INTERACTIVE TOOLBOX FOR 4D-MODELING

Fabian Märki[1], Manfred Vogel[2], Manfred Breit[3] and Martin Fischer[4]

ABSTRACT

In this paper we present an ongoing research project that focuses on the development of an Interactive 4D–Modeling Toolbox for the 4D modeling of buildings. The aim of this toolbox is to support and speed up building of 4D models, to facilitate changes on them and to control progresses during the building process.

Up to now, the Interactive 4D–Modeling Toolbox consists of:

- 4D–Model Builder, tools supporting the set up of 4D models
- DES, a Discrete Event Simulator for the automated sequencing of activities into a network plan
- GAPO, a Genetic Algorithm Process Optimization module that optimizes project schedules in terms of time, cost and resource management
- A 4D–Player for the visualization of building processes

The 4D–Model Builder supports the identification of building components and the definition of structural dependencies between them. The DES successfully proves that the commonly used component-activity-resource relations can be automatically sequenced by taking structural and process constraints into account. Thereafter, GAPO can be used to optimize the generated network plan in terms of time, cost and resource management – forming a multiple criterion optimized project schedule. Finally, the project schedule can be combined with the 3D–Model of the building – forming a 4D–Model – and visualized trough the 4D–Player.

In this paper we give an overview of the entire toolbox and subsequently a more detailed description of GAPO.

KEY WORDS

4D–Model, 4D Process Planning, Activity Sequencing, Optimization, Genetic Algorithms

1 Computer Science Eng. FH and Master Student in Construction Engrg. & Mgt. , Center for Integrated Facility Engineering, Stanford University, Building 550, Stanford, CA 94305-4020, USA, Phone +1 650 497 7843, FAX +1 650 725 6014, markif@stanford.edu

2 Professor, Mathematics. Institute 4D–Technologies and DataSpaces, University of Applied Sciences Northwestern Switzerland, Klosterzelgstrasse 2, Windisch, CH-5210, Switzerland, Phone +41 56 462 40 77, FAX +41 56 462 44 82, manfred.vogel@fhnw.ch

3 Professor, Construction Engrg.. Institute 4D–Technologies and DataSpaces, University of Applied Sciences Northwestern Switzerland, Klosterzelgstrasse 2, Windisch, CH-5210, Switzerland, Phone +41 56 462 40 79, FAX +41 56 462 44 82, manfred.breit@fhnw.ch

4 Assoc. Professor, Civil and Envir. Engrg, Center for Integrated Facility Engineering, Stanford University, Building 550 Room 553H, Stanford, CA 94305-4020, USA, Phone +1 650 725 4649, FAX +1 650 725 6014, fischer@stanford.edu

INTRODUCTION

The aim of our research is to develop tools that speed up assembly of 4D models, to facilitate changes on them and to monitor and control the building process on site. So far, we have developed four different types of tools and these tools have been incorporated into the Interactive 4D–Modeling Toolbox.

The Interactive 4D–Modeling Toolbox provides a framework that is extensible and therefore provides useful mechanisms that can be used for further research in the area of 4D modeling, optimization, project control and risk management. We consider the GAPO as the most innovative tool in the existing collection and therefore GAPO is going to be the main focus of this paper.

INTERACTIVE 4D–MODELING TOOLBOX

OVERVIEW

Currently, the Interactive 4D–Modeling Toolbox incorporates four types of tools.

The 4D–Model Builder supports the set up of 4D models. So far, a basic tool has been developed that partially automates the identification of structural dependencies between the structural components to derive the sequence constraints between activities. In this area, we currently develop mechanisms aiming at a highly automation of 4D information extraction from 3D models. This includes the improvement of the derivation of the structural dependencies; mechanisms that derive dependencies between structural components resulting in process constraints between activities and mechanism that determine work breakdown structures. The final goal is to automate the set up of 4D models as it is shown in Figure 1.

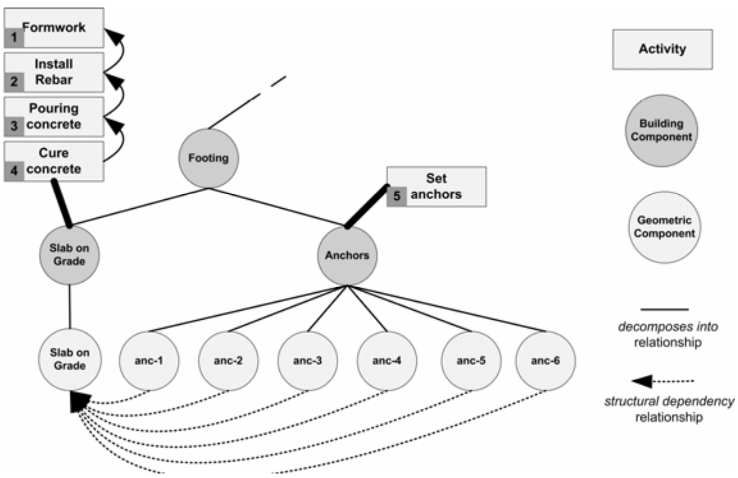

Figure 1: Representation of a 4D Model (Aalami et al. 1998)

The 4D–Model Builder can identify structural components and structural dependencies. The concept behind both tasks of 4D information extraction is the same, a combination of a rule engine and a fuzzy logic (see Figure 2). Depending on the information one wants to extract, the rule engine and the fuzzy logic can be loaded with different sets of rules and similarity functions. If the rule engine cannot come up with a classification, a classification atempt is made using empirical knowledge retrieved from a similarity database. If a classification is too uncertain, the user is asked for a specification. The user input is then used to update the rule or similarity database and a new classification attempt is started. This provides a learning mechnism, allowing an automated classification in the future.

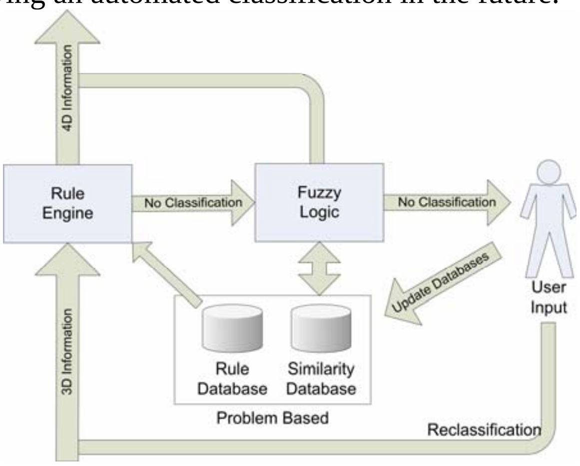

Figure 2: Concept of 4D information extraction

The second tool, the DES (Discrete Event Simulator), performs an automated sequencing of activities into a network plan[5]. It operates on the component-activity-resource relations (Aalami et al. 1998) and takes structural and process constraints into account. The structural constraints are extracted from structural dependencies determined by the 4D–Model Builder.

The third tool optimizes the network plan. The result is a project schedule[6] optimized according to multiple criteria prioritized by the user. Our implementation uses a Genetic Algorithm approach to perform the optimization and therefore it is called GAPO (Genetic Algorithm Process Optimization). GAPO is the main focus of this paper and will be described in more detail.

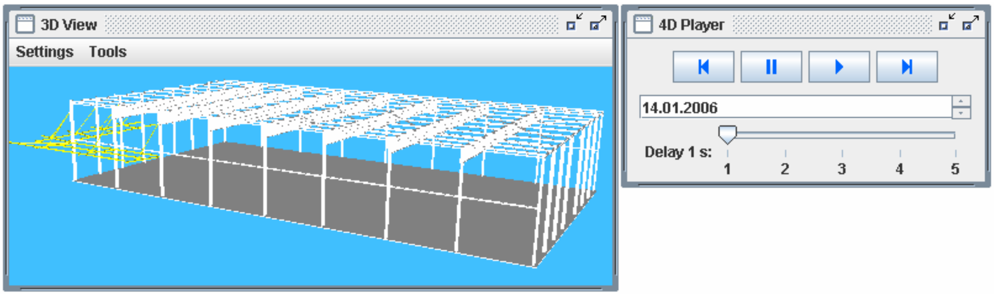

Figure 3: 4D Player Interface

5 Sequenced activities with predecessor relationships.
6 Network plan with time and resource allocations.

The fourth tool visualizes the building process and thereby supports decision making as well as the recognition of possible clashes during the construction phase. Beside the usual 3D visualization we have implemented a 4D–Player, as it is known from other 4D software.

GAPO - GENETIC ALGORITHM PROCESS OPTIMIZATION

Compiling project schedules is a time consuming and complex task. Due to time constraints and the NP-hardness of the scheduling optimization problem[7], one does usually not implement a sensitivity analysis and projects are performed with the first found feasible project schedule. With GAPO, we want to address this problem and show the potential of a software application that supports the compilation of a pareto optimal project schedule.

GAPO is based on a Genetic Algorithm (GA) approach to optimize network plans in terms of time, cost and resource management (Märki and Suter 2003). GAs are a class of heuristic search methods based on the Darwinian principle of evolution[8]. It mimics and exploits the genetic dynamics underlying natural evolution to search for optimal solutions of general combinatorial optimization problems (Coley 1999).

Genetic Algorithms start from a pool of individuals which are scored by Fitness Functions measuring their quality as a candidate solution of a given problem. By some probabilistic mechanism these solutions are exposed to an artificial evolution consisting of Selection, Recombination and Mutation yielding a new generation of candidate solutions which are expected to have a higher quality of Fitness. Genetic Algorithms have been successfully applied to a wide variety of practical problems in diverse fields like chemistry, biology, operations research and many engineering disciplines (Coley 1999).

This innovative method for building project planning has the following advantages:

- The project schedule can be optimized according to specific criteria
- It becomes possible to react efficiently on incidents during the construction phase in order to determine the further optimal procedure
- 4D–Models support the decision-making process and they can be used to control and optimize the planning over the whole lifecycle of a building

GAPO Evolution Model

Our Evolution Model starts with an initial population of randomly generated project schedules. A subsequent population will then be assembled using five strategies which can be weighted by the user. A fraction q of the best individuals will be directly passed to the next population. This guarantees that the quality of the most suited candidates will monotonically increase from generation to generation. A second fraction r of individuals will be passed to the next population after a mutation. On one side, this process opposes early convergence in a local optimum and thereby helps to open new search regions. On the other side, it also allows a fine tuning of suitable solutions by applying small chances on them. A third fraction s of the new population is created by recombining individuals from the old generation. This process forces convergence into an optimum. A fourth fraction t is also created by

7 Wikipedia: http://en.wikipedia.org/wiki/Scheduling (Last accessed: 19th of February 2006)

8 Wikipedia: http://en.wikipedia.org/wiki/Genetic_algorithm (Last accessed: 19th of February 2006)

recombining individuals but instead of passing them directly into the new population the new individual is mutated beforehand. Last, a fraction *u* of the new population is created randomly. This process also helps to open new search regions and prevents early convergence in a local optimum.

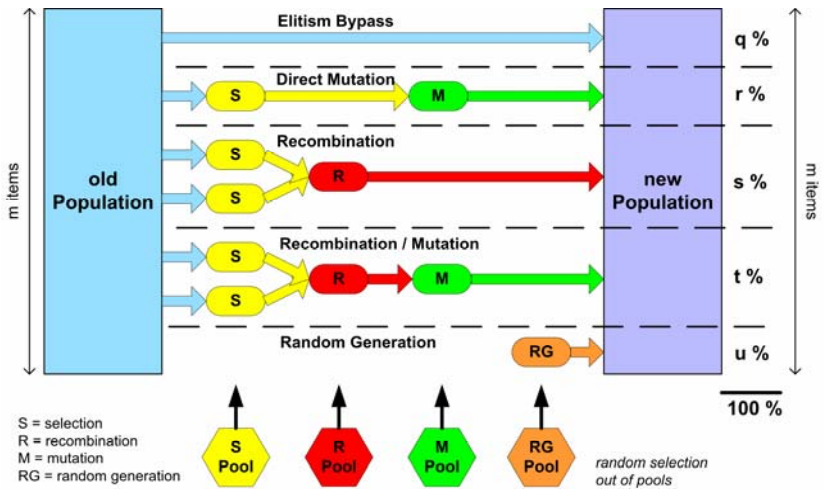

Figure 4: GAPO Evolution Model (Märki et al. 2004)

GAPO Data Structure

The success of GA's in optimization problems crucially depends on an appropriate encoding of the different parameters being optimized. In addition, in process schedule optimization sequencing constraints have to be satisfied. Our representation of a project schedule as a chromosome is shown in Figure 5. The chromosome is defined as a sequence of genes representing the sequencing relation defined by the network plan. All genes differ from each other and have particular characteristics.

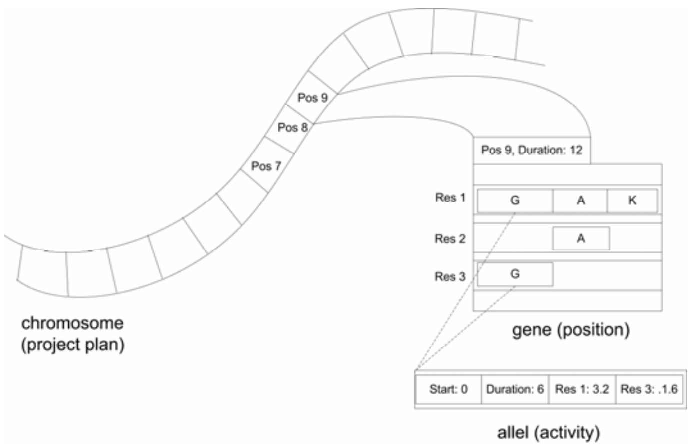

Figure 5: Genetic representation of project schedule

According to the network plan, activities are allocated to positions and to resources as shown in Figure 6. All this information is stored in the genes. In the simple case, an activity can unambiguously be allocated to a well defined position. On the time axis, such an activity will be scheduled somewhere between the start and end time of the corresponding position. Within this position, the activity can be shifted and is only constrained by the resource it is using. In more complicated cases, an activity can be allocated to several positions; such an activity will be called *commuter*. It might be that a commuter starts in one position and extends over some subsequent positions. Again, its particular scheduling on the time axis will be determined by the used resources.

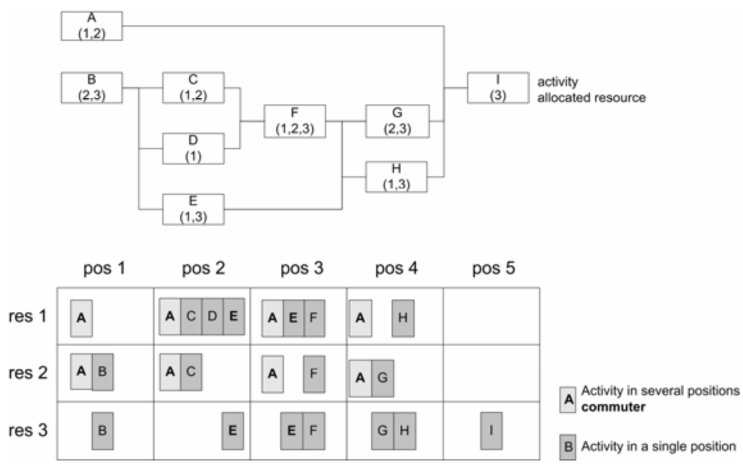

Figure 6: Example of a simple network plan and its allocation to genes (positions)

Genetic Operators

A number of operators are required to manipulate the chromosomes of a given generation in order to construct a new generation of hopefully fitter individuals. In general, we distinguish between three types of operators: selection, recombination and mutation.

The selection determines which project schedule can contribute *genetic material* for the next population. We have implemented the following selection operators:

- *EliteSelector*: Selects randomly from the best *v*% project schedules
- *RandomSelector*: Selects randomly from all project schedules
- *RouletteWheelSelector*: Fitness weighted selection from all project schedules
- *TounamentSelector*: Determines *q* project schedules by random and selects the best one

During recombination, pieces of genetic material are exchanged between a pair of chromosomes. This allows the creation of genetically radically different offspring that are still of the same general flavor. We have implemented the following recombination operators:

- *SimplePositionRecombinator*: Recombines schedules by cutting at arbitrary positions

- *ShorterPositionRecombinator*: Selects from the parents the shorter position for every position
- *ResourceThreadRecombinator*: Selects resource threads and fixes resource conflicts
- *ResourceUseRecombinator*: Selects activities from one and resources from another parent

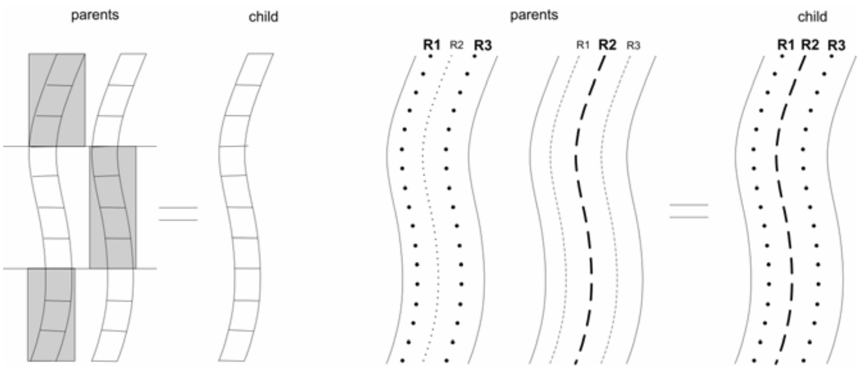

Figure 7: Recombination of positions and resource threads

Mutators operate on activities by either changing their parameters or positions arbitrarily. Possible resource conflicts are automatically resolved. We have implemented the following mutators:

- *ResourceExertionMutator*: Mutates the resources at some positions/activities
- *RandomShuffleMutator*: Mutates the sequence of activities within a position
- *PackerMutator*: Tries to pack activities as close as possible
- *CommuterMoveMutator*: Mutates the position of commuters

In addition to the above operators we also use a random project schedule generator (Gonçalves 2002). The very first population is randomly created by this generator. As already described in the GAPO Evolution Model (see Figure 4), we also use it during the optimization process to add new genetic material into the population. This random generator produces valid project schedules by taking the defined scheduling and resource constraints into account.

Fitness Functions

The quality of every project schedule is measured by fitness functions. In general, a project schedule is superior if its duration is shorter or if the corresponding cost is lower.

For measuring the fitness of a project schedule, several aspects can be taken into account: the total project duration, the duration of resource use, the amount of resources, increase, reduction and change of resources and unused resources. These aspects can be weighted by the user and provide a final fitness consisting of a linear combination of the following fitness functions:

- *ProjectDurationFitnessFunction*: Measures total project duration
- *ResourceDurationFitnessFunction*: Measures duration over which resources are used

- *AreaFitnessFunction*: Integrates amount of resources used during the project
- *ResourceChangeFitnessFunction*: Integrates changes in the amount of used resources
- *GapFitnessFunction*: Measures duration where available resources are unused

A PRACTICAL APPLICATION

Facility for recycling of cars

A one story high steel hall for a planned car recycling center (see Figure 3) is used as a practical test case for GAPO.

Construction issues

The facility has the dimensions 60m/30m/10m. It has a light, hinged structure, stabilized through bracings in the roof and walls. For a basic sensitivity analysis mobile cranes (25t/60t/100t) and hydraulic scissor lifts are available to erect the components. Steel crews are typically set up with three laborers, one to serve the crane, the others to set and fix the structural elements. The erection processes are as follows: frame-wise erection, temporary fixing of bracing and secondary elements; adjustment and final torque.

Application of GAPO

The 4D Model consists of 305 structural 3D elements, 23 conceptual groupings, 63 activities and 5 resources (concrete workers, steel workers, unskilled workers, cranes and scissor lifts). After sequencing, the network plan was optimized using GAPO with the described fitness functions.

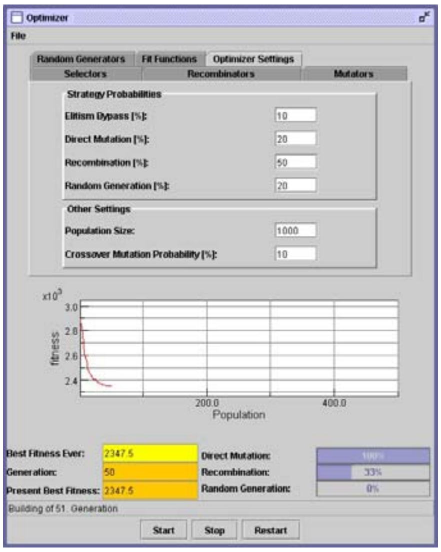

Figure 8: Screenshot of the optimizer dialog box (Optimizer Settings) of GAPO

The typical behavior is that the quality of the project schedule improves rather quickly during the first twenty to thirty generations (see Figure 8). If the total fitness is only determined by the *ProjectDurationFitnessFunction*, the optimization reduces the project duration by about 20% against a first generated project schedule. If only the *AreaFitnessFunction* is considered, the total use of resources is again optimized by about 15% to 20%. Of course, these numbers only apply to the specific problem but we expect a higher optimization potential in bigger and more complex projects.

ONGOING RESEARCH

To further advance research on the optimization of project schedules, collaboration between CIFE[9] and *i*4D*s*[10] has been established. One of the main goals addressed is the development of a GAPO tool allowing the optimization of arbitrary project schedules.

Beside the optimization of the project schedule, we also work on sensitivity analyses that take place during the optimization process. Addressed levels of variations are:

- The first level is a change in the quantities of the resources used within a resource group
- The second level is a change in the resource group assigned to an activity

Figure 9: Variation levels of sensitivity analysis

To enable sensitivity analyses, the resource management must be extended to model relationships between resources and between activities and resources. This resource management must also be general enough to provide the possibility to optimize arbitrary project schedules. Moreover, we plan to introduce a concept that enables GAPO to consider space conditions on the construction site. Since a change in the used resource group can influence the sequence of the activities and this sequence has been considered fixed so far, it is also essential to adapt the genetic representation to this new requirement.

FURTHER ROADMAP

Since the user has limited possibilities to direct the focus of the optimization, we need to introduce mechanisms that enable the user to give the optimization a desired direction and consequently have more control about the outcome of the optimization. Key elements to address are the fitness functions. But also mechanisms like soft constraints combined with the possibility to lock certain parts of the schedule to conserve once found good parts of a project schedule will be considered.

An extension of the sensitivity analysis with a third variation level would be desirable. This level addresses a change in the process method used to perform a specific task.

---

[9] Center for Integrated Facility Engineering, Stanford University
[10] Institute 4D–Technologies & DataSpaces, University of Applied Sciences Northwestern Switzerland

Furthermore, our optimizer requires parameters to be set by the user. This is essential for research but rather cumbersome for field usage. The optimizer should be able to adapt its parameters according the progress during the optimization. We need to develop an evolutionary algorithm so that the user can handle GAPO almost as a black box without knowledge about optimization methods, Genetic Algorithms, etc.

In addition to the described four tools, we are planning to extend the Interactive 4D–Modeling Toolbox with project control instruments and risk management tools. The former will simplify the control of the project during the construction phase and the latter will support the decision making and investment strategies of the management. For project control, we are considering technologies like mobile computing and voice recognition. To explore the potential in voice recognition, we already have a project in progress that addresses the indexation of meeting records based on key words[11]. These two technologies could for example dramatically simplify the monitoring of the project's state, in turn allowing a concurrent project control.

CONCLUSIONS

GAPO shows that project schedules can automatically be created and optimized according multiple criteria. By evaluating our developments on further real world projects, we will be able to substantiate the potential of a tool for project planning that supports a combination of sensitivity analyses with project schedule optimization.

Furthermore, we think that combinations of tools similar to the 4D–Modeling Toolbox will finally enable that planning, decision making and control processes can be fully based on 4D models.

REFERENCES

Aalami, F. B., Kunz, J. C. and Fischer, M. A.; (1998). *Model-Based Mechanisms for Automated Activity Sequencing*. Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305-4020, USA.

Coley, D. A. (1999). *An Introduction to Genetic Algorithms for Scientists and Engineers.* Danvers, USA.

Gonçalves, J. F., Mendes, J.J.M., Resende, M. G. C.; (2002). *A hybrid genetic algorithm for the job shop scheduling problem.* AT&T Labs Research.

Märki, F., Suter, M., Vogel, M. and Breit, M.; (2004). *Optimization of 4D Process Planning using Genetic Algorithms.* Proceedings of the "Xth International Conference on Computing in Civil and Building Engineering", Weimar, 02-04 Juni 2004, p 1-12.

Märki, F. and Suter, M. A.; (2003). *Projektplanoptimierung mit Genetischen Algorithmen.* Diploma thesis, unpublished, University of Applied Sciences Aargau, Switzerland (available at http://www.i4ds.ch/forschung/4d/download/DES.pdf).

Ono I., Yamamura M. and Kobayashi S.; (1996). *A Genetic Algorithm for Job-shop Scheduling Problems Using Job-based Crossover.* Proceedings of IEEE International Conference on Evolutionary Computation, p 547-552.

[11] LAKE: http://www.lake-project.ch/de/main.html (Last accessed: 19th of February 2006)