## Martin Sachenbacher: Research Interests

Below, I have structured my current research interests into four main topics, all of which have to do with model-based and qualitative reasoning, behavior prediction and consistency-based diagnosis. The first two issues are more scientific, while the last two have more to do with applications. Of course, there are several interconnections between these directions, and boundaries are sometimes difficult to define. I also tried to identify their possible impact on and their usefulness for the field of model-based autonomy.

**Automated Qualitative Abstraction**

Modeling is one of the hardest, but also one of the most crucial parts of model-based reasoning; having a model that is adequate for the task at hand is the key to keep reasoning efficient and computationally tractable. Lately, much of my work has been concerned with a modeling problem we called *task-dependent qualitative model abstraction*. The goal is to determine a level of abstraction for a behavior model (composed from a library of generic model fragments) that is "right" for the purpose, i.e. as coarse as possible, but still fine enough to accomplish a given behavior prediction or diagnosis task.

Up to now, we have limited ourselves to domain abstraction only. Task-dependency is characterized by "observable distinctions" defining the granularity of possible inputs to the problem solving process (e.g., observations), and "target distinctions" defining the granularity of its desired outcome (e.g., distinguishing different behavior modes for diagnosis). The problem can then be re-formulated as the search for distinctions within the domains of variables (i.e., qualitative values) that are both necessary and sufficient to achieve the same results as the original model with respect to these task-dependent distinctions.

In my thesis work, I identified preconditions under which a (unique) solution to this problem exists, devised algorithms to automatically determine qualitative values, and developed a prototypic implementation termed AQUA (Automated Qualitative Abstraction). One of its most interesting applications is to turn real-valued models - as commonly used in industry - into qualitative models, making them accessible to automated reasoning methods. In an on-going project together with DaimlerChrysler, Fiat and Renault, AQUA is currently used for constructing qualitative abstractions of numerical (Matlab/Simulink) models with the goal to exploit them for diagnosis.

The impact of this research topic on model-based autonomy might primarily lie in providing "tailor-made" models that improve the chances of meeting the tight space and run-time requirements of on-board applications. In this context, task-dependent domain abstraction is beneficial in two respects: the size of the model is reduced, but also the number of observations is brought down, as only essential (qualitative) changes in the measurements need to be processed. Other benefits include the possibility to exploit both real-valued and qualitative (i.e. "hybrid") behavioral constraints within a common framework, and the ability to re-use models along different tasks, which supports the creation of model libraries.

**Structural CSP Decomposition**

A second major focus is on the challenge to provide efficient solutions for large problems that involve several hundreds or thousands of constraints and variables. Over the last years, I have been concerned in this context with *decomposition techniques* that aim at systematically exploiting structural features of constraint satisfaction problems (CSPs) (reflecting influence of my second advisor, Georg Gottlob, who made major contributions to this area).

Structural decomposition is based on the fact that problems are efficiently solvable if they are acyclic (tree-structured). It provides a generalization of acyclicity, based on the concept of width as an explicit measure of cyclicity. The basic principle is that an arbitrary CSP can be decomposed (or "clustered") into an equivalent tree-structured CSP, with an effort that is determined by its width. The resulting tree-structured CSP can then be solved efficiently and completely using local constraint filtering techniques.

In the context of model-based problem solving, structural CSP decomposition can be thought of as a "pre-compilation" of models, as it can be performed off-line (before observations are put in). In my thesis work, I used a combination of structural decomposition and ordered binary decision diagram (OBDD) methods to encode and solve CSPs that represent behavior models. Although the implementation was prototypical, the results were really amazing, allowing to determine complete solutions for problems that were out of reach for OBDD representations alone. Challenges for future research include the design of diagnosis algorithms based on clustering of constraints and the development of good decomposition strategies.

Benefits of this research direction for model-based autonomy might lie, first, in providing a means to handle real-world problems with very large state spaces; and second, in improving efficiency by the possibility to pre-compile models, which could be particularly helpful within on-board and embedded applications. Third, based on the duality between constraint satisfaction and propositional inference, structural decomposition lays an interesting foundation for deductive algorithms that have guaranteed completeness and complexity properties, which might be useful for real-time applications such as monitoring.

**Towards Model-based Engineering**

A third interest centers around the application of model-based technology to real-world dynamic systems, its integration into devices, and its transfer into industry. In our group, *state-based diagnosis* was put forward as an efficient method to diagnose dynamic systems and turned out successful in a number of applications. However, it makes assumptions about measurability, whose scope can only be clarified in further practical experiments.

For building model-based devices, generating diagnostic candidates or predictions for variables is not the ultimate solution, but rather a sub-problem that needs need to be integrated with other tasks such as *control, planning, testing and recovery*. We began to tackle a limited form of testing and recovery in the domain of vehicle diagnosis, with the goal to determine ("plan") appropriate actions that re-establish the functionality of the respective system as far as possible in the case of a failure. The task-dependent distinctions mentioned above play a role here as a means to express high-level goals explicitly.

To be useful in industry, model-based solutions-in-principle finally have to be turned into tools that support the actual *engineering work process*, and interfaced with established simulation or CAD software. As part of the automotive project mentioned above, we have recently been working on the extraction of structural information (constraint types and their interconnection) from Matlab/Simulink models, to smoothly integrate them in a model-based framework. Another contribution is the standardization of behavioral descriptions as a prerequisite for exchanging models. In this context, I worked on the definition of an XML-based format that allows to represent any relational (real-valued or discrete) constraint.

The possible impact of this topic lies in the transfer and further integration of model-based methods into the pragmatics of current industrial work process (i.e., towards *model-based engineering*); in integrating and interfacing the technology with real-world applications (i.e., going beyond hand-crafted prototypes); and in a closer integration of consistency-based diagnosis with other problem-solving tasks such as control, planning, testing, and therapy.

**Bioinformatics and Automated Modeling from Data**

Turning gene expression data into an actual understanding of the functioning of an organism, i.e. obtaining models of metabolic processes, is considered a major challenge for biotechnology in the next century. I believe that a systematic approach grounded on model-based reasoning, in particular model-based diagnosis, could be instrumental in this context. Recently, I started work on methods for data-driven, automated modeling within a consistency-based framework.

The basic idea is that whenever a discrepancy occurs between predictions and observations (actual gene expression profiles, as obtained e.g. from DNA microarrays), model-based diagnosis techniques can be employed in order to find possible revisions of the model (on its conceptual level) that account for the difference. That is, given an initial, incomplete model describing known pathways of cellular interaction, the diagnostic candidates (corresponding to a subset of the model's conceptual elements) could serve as well-directed "hints" on interactions that have to be newly hypothesized, or need to be dropped, in order to be in accordance with the observations.

This approach of *consistency-based, automated model revision* puts high demands on the involved diagnosis engine in terms of completeness and efficiency. For instance, incomplete prediction or consistency checking procedures would make it difficult to tell whether revision of the model is really necessary or not (observations reported as consistent with the model could in fact be inconsistent, as the algorithm might be unable to detect the inconsistency). Also, model revision lends itself towards qualitative modeling, because it requires a finite number of possibilities. Thus, there are several connections to the topics mentioned above.

I elaborated on these ideas together with Hidde de Jong (Inrialpes, France), who leads a group on modeling and simulation of genetic regulation systems. A former colleague and co-author of me, Ulrich Heller, worked on the extension of consistency-based diagnosis to a process-oriented ontology as one of the theoretical foundations of this approach. Bioinformatics is but one specific target of this research - I like to pursue it because I have some background in biochemistry. However, the methods themselves should be applicable to a broader range of domains involving learning problems and explanation generation.