz M, Kirley M and Halgamuge S 2015 IEEE Transactions on Evolutionary Computation
Acknowledgments

This research was performed due to the Russian Scientific Fund grant (project No. 19-71-00006).