Discrete vs. Continuous Multiobjective Optimization of Continuous Casting of Steel

Miha Mlakar
Jožef Stefan Institute, and
Jožef Stefan International
Postgraduate School,
Jamova cesta 39,
SI-1000 Ljubljana, Slovenia
miha.mlakar@ijs.si

Tea Tušar
Jožef Stefan Institute, and
Jožef Stefan International
Postgraduate School,
Jamova cesta 39,
SI-1000 Ljubljana, Slovenia
tea.tusar@ijs.si

Bogdan Filipič
Jožef Stefan Institute, and
Jožef Stefan International
Postgraduate School,
Jamova cesta 39,
SI-1000 Ljubljana, Slovenia
bogdan.filipic@ijs.si

ABSTRACT

Continuous casting is a widely used steel production process. To yield high-quality steel, the casting parameters have to be tuned with respect to several contradictory criteria. We approached this multiobjective optimization problem in discrete and continuous variants, applying Exhaustive Search (ES) and Differential Evolution for Multiobjective Optimization (DEMO) on the discrete variant, and DEMO on the continuous variant. We analyzed the results in view of effectiveness and efficiency and showed that the most suitable way to solve this optimization problem is to apply the stochastic optimization approach on the finest reasonable discretization.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*global optimization, constrained optimization*

General Terms

Algorithms, Performance, Experimentation

Keywords

Continuous casting of steel, Numerical simulation, Parameter tuning, Multiobjective optimization, Differential evolution, DEMO algorithm

1. INTRODUCTION

Continuous casting is the most widely used procedure to produce steel semi-manufactures. Their quality depends on the casting process parameters, such as the casting temperature and speed, coolant temperatures and flows, etc. To ensure the best possible quality of cast steel, the parameter values need to be properly set. Because of high costs and safety risks, their tuning cannot be carried out on a real casting device, hence the use of a reliable numerical simulator of the casting process is necessary. However, simulation-based evaluation of solutions can still be time-consuming. For this reason we are interested in finding the best parameter settings in as few evaluations as possible.

Improving the quality of cast steel through process parameter optimization has already been treated in the literature both as a single- and multiobjective optimization problem. In the single-objective approach, various methods were used, for example, heuristic search [1] and weighted sum [8]. In the multiobjective approach authors used several criteria with the common goal of improving the steel quality. The applied techniques include Interactive Decision Maps [4], NIMBUS method [5], ant colony system algorithm [3], and DEMO algorithm [2]. However, the authors provide no guidelines on how to formulate and tackle the problem.

In this paper we present multiobjective optimization of process parameters in continuous casting of steel and investigate how to best approach this problem. For this purpose we solve two variants of this optimization problem. The first one is defined on a discrete decision space and the second one on a continuous decision space. On the discrete decision space we deploy Exhaustive Search (ES) and Differential Evolution for Multiobjective Optimization (DEMO), while on continuous decision space only DEMO. After the analysis of the results gained from both algorithms and on both decision spaces we identify the most suitable approach to solving this problem. In addition, we give recommendations on how to efficiently solve the problem with as little of computational effort as possible.

The structure of this paper is as follows. In Section 2 we outline the process of continuous casting of steel. In Section 3 we describe the optimization problem. In Section 4 we present the optimization environment consisting of the numerical simulator and the optimization algorithms. In Section 5 the experimental setup is described, and in Section 6 the results of numerical experiments are reported. Section 7 concludes the paper with a summary of the work done and guidelines for solving the problem.

2. CONTINUOUS CASTING OF STEEL

Continuous casting of steel is a complex metallurgical process where molten steel is cooled and shaped into semi-manufactures of desired dimensions. The main components of the casting system (schematically shown in Figure 1) are the ladle, tundish, mold and cooling subsystems.

The process of steel casting starts with molten steel being

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.

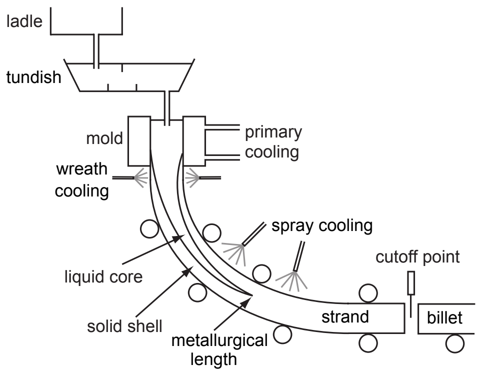

Figure 1: Scheme of the steel casting system

poured into the ladle from the electric furnace and then led through tundish that acts as a buffer for the liquid metal that is than drained into an open-base copper mold. As the mold is water-cooled, the hot steel starts solidifying in contact with it. Cooling is performed by water flowing through the channels built in the walls of the mold. The channels represent the primary cooling subsystem.

Molten steel with a thin solid shell, now called strand, exits the base of the mold into a spray chamber where it is immediately supported by closely spaced water-cooled rollers. To increase the rate of solidification, the strand is sprayed with water in the wreath and spray cooling areas. Together they represent the secondary cooling subsystem.

At the exit from the casting system, the steel is cut to billets of a desired length. The length of the liquid core in the strand is called the metallurgical length. In addition to the metallurgical length, the thickness of the solid shell at the mold exit and the strand surface temperature at the unbending point crucially affect the quality of the cast steel.

3. OPTIMIZATION PROBLEM

The optimization problem involves input variables (process parameters), output variables, and desired output values determined by the domain experts. The task is to find the input variable settings resulting in the values of output variables as close as possible to the desired values. According to empirical knowledge in the steel production domain, such settings result in high-quality steel.

If an output variable y_k violates its boundary constraints, i.e. $y_k \notin [y_k^{min}, y_k^{max}]$, its constraint violation is expressed as

$$v_k = \frac{|y_k - y_k^{border}|}{|y_k^{max} - y_k^{min}|}, \tag{1}$$

where y_k^{border} is the border value closest to y_k. The overall constraint violation v is calculated as

$$v = \sum_{k=1}^{k=n} v_k. \tag{2}$$

where n is the number of considered output variables. Every feasible output value has v_k equal 0.

For each output variable we define the optimization criterion as the difference between the value of the output variable and its desired value. Formally, if y_k is the observed value of an output variable, and y_k^* is its desired value, where

Table 1: Input variables, their bounds and discretization steps

Input variable	Lower bound	Upper bound	Disc. step
Casting speed [m/min]	1.50	2.00	0.01
Mold outlet coolant temp. [°C]	33	35	1
Wreath coolant flow [m^3/h]	10	40	5
Spray coolant flow [m^3/h]	25	65	5

Table 2: Output variables, their bounds and desired values

Output variable	Lower bound	Upper bound	Desired value
Metallurgical length [m]	10	11	10
Shell thickness [mm]	11	15	13
Surface temperature [°C]	1115	1130	1122.5

$y_k* \in [y_k^{min}, y_k^{max}]$, then the corresponding criterion is:

$$\varphi_k = |y_k - y_k^*|, \tag{3}$$

where $k \in \{1,\ 2,\ ...,\ n\}$, and n is the number of output variables. When y_k gets close to y_k^*, φ_k gets close to 0. The goal is to find such values of input variables, that all criteria would be 0 or as close to 0 as possible.

4. OPTIMIZATION ENVIRONMENT

4.1 Casting Process Simulator

To optimize the parameters of the steel casting process, we need a numerical model, because real-world experimentation with parameter settings is expensive, time consuming and could also be dangerous. We use a numerical model of steel casting based on a meshless technique for diffusive heat transport [9]. Given the values of the input variables, the numerical model is used as a process simulator that calculates the temperature field in the strand and returns the values of the output variables.

The input variables subject to optimization in our study are shown in Table 1. The lower and the upper bounds of the variables were determined by domain experts. The observed output variables are presented in Table 2. Their bounds and desired values were also determined by the experts.

The time needed to execute one simulation of the steel casting process was approximately 2 minutes on a 3.4 GHz Intel Core i7 computer with 8 GB RAM.

4.2 Exhaustive Search

ES evaluates every possible solution and returns the best one. For small problems this is an acceptable strategy, but as problems become larger, it is impossible to perform ES in a reasonable amount of time. In our case we applied ES to solve the discrete version of the problem and compare the exact results with the ones found by DEMO which is a stochastic algorithm.

4.3 Differential Evolution for Multiobjective Optimization (DEMO)

DEMO [7] is an evolutionary algorithm based on Differential evolution (DE) [6] and adapted for multiobjective optimization. Like DE, DEMO is easy to understand and imple-

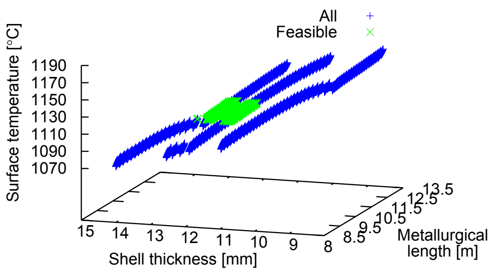

Figure 2: All evaluated solutions and feasible solutions found by ES

ment, and very effective on numerical problems. The main disadvantage of this algorithm is that it is not suited for solving combinatorial problems because the candidate creation uses vector addition and multiplication.

DEMO is a steady state evolutionary algorithm that adds new candidates to the existing population. Since new solutions are immediately used for generating new candidates, the algorithm convergence is accelerated. DEMO is also very effective in uniformly spreading the solutions on the non-dominated front. This is done by removing the solutions from the extended population with the truncation method taken from the SPEA2 algorithm [10].

5. NUMERICAL EXPERIMENTS

The experiments were run on two problem variants. The first one was defined on a discrete and the second one on a continuous decision space. For the first problem variant an appropriate discretization step for every input variable was chosen. The experts suggested the discretization step of 0.1 for casting speed and 1 for other input variables. With this discretization the ES algorithm would run to long. Results of preliminary tests have shown that two input variables, i.e. wreath coolant flow and spray coolant flow, have the least effect on output variables. Therefore, we set the discretization step for these two variables to 5 instead of 1, as shown in Table 1. With this discretization, the ES algorithm took an acceptable amount of time to evaluate all possible solutions.

Using DEMO we solved both variants of the problem. The parameters of DEMO were the following: population size 30, number of evaluations 3200, crossover probability 0.3, scaling factor 0.5, and the environment selection method SPEA2. The chosen number of evaluations is approximately one third of the number of evaluations performed by ES. We ran DEMO five times on both variants of the optimization problem.

6. RESULTS

Let us first look at the results found by ES on the discrete decision space. After solution evaluation for all possible combinations of input variables, we extracted and analyzed the feasible solutions. The proportion of feasible solutions is shown in Figure 2 and is rather small. As we can see, there are three subsets of solutions in the space of output variables. A detailed review shows that each subset corresponds to one of the three mold outlet coolant temperatures, while within each subset the solutions mainly differ according to the casting speed.

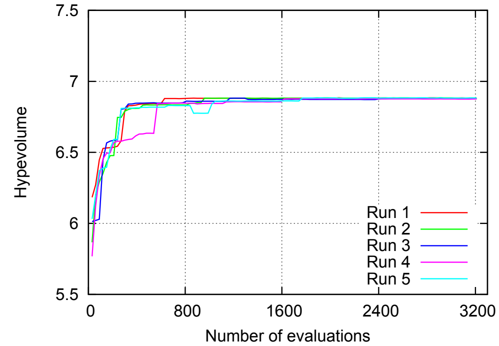

Figure 3: Hypervolume traces of optimization with DEMO on discrete decision space

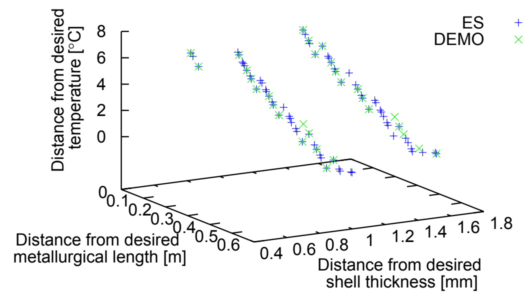

Figure 4: Fronts of non-dominated solutions found by ES and DEMO on discrete decision space

Next, we applied DEMO on discrete decision space. We ran it 5 times and the hypervolume traces gained in these runs are shown in Figure 3. Hypervolume is a measure for assessing the performance of a multiobjective optimization algorithm. Larger values of the hypervolume indicator denote better results. We can see that in all runs DEMO reaches approximately the same hypervolume. In fact, this hypervolume is reached after around 1800 solution evaluations. This means that from there on the front of non-dominated solutions is not improving significantly any more.

The fronts gained from ES and DEMO are very close or, in other words, the DEMO front is close to Pareto optimal front. The two fronts are shown in Figure 4.

The front produced by DEMO consists of 30 solutions which is equal to the population size. But during optimization process the algorithm evaluates numerous solutions, many of which are discarded because of the limited population size. To get a fair comparison with the Pareto optimal front gained with ES, we constructed the front of non-dominated solutions from all the evaluations that algorithm makes during the optimization process. This new front counts around 900 solutions and is almost identical to the front gained with ES.

For the second variant of the optimization problem with continuous decision space only the DEMO algorithm could be used. We run DEMO five times. The hypervolume traces from all runs converge slower than on discrete decision space, but their values are higher. The reason for this is that in the continuous decision space the algorithm has many more candidates to choose from.

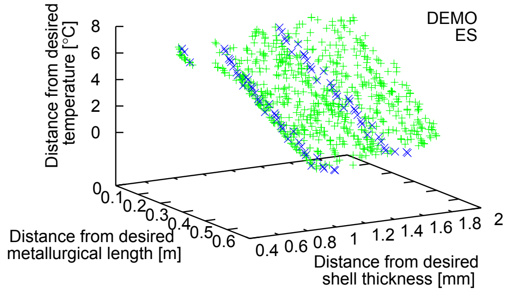

Figure 5: Front of non-dominated solutions found by ES on discrete decision space and front of non-dominated solutions found by DEMO from all evaluated solutions in continuous decision space

The comparison of the front of 30 non-dominated solutions gained with DEMO on continuous decision space and the front gained with ES on discrete decision space show that DEMO provides a better spread of solutions along the front.

The non-dominated front constructed from all evaluated solutions in comparison with front gained with ES is shown in Figure 5. Here we can see that for every solution gained with ES we can find a little better or the same solution gained with this algorithm. In addition, we find a lot of other non-dominated solutions. We can also see just how small proportion of solutions were found because of discretization.

7. CONCLUSIONS

We optimized the parameters of continuous steel casting based on a numerical model. Given the values of the input variables, the model simulated the process and calculated the values of the output variables. From these values we derived empirical optimization criteria. Finding the combination of the values of the input variables to minimize all criteria was the considered multiobjective optimization problem.

We dealt with two variants of this optimization problem. The first one was defined on discrete decision space and was solved with ES and DEMO algorithms, while the second one (continuous) and was solved only with DEMO. The obtained results were compared for effectiveness and efficiency. As expected, ES took the longest time, but the result was the Pareto optimal front for the given discretization. The results obtained with DEMO on the same discretization showed that the front of non-dominated solutions was very close to the front produced by ES. However the calculation time needed for optimization was three times shorter and as the hypervolume indicates, that stopping the optimization earlier, e.g. after about 1800 evaluations, would yield in the same result but in five times less evaluations then with ES. The non-dominated front gained with DEMO on continuous decision space was just a little better compared to the other fronts and the evaluation time was three times shorter than with ES.

In actual steel casting, however, the process parameters can only be set and measured with certain precision, therefore the finest reasonable discretization should be used in optimization. To conclude, the best way to solve this kind of optimization problems is to find the acceptable discretization and use an appropriate stochastic algorithm on this discretization.

Our future work will include optimization of the casting process based on meta models to further reduce the computational burden.

8. ACKNOWLEDGMENTS

The work presented in this paper was supported by the Slovenian Research Agency under projects L2-3651 and P2-0209. The authors are grateful do Dr. Robert Vertnik for making the casting process simulator available, and the Štore Steel company for providing the data.

9. REFERENCES

[1] N. Cheung and A. Garcia. The use of a heuristic search technique for the optimization of quality of steel billets produced by continuous casting. *Engineering Applications of Artificial Intelligence*, 14(2):229–238, April 2001.

[2] B. Filipič, T. Tušar, and E. Laitinen. Preliminary numerical experiments in multiobjective optimization of a metallurgical production process. *Informatica*, 31(2):233–240, June 2007.

[3] Z. Ji and Z. Xie. Multi-objective optimization of continuous casting billet based on ant colony system algorithm. In *PACIIA '08. Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 2008*, pages 262 – 266, 2008.

[4] A. V. Lotov, G. K. Kamenev, V. E. Berezkin, and K. Miettinen. Optimal control of cooling process in continuous casting of steel using a visualization-based multi-criteria approach. *Applied Mathematical Modelling*, 29(7):653–672, July 2005.

[5] K. Miettinen, M. M. Mäkelä, and T. Männikkö. Optimal control of continuous casting by nondifferentiable multiobjective optimization. *Computational Optimization and Applications*, 11(2):177–194, 1998.

[6] K. V. Price and R. Storn. Differential evolution – A simple evolution strategy for fast optimization. *Dr. Dobb's Journal*, 22(4):18–24, 1997.

[7] T. Robič and B. Filipič. DEMO: Differential evolution for multiobjective optimization. In *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization - EMO 2005*, pages 520–533, 2005.

[8] C. Santos, J. Spim, and A. Garcia. Mathematical modeling and optimization strategies (genetic algorithm and knowledge base) applied to the continuous casting of steel. *Engineering Applications of Artificial Intelligence*, 16(5-6):511–527, August-September 2003.

[9] R. Vertnik and B. Šarler. Simulation of continuous casting of steel by a meshless technique. *International Journal of Cast Metals Research*, 22(1-4):311–313, August 2009.

[10] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength pareto evolutionary algorithm. *Computer Engineering in Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland*, (Tik report no. 103), 2001.