# A Constructive Cooperative Coevolutionary Algorithm Applied to Press Line Optimisation

Emile Glorieux[1*], Bo Svensson[1], Fredrik Danielsson[1], and Bengt Lennartson[1,2]

[1]Department of Engineering Science, University West, Trollhättan, Sweden

[2]Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden

***ABSTRACT***

*Simulation-based optimisation often considers computationally expensive problems. Successfully optimising such large scale and complex problems within a practical timeframe is a challenging task. Optimisation techniques to fulfil this need to be developed. A technique to address this involves decomposing the considered problem into smaller subproblems. These subproblems are then optimised separately. In this paper, an efficient algorithm for simulation-based optimisation is proposed. The proposed algorithm extends the cooperative coevolutionary algorithm, which optimises subproblems separately. To optimise the subproblems, the proposed algorithm enables using a deterministic algorithm, next to stochastic genetic algorithms, getting the flexibility of using either type. It also includes a constructive heuristic that creates good initial feasible solutions to reduce the number of fitness calculations. The extension enables solving complex, computationally expensive problems efficiently. The proposed algorithm has been applied on automated sheet metal press lines from the automotive industry. This is a highly complex optimisation problem due to its non-linearity and high dimensionality. The optimisation problem is to find control parameters that maximises the line's production rate. These control parameters determine velocities, time constants, and cam values for critical interactions between components. A simulation model is used for the fitness calculation during the optimisation. The results show that the proposed algorithm manages to solve the press line optimisation problem efficiently. This is a step forward in press line optimisation since this is to the authors' knowledge the first time a press line has been optimised efficiently in this way.*

## 1. INTRODUCTION

Practical optimisation problems often have high dimensionality and the fitness calculations are computationally expensive [1], e.g. in the case of simulation-based optimisation. The "curse-of-dimensionality" for optimisation problems states that high dimensionality results in an exponential difficulty [2]. An increase in difficulty is even more pronounced if there are nonlinear parameter interactions and a multimodal search space [3]. All this makes it very hard to find optimal solutions efficiently, within a practical timeframe. In engineering optimisation applications, the critical limitation for applying optimisation techniques on problems that are represented by computer simulation models, is the computational expense of the fitness calculations [4].

In this work, the considered engineering optimisation application is a sheet metal tandem press line. The optimisation problem is to find optimal values for the control system parameters [5]. The control system manages, among other things, the speed, the robot paths and the start/stop signals of the motions of the material handling robots and the presses in the press line. Nowadays, these process control parameters are determined online, manually, by trial and error. This approach is time consuming, not without risk of damaging the robot grippers or/and the press dies, and relatively unreliable since it is highly dependent on the experience of the operator. By using simulation-based optimisation techniques, optimal process control parameters can be determined more efficiently and reliably, and nearly without risks of damaging the equipment.

The press line problem is subject to the curse-of-dimensionality. Due to its complexity, multimodality and non-linearity, finding a near optimal solution might exceed the human's cognitive capabilities. There are over a hundred different parameters in the control system of an automated sheet metal press line. It is not practical to model it into a set of mathematical expressions. Instead, it was represented by a computer simulation model. The optimisation software works in concert with the simulation model to perform the fitness calculations. The simulation model provides the robots and presses motion profiles to calculate the fitness value for the trial solutions generated by the optimiser. Because of the high dimensionality and the computationally expensive fitness calculations, selecting a suitable optimisation algorithm or search procedure is challenging. It must be able to handle the high number of dimensions, while keeping the number of fitness calculations limited to solve it within the practical timeframe.

[*] Corresponding author: Tel.: +46 520 22 32 69; Fax: +46 520 22 30 99; E-mail: emile.glorieux@hv.se

A sheet metal tandem press line includes a number of press stations. The search procedure, developed in this work, utilises the intrinsic subproblems of the press line. These subproblems are the press stations in the line. As illustrated in Figure 1, there are two separate phases in the procedure, which is adopted from the existing Continuous Greedy Randomised Adaptive Search Procedure (CGRASP) [6]. During the first phase, a constructive heuristic builds up a feasible solution in a stepwise fashion. Starting in the first step with a single subproblem and adding a next subproblem in each step. This feasible solution is then used as an initial solution for the next phase. The second phase is the local improvement phase, during which the cooperative coevolutionary method is used to further improve the initial solution. This method optimises the subproblems separately whilst transferring information between the optimisations to allow collaboration [7]. This second phase is terminated when the search stagnates. Then, the search procedure returns to the first phase to produce a different feasible solution with the constructive heuristic. With this new feasible solution the second phase can be restarted. This is repeated until one of the predefined termination criteria is met, for example the maximum number of fitness calculations.

The main contribution of this paper is the proposed search procedure that integrates an extended version of the cooperative coevolutionary algorithm in the architecture of CGRASP. This proposed search procedure is able to solve the press line problem within a practical time frame. A new novel constructive heuristic for the CGRASP is proposed. Further, a new type of local improvement algorithm, the cooperative coevolutionary algorithm, is also proposed. A key feature is that this makes it possible to use a deterministic algorithm for the optimisation of the subproblems.

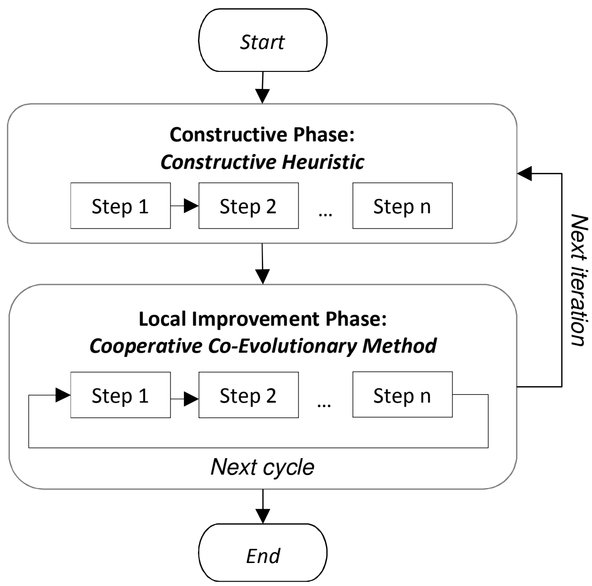


Figure 1. Illustration of the different phases of the proposed search procedure.

This paper includes an overview of relevant background information in Section 2. Also in Section 2, the press line problem is presented and the optimisation problem for the control parameters is described in detail. In Section 3, the proposed search procedure is presented and its different phases are described. Section 4 & 5 describe the implementation of the proposed search procedure and the results of the tests performed in this work.

## 2. BACKGROUND

### *2.1. SIMULATION-BASED OPTIMISATION*

Simulation-based Optimisation, or sometimes called Simulation Optimisation, involves using a computer simulation to model the optimisation problem in order to calculate the fitness of trial solutions [8]. A strength of simulation-based optimisation is that it eliminates the need to model a problem into a set of mathematical expressions. Thereby, it introduces the liberty to apply optimisation techniques on whichever problem that can be accurately represented by a computer simulation model. The fitness calculations are computationally expensive and are thereby the determining factor for the duration of the optimisation.

Simulation-based optimisation techniques have been proposed to solve different practical optimisation problems. Frantzén et al. [9] present an industrial application for scheduling and real-time rescheduling of complex machining lines. In this work, the simulation-based optimisation system is incorporated with a shop floor database system. This allows the simulation to use real time data from the factory floor. A discrete event simulation model is used to represent the production line in detail. In the presented system, different optimisation techniques to solve the scheduling problem were available but limited to discrete optimisation problems only. A decision support tool that uses simulation optimisation for assistance during operational and strategic decision making is presented by Melouk et al. [10]. This tool has been applied on a specific industrial case from the steel manufacturing industry. In this work, an off-the-shelf simulation-based optimisation software was used to solve the optimisation problem. No further investigation on different optimisation methods is presented in their work. These two examples, taken from the vast number of publications on this topic, demonstrate the possibility to use simulation-based optimisation to solve practical optimisation problems using real time data.

Simulation-based optimisation is a valuable technique due to the benefits of the usage of a simulation model rather than mathematical expressions. This benefit leads to the challenge to solve more complex optimisation problems often with a higher dimensionality. The latter, together with the computationally expensive fitness calculations, are among the factors that cause the need for specialised optimisation algorithms [11].

### *2.2. THE PRESS LINE PROBLEM*

A sheet metal press line includes a number of press stations in series, typically between four to six stations, as displayed in Figure 2. A press station includes a press, and material handling robots for loading/unloading sheet metal parts in or out of the press. In a tandem press line, these material handling robots are shared between consecutive stations in the line. The robot in one press station is responsible for loading the press of this station, but is also responsible for unloading the press of the previous station. In Figure 2, the dashed lines indicate the paths of the presses and the material handling robots in the press line. The material handling robots have a gripper that can pick up two plates, one from a table and one from a press, at the same time. The dotted lines represents the paths of how the plates traverse through the press line. Usually a sheet metal press line is fully automated. Hence, discrete events and motion control are handled by multiple industrial programmable logic controllers (PLC) and multiple robot controllers.

The press line problem is concerned with increasing the production rate by optimising the motions of the material handling robots in the line. Osakada et al. [12] showed that optimised motions of the robots can reduce their component handling time and thereby increasing the line's production rate. The individual robot motions must therefore be optimised according to the robot paths, cam values, velocities, etc. There are several critical interactions between the equipment that need to be synchronised to avoid collisions. Svensson et al. [13] indicated that it is critical that the synchronisation of robot motions must also be tuned accordingly to operate collision free.

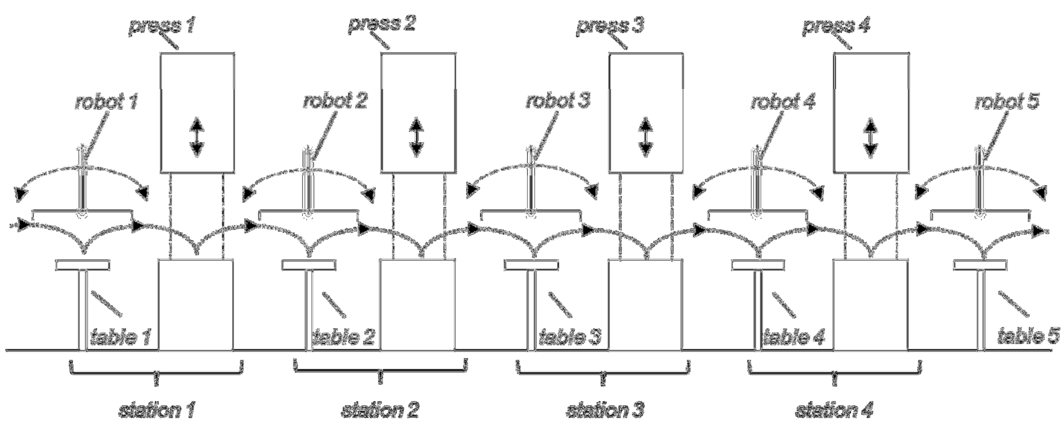


Figure 2. A 2D illustration of a sheet metal tandem press line that includes four stations.

In previous work on sheet metal press lines the focus has been on reducing design time and cost for the press dies by automation of the design and development process [14] and usage of computer-aided tools [15], on process planning of the sheet metal operations [16], [17], and on path simulation [18]–[20]. The optimisation of the synchronisations and the motions of the presses and robots in the line has in general been neglected or proposed to be done manually.

In Svensson et al. [13], a new optimisation algorithm for the press line problem has been presented. This new optimisation algorithm is based on a combination of two existing global and local optimisation algorithms; the Lipschitzian DIRECT and Nelder Mead simplex algorithm. This Combined Lipschitzian and Simplex

(CoLiS) optimisation algorithm is a deterministic algorithm and it was shown that it is well-suited for simulation-based optimisation. Due to the synergy of the global optimisation algorithm (DIRECT) and the local optimisation algorithm (Nelder Mead), the required number of functions evaluations was considerably less for solving the press line problem, compared with the two algorithms separately and also compared with an evolutionary algorithm. However, this work was restricted to an isolated single press station and not an entire press line.

A simulation-based optimisation method for a sheet metal press line has been presented in detail by Svensson et al. [21]. This method is illustrated in Figure 3. The parameter vector $\mathrm{p} = [\, p_1\, p_2 \ldots p_m\,]$ consists of all $m$ parameters $p_j$ of the press line that need to be optimised. The values of the parameter vector p are limited to the constraints $\mathrm{p} \in \mathbb{Q}_p \subseteq \mathbb{R}^m$ of the form $p_{j,min} \leq p_j \leq p_{j,max}$, where $p_{j,min}$ and $p_{j,max}$ are the lower and the upper bounds for the $j^{th}$ parameter value.

The fitness of the parameter vector p is calculated by a simulation model that represents the sheet metal press line. From the response function $h(\mathrm{p})$, including the motion profiles of the individual robots and presses, produced by the simulation model, the individual production performance values $g_i = g_i(h(\mathrm{p}))$ are calculated. In the objective function, these individual production performance values are then combined as follows

$$f(\mathrm{p}) = c_1 g_1(h(\mathrm{p})) + c_2 g_2(h(\mathrm{p})) + \cdots + c_r g_r(h(\mathrm{p}))$$

in which $c_i \geq 0$ are weight values for the individual production performance values. The weight values in the objective function are determined based on the desired industrial targets. For example industrial targets can be production rate, reduced energy consumption, or smooth motions of the robots to reduce the wear.

Global optimisation is the search for the solution vector of the minimum or the maximum in the problem search space. In this paper, the maximisation form of the global optimisation problem is adopted, which can in general be written as $\max f(\mathrm{p})$ , including the parameter constraint $\mathrm{p} \in \mathbb{Q}_p$. Then a feasible solution vector $\mathrm{p}^*$ is a global optimum if and only if $f(\mathrm{p}^*) \geq f(\mathrm{p}),\ \forall\ \mathrm{p} \in \mathbb{Q}_p$.

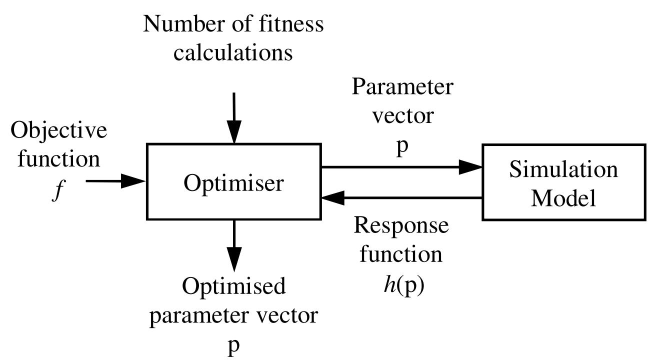


Figure 3. Illustration of the method for simulation-based optimisation, [13].

## 3. The Constructive Cooperative Coevolutionary Optimisation Algorithm

In this section, the proposed search procedure to solve the entire press line problem is described. The constructive cooperative coevolutionary search procedure is based on CGRASP [6] and embeds the cooperative coevolutionary algorithm [7]. Hence as with CGRASP, each iteration of this search procedure includes two phases: a constructive phase and a local improvement phase, as illustrated in Figure 1. The purpose of the constructive phase is to create a feasible solution for the entire problem, by stepwise searching subsolutions. A subsolution is a solution for one or more subproblems. In the second phase, the local improvement phase, this solution is further improved by employing the cooperative coevolutionary algorithm. An iteration is terminated when the local improvement phase, after one or more cycles gets trapped in a local optimum. The best solution $\mathrm{p}^*$ found over all iterations is the final result when the optimisation is terminated.

The optimisation problem, $\max f(\mathrm{p})$ is in this work decomposed into $n$ different subproblems $f(\mathrm{p}) = [f(\mathrm{p}_1)\ f(\mathrm{p}_2) \ldots f(\mathrm{p}_n)]$, where we define $\mathrm{p}_i$ as the subvector of parameters for a single subproblem $i$. Even if the optimisation problem is decomposed, all fitness values can be compared because all subproblems are handled in the same way as the entire optimisation problem.

### *3.1. CONSTRUCTIVE PHASE*

This section describes the proposed novel constructive heuristic that is employed in the constructive phase of the proposed search procedure. A solution for the entire problem is constructed by stepwise optimising all subproblems' subvectors $\mathrm{p}_i$ in the parameter vector p, while keeping $\mathrm{p}_{1,i-1}$ (where $\mathrm{p}_{1,i-1} = [\mathrm{p}_1 \dots \mathrm{p}_{i-1}]$) constant and not considering $\mathrm{p}_{i+1,n}$. In Step $i$, the parameter subvector $\mathrm{p}_i$ belonging to Subproblem $i$ is considered in parameter vector p, and the parameters $\mathrm{p}_{1,i-1}$ are kept fixed to the values found in previous steps. This is done until all $n$ subproblems have been considered. During each subproblem optimisation, the $k$ best subsolutions are recorded and afterwards stored to be further constructed in coming iterations. The best one of these $k$ recorded subsolutions is then selected to be used for the next steps of the current iteration. These stored subsolutions can be organised in a tree structure as shown in Figure 4. There, each level corresponds with a step and therefore represents the number of included subproblems in the subsolutions. The connections between the nodes represent which subsolutions belong together. It is prevented that the same subsolution appears twice in the tree structure. In Figure 4a, the tree of stored subsolution after the first iteration is shown.

At the start of next iteration, after completing the local improvement phase, the best stored unexplored subsolution in the tree is selected. In Figure 4b, $j_{II}$ is the best unexplored subsolution present in the tree structure and is selected to be used for this iteration. The remaining steps (from Step $i+1$ to Step $n$) are completed to obtain new feasible solution for the entire problem. In Figure 4b, the tree of stored subsolutions after the second iteration is shown. There, it can be seen that $k$ new feasible solutions for the entire problem have been constructed from the selected subsolution.

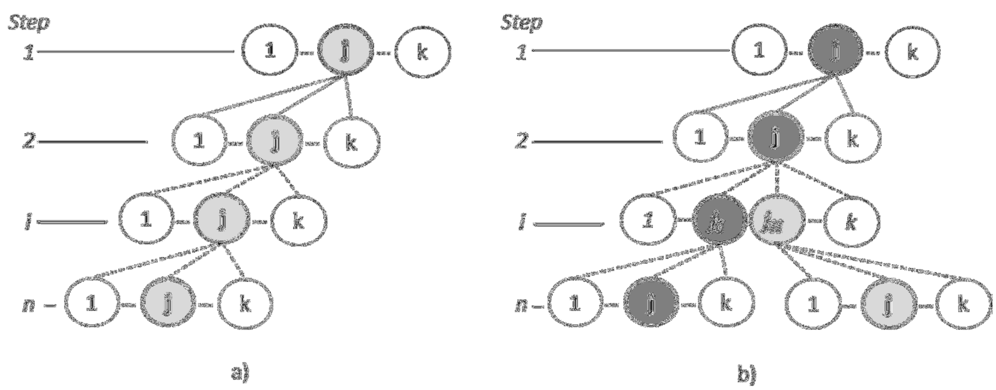


Figure 4. Illustration of the tree structure of the stored subsolutions during the constructive heuristic after the first iteration and second iteration.

This constructive heuristic finds a feasible solution for the entire problem by optimising several the "smaller" subproblems. By further constructing different unexplored subsolutions in each iteration, different regions of the search space are explored.

### *3.2. LOCAL IMPROVEMENT PHASE*

In this phase, a cooperative coevolutionary method is employed that optimises the $n$ subproblems separately. Then all $n$ solutions collaborate to locate the optima for the entire problem. In Figure 5, the different steps of a local improvement cycle are illustrated. This phase consists of one or more cycles. The $l^{\text{th}}$ cycle starts with a blackboard solution $\mathrm{p}^l$, which is either the solution obtained from the constructive phase (when $l = 1$) or is a blackboard solution calculated at the end of the previous cycle (when $l > 1$). The subproblems' parameter vectors are assembled in the blackboard solution $\mathrm{p}^l$ to perform a fitness calculation. Solution $\mathrm{p}_i^l$ is the candidate solution, which was the result of the subproblem optimisation in Step $i$. Note that in candidate solution $\mathrm{p}_i^l$ only the parameters belonging to Subproblem $i$ have a different value compared to in blackboard solution $\mathrm{p}^l$.

The solutions $\mathrm{p}_i^l : \forall\, i \in \{1,2,\cdots,n\}$, found during the $n$ steps of a cycle are the candidates to become the blackboard solution $\mathrm{p}^{l+1}$ for the next cycle. During the collaboration step of the cooperative coevolutionary method, the best solution among the candidate solutions is selected to become the next blackboard solution $p^{l+1}$. In the next cycle, the same is repeated but now the new blackboard solution $p^{l+1}$ is used. The local improvement phase is terminated when the search gets trapped in a local optimum. This is when the fitness value of the new blackboard solution is not better than the fitness value of the previous blackboard solution. When this occurs the iteration is ended, and the next iteration is started.

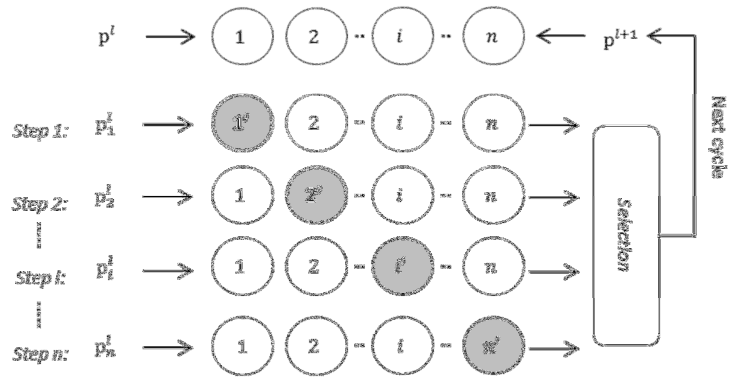


Figure 5. Illustration of the candidate solutions obtained after the different steps during one cycle of the cooperative coevolutionary method used during the local improvement phase.

## 4. IMPLEMENTATION

In this section, the implementation of the proposed search procedure to solve the press line problem is described. A real sheet metal press line from the Swedish car manufacturer, Volvo Car Manufacturing, has been used. The simulation model of the press line is developed in MATLAB and includes the control function for the robots and presses, and their sub functions. The simulated function for the motion planner is based on the robot motion controller used in the actual press line and has been verified for this purpose. In the particular simulation model that was used in this work, a collision detection method proposed by Nia et al. [22], was used to minimise the calculation time and thereby reducing the overall optimisation time. The optimisation algorithm used in this work, and the proposed search procedure, were written in C/C++.

In the tests performed in this work, the production rate of the press line was considered as the only optimisation target. Hence the production rate was the sole individual production performance value in the objective function. The production rate is expressed as the number of sheet metal plates produced per minute. The termination criterion used in the tests performed in this work was a fixed number of fitness calculations. This fixed number of evaluations in the tests was set to 20000. This is because the time required for computing 20000 evaluations (approximately 2.5 days on the used platform in this work) corresponds to the practical timeframe for press line optimisation. The values of $k$, for the number of subsolution that are recorded during each step of the constructive phase, was set to 5. The CoLiS optimisation algorithm was used for the subproblem optimisation.

The press line, which was subject of this study, includes four press stations. Between the presses, a manipulator is located that is responsible for relocation and reorientation of the plates before they are placed into the next press. The press line has five material handling robots standing. One in each station, and one additional robot after the last press to remove the sheet metal parts from the last press and place them onto a conveyor. The parameters of the extra last robot were not included in the optimisation. In this work, a subset of all control parameters of the entire line that need to be optimised was considered, motivated by Svensson et al. [13]. The total number of parameters was 44.

## 5. RESULTS & DISCUSSION

The results of the test with the proposed search procedure are presented in the graph in Figure 6. The vertical axis of the graph represents the line's production rate of the solution and the horizontal axis represents the number of fitness calculations. The shaded areas indicate the constructive phase. The numbers between the constructive phases, or between the shaped areas, are the number of cycles of the local improvement phase. During the constructive phase of the first iteration, no feasible solutions for the entire press line have been found yet, only subsolutions for a subset of subproblems. Therefore the graph shows a production rate of zero plates per minute. After the constructive phase, at the start of the local improvement phase, a relatively good feasible solution for the entire press line is found rapidly. This is because the local improvement phase starts from the feasible solution generated during the constructive phase. It can be assumed that the constructed feasible solution is relatively near an optimum in the search space. During the following iterations, several incrementally improved solutions are found.

It can be noticed that in three out of the four iterations of the local improvement phase, a better solution is found compared to the previously best solution found. This indicates that it is valuable to perform multiple iterations, by iteratively continuing the constructive heuristic and restarting the local improvement phase to explore different regions of the search space. The graph shows an improvement during the local improvement phase of the first and the third iteration. This shows that there are also gains with performing multiple cycles of the local improvement phase.

The termination criterion for the subproblem optimisation with the CoLiS algorithm is intrinsically. Therefore the number of fitness calculations cannot be set to a fixed value. Hence, a desired predefined number of fitness calculations is set and then CoLiS terminates when it has reached a local optimum. Therefore, during the proposed search procedure, the number of evaluations performed by the CoLiS algorithm varies over the different subproblems and during the different phases of the search procedure. On average 445 fitness calculations were done by the CoLiS algorithm on a subproblem before progressing to the next one.

The results also show a restriction with the proposed search procedure. There is a maximum number of fitness calculations for the proposed constructive heuristic to construct a feasible solution. As a rule of thumb, the time required for this amount of fitness calculations should be maximum 25% of the practical time frame for solving the entire problem. Otherwise, this search procedure, in combination with the chosen algorithm for the subproblem optimisation is not suited for the considered problem.

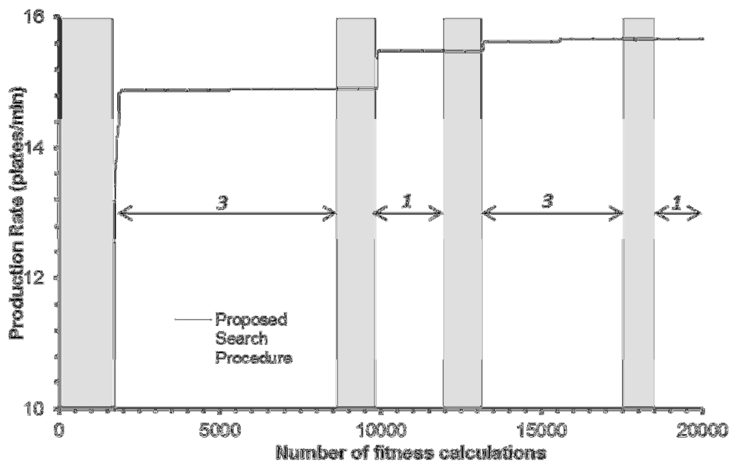


Figure 6. The results of the test with the proposed search procedure.

In this work, tests have also been performed using other classical optimisation methods next to the CoLiS optimisation algorithm. Initial results show that this proposed search procedure has a better performance than the standalone classical optimisation methods.

## 6. Conclusions & Future Work

A constructive cooperative coevolutionary search procedure inspired by CGRASP is presented in this paper and is applied on the press line problem. The results of the tests performed in this work show that the proposed search procedure manages to solve the press line problem within a practical timeframe. To the authors' knowledge, this is the first time the control parameters of an entire press line are optimised at once. Previous work was always limited to an isolated single press station.

The proposed search procedure requires that the optimisation problem can be decomposed into subproblems. An iteration of this search procedure includes a constructive phase and a local improvement phase. The presented constructive heuristic drives the search to explore different regions of the search space in every iteration. The results of the tests performed in this work show that it efficiently finds good initial feasible solutions for a large scale complex optimisation problem.

During the local improvement, every iteration the cooperative coevolutionary algorithm is started from different initial solution. Due to this, it is possible to also use deterministic optimisation algorithms to optimise the subproblems. In other work with the cooperative coevolutionary method, mainly genetic and evolutionary algorithms were used for the optimisation of the subproblems. The results of this work show that the cooperative coevolutionary method can successfully be extended to other algorithms than genetic and evolutionary algorithms.

Future work includes further testing with the proposed search procedure on different optimisation problems. Specifically on other optimisation problems that have a similar structure as the press line problem. This structure includes a set of subproblems which are connected in serial, and there are interdependencies amongst the subproblems. Further work should also include an extensive comparison of the performance of this search procedure with that of other existing optimisation algorithms and search procedures. A third element for future work is concerned with the collaboration mechanism of the cooperative coevolutionary method employed in the local improvement phase. In this work, this mechanism only selects the best candidate solution to become the blackboard solution for the next cycle. A more sophisticated collaboration mechanism should combine the candidate solutions to create an even better solution for the blackboard solution of the next cycle.

## REFERENCES


[1] *J. April, M. Better, F. Glover, and J. Kelly, "New advances and applications for marrying simulation and optimization," in Proceedings of the 2004 Winter Simulation Conference (WSC), Washington, D.C., USA, 2004, pp. 80–86.*

[2] *M. V. F. Pereira and L. M. V. G. Pinto, "Multi-stage stochastic optimization applied to energy planning," Mathematical Programming, vol. 52, no. 1–3, pp. 359–375, May 1991.*

[3] *J. Lundgren, M. Rönnqvist, and P. Värbrand, Optimization, 1:1 ed. Lund: Studentlitteratur AB, 2010.*

[4] *C. C. Tutum and J. Hattel, "State-of-the-Art Multi-Objective Optimisation of Manufacturing Processes Based on Thermo-Mechanical Simulations," in Multi-objective Evolutionary Optimisation for Product Design and Manufacturing, L. Wang, A. H. C. Ng, and K. Deb, Eds. Springer London, 2011, pp. 71–133.*

[5] *B. Svensson, F. Danielsson, and B. Lennartson, "Off-Line Optimisation of Complex Automated Production Lines - Applied on a Sheet-Metal Press Line," in Proceedings of the 2007 IEEE International Symposium on Assembly and Manufacturing (ISAM), Ann Arbor, Michigan, USA, 2007, pp. 82–87.*

[6] *M. J. Hirsch, C. N. Meneses, P. M. Pardalos, and M. G. C. Resende, "Global optimization by continuous grasp," Optimization Letters, vol. 1, no. 2, pp. 201–212, Mar. 2007.*

[7] *M. A. Potter, "The design and analysis of a computational model of cooperative coevolution," George Mason University, 1997.*

[8] *S. Andradóttir, "Simulation optimization," in Banks J (ed) Handbook of simulation: principles, methodology, advances, applications and practice, New York: John Wiley & Sons, 1998, pp. 307–333.*

[9] *M. Frantzén, A. H. C. Ng, and P. Moore, "A simulation-based scheduling system for real-time optimization and decision making support," Robotics and Computer-Integrated Manufacturing, vol. 27, no. 4, pp. 696–705, Aug. 2011.*

[10] *S. H. Melouk, N. K. Freeman, D. Miller, and M. Dunning, "Simulation optimization-based decision support tool for steel manufacturing," International Journal of Production Economics, vol. 141, no. 1, pp. 269–276, Jan. 2013.*

[11] *R. Pasupathy and S. G. Henderson, "SimOpt: A library of simulation optimization problems," in Proceedings of the 2011 Winter Simulation Conference (WSC), Phoenix, AZ, USA, 2011, pp. 4075–4085.*

[12] *K. Osakada, K. Mori, T. Altan, and P. Groche, "Mechanical servo press technology for metal forming," CIRP Annals - Manufacturing Technology, vol. 60, no. 2, pp. 651–672, 2011.*

[13] *B. Svensson, F. Danielsson, and B. Lennartson, "An efficient algorithm for press line optimisation," The International Journal of Advanced Manufacturing Technology, vol. 68, no. 5–8, pp. 1627–1638, Sep. 2013.*

[14] *B.-T. Lin, K.-M. Huang, K.-Y. Su, and C.-Y. Hsu, "Development of an automated structural design system for progressive dies," The International Journal of Advanced Manufacturing Technology, pp. 1–13, Apr. 2013.*

[15] *K. Debray, Y. M. Li, and Y. Q. Guo, "Parametric design and optimization of addendum surfaces for sheet metal forming process," International Journal of Material Forming, pp. 1–11, Jan. 2012.*

[16] *R. Singh and G. S. Sekhon, "PROPLAN: an expert system for optimal planning of sheet metal operations," Journal of Materials Processing Technology, vol. 166, no. 2, pp. 307–312, Aug. 2005.*

[17] *G.-C. Vosniakos and T. Giannakakis, "A knowledge-based manufacturing advisor for pressworked sheet metal parts," Journal of Intelligent Manufacturing, pp. 1–14, Jun. 2012.*

[18] *H. Li and D. Ceglarek, "Optimal Trajectory Planning For Material Handling of Compliant Sheet Metal Parts," Journal of Mechanical Design, vol. 124, no. 2, pp. 213–222, May 2002.*

[19] *X. Liao and G. G. Wang, "Evolutionary path planning for robot assisted part handling in sheet metal bending," Robotics and Computer-Integrated Manufacturing, vol. 19, no. 5, pp. 425–430, Oct. 2003.*

[20] *X. Yin, H. Wang, and G. Wu, "Path planning algorithm for bending robots," in Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guilin, China, 2009, pp. 392–395.*

[21] *B. Svensson, F. Danielsson, and B. Lennartson, "Time-synchronised hardware-in-the-loop simulation — Applied to sheet-metal press line optimisation," Control Engineering Practice, vol. 20, no. 8, pp. 792–804, Aug. 2012.*

[22] *N. K. Nia, F. Danielsson, and B. Lennartson, "A faster collision detection method applied on a sheet metal press line," in Proceedings of 21st International Conference on Flexible Automation and Intelligent Manufacturing (FAIM), Taichung, Taiwan, 2011, pp. 833–840.*