**In:Safeprocess'09, 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, 2009. ISBN: 978-3-902661-46-3**

# Post-mortem Diagnosis of Bottling Plants Based on Recorded Data

**Peter Struss, Benjamin Ertl**

*Comp. Sci. Dept., Tech. Univ. of Munich, Garching, Germany*
*(Ph: +49 89 289 17776; e-mail: {struss, ertlb}@in.tum.de).*

**Abstract:** We present LineMod, a model-based system that performs post-mortem fault localization of food packaging plants and, more specifically, bottling plants based on recorded data. In this project, we need to cope with non-negligible transportation time of objects and capture phenomena like the tailback of units (if transportation is blocked) or the propagation of gaps in the flow of units. Because the application context requires compositionality of the model, i.e. local, context-free models of the individual transportation elements, we are also facing the problem that whether or not a single element produces an output flow (or accepts an input flow) cannot be determined solely by the model of this element, but only through modeling the interaction with the subsequent element, which may block the output (or the previous one not providing the input). The diagnostic system scans a data base with recorded data from the involved machines for fatal symptoms (such as the stop of the filler) and determines the respective causes as a basis for taking actions to improve the performance of the plant. We present results of the evaluation of the first prototype and discuss challenges to future work.

## 1. INTRODUCTION

Modeling the flow of some matter in a system is quite widespread in model-based systems, e.g. in model-based diagnosis of hydraulic or pneumatic systems. At least under certain simplifying assumptions, mathematical first principles models exist, and it appears to be straightforward to abstract them into adequate input to a model-based problem solver.

Typically, such models assume that the flowing matter is continuous and homogeneous and does not have to be modeled as an object or in its detailed structure. And they usually incorporate the analogies to Kirchhoff's and Ohm's Laws, which leads to simultaneous equations that imply instantaneous propagation of pressure and disregard time needed by the matter to be transported through the system.

There are classes of application domains that involve a flow of objects through a plant and, hence, suggest the use of some flow model, but require dropping some of the simplifying assumptions mentioned. One instance of this class is given by food packaging plants, which were subject to our diagnosis project LineMod[1], and, more specifically, by bottling plants, which we will use as an example in this paper. Such plants involve streams of objects of different types, bottles, crates, and pallets being the most prominent ones. As a consequence, we had to develop a model that

- includes transportation times,
- covers interrupted flows,
- handles the exchange of flows between neighbouring elements appropriately.

The diagnostic system scans a data base with recorded data from the involved machines for fatal symptoms (such as the stop of the filler) and determines the respective causes as a basis for taking actions to improve the performance of the plant.

After explaining the application context in section 2, we present the architecture of the diagnostic system (section 3) and the foundational model (section 4). Section 5 discusses the results of the evaluation of the first prototype.

## 2. DISTURBANCES IN BOTTLING PLANTS

A bottling plant for filling beverages into returnable bottles is an assembly of a number of quite different types of specialized machines and conveyors that automatically handles the complete process from supplied pallets with crates containing empty bottles to the final output of pallets with (cleaned) crates and filled and labeled bottles. The plants can be large, distributed over several big halls and

[1] The LineMod project was partially funded by the German Federal Minister of Economics and Technology (BMWi) through the German Federation of Industrial Research Associations (AiF) under FV-Nr.233 ZBG

**Figure 1.** *Conveyors of a bottling plant for returnables*

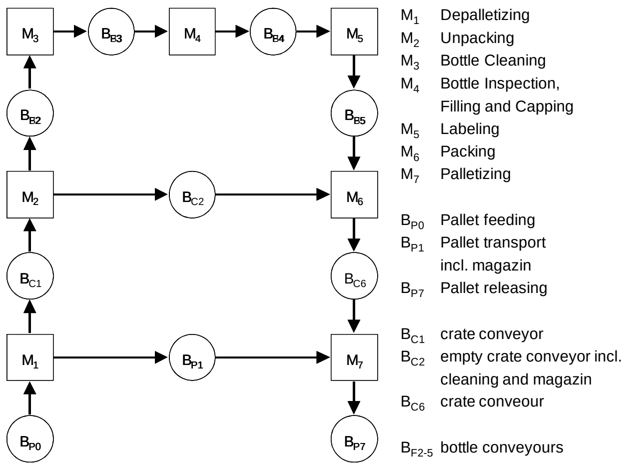


**Figure 2.** *Generic structure of a bottling plant for returnable bottles*

have a complex three-dimensional layout, as illustrated by Figure 1.

From an abstract point of view, which reflects flows and manipulation of different types of objects, the basic schematic topology can be simplified as indicated in Figure 2: there are lines for primary packaging (beverages into bottles), the top section in Figure 2, secondary packaging (into crates), shown as the middle section, and tertiary packaging (e. g. pallets), all organized as an automated branching, but directed flow. Certain backward loops, such as re-submission of improperly cleaned or filled bottles to previous steps, are omitted in the figure. Besides the conveyors and the central filling aggregate, there are machines for de-palletizing and unpacking of returnable bottles, cleaning, inspection, and sorting out of improper objects.

To prevent oxygen intake or microbiological contamination of the beverage, a major objective is to prevent interruptions of the filling process. Besides for internal reasons, the filling aggregate will stop operation if there is a lack of input, i.e. bottles (apart from the beverage itself and caps), or a tailback of filled bottles preventing further output, i.e. disturbances caused by other machines (section 5 presents two examples). Because of the high speeds and output rates (up to 100.000 packages per hour), machines and conveyors are failure-sensitive with an availability degree of 92-98 percent. In order to avoid that each disturbance of single machines in the line results in a filler stop, several conveyor belts are designed as buffers, which should provide a continuous supply and output of other machines and, in particular, the filler, in conjunction with a general operation principle: machines and conveyors upstream and downstream from the filler operate at higher throughput rates than the filler. This principle is usually the only global one; there is **no global control**, and the machines are controlled individually (or, sometimes, as small aggregates).

However, in practice, these measures cannot guarantee avoidance of unwanted idle time of the filler, and (unplanned) downtime of the plant can lie in the range of 10-30 percent. Taking steps to reduce downtime by identifying frequent causes requires statistics and an analysis based on the recorded operating data supplied by (some of) the machines. Because of the interlaced flows of the various object types, time offsets, the large scale of the plants, and the amount and often fragmentary nature of the data this can be difficult and time-consuming. In consequence, bottle filling and packaging industries is highly interested in an automated diagnosis tool for their plants that produces information supporting the identification of bottlenecks and weaknesses in the plant, related to both the physical performance and configuration and the control principles and parameters. Providing such a tool is the goal of the LineMod project described here and its follow-on project.

Because many of the potential end users, e.g. breweries, are small or medium-size enterprises, which could not afford spending many resources on the establishment or adaptation of a tailored diagnosis system for their plant, a **model-based solution** to diagnosis was chosen, which allows performing adaptation to a plant or modification simply by (re-) specifying the plant structure. This also provides a flexible solution that derives the best diagnosis from **whatever data is available** (in contrast, for instance, to decision trees based on a fixed set of observables), which is important, because usually a plant is a combination of machines from various manufacturers with different instrumentation and available data, and there may be temporarily missing data due to technical problems.

Heterogeneity and changes of the set of machines also establishes a requirement on the model: firstly, it has to be machine-centered and **compositional**; secondly, it has to be stated at a level of abstraction that covers **types of machines**, independently of specificities and the manufacturer.

## 3. A MODEL-BASED DIAGNOSTIC SOLUTION

The first demonstrator of the tool addresses the needs explained above by localizing those interruptions of transportation that caused downtime of the filler based on the available recorded data of the machines (over a period of days to months) collected in a data base.

The data base is large and will contain reflections of many disturbances and machine stops that are irrelevant for the overall performance of the plant. Feeding all data to a diagnostic analysis is prohibitive and not interesting. Rather,

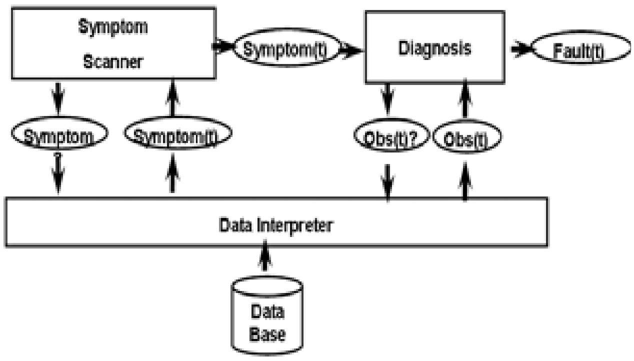


**Figure 3.** *The architecture of the tool*

the diagnostic tool scans the data base for the relevant disturbances to be explained (see Figure 3). In the first version, the only symptom considered is a stop of the filler.

Whenever the presence of a symptom is confirmed by the data for some time period, a run of the diagnosis module is performed using the symptom (and its temporal extent) as the initial observation. Diagnosis is performed in a **cycle of prediction, observation, and computation of diagnoses** (using RAZ'R ([RAZ'R 09]), a consistency-based diagnosis engine ([Struss 08]). Since the control stopped the filler either due to a lack at the input or because of a tailback at the output, this information is (usually) represented in the data (otherwise, both hypotheses would have to be analyzed). This information is propagated to the adjacent component models and triggers queries to the data base in order to find data that confirm a disturbance of the respective component or refute it. In the latter case, the model of correct behavior generates predictions concerning the behavior of the next machine(s), which are then checked against data retrieved from the data base. Propagation is also continued if there is no information available or if it is too weak.

Whenever refuting or confirming observations are found, they and their temporal extensions are used for the next prediction step, replacing the original predictions. This is important, since model-based temporal prediction has to be very conservative, i.e. generate large intervals due to the uncertainty in the delays of propagated effects in order to guarantee that no evidence is missed. Using the **observed time periods** wherever available **restricts** widening of the **time intervals** significantly. Performing a complete prediction would quickly lead to large intervals and create overly ambiguous and spurious results.

Figure 3 shows a **data interpreter** mediating between the diagnosis module and the data base. This module is essential to the objective of creating a flexible, extensible, and adaptable solution. It organizes two important mappings:

- It maps fairly abstract and high-level concepts (called features) the diagnostic model uses to characterize behavior (such as flow, buffer filling degree, lack, tailback) and states queries to the basic and generic quantities represented by the operation data supplied by the machines.
- In a second step, these generic operation data (such as machine states, counter data, speeds) are mapped to the specific records in the data base, which is needed to generate the queries and ensures the adaptation of the tool to a customer-specific data base by simply specifying this mapping, without having to touch the definition of higher-level features in the first mapping.

In the simplest case, high-level features may correspond directly to (conjunctions or disjunctions of) values assigned to operation data records, e.g. "machineX.lack=T" maps to "machineX.status=Lack", and this in turn is mapped to "machineId.typeId=<code for lack>". As discussed in section 5, more complex features will be needed in an extended solution, e.g. "stuttering" of a machine, i.e. frequent oscillation between operating and stops.

A major requirement was to limit the efforts needed for adaptation to a new or modified plant. This is met by the model-based solution. Firstly, the topology has to be entered, mapping the existing machines to component types, which have an associated model in the library. Due to a common typology of machines based on their basic function (washers, inspectors, buffers …) and the genericity of the models in the library, this has been found to be straightforward.

Additionally, some parameters have to be obtained:

- parameters of single machines, as nominal output rate, capacity, time needed to run empty or full, or reaction times
- parameters of single transporters, such as capacities, reaction times, or information about their geometry, if available,

which are currently exploited for temporal prediction only.

## 4. MODELS

The model used for diagnosis has to be

- **compositional and component-oriented** to allow for an easy configuration of the model according to the specific plant topology,
- **abstract** in order to cover general classes of machines, rather than a zoo of different variants from different suppliers,
- **qualitative** in order to cope with the uncertainties and non-determinism of the behavior.

They mainly capture the flows of objects, their separation and combination and how interruptions of the flow propagate across the machines. We first developed a numerical model, implemented and validated it as a Matlab model, and then abstracted it to a qualitative diagnosis model. For more details, see [Struss et al. 08]. Figure 4 displays the model of two basic elements of the model, a transportation element with a buffer and the virtual component that connects two machines.

## 5. RESULTS OF THE FIRST DEMONSTRATOR

### *5.1 Evaluation Method*

We followed two ways to evaluate the diagnostic results of the tool. One was by **comparison** with the fault localization on **two real plants** performed by **human experts**. Since it is difficult and time-consuming for them to do this based on the recorded data (which is actually what motivated the project!), the reference diagnoses were obtained by direct on-site

**Transportation Element with Buffer**

**State variables**

$B(t)$ # objects in buffer

$B_{out}(t)$ # objects buffered for immediate output

$v_{in}(t)$ velocity of input transportation means

$v_{out}(t)$ velocity of output transportation means

$t_d(t)$ minimal transportation time

**Parameters**

$d_0$ diameter of transported object (in transportation plain)

C Capacity (as number of objects)

**Interface variables**

$in.q_{pot}(t)$ potential inflow [objects/s]

$out.q_{pot}(t)$ potential outflow [objects/s]

$in.q_{act}(t)$ actual inflow [objects/s]

$out.q_{act}(t)$ actual outflow [objects/s]

Equations

(1) $in.q_{pot}(t) = v_{in}(t) / d_0$ if $B(t)<C$

$in.q_{pot}(t) = \min (v_{in}(t) / d_0, out.q_{act}(t))$ if $B(t)=C$

(2) $dB/dt = in.q_{act}(t) - out.q_{act}(t)$

(3) $out.q_{pot}(t)= v_{out}(t) / d_0$ if $B_{out}(t) \geq 1$

$out.q_{pot}(t)= \min (in.q_{act}(t - t_d) , v_{out}(t) / d_0)$ else

(4) $dB_{out}(t) / dt = in.q_{act}(t - t_d) - out.q_{act}(t)$

**Connector between Transportation Elements**

**Interface variables**

$TE_{n+1}.in.q_{pot}(t)$ potential inflow of upstream element $TE_{n+1}$

$TE_n.out.q_{pot}(t)$ potential outflow of downstream element $TE_n$

$TE_{n+1}.in.q_{act}(t)$ actual inflow of upstream element $TE_{n+1}$

$TE_n.out.q_{act}(t)$ actual outflow of downstream element $TE_n$

**Equations**

(5) $TE_n.out.q_{act}(t) = \min (TE_{n+1}.in.q_{pot}(t) , TE_n.out.q_{pot}(t))$

$TE_n.out.q_{act}(t) = TE_{n+1}.in.q_{act}(t)$

**Figure 4.** *Equations of buffer and connector*

observation: a number of experts observed the plant and immediately tried to identify the causes for filler stops. Thus, a protocol was created that contains disturbances of the Bottle Filler as symptoms $S_{ref}$ with relevant time intervals, and with manually associated causes $C_{ref}(s)$ for each symptom $s \in S_{ref}$. This protocol serves as reference diagnosis result $result_{ref}$ for assessing the quality of the diagnosis core result $result_{core}$ produced by the LineMod tool. Obviously, this could not be done for longer periods than a few days, but, nevertheless, produced a set of several hundred relevant cases for the evaluation.

Because one cannot expect to encounter all kinds of interesting cases in a limited time (and in the chosen plants), the second way of evaluation was performed by **simulation**. For this purpose, a model of one of the real plants was created for discrete event simulation, which generated a set of 80 cases of disturbances for comparison with the output $result_{core}$ of the tool. It consists of a list of detected symptoms $S_{core}$ with relevant time intervals and with an associated list of pairs of possible causes $C_{core}(s)$ and relevant time intervals for each symptom $s \in S_{core}$ generated by the diagnosis solution.

The causes per symptom instance $s \in S_{ref}$ are called real causes $C_{real}(s)$, which were always singletons in our case.

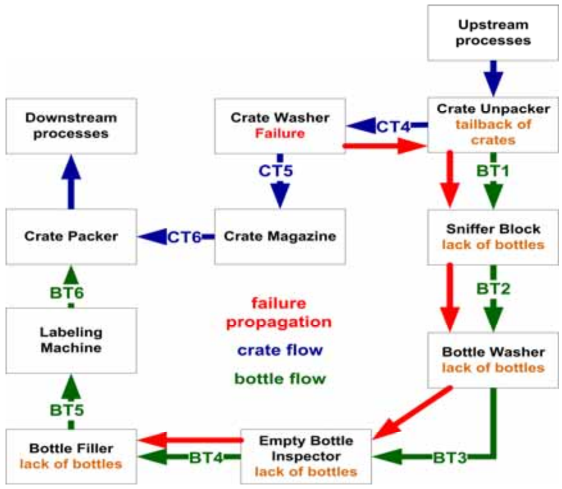


**Figure 5.** *Failure propagation (red arrows) via sub-branch of production line causing a lack of bottles at the Filler*

The evaluation checked for what we call **compliance**, i.e. whether exactly the relevant symptoms (filler stops) were detected and whether the real (observed or simulated) causes were included in the diagnostic result:

$$S_{core} = S_{ref}$$

and

$$\forall\ s \in S_{core} : C_{real}(s) \subseteq C_{core}(s)\ .$$

Except for individual cases, we could not perform a complete check for false positives, again because this would have required a detailed, time-consuming analysis by experts. However, in general, additional diagnoses generated by the tool were caused by incomplete discrimination due to lacking operation data in some areas of the plants.

*5.2 Sample Scenarios*

To illustrate the task, we present two scenarios from the evaluation, which are considered as non-trivial by the experts. While blockage of machines or transportation elements directly downstream or upstream from the filler are frequent causes of lacks or tailbacks, disturbances may propagate via several machines and transportation elements not only on the main-branch of the production line (involving the bottles), but also via its sub-branches.

In a first scenario taken from the simulated plant ("plant C"), a **disturbance at the crate washer** was the origin of a lack of bottles at the bottle filler via the following causal chain displayed in Figure 5 (since, for the real plant A, there is no operation data available for components upstream from the crate unpacker and downstream from the crate packer and, hence, no detailed localization was possible in these parts of the plant, they are represented as aggregates upstream processes and downstream processes, resp.):

- the disturbance at the crate washer causes a **tailback** of crates, which ultimately reaches and stops the **crate unpacker**
- the stopped crate unpacker interrupts the flow of bottles, causing a **lack** which propagates successively to sniffer

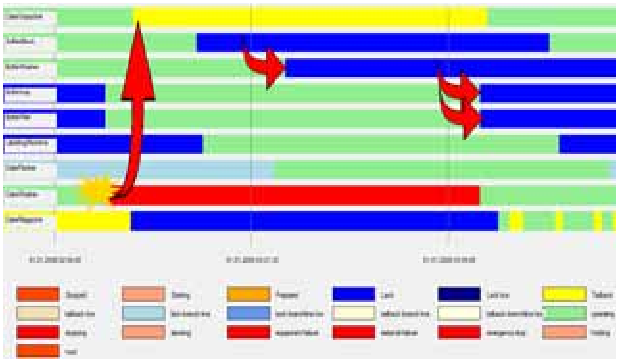

**Figure 6.** Gantt chart of failure scenario1. Legend: red = failure, yellow = tailback, blue = lack

- block, bottle washer, empty bottle inspector and, finally, **bottle filler**.

Figure 6 shows a Gantt chart of the failure scenario. This is the main support for the experts in the diagnosis process. Every line represents the operating states of a certain machine within a defined time interval. Especially the lines marked with arrows are of interest, because they show the failure propagation through the plant.

The manual localization is considered complicated because the disturbance originated from the crate line up to the bottle filler. The diagnosis demonstrator correctly localized the crate washer as the origin of the symptom.

The second scenario (shown in Figure 7) is taken from the diagnostic data of the real plant A, which provided quite bad data quality. It illustrates the capabilities of the model-based diagnosis solution even under such conditions. A defective checkmat (a crate inspection machine) upstream from the crate unpacker caused a tailback of bottles at the bottle filler as follows:

- a **lack** of crates propagates and successively stops crate unpacker, crate washer, crate magazine, and **crate packer**
- the stopped crate packer produces a **tailback** of bottles, which propagates via labeling machine to **bottle filler.**

Despite the completely non-observable crate line between crate unpacker and crate packer (grey-shaded components in the figure), the system model produced predictions for this part and enabled the diagnosis system to produce a proper fault localization, namely the upstream processes component, which contains the defective checkmat, but does not allow for further discrimination due to the lack of data. This example shows that the diagnosis solution provides best possible results, even if the available data is incomplete or not available for some components.

*5.3 Statistics*

**Plant A** was not designed according to the data standard WS2005, but according to the predecessor, and produced a fairly insufficient data base. Only 65% of the original amount of symptoms was considered valid, altogether 49 disturbances of the Bottle Filler, whereof 30 were self-caused faults and 19 were caused by other machines or transportation elements. Bad data included even erroneous state messages of machines, which were eliminated from the reference diagnosis results.

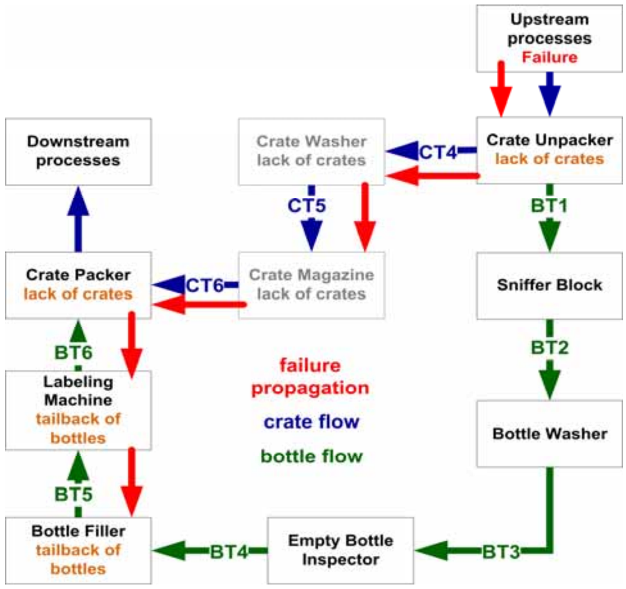


**Figure 7.** *Failure propagation via sub-branch of production line causing a tailback of bottles at the Filler*

Despite this, Figure 8 shows compliance of 88% of the diagnosis results $result_{core}$ with the reference diagnosis results $result_{ref}$. The remaining 12% can mainly be ascribed to missing data making a correct diagnosis impossible.

**Plant B** was designed according to the data standard WS2005 - which resulted in much better statistical evaluation of the diagnosis results. Nevertheless, even for this plant the amount of bad data is significant (17%). Most of the bad data deleted from $result_{ref}$ were imprecision or questionable results in the manually obtained localization, or unusable data produced during maintenance, cleaning, or grade change. After deletion of bad data, 416 symptoms were left in $result_{ref}$ , whereof 265 disturbances were self-caused faults, and 151 disturbances were caused by other machines or transportation elements.

Because of the available data of four production days, the statistical evaluation of the diagnosis results of plant B can be seen as a representative example indicating the quality of results based on an appropriate data base. Figure 8 shows the statistical evaluation of the diagnosis results. The diagram shows that $result_{core}$ complies with $result_{ref}$ in most of the cases.

All of the 80 simulated disturbance cases of **plant C** were correctly diagnosed.

Still there are misdiagnoses, and in case of missing or corrupted data, their number may not be small. However, one has to keep the purpose of the analysis in mind: providing evidence for potential deficiencies of the plant and, hence, possible steps to improving the overall performance. A certain error in the statistics will not trigger a major remedial measure and usually be validated by a more detailed observation and analysis, anyway. This is quite different from, say, diagnosis in a workshop, which is guided by the objective of precisely and unambiguously identifying broken components in order to replace them.

*5.4 Commercial Exploitation*

The results obtained from the first demonstrator have triggered significant interest of both end users (breweries etc.) and suppliers. Two of the latter have started to include results from the project in their analysis tools, which are essentially based on decision trees. A third one is forming a joint venture with a software company to extend its tools by exploiting the model-based solution described in this paper. It is expected that even under the limited scope of the current demonstrator (diagnosing "hard" reasons for filler stops only), a commercial benefit will be obtained.

The following section discusses the practical importance of widening this scope and the resulting challenges to model-based diagnosis, which has lead to a follow-on project.

## 6. CHALLENGES

The diagnostic task solved by the current demonstrator appears to be straightforward. In reality, there are cases in which such an answer is at least questionable or even misleading. This has to do with the fact that the plant achieves some tolerance with respect to limited disturbances of individual machines through buffer elements.

To illustrate this, we consider a sequence of three machines $M_i$ with two intermediate buffers $B_j$ (Figure 9). If buffer $B_2$ is filled to some extent, it will prevent that some short interruption of output from $M_2$ will cause missing input to $M_3$. Now assume that $M_1$ experienced a series of several small disturbances over a longer period of time ("stuttering"), none of which caused a lack of input to $M_2$ or $M_3$.

However, as an accumulated effect of this stuttering, the amount of objects buffered in $B_2$ has been reduced significantly, affecting its capability to compensate for further interruption of flow. As a result, a fairly short fault in $M_2$, which would have had no serious effect under "normal" circumstances, now causes a lack at $M_3$, and the algorithm described would actually detect this and blame solely $M_2$ for the stop of $M_3$.

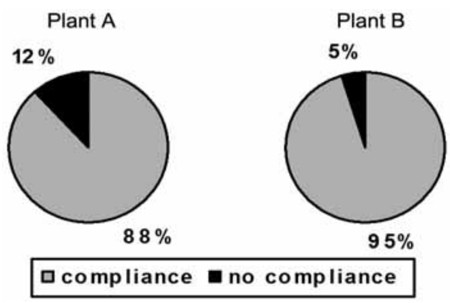


**Figure 8.** *Compliance of core diagnosis results with corrected reference diagnosis results of plant A (left) and B (right)*

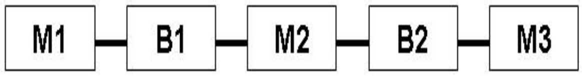


**Figure 9.** *Three machines $M_i$ with buffers $B_j$ in between*

In order to properly understand the origin of the symptom and for determining appropriate remedies, it would be necessary to reveal the role of $M_1$ as the originator or, at least, one contributor to the problem. While it is possible to extend the time windows and the observed behavior patterns in the tool, it is not obvious whether and how to distribute the blame among the components, even more if there are more complex situations of disturbances in several machines.

The core of this problem lies in the fact that there is an element of the designed plant behavior that is not captured by the local component models: although there is no global control, there do exist some principles and intentions how to run the plant properly. For instance, as mentioned in section 2, the output rates of the machines should increase both upstream and downstream from the filler (to avoid filler lacks and tailbacks, respectively). For the same reason, the filling degree of buffers should be "neither too high nor too low". Unfortunately, this healthy degree is not fixed and, more importantly, cannot be measured directly or appropriately estimated from other sensor data.

A solution to this problem is one of the main targets of the follow-on project. Besides a principled way to assess the contribution of several machines, this will require the representation of more complex behavior patterns, such as "stuttering" of machines. The current demonstrator has already been designed to support this by enabling the definition of complex features via more low level variables. Stuttering, for instance, can be specified as the presence of an oscillation between different status values.

## ACKNOWLEDGEMENTS

Our work would have been impossible without the expertise and help of our colleagues from the department of food packaging technology (LVT) and the other members of the Model-based Systems and Qualitative Modeling Group (MQM). We are also indebted to the end users and suppliers for their support, which was well beyond what you can usually expect from industrial partners.

## REFERENCES

[Kather et al. 05] Kather, A.; Voigt, T.; Langowski, H.-C: Plug and Acquire – The New Weihenstephan Standards For Process Data Acquisition In Bottling Plants. Proceedings of the 30th EBC Congress, Prag. Nürnberg: Hans Carl, 2005, on CD

[RAZ'R 09] www.occm.de

[Struss 88] Struss, P.: Extensions to ATMS-based diagnosis. In: J. S. Gero (Ed.) Artificial Intelligence in Engineering: Diagnosis and Learning, pp. 3-27, August 1988

[Struss et al. 08a] Struss, P., Kather, A., Schneider, D., and Voigt, T.: Qualitative Modeling for Diagnosis of Machines Transporting Rigid Objects. In: Bradley, L. and Trave-Massuyes, L (eds.). 22nd International Workshop on Qualitative Reasoning QR08, Boulder, Co, 2008

[Struss et al. 08b] Struss, P., Kather, A., Schneider, D., Voigt, T.: Qualitative Modeling for Diagnosis of Machines Transporting Rigid Objects. In: Grastien, A. and Stumptner, M. (eds.),19th International Workshop on Principles of Diagnosis (DX08), Blue Moutains, Australia

[Williams 86] Williams, B.: Doing Time: Putting Qualitative Reasoning on Firmer Ground." In: Proceedings of the National Conference on Artificial Intelligence, Philadelphia, PA, 1986, pp. 105-113