Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds.

CONSISTENT USE OF EMULATION ACROSS DIFFERENT STAGES OF PLANT DEVELOPMENT – THE CASE OF DEADLOCK AVOIDANCE FOR CYCLIC CUT-TO-SIZE PROCESSES

Ruth Fleisch
Robert Schoech
Thorsten Prante

V-Research GmbH
Stadtstrasse 33
A-6850 Dornbirn, AUSTRIA

Robert Pflegerl

Schelling Anlagenbau GmbH
Gebhard-Schwärzler-Strasse 34
A-6858 Schwarzach, AUSTRIA

ABSTRACT

This paper presents an emulation tool and the way in which it supports the test-driven design of zero-deadlock cut-to-size plants. In contrast to previous plant layouts featuring a linear material flow, panels can now be allocated to the same saw again for their further partitioning. The resulting circles in the material flow of cut-to-size plants impose new challenges as they may cause deadlocks, which means that the manufacturing system or parts of it remains indefinitely blocked. In order to prevent deadlock occurrences, an algorithm is developed and implemented into the software controlling the cut-to-size plant. The necessary tests are performed in an iterative way with the help of our emulation tool, which connects the plant control software with the virtual analogs of the control and the field layers.

1 INTRODUCTION

Cut-to-size plants are complex machinery, the purpose of which is to realize cutting, sorting, and stacking of panels of different sizes and materials (e.g., wood, plastics, or metal). The just mentioned main cut-to-size processes are carried out by cut-to-size saws, sorting carriages, stacking devices, roller tracks, etc., as depicted in Figure 1.

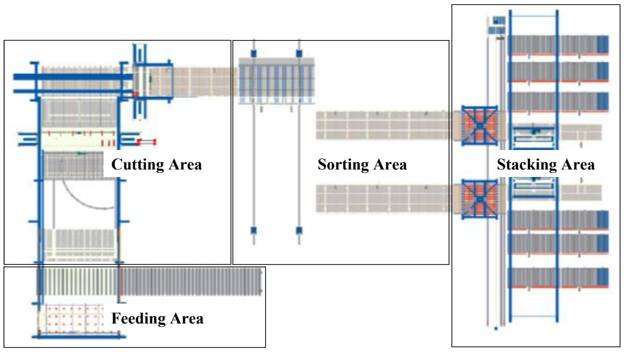

Figure 1: Exemplary cut-to-size plant layout with units for feeding the plant, cutting, sorting, and stacking grouped into areas

The illustrated plant components – more precisely, devices and units – can be combined and configured in a variety of ways, leading to a high degree of *structural* complexity. This diversification by variation is essential for matching the manifold and specialized requirements of a whole range of customers from industry sectors such as furniture manufacturing and circuit board industry. On the other hand, along with the structural complexity comes the exposition of complex *dynamic behavior* by cut-to-size plants. Examples include parallel material movements, buffer areas, a process-inherent increase of panels in such plants, and a large number of simultaneous activities, in general.

In view of these complexities, the ultimate motivation of the here presented work is to incrementally develop a model-based, multi-faceted decision support tool, which consistently uses and refines – across stages of plant design and lifecycle – "an emulation model (as distinct from a pure simulation model) (…), where some functional part of the model is carried out by a part of the real system" (McGregor 2002). To be cost effective, this tool should at best be applicable all the way from concept planning, via bid preparation and sales negotiation, through design, testing, and deployment until operation and adjustments. Concretely, in Schoech, Schmid, and Hillbrand (2011) and Schoech, Schmid, and Hillbrand (to appear), it was already shown that the decision support tool provides added value for both suppliers and clients during sales processes of cut-to-size plants. The particular motivation for that came from the fact that typical customers in this highly specialized business – the discussed plants are built according to customer specifications (lot-size one, make-to-order strategy) – commonly place an order for a turnkey plant only after complex sales procedures, in which the plant supplier needs to demonstrate the capability to fulfill specific customer requirements. Consequently, the following has been demonstrated: Support for iterative requirements specification and plant conception, plus accurate calculation of critical performance indicators, and 3D animation of material flow (amongst others reducing information asymmetries between manufacturer and potential customer).

On the other hand, in this paper, usefulness of the same model-based tool in later phases of design and implementation will be shown. In particular, the focus will be on the support for trade-off decisions such as ensuring absence of deadlocks while still fulfilling performance-indicator requirements for cut-to-size plants, which feature feedback in their material flow. This is closely related to plant control software implementation.

2 RELATED WORK

While the deployment of the here presented emulation approach during sales processes of cut-to-size plants and the related benefits with regard to optimizing plant layout decisions have already been discussed in Schoech, Schmid, and Hillbrand (2011) and Schoech, Schmid, and Hillbrand (to appear), in this paper the use of the emulation tool during a later stage of plant development is described, where it supports establishing a deadlock-handling strategy in the process control layer of a cut-to-size plant. A similar approach of virtually testing upper-level control software is also adopted by Okolnishnikov (2011), who gives examples of developing process control systems with the use of emulation models in various industrial fields such as a railway tunnel or underground coal mining. In Meyer, Pöge, and Mayer (2012) the expansion of an existing simulation object library for the automotive industry to include emulation functionality is described. In that project the simulated plant is linked to the manufacturing execution system (MES) as well, whereas Seidel, Donath, and Haufe (2012) present an integrated simulation and virtual commissioning environment for controls of (intralogistics) material handling systems which comprises emulations not only connected to the upper level of a material flow controller (MFC) but also to the lower level of programmable logic controllers (PLC).

The following section gives an explanation of the emulation approach used in our tool together with the different layers of the control system. Section 4 shows how this conceptual model is implemented resulting in a tool which enables a domain expert of cut-to-size plants to model, emulate, and animate a plant and its processes. In section 5, we take a closer look at plant layouts with feedback in their material flows, before we provide details about deploying our emulation tool to design zero-deadlock plants for such layouts in section 6.

3 EMULATION APPROACH

According to Mertens (2001), the term *emulation* can be defined as the virtual reproduction of certain aspects of hardware or software systems on another system. Therefore, emulation can be seen as a special case of simulation, supplemented with the coupling of real-world functional components. For emulation of cut-to-size plants, orders from the plant server are executed within a virtual plant.

As Figure 2 illustrates, classical plants are centrally and hierarchically controlled real time systems (Günthner and ten Hompel 2010). Within such real-world systems the *field layer*, constituting the lowest level, represents all mechanical components with their actuators and sensors and controls the material handling. The *control layer* resides above the process level. On this layer, sensor data are processed and control signals for the actuators are generated. This level represents the basis for an automated, modular plant. Moreover, it coordinates the handover of load data and controls the material flow of the plant modules. The *process control layer* is the highest layer in the control pyramid and is often called plant server. The control layer and the process control layer are connected by means of a communication channel, which passes the scheduled orders to the control layer and receives confirmation when all actions have been processed.

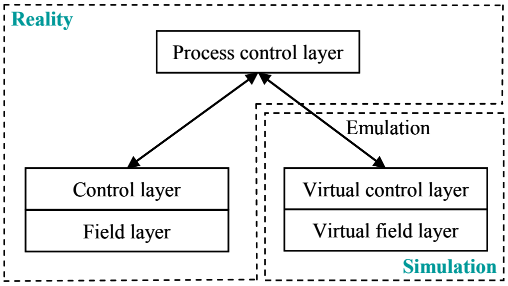

Figure 2: Central hierarchical control systems

The plant logic of the real-world system as described above is mapped to a model within the virtual plant: The *virtual field layer* of the emulated system visualizes all mechanical components and displays the kinematic movements of the material handling within a virtual environment. The *virtual control layer* prepares all orders originating from the process control layer and controls each virtual plant unit. These are one-to-one correspondents of real-world plant units. Since there is a tight coupling between real-world and simulation systems, the virtual system can be called an emulation system (McGregor 2002).

In contrast to simulation, emulation models are mostly used in much more precisely defined ways, for example, “in order to test the operation of the control system under different system loading conditions” (McGregor 2002). The conditions under which the evaluations are carried out can be better controlled, allowing the study of different scenarios the control system has to deal with (McGregor 2002).

4 FRAMEWORK: ARCHITECTURE AND USAGES

This section presents a general overview of the system architecture for our emulation-based decision support tool. The foundation of our system, the in-house developed software framework *Application Platform Simulation* (Dobler, Saler, and Maerz 2008), provides simulation libraries with generic modeling functions which can be used to implement domain-specific modeling environments. A number of domain-specific modeling methods and applications, such as planning of transportation networks (Hillbrand and Schmid 2011) or warehousing structures (Maerz and Saler 2008), have already been implemented on top of this generic platform. For the work presented in this paper, specific methods for cut-to-size plants were implemented. The main intention has been to enable domain experts (e.g., technical sales personnel) to define and project alternative plant configurations within the virtual system and to evaluate them against the predefined critical performance indicators in order to identify an optimal solution. By using a domain-specific modeling environment with an underlying simulation model built-in, the domain expert can take advantage of emulation techniques without needing to have simulation expertise. Thereto, the software architecture of the decision support system for cut-to-size-plants is arranged in two levels (illustrated left and right of the dashed vertical line in the left part of Figure 3): the *modeling and results* level and the *emulation and visualization* level.

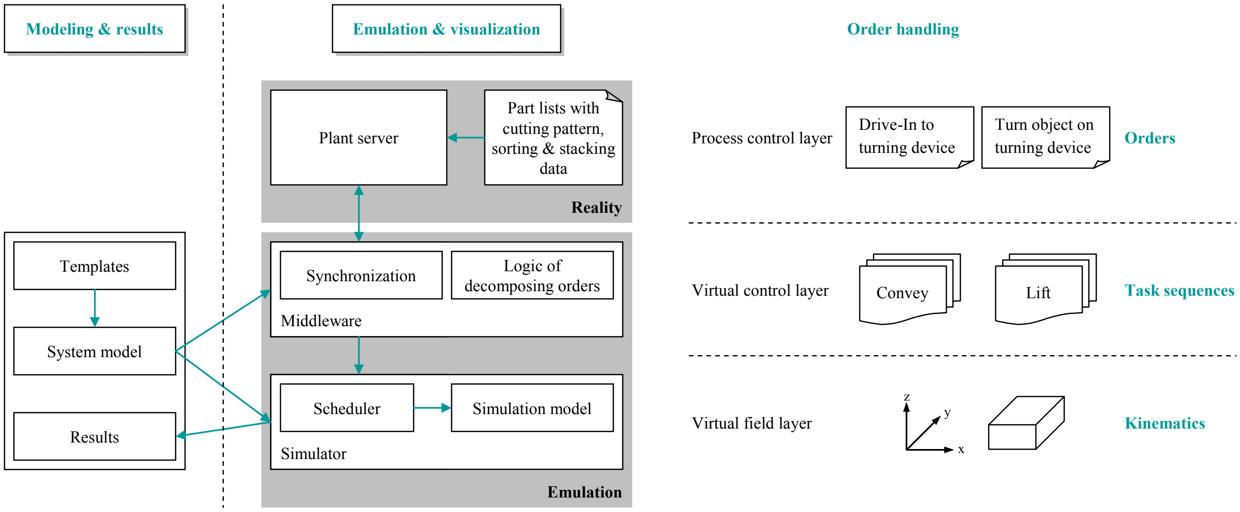

Figure 3: System architecture (left side) and order handling (right side)

Firstly, the *modeling and results* level is used for the development of unit-based system models (Zeigler and Sarjoughian 2003): a plant is designed as a collection of plant units. A *plant unit template* represents a number of variations of a self-contained plant unit, which again consist of a set of devices. A *device* is specified by attributes which describe process-relevant and mechanical information, velocities and meta-information for the instantiation of each class type used in the virtual field layer and in the virtual control layer. These unit templates represent atomic modules of the virtual plant and cannot be further broken down. They are linked together in drag-and-drop manner to form a system model of the plant in question in a graphical editor (see Figure 5 in Schoech, Schmid, and Hillbrand (2011)). The work flow of the plant is modeled implicitly by the coupling of the individual plant units. Once the plant is specified as a unit-based system model, an emulation run and the visualization of the material flow are triggered at the modeling and results level. For transforming the system model into a simulation model, the unit-based system model is rendered persistent as a well-formed XML document. The transformation process of building a simulation model is automated and consists of two parts, depicted by the two arrows crossing the vertical dashed line in Figure 3 from left to right: creation of simulation objects and the assignment of values to all public properties, on the one hand. Additionally, instantiation of the virtual control layer takes place depending on the defined unit-based system model description (upper arrow).

The *emulation and visualization* level represents the plant server and the virtual system. Instead of controlling the real plant, the plant server controls the virtual one. After starting the emulation, the plant server broadcasts orders to the virtual control layer, which are generated on the basis of part lists with cutting patterns as well as sorting and stacking patterns for the panels. On the virtual control layer, task sequences are generated from the orders (see Figure 3 on the right side). A single task describes an activity that imitates a physical mechanism which is executed on a simulation object in the simulation model. The virtual field layer, which is based on the simulation engine FlexSim©, retrieves the task sequences and transforms them into discrete events. The simulation model implements 3D visualization of the devices of the plant units and of the panels as well as of their kinematic flows according to the tasks. After execution of the tasks of an order, the confirmation of the order is sent to the plant server via the virtual control layer. Depending on the order confirmation, new orders are triggered for processing task sequences.

While running the emulation, run time information is logged, which afterwards can be viewed and analyzed by the user in the modeling and results level.

An important issue to be considered with regard to emulation is that the simulation model and process control system with their different clocks for the scheduled arrival times of events have to be synchronized. Thereto, a variation of the *blocking rendezvous pattern* (Douglass 2004) was implemented, ensuring correct chronologies in the transmission of the orders and at the same time allowing execution of a simulation model at least a magnitude faster than real time. Thus, the behavior in the simulation model is independent of simulation speed. This is further supported by not simply employing the OPC protocol (Open Platform Communications) for the communication link between process control and control layers, but instead an own implementation.

Our emulation approach does not contain any stochastic processes nor are uncertainties modeled in another way. The reason is that on the one hand the material flow in the model is controlled by the plant server. This means that orders for feeding the plant with raw panels or for processing the parts are generated by the plant server. Values for different parameters like, for example, cutting speed or transportation speed of a roller track are either included in the orders from the plant server or fixed and stored in the system model and hence available in the virtual control and field layer. On the other hand, the performance indicators are calculated without taking into account probabilities of system errors, as required by the manufacturer of cut-to-size plants who uses the here presented emulation tool.

5 PLANT LAYOUTS WITH FEEDBACK

In this section, an exemplary plant layout with feedback is presented and discussed. Such plant layouts result from the corresponding cut-to-size plants being used for make-to-order production: Processing small quantities, with the special case of lot-size one, results in the cutting patterns becoming more complex (see Figure 4). Cutting-pattern complexity is formalized and quantified in subsection 6.1 below (see Figure 9 for illustration).

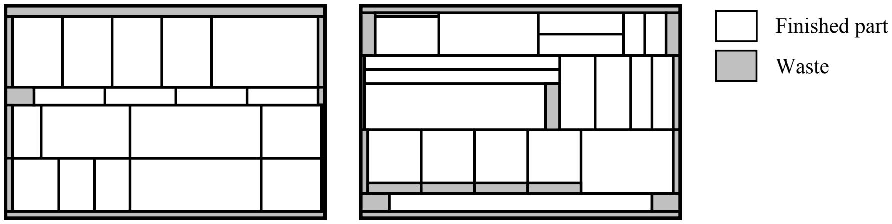

Figure 4: Panel cutting patterns (simple on the left vs. more complex on the right)

While cut-to-size plants featuring a linear layout, with two saws of orthogonal placement – one for rip-cuts and one for cross-cuts, respectively (see Figure 1), can process panels according to cutting patterns such as depicted in Figure 4 on the left, the manufacturing of panels after more complex cutting patterns, like the one shown on the right, exceeds the capabilities of such plants. To overcome this limitation, plant layouts which feature material flows with feedback have been introduced.

In the context of cut-to-size plants, a *feedback in the material flow* means that panels which have been cut by a certain saw will be allocated to the same saw again for further partitioning. Referring to Figure 5, for example, one or more of the panels which result from rip-cutting a raw panel carried out by saw 1 can be assigned to a feedback conveyor, in order to have their cross cuts performed by saw 1, too. More generally speaking, a panel (or parts of it) can be fed back to the same saw again and again as often as required to fulfill the specification given by a certain complex cutting pattern.

The downside of feedbacks in the material flow (saw circles *SC1*, *SC2*, and *SC*3 in Figure 5) towards realizing more complex cutting patterns is that they may cause *deadlocks* (Coffman, Elphick, and Shoshani 1971). Hence, cases may occur such that there is a set *S* of panels, where each panel is occupying a plant unit and waiting for the release of another unit for its advancement. However, at the same time, the requested units are held by other parts of the set *S* and thus cannot be freed.

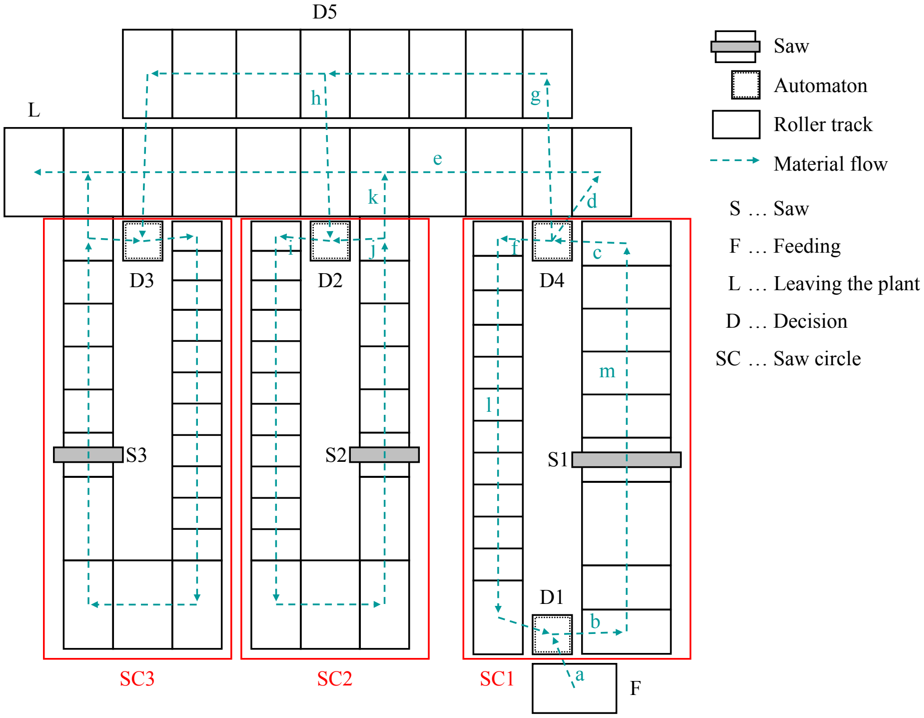

Figure 5: Plant layout with feedbacks

Beyond feedbacks in the material flow, also the *flexible routing of parts* (De Toni and Tonchia 1998) has newly been established in the control of the material flow for cut-to-size plants. Although there are constraints concerning the choice of the saws due to technical specifications (for example, the maximal cutting lengths of saw 2 and 3 in Figure 5 are only 2.3 m respectively, whereas the one of the first saw is 4.3 m), the decision about which saw is to be used next is made in real time on the basis of the current state of the plant in general. Consequently, the way of a part through the plant is not yet determined at the time when the part enters the plant. As a result, the material flow is not predictable anymore, as opposed to cut-to-size plants with simpler (i.e. linear) layouts.

The material flow illustrated in detail in Figure 5 is reduced to its essential stages and presented in Figure 6.

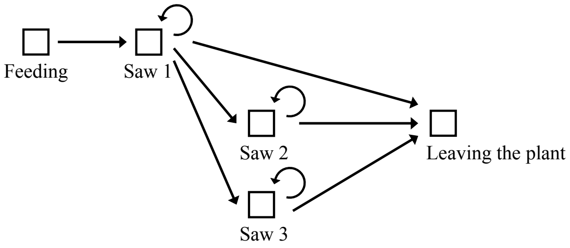

Figure 6: Abstracted material flow of the plant shown in Figure 5

Compared to Figure 1, sorting and stacking areas are not incorporated into the plant layout, because these functions are not as crucial in regard to production of smaller batch sizes as they are for mass production.

The other two issues to be considered with regard to cut-to-size plants with feedbacks in the material flow admittedly also concern cut-to-size plants featuring linear layouts. However, they have to be particularly taken into account with respect to the possible occurrence of deadlocks in cut-to-size plants. The first of these features is the *divergent structure of the product* (Günther and Tempelmeier 2012). As the term 'cut-to-size plant' already indicates, therewith denoted manufacturing systems serve to divide up raw panels. More precisely, two smaller panels arise out of one panel during the process of cutting. Thus, the number of panels in the system can increase even though there are no new raw panels fed into the plant.

The second characteristic of cut-to-size plants to be newly considered concerns the *capacity of a plant unit*. A unit, the purpose of which is to transport panels through the plant, usually consists of one roller track (see legend of Figure 5). Depending on the size of the panels, a varying number of panels fit on a roller track and hence the capacity of the plant unit is variable.

For example, if there aren't any restrictions regarding the quantity of panels on a roller track, panels can be allocated to the unit as long as there is enough space. Vice versa, if the number of panels located on a unit must not exceed one, the situation illustrated in Figure 7 can occur: One panel (p_1) is longer than the roller track (r_3) to which it is assigned, and therefore still partially lies on the preceding unit (r_2). That is why the next part (p_2) cannot enter roller track r_2 even though there is no panel assigned to this unit. Thus, the current capacity of plant unit r_2 is zero.

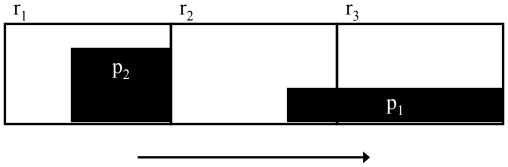

Figure 7: Three roller tracks and two panels

Multiple roller tracks can be consecutively combined for the purpose of locally buffering panels, as it is the case for the cut-to-size plant presented in Figure 5. For each roller track the maximal capacity is one (due to technical peculiarities and procedural specifications) and, mostly, they are of short length.

6 TEST-DRIVEN DESIGN OF ZERO-DEADLOCK PLANTS

As compared to cut-to-size plants with linear layouts (see Figure 1), where the 'circular wait' condition of the four necessary conditions for a deadlock to occur (Coffman, Elphick, and Shoshani 1971) never holds, plants with layouts such as discussed in the preceding section (see Figures 5 and 6) possibly run into deadlocks. For dealing with these deadlocks, the *deadlock avoidance strategy* (Li, Wu, and Zhou 2012) has been chosen for implementation in the plant server software. With this strategy, decisions are made in real time and are based on the analysis of the current state of the system, i.e. the plant. The strategy is intended to not only avoid that deadlocks may occur but also to contribute to that a high utilization of the saws is achieved. According to Zajac (2004), deadlock avoidance is the preferable strategy for deadlock handling in automated manufacturing systems, as opposed to the deadlock-prevention and detection-and-recovery strategies: *Deadlock prevention* is unfavorable, because it often results in poor utilization of manufacturing units due to extensive restrictions on system operation. *Deadlock detection and recovery* is not applicable in this case, because preemption of units from a number of deadlocked processes is not allowed while running a cut-to-size plant (Zajac 2004).

In what follows, the deadlock avoidance strategy for plant layouts with feedbacks in the material flow and its testing by means of emulation are presented in more detail.

6.1 Deadlock-Avoidance Strategy: Implemented Decision Criteria

In order to achieve the objective of operating a cut-to-size plant without deadlocks and at the same time with a high utilization of saws, a deadlock-avoidance strategy is implemented within the process control layer. As just described, it analyzes the current state of the plant and, on this basis, comes to a decision. There are two different *types of decisions*. Firstly, after the release of an automaton which feeds a saw circle, the question has to be answered whether a new panel shall be fed into the saw circle or whether a panel already located in the saw circle shall be allocated to the automaton (decisions *D1, D2,* and *D3* in Figure 5). To this end, the *discrete degree of filling* of the saw circle is used as decision criterion; its calculation is described below. The second decision type concerns the assignment of panels to particular saws in real time – in short, the above mentioned flexible routing of parts. The decision between saw 1 and the other two saw circles is made before a not yet finished panel leaves the automaton placed after saw 1 in material flow direction (decision *D4* in Figure 5). In case the part will not be allocated to saw 1 again, the next decision is between saw 2 and 3 (decision *D5* in Figure 5). These decisions depend on the number of *pending processing steps* of all the panels currently located within a saw circle. They mainly affect the utilization of the saws but they are also relevant with regard to the avoidance of deadlocks, because of the mentioned constraints in relation to the choice of the saws (section 5). Figure 8 shows the abstracted material flow expanded with the discussed decisions.

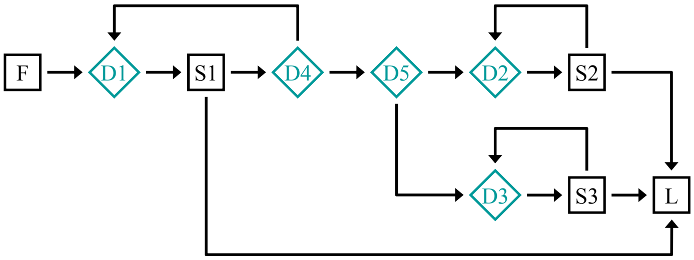

Figure 8: Abstracted material flow with decisions

Both types of decision rules will now be described in more detail. For the decision, whether a new part shall be fed into a saw circle, the *discrete degree of filling* of this saw circle is calculated. That is achieved by way of the following four steps:

- *Listing of the units of the saw circle:* The plant units are listed in the reverse order as to the direction of the material flow, starting with the roller track from which the panels already located in the saw circle are transferred to the automaton, which also feeds the saw circle. For example, the sequence of the units of saw circle 1 in Figure 5 arises from the material flow arrows indicated by *l*, *f*, *c*, and *m*, from the arrowhead to the end of the shaft respectively.
- *Listing of the panels of the saw circle:* The panels currently located in the saw circle are registered in the same order as the units. However, in order to take into account the divergent structure of the product, a panel is actually not listed itself. Instead the parts which will emerge from the next passage of the panel through a saw are listed.
- *Determination of the sizes of the listed panels:* In this context the size of a panel is defined as the length of the longer of both lateral dimensions, regardless of how the panel will be situated on a unit: the longer side parallel to or orthogonal to the material flow direction. The size of a plant unit is given by its length in material flow direction.
- *Assignment of the listed panels to the units:* Both, the list of plant units and the list of panels are related to each other by successively assigning a panel to the next unit, the capacity of which is not exhausted. The panel size determined in the preceding step is compared with the size of the unit in order to determine the capacity of the next units in the list. This is because the panel might lie on them (see the discussion on 'variable capacity' in section 5).

The *discrete degree of filling* is defined as the number of the listed units, the capacities of which are exhausted and it serves as decision criterion. If this number is greater than a pre-assigned threshold value, no new panel is fed into the saw circle.

On the other hand, the decision rule for the choice of saw compares the number of the *pending processing steps* of the saw circles and prioritizes the saw circle with the fewest processing steps: For each panel *p* out of the set *P* of all panels, which are located in a saw circle, the number of the pending processing steps is determined, and then, they are added up. The pending processing steps of one panel p consist of the number of cuts C_p still to be done plus the number of the corresponding saw passages SP_p. One saw passage comprises all the cuts of a panel, which can be executed without the need to turn this panel. Thus, the formula for the calculation of the number of the pending processing steps of a saw circle (*PPS*) reads

$$PPS = \sum_{p \in P} \left(C_p + SP_p \right).$$

For illustration purposes, the number of pending processing steps of the panel, the cutting pattern of which is shown in Figure 9, will now be computed: The number of cuts is sixteen (each cut is numbered in Figure 9), and five saw passages are necessary for the partitioning of the panel. The cuts with numbers 1, 2, and 3 take place when the panel passes a saw for the first time. The two small parts at the edges are waste material and are disposed immediately after sawing, whereas each of the other two parts passes a saw again, but rotated by ninety degrees. They are saw passages number 2 (cuts 4 to 7) and 3 (cuts 10 to 14). The cuts with numbers 8 and 9 belong to the fourth saw passage and the ones with numbers 15 and 16 to the fifth one. Therefore, the number of pending processing steps of the depicted panel is 16 + 5 = 21.

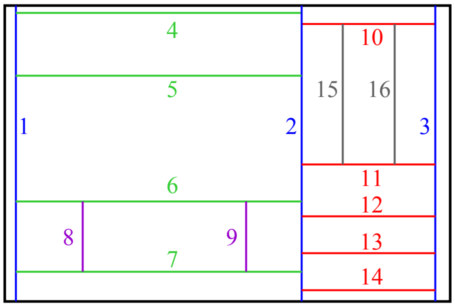

Figure 9: Another complex panel cutting pattern

6.2 Balancing Plant Performance and Absence of Deadlocks

As said above, the main intention of deploying the emulation tool is to enable ‘cut-to-size plant’ domain experts to define a high utilization of deadlock-free saw circles, for then evaluating them against predefined critical performance indicators (such as cycle times, throughput rates, and utilization of plant units) in order to identify an optimal solution. Here, it should be noted that simulation is an evaluation technique and does not automatically produce an optimal solution, unless simulation runs are controlled by an external search loop (see below).

In the process control layer the decision algorithm for deadlock avoidance is activated for the relevant plant units, each of which also contains decision rule descriptions about degree of filling and/or pending processing steps.

After starting the emulation, orders will be served through the process control layer. Before the process control layer sends new orders to designated plant units of the virtual control layer, all plant units with the activated decision algorithm compute the current degree of filling and/or pending processing steps. The computed results of the decision algorithm control and rearrange the new orders for registering all activities as tasks in the virtual control layer. The emulation runs and the subsequent analysis of results can serve as an input for modifications of parameter values of the decision rules. The following procedure illustrates the iterative approximation to an optimal solution of a plant model:

1. *Initialization:* Set low values for decision rules so that a deadlock-free saw circle with low utilization will be achieved.
2. Always *run an emulation job* with constant cutting patterns and device settings for mechanical information and velocities and record data for the new scenario.

3. If a *deadlock-free* emulation job is available, then analyze the utilization of plant units and increase the values for the decision rules. Go to step 2 and run the emulation again.
4. If a *deadlock has occurred* during an emulation job, then decrease the values for the decision rules. Go to step 2 and generate a new scenario.
5. If a *deadlock has occurred* during an emulation job and a valid scenario of lower value settings for decision rules is already available, the optimal solution of the plant model has been found.
6. If a *deadlock has occurred* during an emulation job and decreasing the values for decision rules is not possible, then the capacity of the unit which is used as buffer of the saw circles has to be adjusted for a new plant model.

The optimal solution found is based on the assumption that for real product manufacturing, there are valid plant-specific cutting patterns and hence predefined constraints, such as maximal size of panel, size of part of panel or number of cuts per cycle, are used for pattern formation for parts lists.

7 RESULTS AND FINDINGS

At the moment, the above described, implemented, and deployed emulation tool provides relevant information, models, and methods for acquisition, design and planning, as well as testing and implementation stages of cut-to-size plants, where the underlying models are consistently used across process stages. The demonstrated techniques facilitate the anticipation of real-life plant behavior by means of a virtual system. This in turn allows plant experts to model, emulate, and animate sequences of cut-to-size plant operations, without the need for in-depth simulation expertise: System models are built following a modeling process, characterized by plant-unit template configuration und drag-and-drop style system construction. An automatically generated simulation model allows running emulation jobs, where results are logged and enable a plant expert to meaningfully analyze critical performance indicators (cycle times, throughput, and utilization).

First of all, the presented emulation tool is successfully employed in the rather complex sales procedures typical in regard to cut-to-size plants, with a proven track record of enabling a key competitive advantage via the cost-effective, reliable, and accurate calculation of all the relevant performance indicators, potential clients are interested in. The role of emulation in this process has been extensively discussed in our previous work. Second, and of no less importance, the complexities of cut-to-size plants with feedbacks in the material flow are kept manageable with the presented emulation tool, processes, and procedures. This is another key advantage not only concerning sales procedures but also as to providing a virtual, high-level plant-control system test bed. Summarizing, technical expertise, constancy, and reliability can be demonstrated through virtual plants.

The emulation tool supports the plant experts in planning processes, in modeling and evaluating cutting areas with feedback and in validation of deadlock-free solutions. The comparison of several emulation scenarios allows the plant experts to optimize the solution according to the procedure explained in section 6.2. Overall, with the presented approach, tools, models, and techniques costs and time can be saved in the planning process. The planning instrument not only enables plant experts to model layouts with feedback with multiple parallel cutting areas but during implementation of the real plant, the instrument serves an incremental implementation of the cutting areas. For example, in the ramp-up of a further cutting area, the tool enables the plant experts to find an optimal solution for the plant extension regarding the absence of deadlocks. Also, the tool can serve as training instrument to get familiar with the control system and plant processes. While operating the plant, it is possible to perform impact analysis of modifications of the virtual as well as the real system. This allows for an efficient analysis and resolution of errors, without disrupting the normal course of business. With the realization of the described emulation tool, a planning instrument becomes feasible which reduces acquisition and start-up times for cut-to-size plants and continuously tests and optimizes the control processes.

More activities to further developing and extending the emulation tool are planned in the course of this year, amongst them not only targeting the upper-level MES software but also the lower-level software, i.e. PLC code. Currently, real plant units can often only be tested separately and not as a combined plant system. The idea is to create a complex system where a combination of reality and simulation can be achieved. Therewith, real units of the plant could be tested within the overall virtual system. This approach will help the plant experts to be able to find failures on plant units or in the material flow much earlier as without the emulation tool.

ACKNOWLEDGMENTS

This paper discusses the results and findings of a research project within the K-Project 'Integrated Decision Support Systems for Industrial Processes (ProDSS)' which has been financed under the Austrian funding scheme COMET (COMpetence centers for Excellent Technologies).

REFERENCES

Coffman, E. G., M. J. Elphick, and A. Shoshani. 1971. "System Deadlocks." *Computing Surveys* 3:67–78.

De Toni, A., and S. Tonchia. 1998. "Manufacturing Flexibility: A Literature Review." *International Journal of Production Research* 36:1587–1617.

Dobler, M., M. Saler, and L. Maerz. 2008. "Distribution of Concurrent Simulation Runs on a Service-oriented Network Structure." In *Advances in Simulation for Production and Logistics Applications*, Edited by M. Rabe, 509–517. Stuttgart: Fraunhofer IRB Verlag.

Douglass, B. P. 2004. *Real Time UML: Advances in the UML for Real-time Systems*. Boston: Addison-Wesley.

Günther, H.-O., and H. Tempelmeier. 2012. *Produktion und Logistik*. Berlin, Heidelberg: Springer-Verlag.

Günthner, W., and M. ten Hompel. 2010. *Internet der Dinge in der Intralogistik*. Berlin, Heidelberg: Springer-Verlag.

Hillbrand, C., and S. Schmid. 2011. "Simulation of Multimodal Logistics Networks." In *Proceedings 25th European Conference on Modelling and Simulation*, Edited by T. Burczynski, J. Kolodziej, A. Byrski, and M. Carvalho, 594–600.

Li, Z., N. Wu, and M. Zhou. 2012. "Deadlock Control of Automated Manufacturing Systems Based on Petri Nets–A Literature Review." *IEEE Transactions on Systems, Man, and Cybernetics–Part C: Applications and Reviews* 42:437–462.

Maerz, L., and M. Saler. 2008. "An Analysis, Planning and Decision Tool for Warehouse Applications." In *Advances in Simulation for Production and Logistics Applications*, Edited by M. Rabe, 237–245. Stuttgart: Fraunhofer IRB Verlag.

McGregor, I. 2002. "The Relationship Between Simulation and Emulation." In *Proceedings of the 2002 Winter Simulation Conference*, Edited by E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 1683–1688. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Mertens, P. 2001. *Lexikon der Wirtschaftsinformatik*. Berlin, Heidelberg: Springer-Verlag.

Meyer, T., C. Pöge, and G. Mayer. 2012. "Integration of Emulation Functionality into an Established Simulation Object Library."In *Proceedings of the 2012 Winter Simulation Conference*, Edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher, 2864–2874. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Okolnishnikov, V. 2011. "Development of process control systems with the use of emulation models." *International Journal of Mathematics and Computers in Simulation* 5:553–560.

Schoech, R., S. Schmid, and C. Hillbrand. 2011. "Simulation Based Decision Support System for Cut-to-size Plants." In *Proceedings 25th European Conference on Modelling and Simulation*, Edited by T. Burczynski, J. Kolodziej, A. Byrski, and M. Carvalho, 394–400.

Schoech, R., S. Schmid, and C. Hillbrand. To Appear. "Optimising Plant Layout Decisions Based on Emulation Models - Technical Framework and Practical Insights." *International Journal of Simulation and Process Modelling*, Special Issue on: "Research Challenges in Developing Advanced Solutions in Industry and Supply Chains Modelling and Simulation for Building 21st Century Enterprises." http://www.inderscience.com/info/ingeneral/forthcoming.php?jcode=ijspm.

Seidel, S., U. Donath, and J. Haufe. 2012. "Towards an Integrated Simulation and Virtual Commissioning Environment for Controls of Material Handling Systems." In *Proceedings of the 2012 Winter Simulation Conference*, Edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher, 2852–2863. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Zajac, J. 2004. "A Deadlock Handling Method for Automated Manufacturing Systems." *Annals of the CIRP* 53:367–370.

Zeigler, B. P., and H. S. Sarjoughian. 2003. "Introduction to DEVS Modeling & Simulation with JAVA: Developing Component-based Simulation Models." http://www.cs.gsu.edu/xhu/CSC8840/DEVSJAVA_Manuscript.pdf.

AUTHOR BIOGRAPHIES

RUTH FLEISCH is research associate at V-Research. After receiving her diploma in mathematics from the University of Innsbruck she joined V-Research in 2010. Her research focuses on discrete-event systems as well as RFID technology. Her email address is ruth.fleisch@v-research.at.

ROBERT SCHOECH is project manager at V-Research, an Austrian competence center for industrial research and development. Within the research area "Technical Logistics" the research center focuses on methods to support complex decisions in manufacturing and logistics processes. Robert Schoech's work focuses on simulation-based systems for industrial processes. His research interests also include track & trace solutions based on GPS, GSM, and RFID technology. His email address is robert.schoech@v-research.at.

THORSTEN PRANTE is a project manager at V-Research and the director of the research program "ProDSS" (Integrated Decision Support Systems for Industrial Processes). He has worked on man-machine interaction in the context of ubiquitous computing for a number of years and is a co-founder of the open-source project Zeitgeist on activity-based information management. Thorsten's current work focusses around balancing interaction and automation in the field of engineering design informatics. His email address is thorsten.prante@v-research.at.

ROBERT PFLEGERL is software developer at Schelling, a leading manufacturer of cut-to-size plants. His work focuses on the preparation and administration of process data for the automated procedures of the plants, the development of graphical user interfaces for the visualization of the states of the plant and diagnostic data, as well as the capture and analysis of manufacturing data. His email address is robert.pflegerl@schelling.at.