

What-if Game Simulation in Agent-based Strategic Production Planners

Paulo Leitão^{1,2}, Nelson Rodrigues^{1,2} and José Barbosa¹

¹ Polytechnic Institute of Bragança, Campus Sta Apolónia, Apartado 1134, 5301-857 Bragança, Portugal
{pleitao, nrodrigues, jbarbosa}@ipb.pt

² Artificial Intelligence and Computer Science Laboratory, Rua Campo Alegre 102, 4169-007 Porto, Portugal

Abstract— In the nowadays highly unstable manufacturing market, companies are faced, on a daily basis, with important strategic decisions, such as “does the company has the necessary capacity to accept a high volume order?” or “what measures need to be implemented if the product demand increases $x\%$ a year?”. Decision-makers, i.e. company’s managers, rely on their experience and insights supported by classical tools to take such decisions. Classical mathematical solvers or agent-based systems are typical architectural solutions to implement strategic planning tools to support decision-makers on this important task. Within the ARUM (Adaptive Production Management) project, a hybrid strategic planning tool was specified and developed, combining the optimization features of classical solvers with the flexibility and agility of agent systems. This paper briefly presents such architecture and focuses on the generation of the “what-if game” mechanism to support the generation of more intelligent and dynamic planning solutions.

Keywords— *Strategic production planning; Multi-agent Systems; What-if game simulation*

I. INTRODUCTION

The daily occurrence of unexpected events creates significant risks during the initial phase of the production of a product, e.g. in situations of peak of demand, late change requests and immature technology. This problem becomes harder to be handled when the products are made in short series, e.g., in aircraft and shipbuilding industries, since learning from the unexpected events becomes difficult and slow. The challenge is to consider innovative real-time decision-support tools that provide optimized solutions to respond faster and efficiently to these events, incorporating in a short notice the learning from their occurrence.

In particular, strategic production planning assumes a crucial importance in this scenario to support the decision-makers to take strategic decisions in short time with high confidence for situations where unexpected events may provoke strong impact on the running plans. As example, consider that a major customer asks a company to deliver within only a couple of months a number of products that almost exceeded its annual production volume. In this situation, the time to confirm or decline the request is very short, and any possible response strategy requires complex decisions about deep changes of current production and capacity plans, including the need to abandon a significant volume of orders from other customers.

The production planning refers to the elaboration of a plan, using an algorithm to optimize a problem subject to certain constraints according to a set of criteria, which contains decisions about the products and their quantities to be produced in the planning periods and about the needed capacity of the production environment, i.e., number of production lines, workers and shifts. The objective of the strategic planning is to maximize profit taking into account the finite capacity of the manufacturing system. Typically, these problems are hard to be solved, requiring significant computational resources and time, since they are NP-hard problems.

The production planning is traditionally faced by using a mathematical optimization solver to solve a complex problem determining the optimal solution for given constraints. These solvers are usually mature and robust computational applications that may implement different optimization algorithms, ranging from linear programming to meta-heuristics (see [1] for a comparative analysis). However, in spite of the achieved high optimization levels, these solvers lack the responsiveness to achieve solutions in short term and to dynamically produce different planning solutions by varying the problem constraints.

Bio-inspired algorithms are also being used as the means to address this problematic aiming to achieve solutions in a shorter time. Namely, the Ant Colony Optimization (ACO) method is used for demand forecast [2], and the Genetic Algorithm (GA) method is used to minimize production and inventory costs in the planning horizon [3] and to master production scheduling at the automotive supply chain level of small batches and multi-items aiming the improvement of payment and profit efficiency [4].

Multi-Agent Systems (MAS) principles [5-6] can be used to implement the optimization algorithm to provide a way to achieve an optimized solution in a more robust, flexible and agile manner. Examples of such approaches are reported in [7-10] and particularly an overview is found in [11]. In spite of the introduced responsiveness and flexibility, the MAS approach usually misses the achievement of an optimal solution. In this way, the MAS principles can be integrated with mathematical optimization solvers, combining the maturity, robustness and optimization of the solver with the flexibility and responsiveness of the MAS solutions. An example of the use of this hybrid approach to build optimization production tools is illustrated in [12].

An important issue to address the current requirements in terms of changing conditions and unpredictability is to consider what-if game simulation in the generation of production planning solutions, and particularly to achieve faster and more efficient alternative planning solutions. What-if game strategies were already used in other domains to provide the decision-makers with valuable information to support the decision-making. An example is described in [13], where a what-if game solution is applied to provide Small-Medium Enterprises (SME) with the capabilities for an efficient failure-based maintenance decision support. This approach allows to foresee, by varying hypothetically the inputs, how the system behaves, e.g., to detect the system deterioration.

The what-if game approach, not traditionally used in the production planning, can benefit by using the MAS infrastructure to implement the strategic planning system, allowing to explore the achievement of alternatives planning solutions. With this, the decision-maker is released from the sensitive task of properly selecting the most appropriate combination of problem constraints since the tool is able to iteratively provide them for the user's evaluation.

This paper describes the what-if game simulation functionality provided by a strategic planning system that combines the MAS principles and mathematical optimization solvers, aiming to provide mitigation strategies for the ramp up production phase of complex and highly customized products. This functionality supports the decision-makers to take strategic decisions on short time with higher confidence. Particularly, the insertion of the human-on-the-loop, at the what-if game simulation, enables a more reliable and faster solution achievement. Thus, the process follows a closed-loop organization, where the human serves as the main evaluator, while the tool itself manages to provide the most promising planning solutions in a shorter time.

The rest of the paper is organized as follows: Section II describes the architectural principles of the agent-based strategic planner and Section III introduces the conceptual model of the what-if game simulation. Section IV presents the embedded mechanisms to generate the scenarios and to analyze the planning solutions. Section V presents the experimental validation of the proposed approach and finally, Section VI rounds up the paper with the conclusions and points out the future work.

II. AGENT-BASED STRATEGIC PLANNING ARCHITECTURE

The strategic planner provides the production planning functionality related to the long-term production plans (i.e. from several months to years) that will be refined by scheduling and dispatching activities. In this work, it is assumed that the demand is given, or estimated, for a certain planning horizon, which is divided into planning periods.

The architectural design adopted in this work for the strategic planning combines a mathematical optimization solver, providing maturity, robustness and stability for the optimization algorithm, with a MAS infrastructure that offers the required flexibility to address the complex ramp-up production problem. The proposed approach is composed by a set of agents, each one possessing individual functionalities and knowledge, enabling

the accomplishment of the production planning requirements, particularly the generation of planning scenarios and the computation of the planning itself. The agency comprises the following types of agents [14]:

- *Resource agent (RA)*: represent the physical resources in the enterprise and are responsible to initiate the production planning process. Note that they may represent the enterprise, the facility or the production line according to the scope.
- *Scenario agent (SA)*: responsible to generate scenarios for the production planning, exploring different Degrees of Freedom (DoF), such as capacity expansion (e.g., introduction of new shifts, workers or overtime labor) and costs/revenues.
- *Planning agent (PA)*: responsible to apply optimization algorithms to find a solution for the planning problem based on a certain scenario. They possess the description of the problem to be solved, i.e. the mathematical formulation. The possibility to have several planning agents running simultaneously allows the parallelization of the planning solutions calculation (also with the possibility to use different optimization algorithms).
- *Simulation agent*: responsible to assess the production plans through simulation to anticipate the stochastic behavior of the production system.

The global system behavior, reflecting the functioning of the strategic planning, emerges from the interaction among the agents according to proper cooperation processes, as illustrated in Fig. 1. Basically, the resource agent, considering the problem description (i.e. the mathematical model and the DoF boundaries) provided by the decision-maker, asks the available scenario agents to generate a set of scenarios that can be used during the planning process. The scenario agents apply learning techniques to generate and select the best scenarios for the specific problem based on their experience.

The resource agent receives the list of scenarios and requests the planning agents to solve the planning problem instances associated with the selected scenarios. These agents use a solver, e.g., IBM ILOG CPLEX Optimizer or LP_SOLVER, running an optimization method, e.g., Mixed Integer Programming, considering the specific problem formulation and the defined scenario.

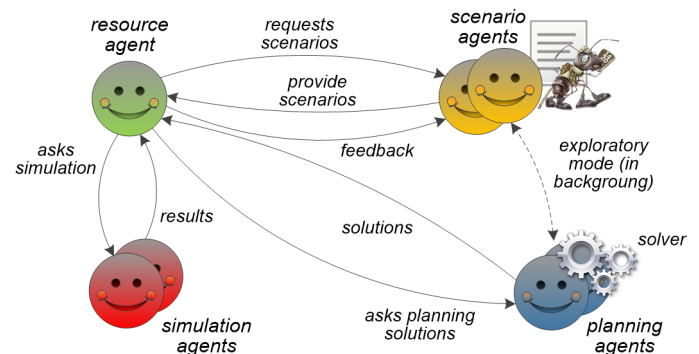


Fig. 1. Agent-based Strategic Planning Architecture.

At the end, the planning agent sends the achieved solution to the resource agent, which compiles and evaluates the received planning solutions and presents them in a sorted manner to the decision-maker. The agent, articulated with the decision-maker, can slightly modify the criteria and iteratively ask new scenarios to scenario agents or planning solutions to planning agents, or at the end to decide accepting one planning solution to be implemented.

An important aspect in such distributed system is to use an ontology to represent the knowledge exchanged among the agents, allowing a common understating during the cooperation processes. Fig. 2 represents, partially, the designed ontology schema for the planning application, [15] developed in the ARUM project [16].

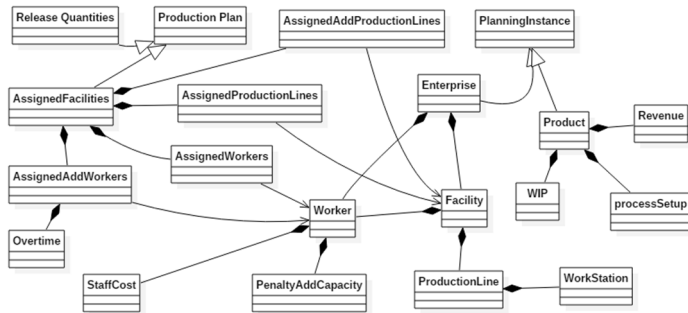


Fig. 2. Ontology schema for the knowledge representation.

Besides of the hybrid architecture, another innovative characteristic of the proposed strategic planner is to provide a what-if simulation playground to support the exploration of the best planning solutions by considering the proper scenarios. The what-if simulation appears by generating scenarios for the DoF boundaries, e.g., number of workers and number of production lines, and by analyzing the alternative planning solutions according to certain Key Performance Indicators (KPIs), e.g., backlog or profit.

III. PLAYGROUND FOR THE WHAT-IF GAME SIMULATION

The main innovation of this hybrid strategic planning approach is centered on using the MAS principles to provide a what-if game simulation playground to explore different DoF to expand the production capacity, supporting the decision-maker to take better and faster strategic decisions.

A. Definitions and Benefits of the What-if Game Simulation

The what-if game functionality supports the decision-maker to analyze what happens if some DoF are changed, and consequently be more prepared to select the best strategy to mitigate the unexpected event. Having this in mind, the elaboration of the production plan can be analyzed and simulated considering several DoF and taking into account various sources of uncertainty, for example, late supplies of components or product demand changes.

In fact, this simulation capability allows playing with certain DoF while answering to several what-if questions related to the planning problem, namely:

- What is the impact of cost and revenue changes on the production plans for specific scenarios?

- Is it worth to extend the capacity, e.g., hiring workers or setup additional production lines, to accommodate a demand increase for the next six months?
- Is it useful to increase the budget for capacity expansions given a certain demand scenario for the next three years?

Based on the obtained results, the decision-maker is able to take strategic decisions on short notice with a higher level of confidence, for example, with respect to the order acceptance or with respect to build a new facility in the next year.

B. Conceptual Model for the What-if Game Simulation

In the proposed approach, the what-if simulation playground comprises the following architectural issues:

- The boundaries of the selected DoF should be played (i.e. relaxed or narrowed) in a manually or automatically manner.
- A sub-set of the most efficient and promising scenarios and planning solutions should be considered by applying reasoning mechanisms and particularly learning techniques.

The conceptual model for the what-if game simulation playground requires a proper interaction among the distributed agents, as illustrated in Fig. 3, each one contributing with its individual behavior.

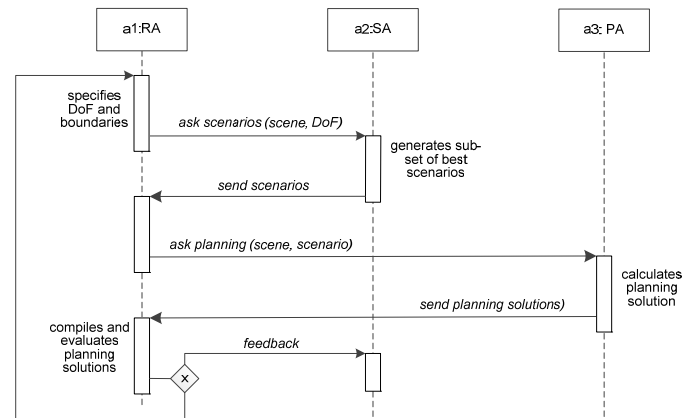


Fig. 3. Interaction pattern for the what-if game simulation.

To play a what-if game, the decision-maker, interacting with the resource agent, needs to define the KPIs and the boundaries of the DoF that will allow to explore different planning scenarios. The scenario agents explores the generation of different scenarios considering the boundaries of the DoF established by the resource agent, e.g., the number of shifts, workers, production lines or factories.

Appropriate reasoning and learning techniques will be applied to select the best scenarios from the search space. Note that the required computational time to solve the planning problems associated to all possible scenarios is usually very high. Considering only a sub-set of the best and promising scenarios, which reduces the exploratory space, the calculation time of the planning solutions is significantly reduced and

consequently being more effective in achieving faster the planning strategies.

For this purpose, the (positive and negative) feedback from previous iterations reflecting the quality of the planning solutions is used by the scenario agent to rank the scenarios according to pre-defined KPIs. Note that this feedback (allowing to reward the scenario according to the quality of the planning solution) is performed by the resource agent after receiving and evaluating the planning solutions. This means that when a future request for a similar problem appears, the scenario agent will search in its historical knowledge base for the best scenarios, avoiding in this way to create scenarios that will not produce good solutions. In background, the scenario agents can run exploratory procedures that consider other scenarios, not so well ranked, to prospect and evaluate their performance for specific situations, preventing also pitfalls of local minimums.

The resource agent uses the sub-set of scenarios (sorted according to rank of the scenario quality) to ask the planning solutions to the planning agents. The algorithm starts by considering the best scenario and stops when: i) all generated scenarios have planning solutions, or ii) there is a decreasing trend in the quality of the planning solutions. Also important is to avoid the recalculation of planning solutions for scenarios already considered in order to reduce the response time. This means that the resource agent should store the planning solutions for each analyzed scenario and should only ask the planning agents to calculate a planning solution for new scenarios.

The resource agents, after gathering the planning solutions for the generated scenarios, should perform an analysis of these alternative solutions and decide to accept them (sorting them according to pre-defined KPIs) or to start a new iterative round by adjusting the boundaries of the DoF if the achieved results are not satisfactory.

In this process, the use of learning mechanisms is important to improve the quality and time of convergence to the candidate scenarios and planning solutions and also to allow a better behavior in case of “bad events”.

IV. INTELLIGENT MECHANISMS SUPPORTING WHAT-IF GAME

This section describes the mechanisms embedded in the resource and scenario agents addressing the what-if simulation functionality. A special attention will be devoted to the description of how the sub-set of scenarios are generated and how the planning solutions are analyzed, sorted and accepted.

A. Generating Scenarios

One of the main responsibilities of the scenario agent is to generate a set of scenarios to be used by the resource agent during the production planning. The scenario agents manipulate the range of the DoF aiming to select a sub-set of scenarios from the search space (i.e. all DoF possible combinations) that better represents the production planning solutions to be tested (aiming to reduce the response time). The idea is to select only the most promising scenarios avoiding the need to test all possible scenarios.

The decisional engine behind the scenario agents will generate scenarios based on the DoF (and particularly their boundaries), and considering historical data from previous

iterations, where similar inputs can produce similar outputs, avoiding the testing of weak or already non-valid scenarios. This is important to narrow the search space to be covered by the strategic planning agent, by only considering the scenarios with high probability of leading to good planning solutions.

Fig. 4 illustrates the process to generate scenarios, particularly their sorting and selection, based on a kind of histogram learning.

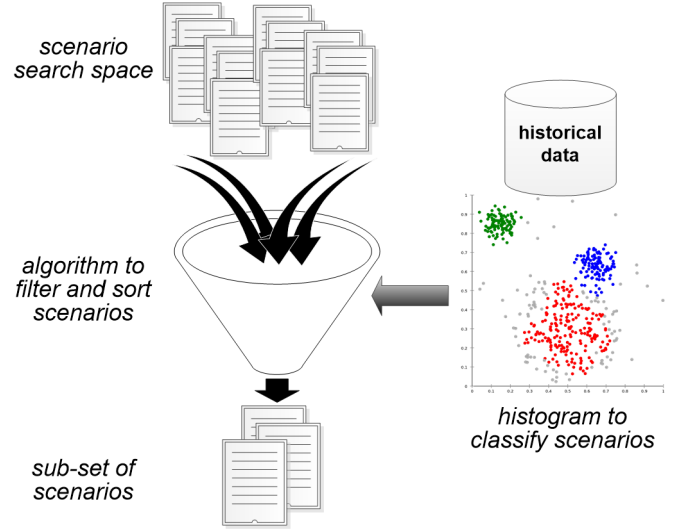


Fig. 4. Scenario generation process.

In this process, the scenario agent uses a matrix for each unexpected problem type (e.g., increase of demand or peak of demand), which contains the learning values for each scenario. Since each scenario is defined by the configuration of the different DoF, the matrix has as many dimensions as the number of DoF considered. The range of each dimension corresponds to the boundaries of the DoF of such dimension.

$$\begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mn} \end{bmatrix}$$

The learning values are reinforced based in the feedback of the quality of the planning solutions according to the selected KPIs:

$$A_{ij_{k+1}} = A_{ij_k} + f_p - f_w$$

where $A_{ij_{k+1}}$ is the new learning value, A_{ij_k} is the previous value, f_p is the positive feedback and f_w is the negative feedback. Note that the feedback values (from the user) are dependent of the quality of the planning solutions and can impact more or less the reinforcement process according to how good or bad the solution is.

By this, learning can be divided in two distinct phases where firstly the MAS is ranking the planned solutions based on the user pre-defined KPIs, e.g., backlog levels and solution cost, and secondly in the user’s solution perception. Both learning mechanisms are using the aforementioned reinforcement technique but with different reinforcement weights. Proper selection of the reinforcement weights must be perform, but the user’s natural solution perception has a higher weight. These

weights' choice is based on the fact that strategic planning is not a straightforward process, where the most profitable solution might not be executable. Note for instance a solution that considers an intermittent usage of production lines that is highly ranked by the MAS system. Since this solution may raise social instability, the decision-maker might prefer to discard it over other alternative solutions.

Aiming to improve the accuracy of the learning values about the scenario search space, being better prepared when a request for generating scenarios arrives, the scenario agent also performs exploratory testing in its idle time (i.e. in background). In this case, scenarios not well ranked or not yet considered in the past should be tested, updating its ranking and avoiding the non-detection of possible good future DoF combinations.

When a request to generate scenarios comes from the resource agent, and in order to select the most promising scenarios, the scenario agent will analyze its knowledge related to the previous scenarios results. For this purpose, the agent selects the proper matrix and extracts a sub matrix, considering the pre-defined DoF boundaries. Using this new matrix, the scenario agent will, based the proper learning matrix, extract the n most ranked scenarios for this planning problem type (sorted in terms of the received feedback reflecting the quality of previous planning solutions) and elaborates the sub-set of more promising scenarios to be sent to the resource agent.

The described approach runs iteratively when an existing problem type already exists in the agent knowledge base. However, when a new problem type appears, the matrix is built from scratch considering all possible scenarios. Using learning by analogy techniques, it is possible to accelerate the learning phase by considering similar existing problems. Another option is to consider the exploratory mode previously described.

B. Evaluating and Sorting Planning Solutions

The set of scenarios generated by the scenario agents are used to perform the production planning solutions by the planning agents using a mathematical solver. Each planning solution (associated to one scenario) is evaluated and ranked according to a KPI, being offered to the decision-maker a summary of the best solutions.

In this approach, the initial step is related to removing the planning solutions that don't fulfil the initial requirements, i.e. are not fulfilling the minimum acceptable values for the KPIs. As example, if the KPIs are the backlog and solution cost, the planning solutions with a backlog above the minimum threshold should not be considered and consequently removed. Note that initially, besides the DoF boundaries, the decision-maker also needs to define the number of planning solutions, n , and the minimum acceptable values for each KPI, i.e.

$$KPI_i < \delta_i$$

The second step is related to sort the achieved planning solutions and select the n best solutions considering individually each KPI and also considering a multi-criteria function that weights several KPIs. The n solutions for each individual KPI and multi-criteria function are shown to the decision-maker using a spider diagram, with the score of each solution for the KPI being shown in one graphic dimension. The decision-maker

can navigate between KPIs and also between planning solutions, and deeply analyze each one.

At the end, the decision-maker can select one planning solution or, if not satisfied, discard some provided solutions and ask again, in an iterative manner, for another set of n best solutions. Additionally, the decision-maker can take a more disruptive decision and adjust the DoF boundaries and start a new iteration in the what-if game simulation. This iterative procedure can be performed manually by the decision maker or automatically by the agent-based system, allowing to adjust the DoF boundaries for each round.

V. EXPERIMENTAL VALIDATION

The described what-if game simulation playground, included in the agent-based strategic planning system, was validated by considering a real case study related to a manufacturing company that produces modular equipment used during the airplanes' flights.

A. Implementation details

The agent-based strategic planning tool was developed using the JADE [17] agent-based development framework (see [14] for more technical implementation details). The planning agents use the ILOG CPLEX Optimizer solver to run the optimization techniques that solves the planning problem. The connection between the planning agent and the solver is performed through the ILOG Java API.

The previously described mechanisms to implement the what-if simulation were developed and embedded in the appropriate agents, i.e. resource and scenario agents. Particularly, a reinforcement learning mechanism was used to implement the learning mechanism considered to update the learning matrix. The resource agent, after receiving and sorting the planning solutions, updates the learning matrix rewarding more the best solutions.

Several User Interfaces (UIs) were developed as desktop applications built on top of the NetBeans platform, being used several charts, developed using the JFreeChart API, to interface the input of data and display the outputs of the planner.

Furthermore, the ARUM architecture is used as mean to integrate the UIs with the MAS strategic planner, offering the proper communication, service registration and services monitoring infrastructure [18]. Amongst others, the standard ESB communication protocols were enhanced with FIPA (Foundation for Intelligent Physical Agents) specifications, enabling their usage in more dynamic scenarios, such as those found during the strategic planning what-if game simulation.

B. Description of the Case Study

The company produces coffee machines, trash compactors, espresso and hot beverage makers, ovens and catering trolleys. Considering specifically the production of coffee makers, four production lines are dedicated to produce around 14 coffee machines per week. Each production line is composed by two working places, equipped with the necessary tools and accessories, where two skilled workers are performing the assembly steps.

The planning problem considered in this study is related to a problem of a coffee machine peak demand, which requires a strategic decision about how to balance the production to fulfil the book of orders with the limited resources. This peak of demand is characterized by a higher demand between the weeks 15 to 30 that is twice the normal demand. Considering the current operational configuration, i.e. 4 production lines and 40 working hours/week, the backlog is shown in Fig. 5. This backlog level is unaffordable and consequently some mitigation measures must be taken.

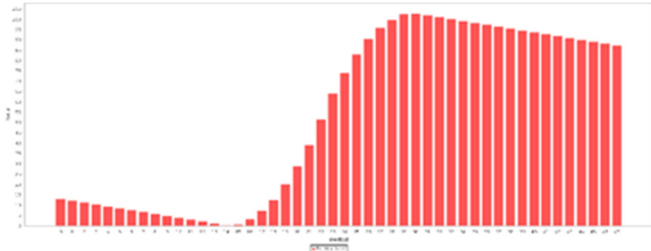


Fig. 5. Backlog for the peak demand without acting in DoF (max value: 102.8 units).

The strategic planner will assist the decision-makers to anticipate, assess and prepare mitigation strategies for this problem. For this purpose, the decision-maker play with the boundaries of some DoF related to the capacity expansion, namely additional production lines, additional workers and extra working hours. In this work, two DoF are considered: production lines with a range of $[0 .. 2]$ and overtime with a range of $[0 .. 10]$, meaning a maximum of two extra production lines and 2 extra working hours per day.

Playing with the DoF, the human decision-maker can reach alternative planning solutions, each one presenting different impact (e.g., in terms of backlog) but also costs. Note that the expansion of the production capacity has costs, e.g., the addition of an extra production line implies a fixed cost related to the setup of the working bench and a variable cost related to the workers' salary.

An UI serves as interface with the decision-maker to insert input data, define the DoF boundaries for the what-if game simulation and visualize alternative solutions.

C. Playing the Manual and iterative What-if Game

The first possibility is the manual adjustment of the DoF by the decision-maker aiming to achieve different alternative solutions. Fig. 6 depicts the backlog considering the possibility to use 2 extra production lines.

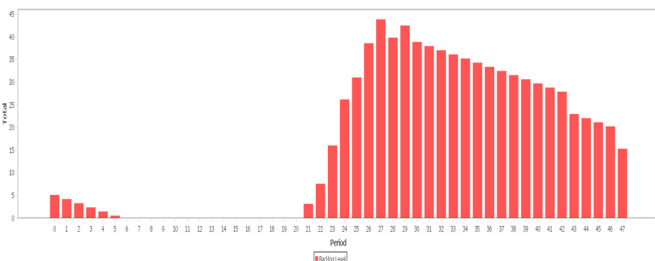


Fig. 6. Backlog for the peak demand considering extra production lines (max value: 43.8 units).

Despite the considerable reduction of the backlog, the use of the extra-production lines is intermittent, as seen in Fig. 7, which may be undesirable due to social instability.

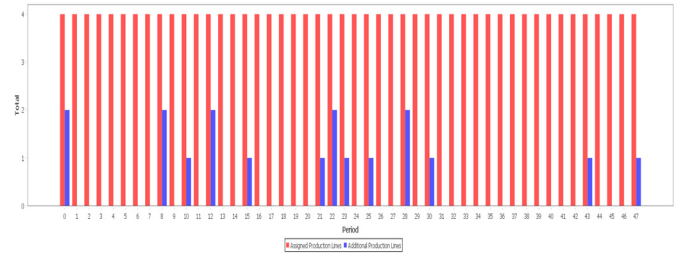


Fig. 7. Usage of production lines for the tested scenario (red: normal production lines, blue: extra production lines)

Further analysis can still be made, e.g. considering the use of overtime as an alternative to the extension of the production lines. Fig. 8 shows the backlog evolution for a scenario considering an overtime of 2 hours per day. The analysis of the results allows to conclude solution viability.

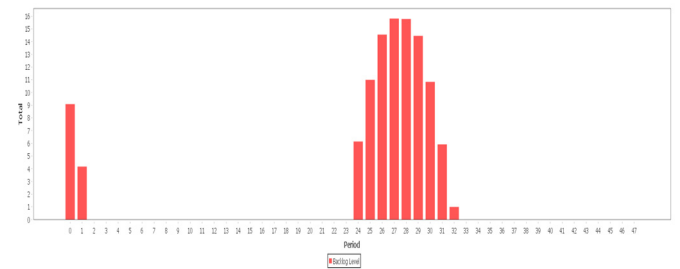


Fig. 8. Backlog for the peak demand considering overtime (max value: 5.8 units).

The exploration of DoF can be more elaborated considering a combination of both DoF. The backlog evolution allowing the usage of 2 extra production lines combined with a maximum of 10 weekly overtime hours is depicted in Fig. 9.

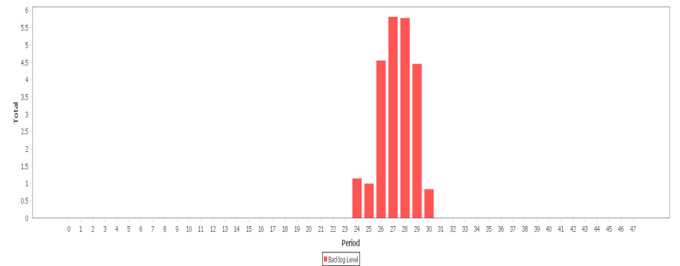


Fig. 9. Backlog for the peak demand considering overtime (max value: 15.8 units).

The tool also provides an estimation for the costs related to the capacity extension solution. In this way, the first case, acting only at the extra production lines, the solution has a cost of 9.640€, while on the second case is of 16.224€. Lastly, the combination of both DoF has an estimated cost of 18.094€, where from these 4.840€ are related to the extra production lines, while the rest is due to the use of overtime. With this information, i.e. backlog and cost of extending the capacity, the decision-maker can take the better decision to solve the peak demand problem.

As observed, the decision-maker is playing a manual what-if game simulation, varying each one of the available DoF until a satisfactory solution is encountered. However, this manual and iterative process is usually complex and very time consuming. An alternative is to use a more intelligent and automatic what-if game playground to achieve faster the sub-set of promising planning solutions.

D. Playing the Automatic What-if Game

In the automatic operating mode, and after the specification of the desired DoF and associated boundaries, the start of the what-if game is done by pressing the “Start Planning” button, which triggers the already described interaction in the MAS strategic planner.

A spider diagram summarizes the reached alternative planning solutions according to the pre-defined set of KPIs, as illustrated in Fig. 10. The decision-maker can have a detailed view of each solution, e.g., related to the number of production lines or workers and costs, by clicking on the solution line.

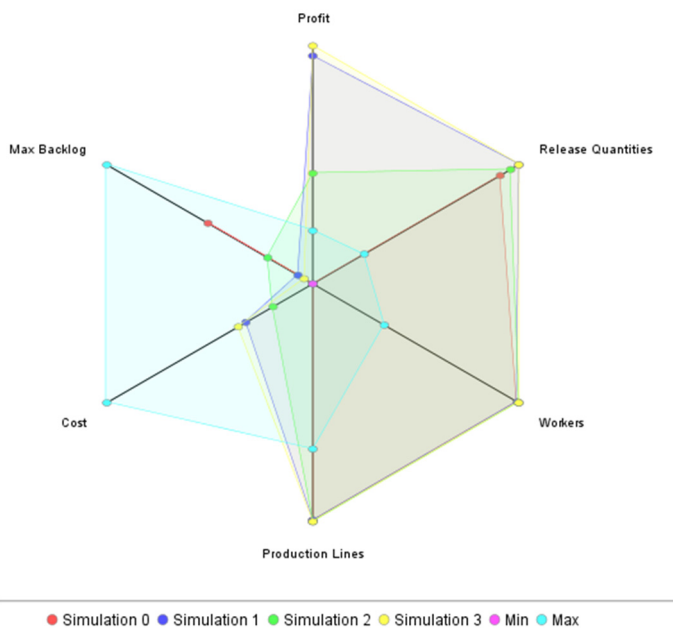


Fig. 10. Spider diagram to show the alternative planning solutions.

The manual what-if game playing is very time consuming. An experienced user, discarding the time needed to setup the initial scenario conditions, which is similar to what happens in the what-if game mode, would take around 1 minute between the parameters adjustment, planning and make a rough solution analysis. Considering that there are around 33 planning combinations (combining the described DoF and boundaries), the decision maker would need more than half an hour to analyze the situation. Additionally, non-valid solutions may appear in-between.

To have a confident what-if game mode, an initial training phase allowed the prospection of the most viable DoF combination to achieve the best planning solutions. This was considered for each type of problem type, namely for the increase, decrease and peak demand.

After this initial phase, the agents are able to provide to the decision-maker with a faster and more confident set of planning solutions. Based on this knowledge and on the number of desired, viable, planning solutions, the tool selects the most promising DoF, and provides the best solutions.

As described, the use of what-if game supports the decision-makers to reach faster and highly confident analysis of the alternative mitigation strategies and in this way to take the better strategic decisions in a short notice and with better confidence. Additionally, by being only provided with the best possible solutions, the decision-maker can concentrate more on selecting the best (on the human perspective) solution and not discarding the worst ones.

Finally, as the training/learning set increases, the tool gains more knowledge about how to combine the DoF for a particular scenario type. It is important to recall that varying the input conditions, e.g., situations for higher to lower demand increase, may impose the combination of DoF has the optimal combination.

VI. CONCLUSIONS

This paper describes an agent-based strategic planning system that combines the robustness and optimization of classical mathematical optimization solvers with the flexibility and responsiveness to achieve faster alternative planning solutions of the MAS infrastructure. This approach allows the achievement of several benefits, namely flexibility, robustness, and the dynamic achievement of exploratory alternative solutions.

Such intelligent agent-based solution allows the playing of iterative what-if game simulations for the most predictable scenarios, supporting decision-makers to take strategic decisions in a faster and efficient manner based on significant and effective alternative planning solutions.

Future work is related to consider more powerful learning algorithms to fuel the automatic and multi-round what-if game simulation playground. Additionally, the development of alternative planning modules, replacing the mathematical planner with either a heuristic-based or pure MAS based, is also in the horizon, making the tool model independent and therefore more generic.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007- 2013 under grant agreement n° 314056.

REFERENCES

- [1] B. Meindl, M. Tempel, “Analysis of Commercial and Free and Open Source Solvers for Linear Optimization Problems”, Eurostat and Statistics Netherlands within the project ESSnet on common tools and harmonised methodology for SDC in the ESS, 2012.
- [2] Z. Wu, C. Zhang, X. Zhu, “An Ant Colony Algorithm for Master Production Scheduling Optimization”, Proceedings of the 16th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD’12), 23-25 May, p.775-779, 2012.
- [3] Y. Qing-he, W. Wei, D. Xue-fei, “Simulation and Optimization of Carbon Enterprises Master Production Scheduling Based on Genetic Algorithm”,

- Proceedings of the International Conference on Computing, Control and Industrial Engineering (CCIE0'10), 5-6 June, pp.208-211, 2010.
- [4] X. Li, D. He, "Dynamic Master Production Scheduling for Automobile Mixed-Mode Assembly Line Based on the Rates of Payment and Profit", Proceedings of the Fifth International Conference on Computational and Information Sciences (ICCIS'13), 21-23 June, pp.1186-1189, 2013.
- [5] M. Wooldridge, "An Introduction to Multi-Agent Systems", John Wiley & Sons, 2002.
- [6] P. Leitão, "Agent-based Distributed Manufacturing Control: A State-of-the-art Survey", Engineering Applications of Artificial Intelligence, vol. 22, n. 7, pp. 979-991, 2009.
- [7] T. Hasegawa, L. Gou, S. Tamura, P.B. Luh, J.M. Oblak, "Holonc Planning and Scheduling Architecture for Manufacturing", Proc. of the International Working Conference on Cooperating Knowledge Based Systems, pp. 125-139, 1994.
- [8] B. Denkena, H.K. Tonshoff, M. Zwick, P. Woelk, "Process Planning and Scheduling with Multi-agent Systems" Knowledge and Technology Integration in Production and Services: Balancing Knowledge and Technology in Product and Service Life Cycle, pp. 339-348, 2002.
- [9] M. Caridi, S. Cavalieri. "Multi-agent Systems in Production Planning and Control: an Overview", Production Planning & Control, vol. 15, n. 2, pp. 106-118, 2004.
- [10] L. Florin, "Design of a Multiagent System for Solving Search Problems", Journal of Engineering Studies and Research, vol. 16, n. 3, pp. 51-64, 2010.
- [11] M. Caridi, S. Cavalieri, "Multi-agent systems in production planning and control: an overview", Production Planning & Control, vol. 15, n. 2, pp. 106-118, 2004.
- [12] L. Qin, Q. Li, "A New Construction of Job-Shop Scheduling System Integrating ILOG and MAS", Journal of Software, vol. 7, n. 2, pp. 269-276, 2012.
- [13] M. A. Burhanuddin, A. R. Ahmad, S. M. Halawani, "An Efficient Failure-Based Maintenance Decision Support System for Small and Medium Industries", INTECH Open Access Publisher, 2011.
- [14] A. Ferreira, A. Pereira, N. Rodrigues, J. Barbosa, P. Leitão, "Integration of an Agent-based Strategic Planner in an Enterprise Service Bus Ecosystem", Proceedings of the IEEE International Conference on Industrial Informatics (INDIN'15), 22-24 July, Cambridge, UK, 2015.
- [15] ARUM (Adaptive Production Management), "ARUM Ontology version-1", Deliverable 3.3.2, 30th September 2013.
- [16] ARUM - Adaptive Production Management, at: <http://www.arum-project.eu/> (accessed in 13/02/15).
- [17] F. Bellifemine, G. Caire, D. Greenwood, "Developing Multi-Agent Systems with JADE", Wiley, 2007.
- [18] C. Marin, L. Moench, P. Leitao, P. Vrba, D. Kazanskaia, V. Chepegin, L. Liu, N. Mehandjiev, "A Conceptual Architecture Based on Intelligent Services for Manufacturing Support Systems", Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC'13), 13-16 October, Manchester, UK, pp. 4749-4754, 2013.