OPTIMIZATION OF THE GAS TRANSPORT IN PIPELINE SYSTEMS

Anton Sedliak—Tibor Žáčik

ABSTRACT. The purpose of this work is to design a suitable methodology to solve selected optimization tasks from the field of gas transport in pipeline systems. The modifications of evolution strategies algorithm to solve such optimization problems was developed. Testing of algorithm has been realized by a software implementation on the model of the real transmission pipeline system.

1. Introduction

According to predictions of renowned institutions it is expected that natural gas will be one of the most important energy resources in the future. Findings of International Gas Union [1], The European Union of the Natural Gas Industry [2] or U.S. Energy Information Administration [3] anticipate approximately linear increase in the gas consumption. The increasing demand leads to growing gas production and therefore growing transport to customers. Thus transport costs are rising up and it is necessary to optimize the effectiveness of the gas transportation.

In this paper, we are focusing on the problems connected with high pressure gas transmission systems (the gas pressure in pipelines is higher than 0.4 MPa), namely, on problems related to finding maximal flow through the network, minimal energy or gas consumption, minimal operational costs at requested transport conditions, and problems leading to multi-objective optimization.

The aim of this paper was to find an appropriate methodology for solving selected optimization tasks arising during gas transportation. We propose a modified algorithm of evolution strategies, which can solve a large class of steady-state optimization problems. The software implementation of the algorithm was tested on the model of real gas transmission network.

2. Related works

Steady-state optimization problems in gas pipelines transportation comprise a relatively well-known area of research. Coherent overview can be found in [4]. In almost every case, the problem is defined as a problem of mathematical programming, which is solved by various deterministic and stochastic optimization methods. For instance P r a t t and W i l s o n in [5] apply the mixed integer linear programming, P a r c e l and R y a n in [6] apply the generalized reduced gradient method (GRG) and F l o r e s - V i l l a r r e a l and R í o s - M e r c a d o in [7] apply GRG for cyclic and non-cyclic pipeline topology.

The most successful approach till now is based on dynamic programming methods (DP). In 1968 authors W o n g and L a r s o n published papers [8], [9], where they for the first time applied DP to solve stationary optimization problems on linear and tree pipeline topologies. First commercially successful program based on DP was developed by Z i m m e r [10] in 1975. Works [11], [12] from the field of chemical engineering inspired to try using DP also for cyclic pipeline topologies. Compact review of development in this field can be found in C a r t e r's work [13]. From the point of view of practical use, the DP based approaches are suitable and widely employed especially for simple problems on linear and tree pipeline topologies. On cyclic pipeline topologies, there is an interesting approach that combines DB with a network-based heuristic [14].

Authors in [15] solved multi-objective optimization with two objectives: maximal outflow and minimal energy consumption. They defined mathematical programming problem and solved it by a particle swarm optimization and a genetic algorithm. Model was tested on real-world problem, however, the network consisted from only four identical parallel compressors with coolers and one pipe.

From our point of view, previously mentioned approaches have some disadvantages. A physical model of gas flow was heavily simplified. It takes into account neither temperature changes due to compression and surrounding environment interaction nor elevation of pipes. But, for instance, in [15] authors showed that almost 25 % of saving energy was achieved by optimal cooling gas after its compression. Moreover, the model of compressor station is usually significantly simplified and therefore markedly inaccurate. Finally, listed approaches were designed to solve specific problems mostly for specific pipeline topologies.

3. Motivation

For better understanding our motivation we now briefly show the basics of the models mentioned earlier. A pipeline system is modeled as a directed graph

$$G = (E, V),$$

where the set of edges E represents components (compressor stations, pipes, valves, check-valves, control-valves) and the set of nodes V represents intersection points of segments, inlet nodes (delivery points) and outlet nodes (gas demanding points). Now consider pressure variables p_v for all $v \in V$ and flow variables q_e for all edges $e \in E$. Further, consider variables f_c and N_c that represent gas consumption and power of all $c \in E_c$ where $E_c \subset E$ is the set of all compressor stations. Finally consider binary variables $s \in \{0, 1\}$ for all components that can be switched on and off (compressors, valves).

There are many conditions which have to be satisfied in form of equalities and inequalities. They represent physical properties of gas flow, physical properties of components or special conditions of concrete solving problem. For example, for all nodes $i \in V$ the first law of Kirchhoff must be observed:

$$\sum_{e \in \delta^+(i)} q_e = \sum_{e \in \delta^-(i)} q_e,$$

where $\delta^+(i)$ means the set of ingoing segments of node i and $\delta^-(i)$ means the set of outgoing segments of node i. Further, for all nodes $i \in V$ we set lower and upper bounds for the pressure $p_i^{\min} \leq p_i \leq p_i^{\max}$ and lower and upper bounds for the gas flow $q_i^{\min} \leq q_i \leq q_i^{\max}$. From the momentum equation, one can derive formula for steady-state flow in one pipe without declination:

$$p_{\text{in}}^2 - p_{\text{out}}^2 = k\,q\,|q|,$$

where p_{in} is the pressure at the inlet point of the pipe, p_{out} is the pressure at the outlet point of the pipe, q is the gas flow through the pipe and the coefficient k represents physical properties of the pipe.

The most complex set of boundaries is for compressor stations. The inlet pressure p_{in_c}, outlet pressure p_{out_c} of a compressor station and the gas flow q_c through the compressor station have to satisfy:

$$(p_{\text{in}_c}, p_{\text{out}_c}, q_c) \in D_c \subset \mathbb{R}^3, \qquad c \in E_c,$$

where D_c is the feasible domain for compressor station c. D_c is non-convex domain determined by working map of all compressor units. Also, for each compressor unit there are several other technical constrains, such as minimum and maximum revolution, power constrains, etc.

For instance, in [16] there are 43 different equalities and inequalities constrains that have to be satisfied by all or some subset of pressure and flow variables. With the model of the Slovak transit pipeline system, where there are 574 edges and 400 nodes, it would be more than 2,000 variables and more than 10,000 constrains to be satisfied. And we still do not consider the gas temperature variables, pipeline elevation, physical properties of surrounding environment, etc. Furthermore, in the work by M ö l l e r [16] all nonlinear relations have to be

linearized to enable using the mixed integer linear programming. To avoid the problem with simplification of physical properties and relations and to reduce dimensionality of optimizing problem, we have developed another methodology described below.

4. The principle of proposed optimization method

Our approach to find optimal solution of steady-state optimization problems is based on dividing problems into two levels: local level—an optimization of compressor station itself, and global level—an optimization of pipeline network as a whole. These two levels lead to the following decomposition for finding solution:

- deterministic part, represented by steady-state simulation, which includes local optimization of compressor stations;
- stochastic part, in our case represented by evolution strategies algorithm (ES), employing results of the deterministic part to assign value of fitness function to candidate solution and verify its feasibility.

In our approach we use steady-state simulation as a complex highly complicated fitness function. There are some assumptions which need to be satisfied. Namely, the steady-state simulation has to be able to calculate steady-state, where the following items are set:

- pressure or flow boundary condition at every inlet and outlet node;
- temperature boundary condition at every inlet node;
- pressure restrictions at all nodes;
- compression ratios of all compressor stations;
- for every type of compressor – its minimum and maximum number, minimum and maximum revolution, surge and choke safety limits;
- power restriction for power unit of every type of compressor;
- ambient temperature that determines restriction of its drive in compressor stations;
- control pressure of every control valve, etc.

Furthermore, for a given set of compression ratios of compressor stations, a steady-state simulation has to be able to find locally optimal state of each compressor station. In other words, for a chosen criterion (energy consumption, gas consumption, operation cost) and given compression ratio on a compressor station, the steady-state simulation has to be able to find optimal numbers and types of running compressor units, its revolutions and their mutual connections.

Last but not least assumption is that the steady-state simulation has to be reasonably fast due to a very high required number of evaluations of fitness function.

For a purpose of this paper we consider the steady-state simulation with local optimization as a black box. For decision parameters, together with all preassigned parameters (boundary conditions, compression ratios, settings of regulators, state of the valves, etc.), this one determines whether these input parameters define feasible solution and assigns a value of fitness function to candidate solution.

We just sum up that we use a steady-state simulation based on a complete one-dimensional model of turbulent gas flow. Its implementation is based on the modified Hardy Cross method, and we achieve an acceleration of the simulation by a hierarchical simplification of a network (a multilevel network method). One can find further details in [17] and [18]. Of course, it is possible to use other implementation of steady-state simulation satisfying all given assumptions.

In the stochastic part, we employ the method of evolution strategies to find a set of decision parameters that minimize (maximize) value of the fitness function.

4.1. Single-objective optimization

By a number of numerical tests, we choose for implementation of ES modified $(\mu + \lambda)$-ES with self-adaptive standard deviations and without correlated mutation. The symbol μ represents the number of parents in population and λ represents the number of offspring in population. The symbol "+" means that all parents and all offspring fight against each other for survival. There is also another representation of selection pressure in ES, denoted by symbol ",", denoting that only new offspring fight against each other for their survival. By "self-adaptive standard deviations" we mean that the standard deviation σ is also evolved.

An individual (a candidate solution) in the ES is represented by a vector of real numbers. If the number of decision parameters is n then the individual will be represented by $2n$ real numbers $X = (\mathbf{x}, \boldsymbol{\sigma})$ where first n numbers represent decision parameters and the rest n numbers appropriate standard deviations.

As it was mentioned before, the value of fitness function for individual X will be obtained by calculation of steady-state simulation together with information whether the parameter $\mathbf{x}$ represents feasible solution or not.

It is suitable to have initial population well distributed over subset of design space representing feasible solutions. In the Fig. 1, there is a fitness function over domain of feasible solutions for particular problem on the Slovak transmission pipeline system, where we restrict the problem to just two dimensions—only two compressor stations (CS 1 and CS 3) are running out of four. Compression ratios $K1$ and $K3$ are on the horizontal axes and on the vertical axis there is
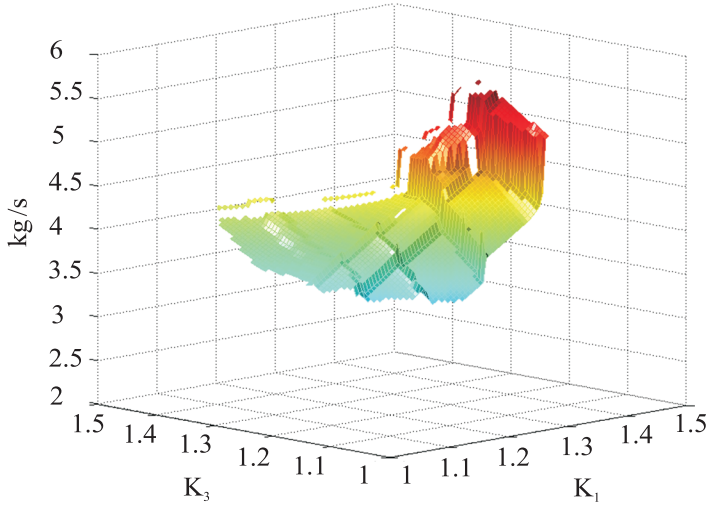
FIGURE 1. Gas consumption of compressor stations CS 1 and CS 3 (in kg/s), while compressor stations CS 2 and CS 4 are bypassed.

gas consumption in kilograms per second. As one can see, the fitness function is quite complicated and non-continuous. In many cases, its domain is too narrow (cf. Fig. 3).

Therefore, there can be a problem to find an initial population by random generating of individuals. But numerical tests show that if an initial generation consists of just one individual, it is sufficient to start evolution without losing search and convergence properties of the algorithm.

Numerical tests also show that, for creating a new individual, an approach of one parent and self-adaptive standard deviation seems to be the most appropriate. The parent is chosen randomly from populations of parents by uniform distribution. Subsequently, the mutation of the standard deviation $\boldsymbol{\sigma}'$ is calculated in the first step from $\boldsymbol{\sigma}$ by

$$\boldsymbol{\sigma}' = \boldsymbol{\sigma} e^{N(0,\,\Delta\sigma)}, \tag{1}$$

where $\Delta\sigma$ is the parameter of the method. S c h w e f e l in [19] recommended to set it to

$$\Delta\sigma = \frac{1}{\sqrt{2\sqrt{n}}},$$

where n is the number of decision parameters. Expression $N(0, \Delta\sigma)$ represents the normal distribution with mean 0 and the standard deviation $\Delta\sigma$. After that we can compute decision parameters for new offspring by formula:

$$\mathbf{x}' = \mathbf{x} + N(0, \boldsymbol{\sigma}'). \tag{2}$$

Now, it is necessary to decide whether the new decision parameters $\mathbf{x}'$ define feasible solution and in that case assign value of the fitness function to this new offspring. This means that we have to run the steady-state simulation with decision parameters $\mathbf{x}'$ and with all other preassigned parameters. If these input parameters do not represent feasible solution, it is necessary to calculate from (1) and (2) new decision parameters $\mathbf{x}'$ and run another steady-state simulation.

It is obvious that a calculation of steady-state simulations is the most time consuming component of suggested optimization approach. When it is not possible to accelerate steady-state simulation by itself, there are fundamentally two ways to accelerate optimization calculation as a whole: by reducing number of required steady-state simulations and by parallel run of steady-state simulations.

- *Optimal generating of candidate solutions:* the goal is to find an optimal modification of ES and optimal value of control parameters of ES algorithm to minimize the required number of steady-state optimizations. To enable comparison between different modifications of ES and different values of control parameters, we have introduced the parameter *successfulness*. It is the ratio between the number of all feasible solutions and the number of all candidates for solution during whole optimization calculations. Using this parameter, various types of offspring production were numerically tested and control parameters were analyzed;
- *Parallel computing:* modern multiprocessor architecture allows running independent steady-state simulations. It substantially accelerates the optimization.

To prevent the algorithm from premature convergence, we have introduced the modification of ES. It is a combination of principle of $(\mu + \lambda)$-ES and (μ, λ)-ES modifications. There is a new control parameter called *maximal age* and every individual has a new integer parameter *age* initially randomly selected from the interval $(\frac{\text{maximal age}}{2}, \text{maximal age})$ by uniform distribution. When a new individual representing feasible solution occurs, it gets a randomly chosen integer representing the age that can be achieved by it. Every time this individual becomes parent in a new generation, its age is decreased by one. When its age reaches zero, it "dies" and cannot become a parent in the next generation.

Since, in general, a value of optimal solution is unknown, as a termination condition is chosen either a total number of generations or an overall computing time. After satisfying one of the terminal conditions, the individual with the best value of fitness function till now will be set as the solution of the optimization process.

4.2. Multi-objective optimization

It is obvious that most of real-world engineering problems lead to multi-objective optimization problems (MOP). For example, in area of stationary optimization problems in gas transmission pipeline systems, it is the common problem to find minimal gas consumption together with preservation of the current line pack (a total mass of gas in a whole pipeline system). Optimal solution which significantly changes the line pack is often inapplicable.

Despite the generality and robustness of our approach, it is necessary to make some modifications that enable us to solve MOP. Representation of an individual does not need any modification, except that the fitness value is no longer represented by only one real number but rather by a vector of real numbers. It is due to the fact that in the case of k criteria, the value of fitness function should be represented by a vector of k real numbers. Due to this, some modification has to be done, namely the process of choosing new parents from population. Usually, we are searching for a set of solutions, called *Pareto optimal* solutions. It has been established by Italian economist Vilfredo P a r e t o.

We will say that feasible solution of MOP **x** *dominates* other feasible solution **y** if, for $i = 1, 2, \ldots, k$, $f_i(\mathbf{x}) \geq f_i(\mathbf{y})$ and simultaneously there exist at least one i such as $f_i(\mathbf{x}) > f_i(\mathbf{y})$. Let's assume P is the subset of set of feasible solutions. Feasible solution that is not dominated by any other feasible solution from P will be called *non-dominated* according to set P. A subset of all non-dominated solutions from set of all feasible solutions is called the *Pareto optimal set* and its image in domain of the fitness function is called the *Pareto front*.

Consider results from various fields of research [20] that solved MOP by using evolutionary and stochastic methods on different real world engineering MOP. A mix of two ideas: elitism and diversification of a population can be observed.

Note: The *elitism* is such a principle that just the best individuals should become parents.

In our single-objective optimization, a new parent is selected according to its fitness function value. In MOP modification we will select parents from non--dominated individuals according to all feasible solutions obtained so far. The set of all non-dominated solutions will be an approximation of the Pareto front. If the number of non-dominated individuals is equal to μ, these individuals become parents. But how to perform selection when the number of non-dominated individuals is smaller or higher than μ? In these two cases we suggest solutions in agreement with the principle of elitism and diversification of population:

- the number of non-dominated solutions is higher than μ: parents are chosen from set of non-dominated individuals to achieve a uniform cover of the Pareto front. It means that we skip non-dominated solutions which have the nearest neighbor the non-dominated solution;

- the number of non-dominated solutions is lower than μ: required parents are added as the nearest feasible solutions to the approximation of the Pareto front.

After satisfying one of the terminal conditions, all non-dominated individuals obtained so far will be the result—the approximation of the Pareto optimal solutions.

5. Numerical tests

In this section we present numerical tests and convergence properties of selected problems related to the real-world transmission pipeline system on a model of the Slovak pipeline system (Fig. 2). It consists of 161 pipes with 2268 km of total length, 400 nodes including approximately 20 inlet and/or outlet nodes, 395 valves and 14 control valves. There are four complex compression stations in line ("gunbarrel" or linear network topology). Every compressor station consists of some number of 6 different types of compressor units that can be connected parallel, serial or parallel-serial.
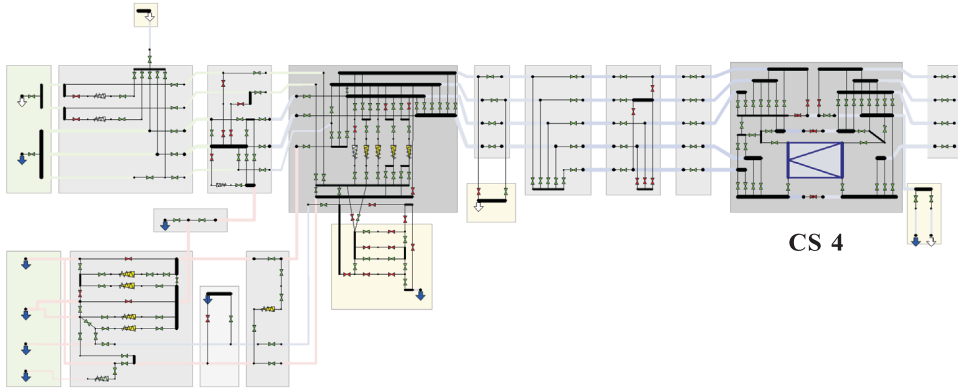

CS 4

FIGURE 2. A segment of a model of the Slovak transmission pipeline system.

Algorithm and its modifications were written in C++ language as an embedded module of the software MARTI Studio. Tests were performed on PC with Intel Core i7-4771 CPU 3.5 GHz processor with 16 GB RAM. All results were processed in Excel and Matlab.

Although the proposed algorithm is able to solve wide class of steady-state optimization problems in gas pipeline transport, we had chosen minimal energy consumption problem to enable a comparison of results with the dynamic programming algorithm.

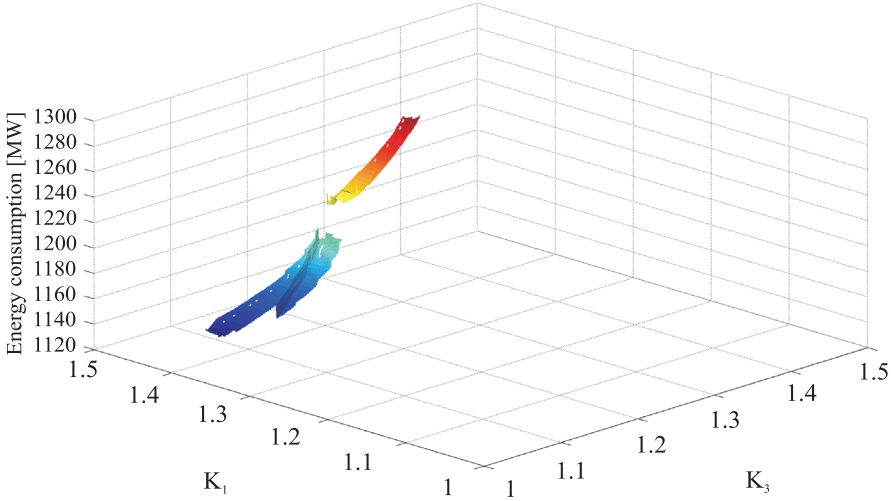## 5.1. Numerical analysis of control parameters for single-objective optimization

FIGURE 3. The section of the fitness function for fixed compression ratios in compressor stations CS 2 and CS 4 (K2 = 1.25, K4 = 1.25).

The proposed algorithm has five basic control parameters: the number of generations, number of parents, number of offspring, maximal age and initial value of the standard deviation σ. Tested problem was to find the minimal energy consumption at fixed transport conditions. The decision parameters are compression ratios of four compressor stations.

The problem simulates the real task and gives us a good picture of typical problems for which the proposed optimization approach was developed. A section of fitness function is showed in the Fig. 3. We can see that the fitness function and its domain of feasible solutions are neither continuous nor smooth. Also, the domain is too narrow.

Different values of control parameters were tested by 100 independent optimization runs with the same conditions. The full results of numerical tests (including the problem of maximal outlet and the multi-objective problem) can be found in the PhD thesis [21].

Number of generations

We performed 100 independent runs for each of following numbers of generations: 5, 10, 25, 50, 75, 100 and 150. Each time control parameters were:

5 parents, 10 offspring, maximal age 10 and starting sigma 0.05. As it was expected, convergence properties of the proposed algorithm were getting better with an increasing number of generations but the overall computation time was increasing as well. With the number of generations 50, the standard deviation of solutions was approximately 0.6 (Table 1). That means that the maximal difference between the best and the worst solution was under 5 ‰ and from the practical point of view it was an acceptable precision. With the number of generations 50, the overall computation time was, on average, less than 3 minutes. Numerical tests indicate that average overall computation time was growing approximately linearly with the number of generations but the standard deviation of overall computation time was growing approximately exponentially. A successfulness of generating a new individual was slightly rising up with the growing number of generations, but the standard deviation of successfulness of generating new individual was growing too. Therefore, the optimal value of the number of generations was between 50 and 100. Although the standard deviation of solutions was still decreasing for value higher than 100 generations, it led to disproportionately long computation times.

TABLE 1. The standard deviation of optimal solutions obtained in dependency on the number of generations.

Numbers of generations	5	10	25	50	75	100	150
Standard deviation [MW]	6.411	5.426	1.622	0.607	0.470	0.224	0.121

Number of individuals in every generation

We performed 100 independent runs for each of following pairs of parents/offspring: 1/1, 1/2, 1/5, 2/2, 2/4, 5/50, 5/10, 5/20, 10/20 and 10/50. Each time the control parameters were: 50 generations, maximal age 50 and 0.05 starting σ. The number of individuals has similar influence on convergence properties as the number of generations. In general, bigger populations lead to better convergence properties but also to longer computation time, which is unfavorable (Fig. 4, Fig. 5). When the number of parents was at least 5 and the number of offspring was at least 10, the standard deviation of solutions was less than 0.75. That means that the maximal difference between optimal solutions was less than 1 %. Furthermore, numerical tests indicate that the number of offspring would be at least twice as large as the number of parents. Successfulness of generating new individuals with growing population was slowly increasing, but it led to more requiring steady-state optimizations and thus to longer overall computation time. It follows that as optimal values we can put the number of parents 5 and the number of offspring 10.
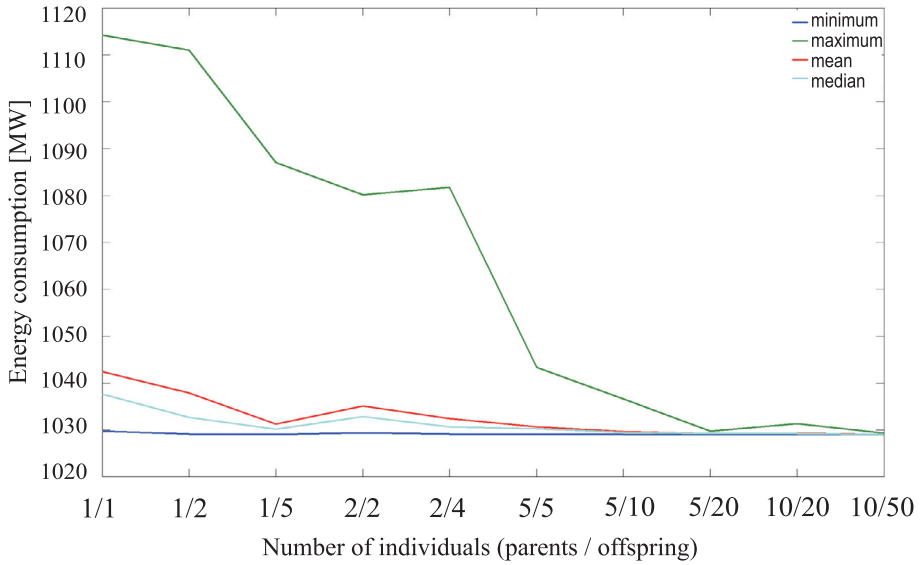
FIGURE 4. Statistical properties of optimal solutions in dependency on the number of individuals.
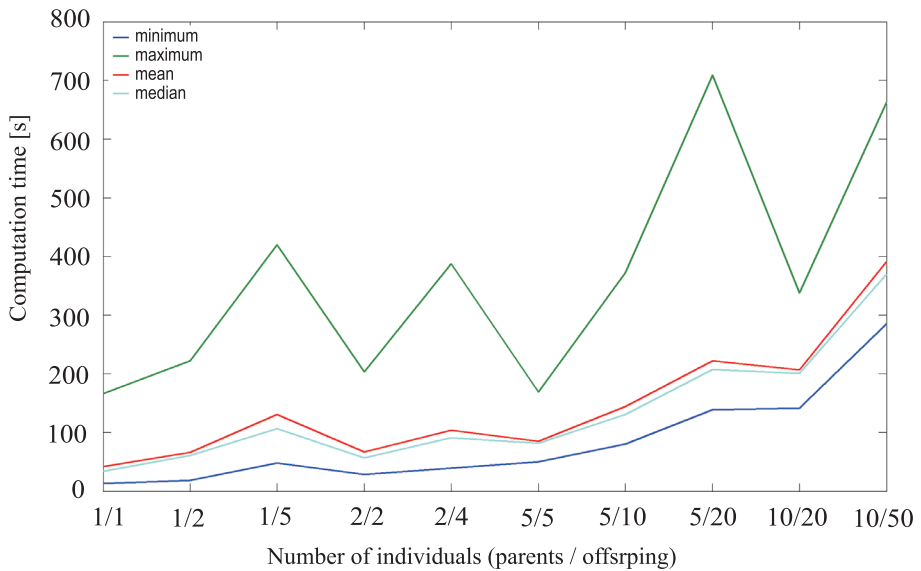

FIGURE 5. Statistical properties of overall computation time in dependency on the number of individuals.

Maximal age of parents

We performed 100 independent runs for each of the following numbers of maximal age: 2, 5, 10, 20, 30, 50 and 75. Each time the control parameters were: 75 generations, 5 parents, 10 offspring and 0.05 starting σ. Numerical tests have shown that when the maximal age of parents was less than 5 then convergent properties of the proposed algorithm were significantly getting worse (Table 2). On the other hand, maximal age of parents greater than 20 significantly worsened the standard deviation of overall computation time (Table 3). From the statistical point of view, there was no influence of maximal age of parents on successfulness of generating new individuals. It follows that an optimal value of maximal age seems to be around 10 % of the number of generations.

TABLE 2. The standard deviation of optimal solutions in dependency on the maximal age of parents.

Maximal age of parents	2	5	10	20	30	50	75
Standard deviation [MW]	1.676	0.598	0.460	0.312	0.385	0.323	0.470

TABLE 3. The standard deviation of overall computation time in dependency on the maximal age of parents.

Maximal age of parents	2	5	10	20	30	50	75
Standard deviation [s]	36.33	35.99	36.61	52.87	60.30	54.25	65.82

Initial value of the standard deviation

We performed 100 independent runs for each of following numbers of initial σ: 0.005, 0.01, 0.025, 0.05, 0.075, 0.1 and 0.5. Each time the control parameters were: 75 generations, 5 parents, 10 offspring and 10 maximal age. For a value of initial σ it is necessary to take into account a range of feasible values of design parameters. In this particular problem, we had four design parameters that represent compressor ratios on four compressor stations. Every design parameter was limited by interval $[1, 1.5]$. Therefore, according to (2), it was not suitable to use an initial value of σ higher than 1. It follows from numerical tests that the standard deviation of optimal solutions (Table 4) is reasonable for initial sigma from interval $[0.025, 0.5]$. Overall computation time was slowly increasing (Fig. 6) with increasing value of initial σ. For σ greater than 0.1 overall computation time began to increase faster also with the standard deviation of overall computation time. The successfulness of generating

new individual was slowly decreasing for growing value of initial σ. When we sum up these observations, as an optimum value of initial σ can be considered value 0.05. It is 1 % of range of boundary interval for design parameters.

TABLE 4. The standard deviation of optimal solutions in dependency on value of initial σ.

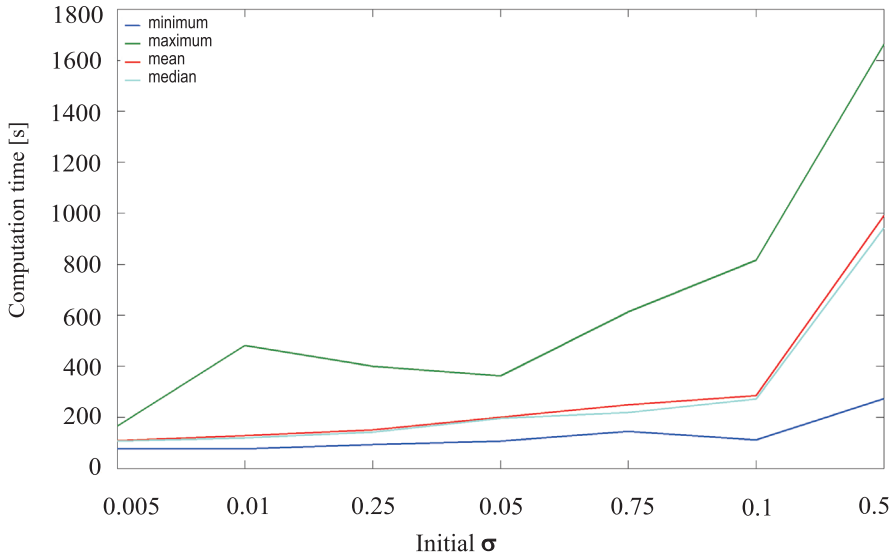| Initial value of σ	0.005	0.01	0.025	0.05	0.075	0.1	0.5
Standard deviation [MW]	6.442	6.038	1.050	0.378	0.294	0.484	2.354

FIGURE 6. Statistical properties of overall computation time in dependency on initial σ.

Summary

When one set the recommended values of control parameters: 75 generations, 5 parents, 10 offspring, 0.05 initial σ and 10 maximal age, then 100 independent runs of proposed algorithm typically look like in Fig. 7, where the value of best solutions over generations can be observed. In the Table 5, there are statistical properties of considered 100 independent runs. In comparison to an optimal solution obtained by algorithm of dynamic programming, we obtained solutions with precision of 5 significant digits. Furthermore, the proposed algorithm and

116

the algorithm of dynamic programming found exactly the same numbers and types of running compressor units and their mutual connections in compressor stations. Differences are only in revolutions of running compressors, but the maximal difference was 28 revolutions per minute.

TABLE 5. Statistical properties of optimal solutions, overall computation time and successfulness after 100 independent runs with control parameters as follows: 75 generations, 5 parents, 10 offspring, 0.05 initial σ and 10 maximal age.

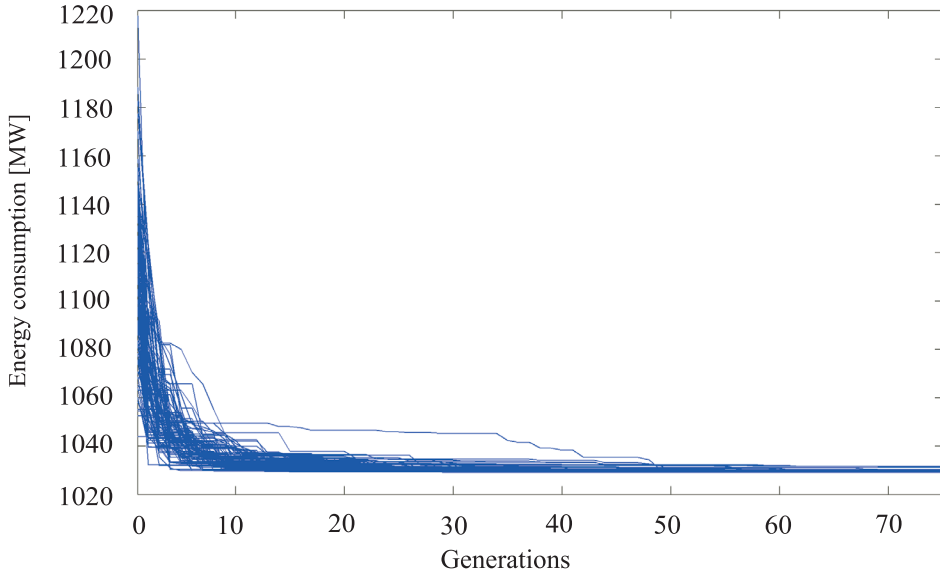|	Energy consumption [MW]	K1	K2	K3	K4	Overall time [s]	Success-fulness [%]
Minimum	1029.00	1.439	1.301	1.139	1.439	102.3	8.35
Maximum	1031.49	1.440	1.313	1.166	1.440	306.6	27.47
Mean	1029.32	1.439	1.307	1.157	1.439	167.5	16.50
Median	1029.18	1.440	1.306	1.157	1.440	163.1	16.24
Std. dev.	0.4604	7.8e-5	2.6e-3	4.9e-3	7.8e-5	36.6	3.43

FIGURE 7. Values of the best candidates for solution in dependence on generations for 100 independent runs with control parameters as follows: 75 generations, 5 parents, 10 offspring, 0.05 initial σ and 10 maximal age.

6. Conclusion

In this paper an implementation of the stochastic population method of evolution strategies has been considered for solving a wide class of stationary optimization problems in the field of a gas transport in transmission pipeline systems. Namely, we developed it with focus on following problems:

- finding of maximal outflow;
- minimization problems (e.g., finding minimal gas consumption at certain transport conditions);
- multi-objective optimization (e.g., minimal energy consumption and prescribed line pack).

In the well-known approaches (in most cases, it is the model of nonlinear programming or mixed-integer linear programming) a treatment of systems with complicated compressor stations leads to a problem with too many design parameters. This could lead to enormous increase of computation time and to non-practical solutions.

One of the basic results of this work is the effective elimination of large number of design parameters and its substitution by smaller—even optimal—number of design parameters. Specifically, instead of searching for optimal numbers and types of running compressor units, its revolutions and their mutual connections in compressor stations directly, we try to find out a set of compression ratios for all compressor stations. As a consequence, this approach leads to nontrivial acceleration of calculation.

Further properties of an implementation of the proposed algorithm:

Implementation in a form of end product

Proposed algorithm was implemented in C++ programming language as an embedded module in software MARTI Studio—a general tool for solving tasks in gas industry. It was designed to allow users to define their own design vector and fitness function for individual optimization problems.

Parallel computing

Implementation on multiprocessor architecture enables to keep computation time small enough to preserve its usability for practical use.

Heuristics for an improvement of convergence properties

To improve convergence properties of proposed algorithm we have included the notion of maximal age. It is combination of $(\mu + \lambda)$-ES and (μ, λ)-ES and had been developed to prevent the algorithm from premature convergence.

Testing of proposed algorithm on real-world problems and numerical analysis of control parameters

Due to long-term cooperation with the gas transport industry we had possibility of testing algorithm on models and data from practice and solving practical problems. We had performed complex numerical analysis of control parameters on these problems to estimate optimal values of control parameters. The results were verified by dynamic programming algorithm as it guarantees (see [4]) reaching the global optimum.

Modifications of algorithm for solving multi-objective optimization problems

It is obvious that real-world optimizing problems are in most cases of a multi-objective nature. Therefore it is natural to modify the proposed algorithm to enable solving such multi-objective problems.

Universality of proposed solution

It is easy to modify the proposed algorithm. In this paper we have introduced a method based on an algorithm of evolution strategies. But the principle of proposed algorithm enables us to use other global optimization methods as well. For instance, we start experiment with algorithm of differential evolution and with hybrid method: evolution strategies are utilized to create population that can serve as the initial generation for differential evolution.

REFERENCES

[1] *International Gas Union, Global Vision for Gas, The Pathway Towards a Sustainable Energy Future*, June 2012.

[2] *The European Union of the Natural Gas Industry, Long-term Outlook for Gas to 2035*, October 2013.

[3] *U.S. Energy Information Administration, Annual Energy Outlook 2012 with Projection to 2035*, June 2012.

[4] RÍOS-MERCADO, R. Z.—BORRAZ-SÁNCHEZ, C.: *Optimization Problems in Natural Gas Transmission Systems: A State of the Art Survey*, Applied Energy **147** (2015), 536–555.

[5] PRATT, K. F.—WILSON, J. G.: *Optimization of the operation of gas transmission systems*, Transactions of the Institute of Measurement and Control (1984), 261–269.

[6] PERCELL, F. B.—RYAN, M. J.: *Steady-state optimization of gas pipeline network operation*, in: PSIG Annual Meeting, article no. PSIG–8703, Tucson, U.S.A., 1987.

[7] FLORES-VILLARREAL, H. J.—RÍOS-MERCADO, R. Z.: *Computational experience with a GRG method for minimizing fuel consumption on cyclic natural gas networks*, Computational Methods in Circuits and Systems Applications (2003), 90–94.

[8] WONG, P. J.—LARSON, R. E.: *Optimization of natural-gas pipeline systems via dynamic programming*, IEEE Transactions on Automatic Control, AC **13** (1968), 457–481.

[9] WONG, P. J.—LARSON, R. E.: *Optimization of tree-structured natural-gas transmission networks*, J. Math. Anal. Appl. **24** (1968), 613–326.

[10] ZIMMER, H. I.: *Calculating optimum pipeline operations*, Presented at the AGA Transmission Conference, 1975.

[11] WILDE, D. J.—LODERER, M.—ZAJAC, P.: *Strategies for optimizing macrosystems*, Chemical Engineering Progress **61** (1965), 86–93.

[12] ARIS, R.—NEMHAUSER, G. L.: *Optimization of multistage cyclic and branching systems by serial procedures*, A.I.Ch.E. Journal, **10** (1965), 913–919.

[13] CARTER, R. G.: *Pipeline optimization: Dynamic programming after 30 years*, in: PSIG Annual Meeting, article no. PSIG–9803, Denver, U.S.A., 1998.

[14] RÍOS-MERCADO, R. Z.—KIM, S.—BOYD, E. A.: *Efficient operation of natural gas transmission systems: a network-based heuristic for cyclic structures*, Comp. Oper. Res. **33** (2013), 2323–2354.

[15] HADDAD, J.—BEHBAHANI, R. M.: *Optimization of natural gas transmission system*, Internat. J. Comput. Appl. **66** (2013) 35–42.

[16] MÖLLER, M.: *Mixed Integer Models for the Optimization of Gas Networks in the Stationary Case*, Dissertation Thesis, University of Darmstadt, 2004.

[17] HAJOSSY, R.—HUČEK, A.—ŽÁČIK, T.: *Acceleration of the computations in gas transport optimization*, in: Conference Proceedings of Inter-Academia, Comenius University and Union of Slovak Mathematicians and Physicist, Bratislava, 2002, pp. 90–94.

[18] ŽÁČIK, T.—SOMORA, P.—HAJOSSY, R.: *Modeling and optimizing in Slovak gas transmission network*, in: PSIG Annual Meeting, article no. PSIG–1319, Prague, Czech Republic, 2013.

[19] BÄCK, T.—HOFFMEISTER, F.—SCHWEFEL, H. P.: *Survey of evolution strategies*, in: The Fourth International Conference on Genetic Algorithms, San Mateo, CA, 1991.

[20] COELLO COELLO, C. A.: *A short tutorial on evolutionary multiobjective optimization*, in: The First International Conference on Evolutionary Multi-Criterion Optimization, Lecture Notes in Comput. Sci. **1993** (2001), 21–40.

[21] SEDLIAK, A.: *Optimization of the Fluid Transport in Pipeline Systems*, Dissertation Thesis, Comenius University in Bratislava, 2016.

Department of Applied Mathematics
Mathematical Institute
Slovak Academy of Science
Štefánikova 49
SK–814-73 Bratislava
SLOVAKIA

E-mail: Anton.Sedliak@mat.savba.sk
　　　　 Tibor.Zacik@mat.savba.sk