

# IOSO OPTIMIZATION TOOLKIT – NOVEL SOFTWARE TO CREATE BETTER DESIGN.

Egorov, I.N., Kretinin, G.V., Leshchenko, I.A. and Kuptzov, S.V.

IOSO Technology center, Moscow, Russia  
e-mail: [company@iosotech.com](mailto:company@iosotech.com)

## Abstract

Optimization has always been around. Designing of new object or system in any case leads to the necessity to provide minimum or maximum value of some indicator characterizing the efficiency of the system. To find optimum solution one can go in two ways: do a series of parameters variations and pick the best solution from the results, or employ numerical optimization algorithms. The latter way is much more preferable for design optimization problems with more than two variables. It allows a designer to find mathematical optimum faster and more accurately.

IOSO optimization software package facilitates better design by allowing search for object function optimum in design space taking into account design constraints. All IOSO software packages use response surface technology in the algorithms of search for optimum. Its response surface implements a complimentary set of approximation algorithms. The strategy of response surface building is set up adaptively during search for optimum, depending on features of object being optimized and current state of search.

The paper presents main features of IOSO optimization software and examples of some practical problems solution. In particular, it demonstrates the example of solution of real-life optimization problem for parameters of gas turbine engine being developed for a commercial aircraft.

## Introduction

Creation of any complex technical object or system assumes search of combination of parameters for that system that would provide maximum or minimum values of one or more efficiency indicators. To accomplish this one should, at the designing stage, conduct a complex research aimed to evaluate influence of design parameters on the object's efficiency. A distinctive feature of solving such tasks consists in the fact that the object can be investigated at different levels of complexity. For aircraft engine, for example, both internal efficiency indicators (such as specific fuel consumption or specific weight) and higher-level indicators (such as lifecycle cost or direct maintenance charges for aircraft equipped with the engine being

investigated) can be used. Obviously, the most reliable results can be obtained by using high-level criteria, which take into consideration efficiency of the whole system and not of one of its particular element. Hence, to optimize parameters of an object that is part of a higher-level system, a multidisciplinary approach should be used, that is the one that consider multiple disciplines of analysis when designing the system.

Conduction of any optimization research requires at least two components. First, one needs the object itself or its analogue (a mathematical model) that will make it possible to quantitatively evaluate how design parameters affect efficiency of the object. Second, an appropriate technique for optimum solution search is required.

Multidisciplinary optimization problems are distinguished from the others by the fact that mathematical analysis models for different disciplines are usually developed by different authors, and are implemented using different algorithmic programming languages and hardware platforms. These factors significantly complicate solution of such problems since implementing the scheme "mathematical model - optimization algorithms" inevitably encounters with technical or organizational problems. This explains why software products aimed to search for extremum using different-class mathematical models are being developed so intensively. Achieving this goal assumes proper customizing data exchange between mathematical model and the optimization program. The well-known software products on the market are iSIGHT by Engineous Software Co, VisualDoc by Vanderplaats Research & Development, Inc. (VR&D), Pointer by Synaps Co and some others. These packages have proved well when solving real life optimization problems and are used extensively.

This paper presents basic capabilities and features of software and tools that utilize various IOSO (Indirect Optimization based on Self-Organization) Technology algorithms<sup>1</sup>. This software product is designed to solve wide range of real-life problems in various fields of science and technology<sup>2,3,4</sup>. Currently, our algorithms and software are effectively applied to the following problems:

- Design optimization (obtaining the most effective technical solutions by means of multidimensional optimization of design parameters of the system being investigated according to one or more criteria);
- Obtaining optimal management of complex systems at a wide range of varying conditions and purposes of functioning;
- Design optimization of controllable systems (simultaneous optimization of design parameters and control laws);
- Comparative analysis of different solutions and grounds for choosing a particular technical option.

It is noticeable that while solving MDO problems, when a combination of different mathematical models is employed, it is quite difficult to predict topology of objective function and constraints. The function can be multiextremum, non-differentiable, with incomputability areas or with multiple interactions of solution area.

This paper presents a sample usage of one of our software products, IOSO NS<sup>5</sup>, to solve various-class problems. An example of using this software to pick thermodynamic parameters of a newly designed gas turbine engine for a medium-range airliner is given. This problem was solved in cooperation with Central Institute of Aircraft Motors (CIAM).

### **Distinctive features of**

#### **IOSO Technology Software and Tools**

Software and Tools of IOSO Technology consist of several independent algorithms intended for solving the following nonlinear optimization problems:

- Single-objective
- Multiobjective
- Parallel single and multi-objective
- Multilevel with adaptive change of the modeled object fidelity (low-, middle-, high fidelity models)
- Robust design optimization and robust optimal control

All IOSO technology algorithms were developed according to the single concept of formulating optimization problem, providing initial data, data exchange with the user's program, and analysis of the obtained results.

IOSO technology algorithms implement a new evolutionary response surface strategy. This strategy differs significantly from both the traditional approaches of nonlinear programming and the traditional response surface methodology. Because of that, IOSO algorithms have higher efficiency, provide wider range of capabilities, and are practically insensitive with respect to the types of objective

function and constraints. They could be smooth, non-differentiable, stochastic, with multiple optima, with the portions of the design space where objective function and constraints could not be evaluated at all, with the objective function and constraints dependent on mixed variables, etc.<sup>6</sup>.

### **Basic algorithm**

Each iteration of IOSO consists of two steps. The first step is creation of an approximation of the objective function(s). Each iteration in this step represents a decomposition of initial approximation function(s) into a set of simple approximation functions so that the final response function is a multi-level graph. The second step is the optimization of this approximation function. This approach allows for corrective updates of the structure and the parameters of the response surface approximation. The distinctive feature of this approach is an extremely low number of trial points to initialize the algorithm (30-50 points for the optimization problems with nearly 100 design variables). The obtained response functions are used in the process of optimization. During each iteration of IOSO, the optimization of the response function is performed only within the current search area. This step is followed by a direct call to the mathematical analysis model for the obtained point. During the IOSO operation, the information concerning the behavior of the objective function in the vicinity of the extremum is stored, and the response function is made more accurate only for this search area. While proceeding from one iteration to the next, the following steps are carried out: modification of the experiment plan; adaptive selection of current extremum search area; choice of the response function type (global or middle-range); transformation of the response function; modification of both parameters and structure of the optimization algorithms; and, if necessary, selection of new promising points within the researched area.

IOSO Technology algorithms designed to solve particular-type optimization problems (single-objective, multiobjective, parallel, multilevel and robust design optimization), have certain peculiarities, which provide high efficiency while solving concrete problems. Information on these algorithms and practical solution examples can be found in<sup>1,2,3,4</sup>.

### **Invariant Features of IOSO Technology Software and Tools**

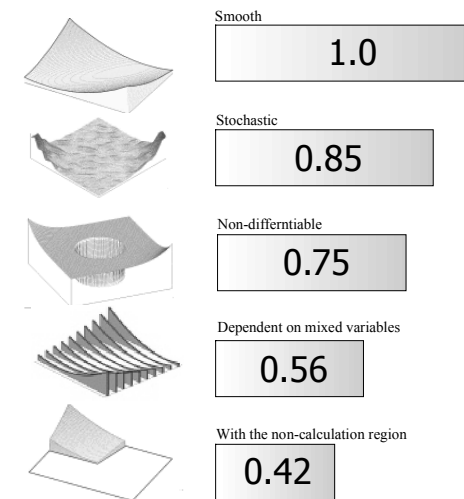
An important feature of IOSO Technology software is its capability to solve a wide range of optimization problems having different types of objective functions.

IOSO Technology Tools implement highly efficient evolutionary self-organizing algorithms. The efficiency

is guaranteed by internal adaptive choice of the algorithm suitable for each particular problem. This feature results in solving complex optimization problems with minimal number of evaluations of the system mathematical model.<sup>1,2,3,4,5</sup>

Optimization procedure is universal. It is uniquely powerful according to relation between required number of call to analysis module and response topography complexity. On smooth object function it works not worse than gradient methods. However for complex (and more probable to be faced to a designer in practice) object functions, having incomputability areas, discontinuities, multiple extremums and noise, the number of function calls required to find global extremum is being increased inconsiderably, while gradient methods are inapplicable for such tasks solution. Results of comparative analysis of IOSO Technology Tools and known gradient methods can be found in<sup>7</sup>.

We have conducted special research aimed to assess efficiency of IOSO Technology algorithms when solving optimization problems of different classes and types. Fig. 1 shows a comparative rating obtained after extensive trial of IOSO algorithms with a number of test functions. The rate decreases with complication of objective function topology proportionally to increasing of number of calls necessary to search for an optimal solution.



**Fig. 1 IOSO algorithms efficiency for different objective functions.**

In our research we used known test functions for constrained and unconstrained optimization presented in<sup>8,9</sup>.

All IOSO Technology algorithms have undergone extensive tests with functions of different topological complexity. The results of the trials prove high

efficiency of the IOSO NS algorithms<sup>1,5,6,7</sup>. This paper presents some results of application of these algorithms<sup>5</sup>.

For example, modified Wood Test Problem. The given test problem is sited in<sup>8</sup>. This test problem has 4 design variables. Variables bounds are following:  
 $-10 \leq x_j \leq 10, j = \overline{1,4}$ .

The analytical expression of original goal function is following:

$$y = 100 \cdot (x_2 - x_1^2)^2 + (1 - x_1)^2 + 90 \cdot (x_4 - x_3^2)^2 + (1 - x_3)^2 + 10.1 \cdot [(x_2 - 1)^2 + (x_4 - x_1)^2] + 19.8 \cdot (x_2 - 1) \cdot (x_4 - x_1)$$

The modified problem has 5 inequality constraints, such as:

- Constraint(i)=x(i)< 1.0, i=1,4;
  - Constraint(5)=2\*x(1)-x(2)-1>0.0
- (All constraints are active).

The solution is following:

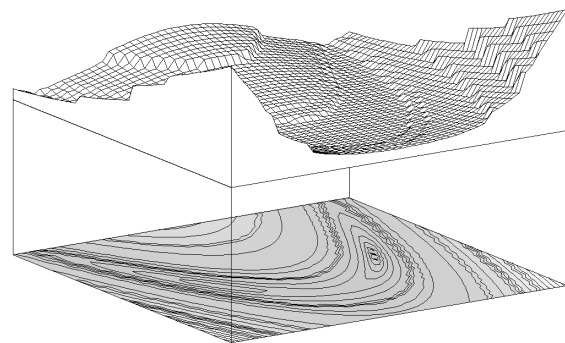
$$y^* = 0, \text{ for } x_i^* = 1, i = \overline{1,4}$$

When solving this problem using IOSO NS package, we obtained best value of object function as 0.29e-12 with only 104 direct calls to mathematical model.

To evaluate the stability of IOSO NS the artificial non-differentiability has been added to the goal function, such as:

$$\tilde{y} = \begin{cases} y + a_i & \text{for } y > 10^{5-i}, a_i = 10^{6-i}, i = \overline{1,11} \\ y & \text{for another cases} \end{cases}$$

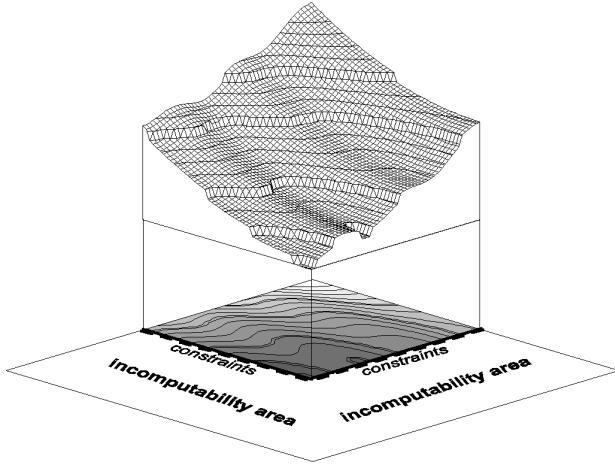
The topology of resulting objective is shown at Figure 2.



**Figure 2. The example of modified Wood function topology.**

It was difficult enough to find optimal solution, but only 623 direct calls to mathematical model were required for IOSO NS to reach the level of objective as 0.53e-16.

When modifying the Wood function making it incomputable, we simulated the crash of executable module when  $x_i < 1, i = \overline{1,4}$ . The example of topology is shown at Figure 3.



**Figure 3. The example of modified Wood function topology with artificial incomputability areas.**

This modification made the test function much more difficult for IOSO NS, but it was still solvable. For this example of test the number of direct calls as 980 was enough to reach the value of objective  $0.35e-14$ .

Let us consider another function - test problem of Levy # 9.

The given test problem is sited in <sup>9</sup>. The problem has 4 design variables with 8 variable bounds:  $-10 \leq x_j \leq 10, j = \overline{1,4}$ .

Analytical expression is following:

$$y = \sin^2[\pi \cdot z_1] + \sum_{i=1}^{n-1} [z_i - 1]^2 \cdot [1 + 10 \cdot \sin^2(\pi \cdot z_{i+1})] + (z_4 - 1)^2,$$

$$\text{where } z_i = 1 + \frac{(x_i - 1)}{4}, (i = \overline{1,4}).$$

The solution (global minimum) is following:

$$y^* = 0.0 \text{ for } x_i^* = 1.0, i = \overline{1,4}$$

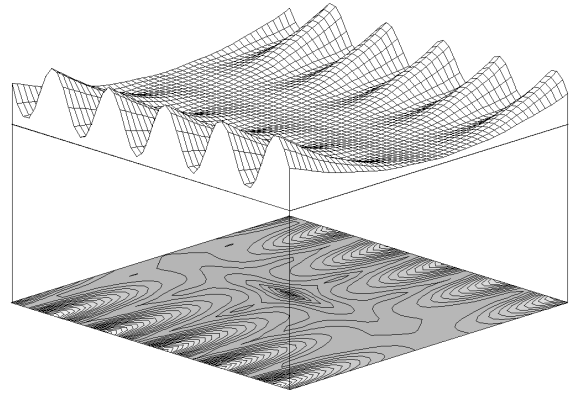
Goal function is multiextremum, it has 625 local minima. The example of its topology is shown at Figure 4.

This function global optimum was successfully found, the solution  $0.71e-13$  was found using 147 direct calls to mathematical model.

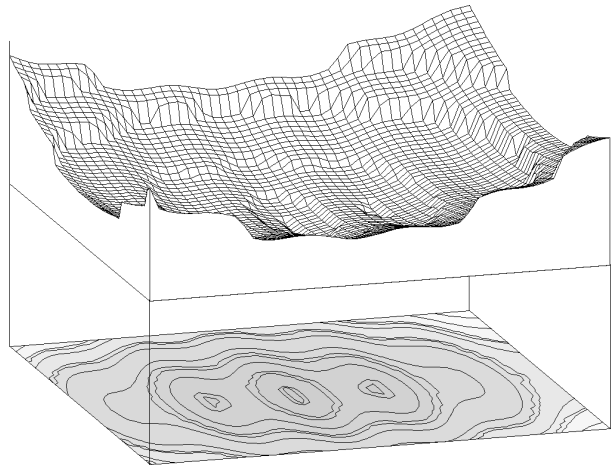
Modified Levy#9 function has artificial non-differentiability such as:

$$\tilde{y} = \begin{cases} y + a_i, & \text{for } y > 10^{5-i}, a_i = 10^{6-i}, i = \overline{1,14} \\ y & \text{for another cases} \end{cases}$$

The topology of modified function is shown at Figure 5.



**Figure 4. The example of Levy#9 test function topology.**



**Figure 5. Modified Levy#9 test function topology.**

Using 420 calls to mathematical model, the value of objective  $0.86e-13$  was found.

Thus, testing of IOSO NS has proved high efficiency and stability of the algorithm when applied to complicated test problems.

### Some features of IOSO NS

The IOSO NS package is designed to solve complex problems of constrained and unconstrained optimization with various classes of objective function: smooth, non-differentiable, stochastic, with multiple optima, with the portions of the design space where objective function and constraints could not be evaluated at all, with the objective function and constraints dependent on mixed variables, etc. Algorithms of IOSO NS have good invariant features,

high level of stability of calculation while optimization of complex objects; they also ensure search for extremum with presence of incomputability areas. These features of the algorithms make it possible to substantially expand classes of problems solved, facilitating the use of this software for complex practical problems.

IOSO NS package works with only executable modules written to represent mathematical models. This significantly facilitates customizing of interaction of user's model and the optimization procedure since it does not require either shared PC memory spaces for data exchange or specific programming language to write the analysis code. Data exchange is provided by means of text files on a disk drive, making it easy to integrate the analysis codes into IOSO NS package.

IOSO NS has user friendly GUI and is simple to use. The software provides all necessary information to the user interactively. Parameters of IOSO technology are pre-programmed and are adaptively changing during the search for extremum without user's intervention. Most of the algorithm tunings are done internally, i.e. are hidden from the user who is not required to have any knowledge of nonlinear programming or optimization procedures.

The only important thing for the user is to understand the physics of the problem and to have a mathematical model of the system. Creating an interface between IOSO and mathematical model typically takes just several minutes.

Optimization process is visually represented in real time (there are displayed current values of the design variables and their bounds, objective function history). User is able to control optimization process (interruption of the optimization process to tune up parameters with ability to restart from the specified point, cleaning up "hanged" or crashed user's mathematical model).

#### **Optimization problem statement for parameters of airliner's power plant**

For turbofan engine it is necessary to select design parameters for performance that will provide extremal value of the chosen indicator of its efficiency. The following main parameters and constraints for the engine were set:

- take-off thrust in conditions of international standard atmosphere (ISA) – 8000 kg;
- cruising thrust – no less than 1750 kg;
- maximum allowed turbine inlet temperature - no more than 1970K;
- air flow rate during take-off in conditions of ISA – no more than 250 kg/s.

Proceeding from the level of present-day propulsion engineering the following parameters were taken for values: efficiency of compressors and turbines, power takeoff from shafts and air takeoff through compressor for both cooling system and airliner's needs, levels of hydraulic losses in flow-through parts of engine.

The design values of the following thermodynamic parameters were to be calculated in the research:

- total compressor pressure ratio  $\pi_{c\Sigma}^*$ ;
- low pressure compressor (fan) pressure ratio  $\pi_{LPC}^*$ ;
- bypass ratio  $m$ ;
- temperature before turbine  $T_{T1}^*$ ;
- air flow rate through the engine  $G_I$ .

The criterion for efficiency indicator of the engine was the value of specific fuel consumption during cruising flight (H=11km, Mach=0.8) or direct airliner maintenance costs (\$/passenger\*kilometer)).

#### **Mathematical models**

To calculate performance specifications of the engine we have developed a mathematical model. This model is based on generalized characteristics of compressors and turbines. For given parameters of operation process ( $\pi_{c\Sigma}^*$ ,  $\pi_{LPC}^*$ ,  $m$ ,  $T_{T1}^*$  and  $G_I$ ) the engine is considered by means of a thorough thermodynamic calculation. This defines areas of nozzle diaphragm of the turbines and the nozzles of core and bypass ducts. Then characteristics of compressors and turbines are calculated by generalized procedure of Pr. Fyedorov<sup>10</sup>.

Performance characteristics of the engine are calculated by solution of nonlinear algebraic equations describing balances of air and gas consumption in characteristic sections of flow-through part and the powers on shafts of compressors and turbines. The equation set is solved by Newton's numerical method. The equation set contains explicitly performance constraints of the engine (according to maximum reduced and physical rotor frequencies and temperature before turbine.) Mathematical model is implemented as a Windows-based application, allowing to compute altitude and velocity characteristics as well as throttle performances of the engine for the whole range of flight altitudes and velocities without consideration of power-plant external aerodynamic drag.

To calculate engine's efficiency a mathematical model developed by Central Institute of Aircraft Motors (CIAM) was used. This model is based on statistical data reflecting dependence of efficiency indicators of the power plant on its design parameters and performance characteristics<sup>11</sup>. For research purposes we have provided batch-mode executable modules. The

model consists of independent calculation modules which allow to determine: performance characteristics of the power plant (with external resistance included), engine's life period, flight specifications, maintenance costs, direct performance costs and fuel costs, etc.

The initial data for CIAM model are engine's design parameters and performance characteristics (field of altitude and velocity characteristics for the whole range of flight altitudes and velocities as well as thrust curve) without inclusion of external resistance.

The calculation with the use of all mathematical models was implemented as a bat-file which ensured sequential launch of all executable applications of complex mathematical model. The general scheme of all applications interaction and data exchange is shown in Figure 6.

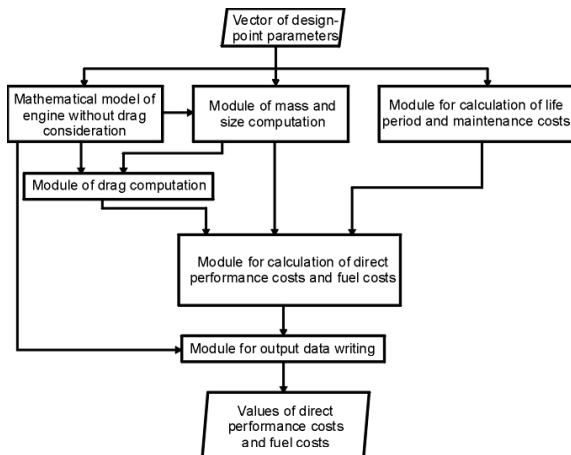


Figure 6: Block-scheme of modules scheduling and data exchange

### Optimization problem statement and solution according to engine's internal efficiency indicators

Operational fuel consumption of medium-range airliner is one of its efficiency indicators that is to be ensured in our research. The engine's fuel consumption rate during cruising flight has, apparently, direct connection with the value of operational fuel consumption of the plane, that is why minimization of this parameters can be used as a criterion of optimum during primary definition of area of solution search.

The mathematical model is formulated in as follows: to find design parameters of engine that will ensure minimal fuel consumption rate at given cruising thrust and constraints set.

The variable parameters are the following:

1. value of total compressor pressure ratio  $\pi_{c\Sigma}^*$  during take off in ISA;

2. value of low pressure compressor pressure ratio  $\pi_{LPC}^*$  during take off in ISA;

3. bypass ratio  $m$  during take off in ISA;

4. temperature before turbine  $T_{II}^*$  during take off in ISA;

5. air flow rate through the engine  $G_I$  during take off in ISA;

6. relative frequency of low pressure rotor in conditions of cruising flight (H=11km, Mach=0.8).

In this research the objective function was considered to be the value of specific fuel consumption during cruising flight (H=11km, Mach=0.8).

The following constraints were chosen in the task:

1. thrust during take off in ISA – no less than 8000 kgf;

2. thrust during cruising flight (H=11km, Mach=0.8) – no less than 1750 kgf.

Setting up an optimization task in IOSO NS environment consists of five easy steps.

At the first step the user selects direction of search for extremum (minimization of objective function, in the case being considered).

At the second step the user defines design variables and their variation bounds. Fig. 7 shows variation bounds of design variables for this particular task. It is noticeable that variation range has been chosen quite broad.

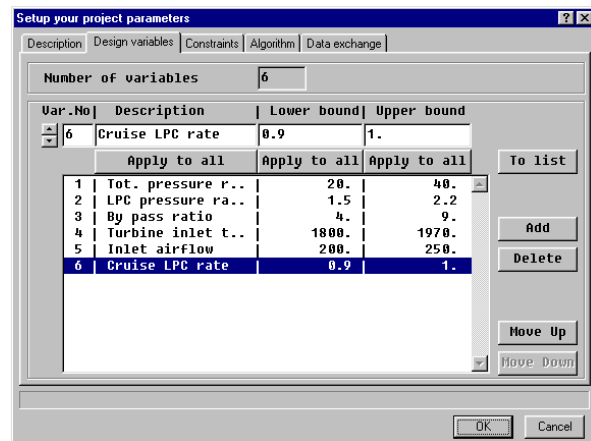


Figure 7: Setting of design variables.

At the third step the constraints are set. Fig 8. shows the constraints, their bound values, and relation characters.

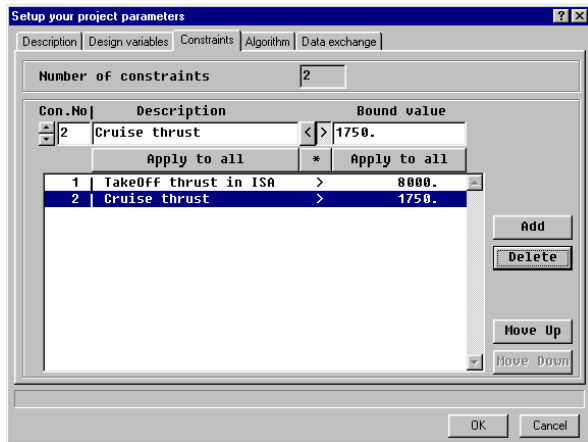


Figure 8: Setting of constraints.

On the fourth step parameters of the algorithm are set. IOSO NS package allows changing of 5 parameters of the algorithm:

- maximum number of iterations;
- accuracy of solution according to criterion set;
- accuracy of constraints observation;
- number of points in experiment plan;
- convergence ratio.

A newly created project takes default values for all of the algorithm parameters. The user is usually required to alter only first three ones.

On the final step the user sets names of analysis executable files and the files for data exchange between the optimizer and the mathematical model.

After the project is saved, the optimization task is launched. The project optimizer window is shown on Fig. 9.

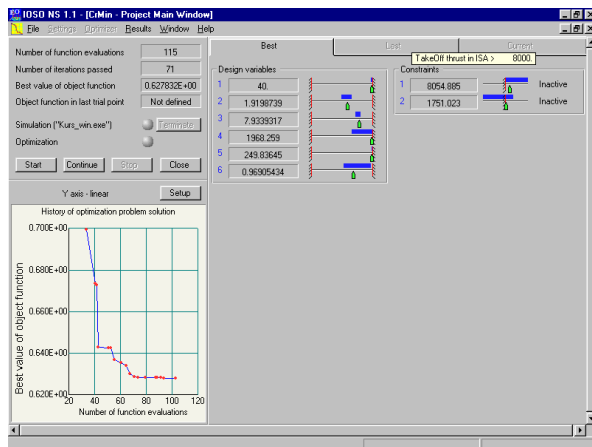


Figure 9: IOSO NS 1.1 optimizer window.

The optimizer window can be used to manage solution search process. It also displays useful information at runtime. The upper left area of the window displays values of objective function at the best and the last search points as well as the number of iterations passed and the number of direct calls to mathematical model. The control buttons allow, if necessary, to stop the process at any moment, or resume it.

Below is the graph reflecting history of object function improvement with all the constraints observed. The graph becomes flat as number of calls to the model increases. This is conditioned by the fact that at the beginning of search the IOSO NS algorithm determines the extremum area fast. In principle, an acceptable solution can be obtained within 70 calls to the model.

On the right the window shows values of design variables and constraints as well as the bounds of current search areas. By switching between tabs values for the best and the last points can be displayed. Also, the vector of design variables for the running user's executable module can be shown.

Working with user's executable modules IOSO NS creates a temporary file on a hard disk with the values of design variables. It then starts user's executable module and waits for its termination, after which values of calls are read. While solving design problems (the one of aircraft turbine engine, for example) incomputable areas of values of objective function and constraints may exist. This can be conditioned by both impossibility of project existence at certain combination of design variables, and instability of numerical schemes used as mathematical models. This can even lead to the crash of user's application. Fig. 10 shows topology of objective function and constraints in the area of extremum as dependent on the bypass ratio and LPC pressure ratio. The dark area on the right corresponds to incomputability areas of mathematical model. In this case the incomputability area is conditioned by the fact that at certain combinations of bypass ratio and fan pressure ratio it is impossible to ensure coordinated work of the engine's elements (increasing of both of these parameters the turbine's power is insufficient to move the fan). Existence of such areas usually substantially complicates solution of optimization tasks and in some cases makes finding extremum impossible.

However, algorithms based on response surface technology, implemented in IOSO NS, make it possible to successfully solve problems with such complex topologies.



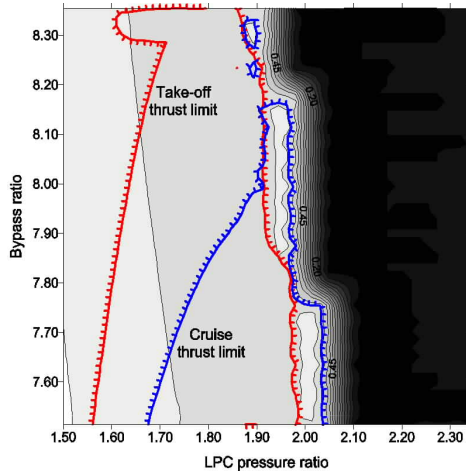


Figure 10: Topology of objective and constraints.

Fig. 9 shows that the solution close to optimum has been found in 115 direct calls to mathematical model. This proves quite a high efficiency of IOSO NS algorithms (the optimization task was characterized by 6 variables, non-differentiable objective function and existence of incomputability areas).

During problem solution the obtained results are displayed on a screen (Fig 11.) and written to a text file.

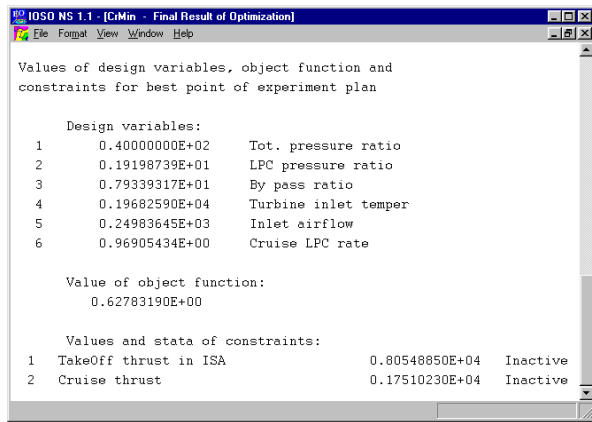


Figure 11: Results viewer window

The analysis of results shows that optimal solution agrees well with physical representations of types of running processes in the engine. The following points can be marked out:

Minimum fuel consumption rate during cruising flight is reached at a maximum total compressor pressure ratio. The design-point air flow rate also approaches the maximum permitted value, which ensures increase of engine's bypass ratio, and consequently, decreases its specific fuel consumption value.

Thus, as a result of this task solution the following optimal design parameters of the engine were determined:

1. total compressor pressure ratio in ISA conditions–40;
2. fan pressure ratio in ISA – 1.92;
3. bypass ratio in ISA – 7.93;
4. temperature before turbine in ISA – 1968 K;
5. air flow rate in ISA – 250 kg/s;
6. relative frequency of low pressure rotor during cruising flight (H=11km, Mach=0.8) – 0.969.

At given engine parameters and with all inactive constraints observed a minimum value of fuel consumption rate during cruising flight (H=11km, Mach=0.8) equals 0.6978 kg/[kgf\*hour].

### Optimization of airliner's power plant

This research used the following problem statement: to find engine's design parameters that would ensure minimal direct operational costs of the airliner of given mass during flight on a given range with constraints observed.

The design variables and constrains were the same as in the first problem. In this research the mathematical model was the compound of "power plant – airliner" models block-scheme of which is shown in Fig. 6.

The solution of this problem has yielded the following results:

1.  $\pi_{C\Sigma}^*$  in ISA conditions– 31.6;
2.  $\pi_{LPC}^*$  in ISA – 1.89;
3. bypass ratio in ISA – 7.73;
4. temperature before turbine during take-off in ISA – 1913 K;
5. air flow rate in ISA – 250 kg/s;
6. relative frequency of low pressure rotor during cruising flight (H=11km, Mach=0.8) – 0.971.

With all this, minimum operational costs with all constraints observed was equal to 0.0232, \$/(passenger\*kilometer).

It is noticeable that the engine parameters providing minimal specific fuel consumption during cruising flight and the minimal direct operational costs are different. This means that the most economical engine, in terms of fuel consumption rate, is not the cheapest one to utilize. Thus, in the first problem the total increase of gas pressure and temperature before turbine was obtained. But the model in the first problem did not deal with the fact that high-pressure compressor would have higher mass and the high-temperature turbine would have low life period. These results confirm that



to ensure high efficiency of aircraft engine it is necessary to define its optimum parameters by means of high-level criteria.

### **Conclusion**

This article has presented main feature of **IOSO NS** software when applied to different-class optimization problems. It also included a sample usage of the software for picking parameters of gas turbine engine of medium-range airliner. These results prove high invariant features of the algorithms employed by IOSO NS. They show high stability of calculation during optimization of complex objects, allowing also an effective search for extremum with existence of incomputability areas.

### **References**

- [1]. Egorov I.N., "Indirect Optimization Method on the Basis of Self-Organization", Curtin University of Technology, Optimization Techniques and Applications (ICOTA'98), Vol.2, pp. 683-691, Perth, Australia, 1998.
- [2]. Egorov I.N., Kretinin G.V., "Search for Compromise Solution of the Multistage Axial Compressor's Stochastic Optimization Problem", World Publishing Corporation, Aerothermodynamics of internal flows III, pp. 112-120, Beijing, China, 1996.
- [3]. Egorov I.N., Kretinin G.V., Leshchenko I.A., Babiy Y.I., "The Use of Variable Fidelity Models in the Optimization Problems for Aircraft Engines Parameters and Control Laws", ASME, 2000-GT-3, 2000.
- [4]. Egorov I.N., Kretinin G.V., Leshchenko I.A., "Two Approaches to Multidisciplinary Optimization Problems", presented at European Congress on Computational Methods in Applied Sciences and Engineering – ECCOMAS-2000, Barselona, Spain, 2000.
- [5]. User Guide, IOSO NS version 1.1, IOSO Technology Center, Moscow, 2002.
- [6]. Egorov, I.N., Kretinin, G.V., Leshchenko, I.A., "Robust Design Optimization Strategy of IOSO Technology". WCCM V, July 7...12, 2002, Vienna, Austria.
- [7]. Egorov, I.N., Kretinin, G.V., Leshchenko, I.A., "Optimization Algorithms as Tools for the Solution of Inverse Problems". WCCM V, July 7...12, 2002, Vienna, Austria.
- [8]. Sandgren, Eric. "The Utility of Nonlinear Programming Algorithms", a thesis submitted for the Degree of Doctor of Philosophy, Purdue University, 1977.

[9]. J.J. More, B.S. Gabow, and K.E.Hillstrom, "Testing Unconstrained Optimization Software," ACM Trans. Meth. Software, (1981), 17-41.

[10]. Fedorov, R.M. "Approximate Computation of Fixed-Geometry Axial Compressors Characteristics in a Wide Range of Relative Rotation Rate and Reynolds Number". Aircraft Industry, No 34, p. 32, Moscow, Russia, 1995,

[11]. Yugov, J.R., Selivanov, O.D., "Principles of Aircraft and Engine Integration". "Mashinostroyenie", Moscow, Russia, 1989.