Pareto Optimization in Oil Refinery

Dmitri Kostenko^(✉), Dmitriy Arseniev, Vyacheslav Shkodyrev, and Vadim Onufriev

Peter the Great St. Petersburg Polytechnic University, Saint-Petersburg, Russia
Zaba-1@bk.ru

Abstract. This article describes the process of multicriteria optimization of a complex industrial control object using Pareto efficiency. The object is being decomposed and viewed as a hierarchy of embedded orgraphs. Performance indicators and controlling factors lists are created based on the orgraphs and technical specifications of an object, thus allowing to systematize sources of influence. Using statistical data archives to train, the neural network approximates key sensors data to identify the model of the controllable object and optimize it.

Keywords: Decomposition · Pareto efficiency · Multicriteria optimization · Identification · SPEA2 · Neural network

1 Introduction

Multicriteria optimization is a process of simultaneous optimization for two or more conflicting functions within one point [1]. The need to simultaneously optimize different key performance indicators (KPI) exists in both business and industrial production, such as oil refinement facilities.

Oil refinement is an advantageous direction for multicriteria optimization because of its complexity. Optimisation process is preceded by decomposition [2] of the refinement sequence, which is followed by model identification [3]. Pareto optimality principle [4] is used in conjunction with aforementioned models to build a front of optimal values.

Part of the refinement process takes place inside a refraction unit (RU). One of these units was taken as a prototype for our model. The model got the following KPIs assigned [5]: Quality (matching degree between output and established norms), Performance (output product volume), Efficiency (resource usage potency), Reliability (equipment failures per unit of time), Safety (emergencies per unit of time).

Rectification consists of a wide array of parameters, up to several hundreds of characteristics per one refraction unit. These include the splitting section, column head and column plates temperature and pressure. Inside and outside of the rectification column both sequential (multi-layered raw oil refinement, raw oil heating, raw oil pumping, etc.) and parallel (vapor condensation) processes take place. Rectification technology also includes transition products into the process, thus applying an additional, horizontal level of hierarchy between the operations. It is also essential to account for the time delay, added by the inertia of the system itself and enforced by the continuous operating mode.

Consequently, processes from the highest levels of the hierarchy have unobvious connections with the lower-level processes thus making it impossible to use simple functions like $y = f(g, u)$ to represent dependencies between them. Yet it is essential to influence top-level processes by changing parameters of the low-level processes and vice-versa.

In this work the aforementioned problem is resolved by decomposing a complex system (such as refraction unit) down to individual units and processes. The resulting structure is represented as a graph. The KPI set takes the top level of the hierarchy. Every KPI is divided into several summands of a lower hierarchy levels. The step is repeated until the summand can be unambiguously interpreted by the $y = f(x)$ type of dependency. Dependencies are identified using a neural network trained on the RU statistical data archive. Going up by one hierarchy level changes dependency to $y = g(f(x))$. Ascending by the hierarchical tree allows to determinate a clean dependency between a KPI and an input parameter from the bottom of the hierarchy.

However, the top-level key performance indicators may directly contradict each other. For instance, raising Performance by forcing aggressive operating parameters will inevitably cause the growth in equipment failures and an overall reduction of Reliability. Which in turn damages Efficiency of the refraction unit or a refinery as a whole. The “Good – Fast – Cheap” triangle encourages us to use a multicriteria optimization algorithm to balance out conflicting key performance indicators.

The aim of this work is to perform a multicriteria optimization to find a Pareto optimal solution. This allows us to find a safe combination of controllable parameters able to keep the target indicators inside the given target intervals.

This analysis is based on statistical data archive taken from a working refinery. It was used to build the graphical representations of dependencies between performance and temperature, characterising the lowest hierarchy level. In order to optimize the top-level KPIs by changing the bottom-level controllable parameters, a strong correlation must be revealed. To grant it, a dependence model identification has been performed [6].

2 Data Identification

The refraction unit consists of an oil pre-heater with a heat-exchange unit, a fractionating column, a refrigerator and a boiler. The pre-heated oil is injected into the column feeding zone to be divided into vapour and solid phases. During the rectification process isopentane is extracted from the top part of the column as a fractionator overhead. Heavier fractions are taken from the plates in the middle of the rectification column. The heaviest part, the long residuum, gets extracted from the bottom part of the column [7]. A simplified scheme, showing distillation inputs and outputs targeted by this work, is present on Fig. 1.

Fig. 1. Simplified scheme of the rectification process

The stripper temperature (U_4 in Table 1) has been chosen for optimization. Temperature variation was aimed at maximizing fractions 240–300 and 300–350 output volumes (G_2 and G_3 in Table 1).

Table 1. Hidden neural layer for fraction 240–300

Neyron	Input (IN)	Output (ON)	Input weight (IHN)	Output weight (OHN)
0	0,5329	0,4675	0,9646	0,5356
1	−0,4489	0,4218	−0,8125	−1,0897
2	0,6221	0,6083	1,1261	0,6041
3	−0,1149	0,5999	−0,2079	−0,7524
4	0,7458	0,5611	1,3499	0,8198
5	2,5314	0,7924	4,5817	3,1874
6	−1,0151	0,3031	−1,8373	−1,8294
7	1,5302	0,7177	2,7695	1,8181
8	−0,0096	0,4126	−0,0174	−0,3680
9	−0,3281	0,3505	−0,5939	−0,8191

To identify dependencies between the stripper temperature and fraction outputs (see Fig. 1), a neural network (NN) was utilized. It was trained on statistical data obtained during 24 h of the prototype column work.

The neural network consists of 1 input, 1 hidden and 1 output layers. Input and output layers both contain one neuron, while the hidden layer contains 10. The NN was trained by the backward propagation of errors method. Hidden and output layers use a sigmoid activation function (1):

$$F(x) = \frac{e^x}{e^x + 1} \quad (1)$$

The general formula for the NN is (2):

$$ON = f \left(\sum_{k=0}^{10} f \left(\sum_{k=0}^{10} IN * IHN \right) * OHN \right) \quad (2)$$

Here f stands for sigmoid activation function (1), while IN, ON, IHN and OHN are present in Tables 1 and 2.

Table 2. Hidden neural layer for fraction 300–350

Neyron	Input (IN)	Output (ON)	Input weight (IHN)	Output weight (OHN)
0	−0,4983	0,2103	−0,9457	0,7026
1	−3,5414	0,0843	−6,7410	−3,9722
2	−0,7764	0,3018	−1,4716	1,3219
3	0,7931	0,4216	1,5021	−1,4161
4	−0,7696	0,3039	−1,4587	1,3393
5	−0,1638	0,1935	−0,3109	0,2529
6	−0,1228	0,2843	−0,2331	0,1390
7	−0,2815	0,2045	−0,5342	0,4098
8	−0,5891	0,2605	−1,1171	0,9753
9	0,0778	0,2562	0,1475	−0,1713

Application of the aforementioned neural network to input data allowed to identify dependencies between the stripper temperature and output volumes of 240–300 and 300–350 fractions (see Fig. 2).

Fig. 2. Graphical representation of the identified models

To verify correlation between the models (lines on Fig. 3) and statistical data (crosses on Fig. 2) correlation coefficients $\rho_{x,y}$ have been calculated. The left graph $\rho_{x,y} = 0,76312$ and the right graph $\rho_{x,y} = 0,90781$. Coefficient formula (3) is present below, σ_x and σ_y are the mean values of the corresponding selections.

$$\rho_{x,y} = \frac{\frac{1}{n} \sum_{t=1}^n X_t Y_t - \left(\frac{1}{n} \sum_{t=1}^n X_t \right) \left(\frac{1}{n} \sum_{t=1}^n Y_t \right)}{\sigma_x \sigma_y} \quad (3)$$

Correlation numbers could be improved by increasing the size of the training selection for the neural network. Amount of NN's hidden layers and/or hidden neurons are also subjects to change. But in order not to deviate from the main theme of the work, achieved correlations have been accepted.

Having mathematical models of interconnections between the basic parameters of the oil refinement process in place, it is now possible to compare them and define the optimal points according to the multicriteria optimization method.

3 Pareto Optimality

Pareto optimality is a state of allocation of resources from which it is impossible to reallocate so as to make any one individual or preference criterion better off without making at least one individual or preference criterion worse off [8].

Pareto front within the range of target functions is a combination of solutions, which do not dominate each other, but dominate every other solutions within the search space at the same time. It means that it is impossible to find a single solution able to excel every other solution at reaching every target. Mathematically such problem can be formulated as follows: one must find a vector $X^* = [x_1^*, x_2^*, \dots, x_n^*]^T$, that would optimize a vector of target functions $F(X) = [f_1(X), f_2(x), \dots, f_k(x)]^T$ while having m inequality constraints $g_i(X) \leq 0, i = \overline{1, m}$, and p equality constraints $h_j(X) \leq 0, j = \overline{1, p}$.

Here $X^* \in R^n$ is a solution vector; $F(X) \in R^k$ is a vector of target functions every single one of which must be optimized [9].

Strength Pareto Evolutionary Algorithm 2 (SPEA2) [10] was used for Pareto optimization. Despite it's relatively old age, it is a well-tested algorithm, effective for select applications [11], including more representative spread of non-dominated solutions [12], and was chosen over others, including VEGA, FFGA [13] and NPGA [14].

SPEA2 algorithm can be summarized in 6 steps:

- Step 1, Initialization: Generate an initial population P_0 and create the empty archive (external set) $\overline{P}_0 = \emptyset$. Set $t = 0$.
- Step 2, Fitness assignment: Calculate fitness values of individuals in P_t and $\overline{P}_t$.
- Step 3, Environmental selection: Copy all nondominated individuals in P_t and $\overline{P}_t$ to P_{t+1} . If size of P_{t+1} exceeds $\overline{N}$, then reduce P_{t+1} by means of the truncation operator, otherwise if size of P_{t+1} is less than $\overline{N}$, then fill P_{t+1} with dominated individuals in P_t and $\overline{P}_t$.
- Step 4, Termination: If $t \geq T$ or another stopping criterion is satisfied then set A to the set of decision vectors represented by the nondominated individuals in P_{t+1} . Stop.

- Step 5, Mating selection: Perform binary tournament selection with replacement on P_{t+1} in order to fill the mating pool.
- Step 6, Variation: Apply recombination and mutation operators to the mating pool and set P_{t+1} to the resulting population. Increment generation counter ($t = t + 1$) and go to Step 2.

SPEA2 algorithm has been utilized to find a front of temperatures, maximizing the output of fractions 240–300 and 300–350. Previously identified models have been used as “experimental data” on the graph seen on Fig. 3.

Fig. 3. Pareto front

As a result, we got a Pareto front, consisting of eight points (see Table 3). The points are sorted in descending order of preference, thus making the first point the most optimal.

Table 3. Complete stats of the Pareto front

Point number	Temperature (°C)	Fraction 240–300 output volume (M ³)	Fraction 300–350 output volume (M ³)
1	344,823852539063	255,869043673089	313,914703058703
2	343,325988769531	255,623350989602	314,046780755533

(continued)

Table 3. (continued)

Point number	Temperature (°C)	Fraction 240–300 output volume (M ³)	Fraction 300–350 output volume (M ³)
3	343,371765136719	255,636549610692	314,046650398148
4	344,712951660156	255,866950335304	313,918779855307
5	343,374877929688	255,661217222795	314,045467958112
6	343,35595703125	255,672724869121	314,044445521759
7	343,364624023438	255,649165327318	314,046207859264
8	343,407836914063	255,683708140522	314,043155299305

4 Conclusion

In the course of the work a refraction unit has been decomposed, and it's input, controllable and target parameters were extracted and listed. Based on statistical data analysis the neural network was trained and then used to identify dependency models between the RU parameters. The models allowed us to define eight points of Pareto front.

We are planning on extend the sphere of practical application of this method. First of all, we'll have to decompose complete sets of dependencies from the basic controllable parameters up to top level KPIs. This would allow us to see key performance indicators of separate units and the whole refinery changing in real time.

Further software development enables us to revert the process and guess controllable parameters, able to sustain a pre-defined set of high-level KPIs. Such instrument will not only allow to optimize complex processes, but will also ensure much more effective control over them.

References

1. Steuer, R.: Multiple Criteria Optimization: Theory, Computations, and Application. Wiley, New-York (1986)
2. Yaochu, J.: Pareto-optimality is everywhere: From engineering design, machine learning, to biological systems. In: 2008 3rd International Workshop on Genetic and Evolving Systems, p. 1. IEEE (2008)
3. Kostenko, D., Kudryashov, N., Maystrishin, M., Onufriev, V., Potekhin, V., Vasiliev, A.: Digital twin applications: diagnostics, optimisation and prediction. In: Proceedings of the 29th DAAAM International Symposium, pp. 574–581. DAAAM International, Vienna (2018)
4. Evans, G.W., Stuckman, B., Mollaghasemi, M.: Multicriteria optimization of simulation models. In: 1991 Winter Simulation Conference Proceedings, pp. 894–900. Phoenix, Arizona (1991)
5. Key performance indicators in the oil & gas industry. <https://www.performancemagazine.org/key-performance-indicators-oil-bp/>. Accessed 23 Jan 2019
6. Schleicha, B., Anwer, N., Mathieu, L., Wartack, S.: Shaping the digital twin for design and production engineering. CIRP Ann. – Manuf. Technol. **66**(1), 141–144 (2017)

7. Alfke, G., Irion, W.W., Neuwirth, O.S.: Oil refining. In: Ullmann's Encyclopedia of Industrial Chemistry (2007)
8. Keeney, R., Raiffa, H.: Decisions with Multiple Objectives: Preferences and Value Trade-Offs. Cambridge University Press, Cambridge (1993)
9. Coello Coello, C.A., Christiansen, A.D.: Multi-objective optimization of trusses using genetic algorithms. *Comput. Struct.* **75**(6), 647–660 (2000)
10. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. In: TIK report № 103. Switzerland, Zurich (2001)
11. Tang, Y., Reed, P., Wagener, T.: How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration? *Hydrol. Earth Syst. Sci. Discuss. Eur. Geosci. Union* **10**(2), 289–307 (2006)
12. Chołodowicz, E., Orlowski, P.: Comparison of SPEA2 and NSGA-II applied to automatic inventory control system using hypervolume indicator. *Stud. Inform. Control* **26**(1), 67–74 (2017)
13. Fonseca, C.M., Fleming, P.J.: Genetic algorithm for multiobjective optimization, formulation, discussion and generalization. In: Genetic Algorithms: Proceeding of the Fifth International Conference, California, pp. 416–423 (1993)
14. Horn, J.N., Nafpliotis, A.L., Goldberg, D.E.: A niched Pareto genetic algorithm for multiobjective optimization. In: Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, pp. 82–87. IEEE Service Center, Piscataway (1994)