A Multi-Objective Coil Route Planning System for the Steelmaking Industry Based on Evolutionary Algorithms

Gianluca Nastasi, Valentina Colla and Marco Del Seppia
Scuola Superiore Sant'Anna
Institute of Communication, Information and Perception Technologies, Pisa, Italy
Email: g.nastasifv.colla, m.delseppiag@sssup.it

Abstract **- In this paper a novel route planning system for steel coils that must pass through different processing steps of a generic steelmaking plant will be presented. Production times and costs are often the only considered indicators by traditional planning systems, while, with this newly proposed approach, customers' quality requirements are also taken into account. In facts, in medium/large steelmaking plants there could be different processing line that perform the same processing step, but with different characteristics (e.g. in terms of flatness, crossbow, etc.). Therefore, the final quality of a coil greatly depends on the route it follows among the different processing lines. Moreover, over-quality, i.e. assigning a high quality coil to a less demanding customer order, must be avoided too. The proposed system is based on Multi-Objective Optimisation and, in particular, it can exploit different paradigms of Multi-Objective Evolutionary Algorithms. The planning system has been developed in C++ (for the optimisation module) and C# (for the graphical user interface). One of its key features is that it is highly configurable so that it can be easily adapted to several real industrial scenarios by means of simple XML configurations file describing the plant and the quality indicators to take into account during the optimisation process.**

Keywords - optimisation; planning system; steelmaking; route planning; genetic algorithms; evolutionary algorithm; multi-objective optimisation; cold rolling; annealing; temper mill; finishing.

I. INTRODUCTION

In the steelmaking industry, as in other industrial fields, production planning is an essential key point for minimising costs and maximising productivity. Production planning systems appeared in metal and steel industry in the early '80s and are nowadays quite diffused, as many plant builders and automation suppliers offer such systems (e.g. Siemens VAI [1] and PSI [2]). However planning is typically performed in a suboptimal way, sometimes applying heuristic solutions [3, 4], without taking into account all the multiple, and sometime conflicting, objective functions. In many practical cases the planning is even manually performed by the people of the planning department. Moreover, even if an optimisation algorithm is applied, the objective is often targeted on production time or costs, while neglecting quality aspects. A review of the production planning issues faced by the steelmaking industry is provided in [5].

In this paper a novel planning system based on Multi-Objective Optimization (MOO) [6] is described, which aims at optimising production routes for the cold rolling area of a generic steelmaking plant. The system relies on the development of prediction models for different Key Quality Indicators (KQIs) that estimate the occurrence of quality issues, such as flatness and crossbow defects and on the estimation of the throughput for each production line. In this way, customer requirements, quality issues and production capabilities and constraints can be taken into account.

In literature two major approaches to MOO can be found: in the first one, individual objective functions are combined into a single composite function transforming a MOO into a Single-Objective Optimisation (SOO), while in the second approach a set of optimal solutions is determined by exploiting the Pareto dominance definition, which aims at pointing out the set of solutions representing different tradeoffs between the objectives [7]. Both approaches present advantages and some drawbacks. As far as the first approach is concerned, the greater advantage lies in the simple formulation of the fitness function. On the other hand, the disadvantage lies both in the impossibility sometimes to combine all objective functions into a single function and in the proper selection of weighting functions characterising decision-makers preferences. Moreover, solutions found by means of a weighted function strongly depend on how good is the choice of the weights themselves [7]. Instead, by means of the second approach, an entire set of Paretooptimal solutions can be obtained with a single run of the algorithm; in this way the decision-maker can select the preferred solution among a set of optimal ones. The main drawback of this solution is that with the increase of the number of objective functions, the definition of Pareto optimality begins to lose effectiveness.

Evolutionary algorithms (EA) techniques [8] have been applied to MOO (multi-objectives Evolutionary Algorithms - MOEAs) with the main advantage that the cost functions to be minimised can be arbitrary complex, non-linear and with complex constraints. Both cost functions and constraints can even be represented by means of software models/simulators of complex systems, because an analytical definition of the problem is not necessary.

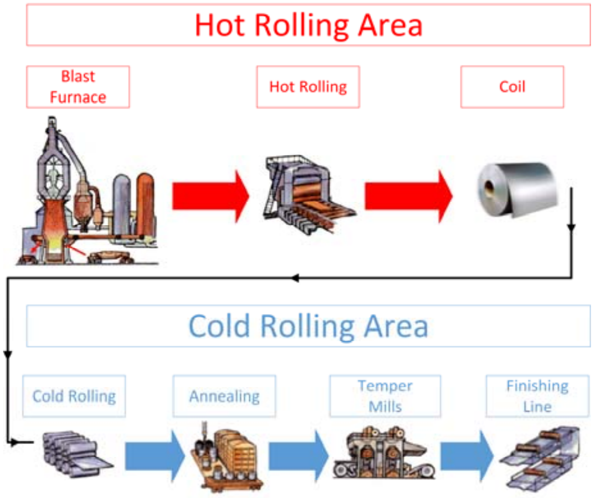

Figure 1. Plant diagram.

Several MOEAs paradigms based on the Pareto approach can be found in literature [9]: the main differences between them lie in the different strategies they employ to explore the solutions space as much as possible, preventing the algorithm from converging to a single region of the Pareto front. In the field of intelligent manufacturing, many applications of EA-based MOO can be found, also related to the steelmaking sector [10–18].

In the method presented in this paper, the Strength Pareto Evolutionary Algorithm 2 (SPEA2) [19, 20] has been successfully applied to the route planning optimisation problem. Nevertheless, the system has been designed in a modular way such that it is easy to implement and plug-in new MOEA paradigms.

In Sec. II the structure of a generic steelmaking plant is described, with particular focus on the cold rolling area; in Sec. III the optimisation problem is stated, describing the involved objective functions and constraints; in Sec. IV an insight on the software implementation is provided, while in Sec. V some results will be presented.

II. PLANT DESCRIPTION

Steelmaking involves several processes that transform raw materials into a finished product. Depending on the particular processing route followed by by-products, finished products with different characteristics may be produced (e.g zinc coated coils, annealed steel, etc.).

In this paper, the focus is on the set of processes that compose the so-called cold rolling area 1: the cold rolling mill (CRM), the continuous annealing (CA), the temper mill (TM) and the finishing line (FL). In the CRM a hot rolled coil is made thinner by passing it between two rolls at environment temperature, i.e. below the steel recrystallization temperature (here the name cold rolling, in opposition to hot rolling, where there is heat supply). The amount of strain introduced during the cold rolling process determines the hardness and other mechanical properties of the final product. The CA is a heat treatment that alters the physical properties of the steel strip and that is aimed at increasing its ductility, workability and homogeneity. In the CA the strip passes a series of furnaces with predetermined temperature profiles, according to the steel grade, where the strip is firstly pre-heated, then heated above a critical temperature for a certain amount of time and finally cooled down again. The main component of a TM is a cold rolling stand that produces the so-called temper pass, i.e. a limited rolling action that introduces an elongation typically between 0:5% and 2%. The tempering influences several physical and mechanical properties of the steel, such as increased yield strength, improved flatness and improved surface finishing, by making the steel useful for a wide variety of applications. Finally, FLs may involve different actions such as tension levelling, side trimming, cutting, welding, recoiling, etc. They do not modify the physical properties of the steel strip and they are employed mainly for packaging.

In a typical medium/large plant there could be different lines for each of the above described processes. Generally these lines may be produced by different suppliers and installed in different time periods so that they are different in terms of efficiency, quality outcomes and the severity of the defects they may introduce on the strip.

When scheduling routes among different possible lines for the same process, the operator or the planning system could have the tendency to saturate the line with best performances, because it ensures better results and products with fewer defects. Nevertheless, there could be customer orders with lower quality demand, which could be also satisfied by coils produced on less efficient lines. As a result, there is the risk to sell to these less-demanding customers high quality coils at a lower cost, so running into the socalled over-quality production, which should be avoided. In the next sections a novel planning system based on quality prediction and customers' demand is presented and described.

III. PROBLEM STATEMENT

The main objective of this new planning system is to optimise the routing of steel coils among different processing lines of the above described processes in order to satisfy a set of KQIs defined on the basis of the customer orders and on production constraints. Thus, the

DOI 10.5013/IJSSST.a.16.01.06

ISSN: 1473-804x online, 1473-8031 print

quality demands expressed by the customer orders influence the routing chosen by the planning system. Let us consider an example where customer A wants to produce tubes, while customer B wants to produce panels for kitchens.

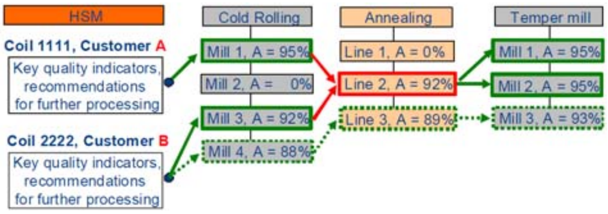

Figure 2. Multi-criteria optimisation of production routes considering through process quality and production output [23].

Both of them required the same kind of steel, but with different quality requirements: customer A needs small thickness tolerance for welding, while customer B requires a good surface quality and good flatness for aesthetic reasons. The figure shows that the final KQIs may vary depending on the particular chosen route, so that the route chosen for customer A may not meet the requirements of customer B and vice versa. Moreover, as described in the previous section, over-quality production must be avoided and therefore the route with the least deviation from the quality goal should be chosen. This has two effects: better processing lines (e.g. those that are newer or that had a better or more recent maintenance) are not saturated and products are not sold at a lower price with respect to their quality.

In order to develop such a planning system KQIs prediction models must be developed for each process step and each processing line. They could be based for example on statistic considerations, neural networks or fuzzy systems [21, 22]. However, the development of these prediction models is not object of this paper, so that they will be assumed as additional inputs to the planning system (see Sec. IV). The input of these models are coil characteristics (e.g. steel grade, thickness, width, KQIs measured or estimated at the previous step, etc.) and process parameters (e.g. rolling force, temperature profiles, etc.), while their output is a value in the range [0; 1], expressing the amount and severity of that particular defect they are modelling (e.g. flatness, crossbow, crown, etc.), where 0 means absence of defect and 1 the maximum severity. A model of the amount of time required to process a coil in each particular line is also necessary to assess their workload and, thus, to control their saturation. Fig. 2 shows an example where a possible saturation of the second line of the CA could be avoided by considering the workload as an additional objective for the optimization.

The other inputs to the planning system are the list of the hot rolled coils that must be worked in the cold rolling area, where their physical characteristics and the

KQIs actually measured at the exit of the hot rolling mill (HRM) are reported, and the list of customer orders to satisfy, which specify the characteristics of the desired coil (such as thickness, steel grade, etc.) and, for each KQI, the range of acceptable values (e.g flatness between 0:20 and 0:25 and crossbow less than 0:15). In this way the planning system can assess, for each potential route of each coil to process, their final KQIs and it can try to match in the best way final products with customer orders.

In this paper the KQIs for flatness, crossbow and workload will be considered as objective functions of the optimisation algorithm. However, the main target is not to minimize them (otherwise the solution would run into over-quality production), but to respect certain limits. Therefore the problem has been defined as a goal programming formulation, where the deviations of a solution from the requirements expressed by customer orders is assessed by means of KQIs models and then they are used as variables to minimise.

A. Maximum Bipartite Matching

The described routing optimisation problem can be formulated in the following manner. Let us consider a set of N hot-rolled coils that must be processed in the cold rolling area. First of all, coils must be matched to orders on the basis of physical characteristics. For instance, during the cold rolling process, coil thickness reduction is typically constrained between the 60% and the 80%; thus a hot rolled coil can be assigned to an order if and only if the cold rolling is able to produce a final product with the requested thickness. Other matching constraints concern steel grade, which must be the same, the length of the coil, which must be longer or equal, and other physical parameters. The final objective of this procedure is to assure that a coil can be transformed into the required final product, respecting the whole set of constraints. In order to solve this multi-criteria matching problem, a Maximum Bipartite Matching (MBM) algorithm has been employed. MBM is based on graph theory: hot-rolled coils form the set of source nodes, while orders forms the set of sink nodes. If a coil satisfy all the constraints defined by an order, an arc can be drawn between the two. The graph obtained at the end of this procedure represents all the possible matching between coils and orders (Fig. 3a). The problem can be solved by transforming it in a flow network and by applying a Maximum Flow algorithms, such as the Ford-Fulkerson algorithm [24] (Fig. 3b), in order to select the optimal set of arc that maximizes the number of matches considering that only one arc can insist on a coil or a customer order. Furthermore, a cost can be also assigned to each edge in order to give priority to certain orders (e.g. orders with a close delivery deadline, or coils with higher prices). In this second scenario MBM minimizes the total cost.

Thus, after the MBM algorithm is applied, a list of coilorder pairs is obtained. Of course it is not guaranteed that coils can satisfy all orders and it mainly depends on the hot-rolled coils production planning performed in the hotrolling area. Only coils and orders that can be matched will be considered in the next steps of the route planning optimisation.

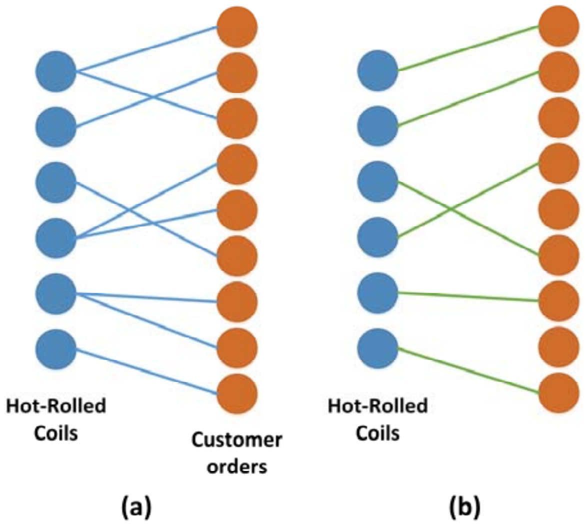

Figure 3. Maximum Bipartite Matching algorithm

B. Planning optimisation

Let us consider the variable Ri as a possible routing for the i-th coil: Ri can be represented by an integer array with a number of element equal to the processing steps a coil should pass (in the case described in this paper it has 4 elements). Each element value represents the particular processing line where the coil should pass in that processing step (e.g. the second line of the CRM). The set of routings for all the N hot rolled coils is thus $\mathbf{R} = \{R_1, R_2, \ldots R_N\}$. On the base of R_i a set of M final KQIs for the i-th coil route $K(Ri) = \{KQI_1(Ri), KQI_2(Ri), \ldots, KQI_M(Ri)\}$ can be evaluated using a "waterfall" prediction approach (Fig. 4), i.e. at each step KQIs are calculated on the basis of KQIs estimated at the previous step. For each coil and each KQI a deviation from the goal defined by the customers' requirements can be calculated as $D_j(R_i) = K_j(R_i) - G_{ij}$, where $D_j(R_i)$ is deviation of the j-th KQI of the i-th coil route from its goal G_{ij} . The average deviation of the j-th KQI considering the complete routing of the N coils $\mathbf{R}$ is thus calculated as

$$\bar{D}_j(\mathbf{R}) = \frac{\sum_{i=1}^{N} D_j(R_i)}{N} = \frac{\sum_{i=1}^{N} K_j(R_i) - G_{ij}}{N} \qquad (1)$$

Moreover, the workload KQI is evaluated for each line by means of an ad hoc model, which outputs a value Wsl 2 [0; 1] expressing the percentage of the total working time that the l-th line of the s-th processing step had worked. For each line the workload goal is equal to 1=Ls where Ls is the number of processing line in the step s, i.e. the goal is to have the workload balanced among all the lines composing a processing step. So in this case the average deviation is defined as

$$\bar{D}_w(\mathbf{R}) = \max_s \frac{\sum_{l=1}^{L} W_s l(\mathbf{R}) - 1/L_s}{L_s} \qquad (2)$$

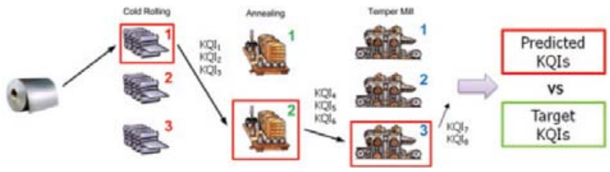

Figure 4. Example of a process route simulation.

The set:

$$\bar{\mathbf{D}} = \{\bar{D}_1, \bar{D}_2, \ldots, \bar{D}_M, \bar{D}_w\}$$

represents the set of cost functions to be minimized by the MOO algorithm.

C. Strength Pareto Evolutionary Algorithm 2

In order to cope with such a problem, a particular paradigm of MOEAs, i.e. the SPEA2, has been employed [20]. MOEAs are are iterative algorithms that explore the solutions space by means of mechanisms inspired by biology and evolution theory, such as mutation and crossover. In these methods, a so-called population of candidate solutions evolves generation by generation (i.e. step by step) by means of the previously cited genetic operators. The best solutions, the ones with higher fitness values (i.e. those that minimize more the objective functions), have more chances to survive to the next generation (elitarism) and to mate with other solutions.

In non-trivial MOO problems there isn't a single optimum, but a possibly infinite set of optimal solution, each one representing a different trade-off between the objective functions. The most of MOEAs paradigms are in fact based on the Pareto dominance definition [19], which is used toidentify non-dominated solutions in the objectives space. This set of solutions is generally refined and improved at each generation of the MOEA, until it doesn't approximates the actual Pareto front in the

DOI 10.5013/IJSSST.a.16.01.06

ISSN: 1473-804x online, 1473-8031 print

objectives space, and thus finding the corresponding Pareto set of optimal solutions.

The main characteristics of MOEAs are: they approximate an entire Pareto front in a single run of the algorithm; they can be employed also in presence of complex constraints and, moreover, they can be employed when an analytical description of he problem is unavailable. This last feature allows the employment of software simulator in place of the analytical formulation and it represent the main reason why MOEAs has been chosen for the routing optimisation task.

The steps on which SPEA2, and thus the route planning optimisation algorithm, are based are the following (Fig. 5):

1) An initial population R of randomly chosen routings is generated for each coil to be processed;

2) The fitness (objective) functions D are evaluated;

3) An archive of non-dominated solutions (Pareto front) is created/updated with this new generation;

4) Candidate solutions are divided into mating pools and then mutated and/or combined by means of genetic operators in order to create a new generation of solutions;

5) Go back to step 2 until some stop criterion is met;

6) At the last generation the archive of non-dominated solutions contains an approximation of the Pareto front and the corresponding Pareto set.

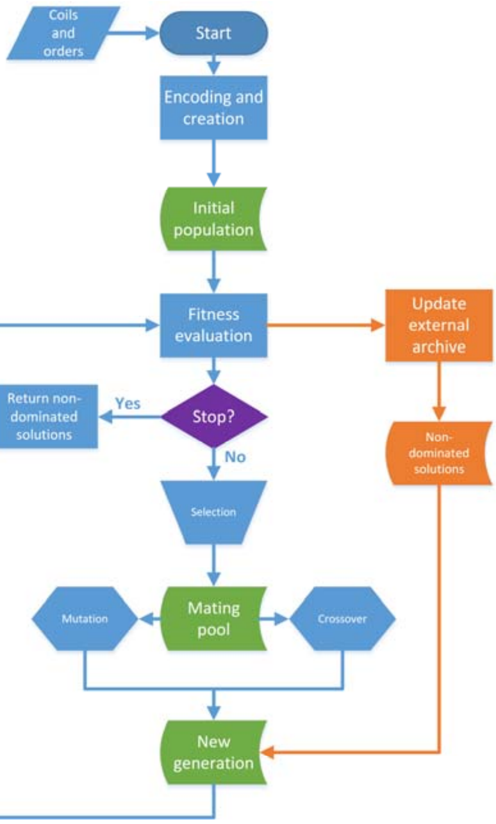

Figure 5. SPEA2 flow chart.

The main difference of SPEA2 with respect to other MOEA paradigms, is that it stores non-dominated solutions in an external archive as they are found. Besides, the archive can be truncated by means of a clustering algorithm if it overflows a certain predefined capacity. Solutions stored in the archive then concur to assign fitness values to the population, as well as they concur to tournament selections.

IV. IMPLEMENTATION

The planning system has been implemented in two main modules: on the one side, he plant simulator and the optimisation algorithm has been implemented as a Dynamic-Link Library (DLL) developed in C++, which ensures high elaboration efficiency and the possibility to be included in different, even already existing, systems, while, on the other side, the graphical user interface (GUI) has been implemented in C#.

```xml
<?xml version="1.0" encoding="utf-8"?>
<Tecplan>
    <GeneralInfo>
        <AppName>
            TecPlanSimulator
        </AppName>
        <Author>
            Scuola Superiore Sant'Anna
        </Author>
    </GeneralInfo>
    <PlantConfig>
        <ColdRolling lineNumbers="3" />
        <Annealing lineNumbers="3" />
        <TemperMill lineNumbers="3" />
        <Finishing lineNumbers="3" />
    </PlantConfig>
    <KQIConfig>
        <FlatnessModel path="flatness.dll" />
        <CrossbowModel path="flatness.dll" />
    </KQIConfig>
</Tecplan>
```

Figure 6. XML plant configuration file example.

The plant simulator is employed to predict the final KQIs of each coil and the workload of each processing line. It comprises different objects that can be further extended and specialized, which can be employed and combined in order to represent any plant. The plant is in facts dynamically configured by means of an XML (eXtended Mark-up Language) file where the plant structure is defined (Fig. 6). Moreover, in the XML configuration file external KQI models can be specified and plugged into the simulator. Therefore, the plant simulator is highly configurable and can be easily adapted to different industrial scenarios.

As far as the optimisation algorithm is considered, the SPEA2 paradigm has been implemented in C++ too and its use has been abstracted by means of the employment

DOI 10.5013/IJSSST.a.16.01.06

ISSN: 1473-804x online, 1473-8031 print

of the Startegy pattern (Fig. 7). It is a behavioural design pattern [25] that encapsulates an algorithm in an object with a determined interface. In this way different algorithms may be encapsulated in different objects with the same interface, making them interchangeable. The users of such objects don't need to make any assumptions on the particular strategy that has been instantiated and it can also be exchanged dynamically. In this particular implementation, the Context represented in

Fig. 7 in the planning system, Strategy is the common abstract interface for the concrete implementations (X and Y) of MOEAs paradigms (e.g. SPEA2 [20] and Non-dominated Sorting Genetic Algorithm II - NSGA-II [19, 26]). Thus more optimisation algorithms may be added and included also in future.

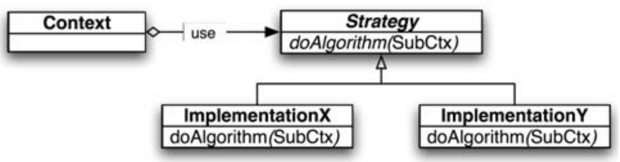

Figure 7. Strategy design pattern.

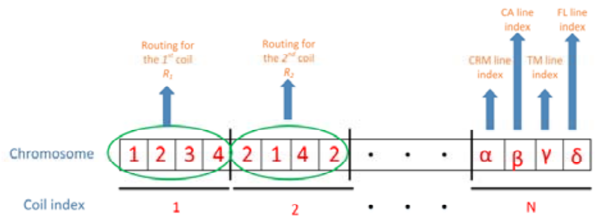

Figure 8. Chromosome encoding.

In order to apply MOEAs, two main functions have to be implemented:

- A encoding function that transforms a candidate solution into a chromosome (e.g. an array of bytes, of integers or of real values);
- A fitness function that calculates the values of the objective functions.

Moreover, if a particular chromosome structure has been defined, standard mutation and crossover functions need to be redefined.

In this scenario, the chromosome structure represent a particular $\mathbf{R}^*$, i.e. the set of the particular routings R_i^* for each coil to be worked. Each R_i^* is encoded by an integer array as described in Sec. III, and a chromosome is encoded as the sequence of all R_i^* , i.e. it is a sequence of integer arrays, and thus an integer array itself. The so defined structure is depicted in Fig. 8 The fitness function is called for each chromosome in the current generation. It decodes the chromosome, initialises the plant simulator by means of the XML configuration file and simulates the routes, obtaining the vector $\mathbf{D}^*$ of objective functions values (see Sec. III). The found nondominated solutions are added to an archive that, at the end of the iterations, will form the approximation of the Pareto Front.

A dedicated GUI has been developed to control and test the planning system optimiser. This GUI, depicted in Fig. 9, allows inserting the lists of coils to work and orders to satisfy, as well as the plant configuration file (Fig. 6). Once the optimisation process is finished, the GUI shows the results in different ways: the Pareto Set is shown in a table reporting the list of chromosomes as they are generated by the optimisation algorithm, while the Pareto Front is shown in a chart and the detailed routing information about each single line is accessible by means of the cold rolling area diagram. Moreover, results may be exported to text or Excel files for further analyses.

TABLE I. CUSTOMER ORDERS SPECIFICATION

Id_Order	CB_Target	FL_Target
1	0.45	0.2
2	0.1	0.05
3	0.3	0.3
4	0.4	0.4
5	0.12	0.45
6	0.15	0.2

V. RESULTS

After the implementation phase, tests to verify the correctness of the planning system have been performed, i.e. to check if the proposed optimal solutions were close to the actual optimal Pareto front. This condition is met when it is no more possible to minimise the value of an objective function without increasing the value of another one because of the presence of conflicting trade-offs.

In order to test the software different artificial input dataset has been created by varying the number N of hot rolled coils to process in 3 incremental steps (N = {3; 6; 9}), while fixing the following parameters:

- Plant configuration (i.e. 3 lines for each processing step);
- Steel grade of the coil;
- KQIs to evaluate (i.e. workload, flatness and crossbow);
- KQIs predicted values (in order to have reproducible runs);
- Employed MOEA pradigm (i.e. SPEA2);
- Maximum number of iteration of the evolutionary algorithm (equal to 100).

The file containing the order list has the format described in Table I in the case of N = 6: the first column indicates the order identification number, while the other two contain the customer requirements for crossbow and flatness. The coils list is simpler and contains just the IDs of the coils to be worked and the actual measured KQIs for flatness and crossbow, sampled at the exit of the hot rolling mill.

An example of obtained results for $N = 6$ is shown in Table II, where each row represents an optimal solution **R**, which is composed by the following columns:

- The list of matchings between coils and orders, composed by couples (*x*,*y*), where *x* is the coil ID and *y* is the order ID;
- The workload average deviation $\bar{D}_w$;
- The crossbow average deviation $\bar{D}_C B$;
- The flatness average deviation $\bar{D}_F L$;
- The optimal routing **R**, corresponding to the generated chromosome.

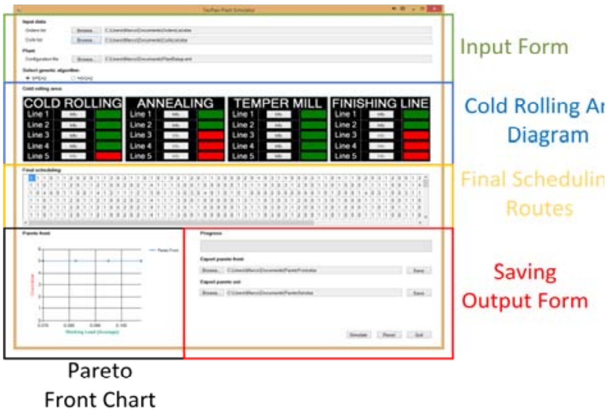

Figure 9. A screenshot of the graphical user interface.

TABLE II. TEST RESULTS

Matches	$\bar{D}_w$	$\bar{D}_{CB}$	$\bar{D}_{FL}$	R
(1,1) (2,2) (3,3) (4,4) (5,5) (6,6)	0.00	-0.03	-0.03	101101102202112200002221
(1,1) (2,2) (3,3) (4,4) (5,5) (6,6)	0.12	-0.25	0.05	101101102202101200002120
(1,1) (2,2) (3,3) (4,4) (5,5) (6,6)	0.08	-0.21	0.22	101101102202112200002120
(1,1) (2,2) (3,3) (4,4) (5,5) (6,6)	0.04	-0.05	-0.06	101101102202112200002220

Table II shows that the planning system has been able to equally distribute the workload between the various lines (the D w column has values close to 0). The global deviation is remarkably low for the first, the third and the fourth solution. The deviation for the crossbow is in average higher the the other KQIs. This can be due to the fact that the input coils had already too bad crossbow defects at the exit of the hot rolling mill to satisfy the customer requirements.

VI. CONCLUSIONS

This new planning system represents a new approach to coils route optimisation. It is deeply configurable, so that it is applicable to different real industrial scenarios, not only belonging to the steelmaking sector. One of its key feature is that it includes the evaluation of different KQIs representing customers' quality demands, which has to be satisfied by the system, while optimising also other variable, such as an equally distributed workload among the different production lines. The results are encouraging and, as soon as refined KQI models will be available, deeper tests can be performed, also on real scenarios.

ACKNOWLEDGEMENTS

The work described in the present paper was developed within the project entitled "TECPLAN - Technology-based assistance system for production planning in stainless steel mills" (Contract No. RFSR-CT-2011-00040) [23], and has received funding from the Research Fund for Coal and Steel of the European Union, which is gratefully acknowledged. The sole responsibility of the issues treated in the present paper lies with the authors; the Union is not responsible for any use that may be made of the information contained therein.

REFERENCES

[1] "Siemens vai," http://www.industry.siemens.com/verticals/metalsindustry, accessed: 2014-09-01.
[2] "PSI," http://www.psimetals.de/en/met-metals-solutions/metplanning, accessed: 2014-09-01.
[3] J. Miltenburg, "Production planning problem where products have alternative routings and bills-of-material," International Journal of Production Research, Vol. 39, 2001.
[4] L. Tang and C. Gao, "A modelling and tabu search heuristic for a continuous galvanizing line scheduling problem," ISIJ International, Vol. 49, pp. 375–384, 2009.
[5] L. Tang, J. Liu, A. Rong, and Z. Yang, "A review of planning and scheduling systems and methods for integrated steel production," European Journal of Operational Research, Vol.133, pp. 1–20, 2001.
[6] R. Steuer, "Multiple criteria optimization: Theory, computations, and application," 1986.
[7] P. Ngatchou, A. Zarei, and M. El-Sharkawi, "Pareto multi objective optimization," p. 84–91, 2005.
[8] D. E. Goldberg, "Genetic algorithms in search, optimisation, and machine learning," 1989.
[9] N. Srinivas and K. Deb, "Multiobjective optimization using non-dominated sorting in genetic algorithms," Journal of Evolutionary Computation, Vol. 2, pp. 221–248, 1995.
[10] V. Colla, G. Nastasi, N. Matarese, and L. Reyneri, "Ga-based solutions comparison for storage strategies optimization for an

automated warehouse," Proc. 9th Int. Conf. on Intelligent Systems Design and Applications (ISDA 2009), Pisa, Italy, Dec. 2009.

[11] V. Colla, G. Nastasi, N. Matarese, and A. Ucci, "Simulation of an automated warehouse for steel tubes," Proceedings – UKSim 10th International Conference on Computer Modelling and Simulation, EUROSIM/UKSim2008, pp. 150–155, 2008.

[12] S. Cateni, V. Colla, and M. Vannucci, "Variable selection through genetic algorithms for classification purposes," Proceedings of the 10th IASTED Int. Conf. on Artificial Intelligence and Applications, AIA 2010, Innsbruck, Austria, pp.6–11, Feb. 2010.

[13] V. Colla, G. Bioli, and M. Vannucci, "Model parameters optimisation for an industrial application: A comparison between traditional approaches and genetic algorithms," Proceedings - EMS 2008, European Modelling Symp., 2nd UKSim European Symp. on Computer Modelling and Simulation, Liverpool, UK, pp. 34–39, Sept. 2008.

[14] M. Vannucci, V. Colla, G. Bioli, and R. Valentini, "Two industrial problems solved through a novel optimization algorithm," Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), vol. 2774 PART 2, pp. 622–628, 2003.

[15] G. Porzio, B. Fornai, A. Amato, N. Matarese, M. Vaannucci, L. Chiappelli, and V. Colla, "Reducing the energy consumption and co2 emissions of energy intensive industries through decision support systems - an example of application to the steel industry," Applied Energy, vol. 112, pp. 818–833, Dec. 2013.

[16] G. Porzio, G. Nastasi, V. Colla, M. Vannucci, and M. Branca, "Comparison of multi-objective optimization techniques applied to off-gas management within an integrated steelwork," Applied Energy, vol. 136, pp. 1085–1097, 2014.

[17] G. Porzio, V. Colla, N. Matares, G. Nastasi, M. Branca, A. Amato, B. Fornai, M. Vannucci, and M. Bergamasco, "Process integration in energy and carbon intensive industries: an example of exploitation of optimization techniques and decision support," Applied Thermal Engineering, vol. 70, pp. 1148–1155, Sept. 2014.

[18] A. Maddaloni, G. Porzio, G.F.and Nastasi, M. Branca, and V. Colla, "Exploitation of multi-objective optimization in retrofit analysis: a case study for the iron and steel production," Energy Procedia, vol. 70, pp. 1148–1155, Sept. 2014.

[19] C. Coello, G. Lamont, and D. Van Veldhuizen, Evolutionary Algorithms For Solving Multi-Objective Problems, 2nd ed. Springer, 2002.

[20] E. Zitzler, M. Laumanns, and L. Thiele, "Spea2: Improving the strength pareto evolutionary algorithm," Kanpur Genetic Algorithms Laboratory, Tech. Rep., 2001.

[21] M. Vannucci, V. Colla, and G. Porzio, "Optimization of complex time consuming models of steel plants by means of ai techniques," IFAC Proceedings Volumes, pp. 202–207, 2012.

[22] R. Valentini, V. Colla, M. Sgarbi, and L. Reyneri, "Parametric jominy profiles predictor based on neural networks," Revista de Metalurgia (Madrid), vol. SPEC. VOL., pp. 314–317, 2005.

[23] "Tecplan - technology-based assistance system for production planning in stainless steel mills," EU RFCS Project, Tech. Rep., 2011, contract No. RFSR-CT-2011-00040.

[24] L. R. Ford Jr. and D. R. Fulkerson, "Maximal flow through a network," Canadian Journal of Mathematics, vol. 8, pp. 399–404, 1956.

[25] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, Design Patterns: Elements of Reusable Object-oriented Software. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1995.

[26] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," Evolutionary Computation, IEEE Transactions on, vol. 6, no. 2, pp. 182–197, 2002.