Evolutionary Algorithms for Optimizing Traffic Signal Operation

Andrea Vogel, Christian Goerick, Werner von Seelen

2000

Evolutionary Algorithms for Optimizing Traffic Signal Operation

Andrea Vogel,* Christian Goerick,† and Werner von Seelen
Institut für Neuroinformatik, Ruhr-Universität Bochum
44801 Bochum, Germany
Phone: +49-234-3225567, Fax: +49-234-14209
e-mail: andrea.vogel@neuroinformatik.ruhr-uni-bochum.de

ABSTRACT: Maintaining efficient transportation systems has become a key goal for government-based transportation agencies in this age of serious energy shortages, rising costs and competing recipients of public funds. The emphasis has shifted to optimizing traffic movement on the existing surface street network rather than to provide more freeways. As described in Schnabel and Lohse (1997), the network capacity is limited by intersections. In this paper, we present an approach to automated optimization of traffic signal operation. We present a problem decomposition and encoding for the use of different traffic models as well as a variety of optimization tools. First experimental results show that an automated design and optimization of intersection signal operations is not only possible but yields good results. This paves the way for on-line self-adapting signal operation: combined with a traffic observation system it is not only possible to react to short-term variations in traffic flow but also to adapt to long-term changes of the situation.

KEYWORDS: traffic, optimization, evolutionary algorithms

INTRODUCTION

The increase of urban traffic is causing significant problems. Since the possibility of providing more streets is limited by economical and ecological factors and thus cannot keep up with traffic growth, it is necessary to optimize traffic movements on the existing street network. There are two approaches to this aim:

1. Optimizing the network. This means on one hand optimizing the network topology by building new roads, changing their use, or introducing a clever routing, on the other hand it means smoothing traffic flow by a system of coupled intersection signal operations.

2. Optimizing intersections. Since intersection parameters are local, an intersection optimization can be based on local traffic attributes. Of course these local traffic parameters are influenced by the intersection's environment. Surrounding traffic signal installations for example can form platoons out of a continuous traffic flow and determine the arrival times of these platoons. Thus adjacent intersections are coupled by the traffic and therefore functionally interlinked to a network.

As described in Schnabel and Lohse (1997), intersections mainly determine the capacity of urban traffic networks. Therefore optimizing intersection signal operation is a quick and efficient way to cope with the characteristic problems of urban traffic: enhancing the permeability of intersections leads to lower traffic densities and reduced travel times. Since shorter travel times and less stop and go traffic reduces the fuel consumption, the urban ecology will benefit due to reduced pollution and noise, too.

Hence, a variety of different signal operation systems has been devised. Parameters of the traffic flow are being monitored more and more detailed in order to react to changes of the traffic situation more flexibly. Since each new signal operation system and each new traffic observation leads to a new and often larger set of parameters, the adaptation to specific intersections and traffic situations becomes more and more complex and obscure. Our aim is to show a constructive way to an appropriate adaptation in order to bring signal operation systems to their full efficiency. In this paper we present an approach to automated optimization of traffic signal operation. After a short description of the optimization task, we present a problem decomposition and encoding for using different signal operation systems and traffic models as well as a variety of optimization tools. The last part of this paper shows first experimental results.

*This work was sponsored by BMB+F as part of the LEONET project, grant number 01 IB 802 C4.
†current address: Future Technology Research Division, Honda R&D Europe (Deutschland) GmbH, Carl-Legien-Straße 30, D-63073 Offenbach

OPTIMIZING SIGNALIZED INTERSECTIONS

Optimizing signalized intersections is a task that usually cannot be solved analytically. Choosing adequate green times alone leads to a system of coupled differential equations that usually has no closed form solution. Models leading to such a closed form solution are strongly simplified. Therefore, they are only valid for limited parts of the state space: for high traffic densities, i.e. when good signal operations are crucial, they are not valid. Other macroscopic models normally do not account for all necessary details. Unlike these closed form models, traffic simulation models are always applicable. Unfortunately, they do not provide us with a way for finding a signal operation in an analytical way. This leads to a trial-and-error design process: Based on heuristics and expert knowledge, the traffic engineer proposes a signal operation and tests its effectiveness by means of simulations. He may test one or two other configurations, but due to the duration of this process he has no possibility for exuberant experiments. This design process usually yields reasonable results. But it has two significant drawbacks:

- The design relies heavily on individual experience. Solutions beyond the traditional way usually will not be found, even though special problems often need special answers. Especially for peak demand periods, the risk of being remote from an optimal signal operation is high. (For a similar problem of parameter adaptation see Bergener et al. (1999))
- The process is not accessible to self adaptation. The design of signal operations is difficult and expensive and therefore mostly is performed only when traffic signals are installed. During a signal system's lifetime traffic usually changes significantly. Due to the complicated and expensive design it is uncommon to react to long term changes of traffic conditions. This leads to a large amount of ill conditioned intersections which are remote from optimal efficiency.

Automated design and optimization is a remedy. Especially evolutionary algorithms are known to efficiently search high dimensional data spaces. Combined with a traffic observation system (see e.g. Bücher (2000)) it paves the way for an on-line adaptation on changing traffic conditions.

In our experiments, we concentrate on a classic system of signal operation: Traffic movements are grouped to a certain number of phases, phases are put into a sequence and assigned a green time. Time gap detectors and vehicle detectors are employed to provide the necessary information in order to respond to short-time traffic fluctuations. The described system is a standard configuration that includes all principle aspects of a common real world intersection that have to be considered. Even though it might not be the most modern system, it is generally used and provides a sufficiently representative optimization task.

The design and optimization of signal operations can be split up in a combinatorial task of structure evolution and a continuous valued parameter adaptation process. In the following, we will describe the meanings of structure and parameters in terms of intersection optimization.

STRUCTURE

Finding an adequate structure means to decide which traffic movements should get the right-of-way simultaneously, i.e. building phases, and to put them into a sequence. Additionally, it must be determined in how far the signal operation should adapt to short term changes in the traffic situation.

- Phases: Traditionally traffic movements are grouped as described in figure 1.

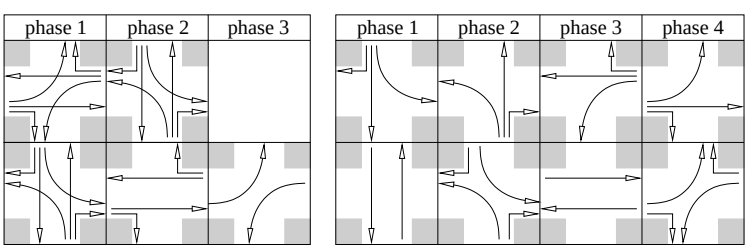


Figure 1: classic phase operations

Other phase systems are possible. A high amount of pedestrians for instance can introduce the necessity of an extra phase in which the complete motorized traffic is stopped and the intersection is reserved for walking. Similar situations can emerge from public transportation that takes priority over the rest of traffic, or, in case of railway traffic, new conflicts can be caused that enforce a regularization differing from the above mentioned standard solutions. In our experiments we will not restrict ourselves on a given set of phase operations. Other restrictions, as from driving physics and law can be implemented as boundary conditions.

- Order of phases: Once the phases are defined they have to be put into a sequence. The order determines the loss time between two green times, which is the clearing time. The number of possible sequences increases dramatically with the numbers of phases: while a two-phase-operation has one single sequence, a four-phase-operation leads to 24 possible phase sequences.
- Traffic Dependence: The search space gets additional dimensions by taking into account the possibility of traffic responsive signal operation. Furthermore this decision has consequences that surpass optimization: A well adapted traffic responsive signal operation can certainly enhance the degree of service, but the need of traffic supervision and complex control software increases the costs of the signal installation.

PARAMETERS

In the framework of signal operation, optimizing parameters means to choose a timing plan adequate to the traffic situation. The dimension of the search space for parameter optimization, i.e. the number of parameters to optimize is determined by the chosen structure. The minimum is one green time per phase to determine. Further parameters occur in case of traffic actuated signalization: The green time adaption due to time gaps between vehicles for example needs a maximum green time and a traffic density threshold as condition when to switch to the next phase. Our approach is not restricted to the above mentioned parameters and can cope with qualitatively different parameter sets as well.

BOUNDARY CONDITIONS

Classical boundary conditions are facts that are, if not explicitly stated, invisible to the optimization process. Typical boundary conditions for signal operations are restrictions by law, e.g. a minimum green time per phase or a maximum cycle length. Other boundary conditions can be given by the vehicle's physics: For a given speed, the time gap between two cars has a minimum value determined by reaction time and deceleration time. These structure or parameter restrictions are rather obvious and can easily be implemented as part of the operators.

Less obvious are those boundary conditions that are part of the fitness function, such as the actual traffic density. In order to gain results of the most possible significance, it can be very useful to clearly identify those hidden boundary conditions. In many cases, an adaptation to changing boundary conditions can be done extraordinarily fast (see Hüsken et al. (2000)).

OPTIMIZATION

Since the design and optimization of signal operations involves combinatorial as well as continuous-valued problems, we decided to use evolutionary algorithms. These techniques have proven to be powerful tools for optimization of both parameters and structure and are known to efficiently search higher-dimensional data spaces. Evolutionary algorithms are able to handle non-analytical fitness functions (such as simulation models). They can be forced to a global search or a hill climbing behaviour towards the nearest (possibly local) optimum.

The idea of evolutionary algorithms is to describe a number of solutions as a population. Each individual of this population represents a solution, i.e. a phase system with its parameters. Each individual has its own quality, its fitness. The fittest individuals will become parents and reproduce: they create offsprings, the next generation of individuals. The reproduction is superimposed by mutations, i.e. a systematic error. So each offspring differs from its parent. Since the fittest survives and generates similar but not identical children, the search converges towards a global optimum. For a description of evolutionary algorithms, see Rechenberg (1994).

Structure evolution and parameter adaption are processes which extract information from the environment in an iterative process: In every time step (generation) the algorithm proposes a quantity of solutions. These solutions are tested in the environment, i.e. in our case the intersection traffic model. In terms of evolutionary algorithms, the solutions quality is called fitness; the environment is described by the fitness function.

Structure evolution and parameter adaptation proceed on different time scales. Parameter adaptation is a fast process adapting a given structure fast to the environment and boundary conditions. Fast parameter adaptation means flexibility and thus the ability to adapt quickly to changing conditions. Structure evolution provides the basis for parameter adaptation. It takes place on a long time scale. Which structure is stated as good depends strongly on the optimization's aim: Adaptability in mind, we require a structure on which the given parameter adaptation is fast and therefore yields in good fitness values for different boundary conditions. An alternative approach could be a specialized structure that gives the maximum fitness for a single condition. In this case, changing boundary conditions, such as fluctuating traffic densities, enforce a new structure evolution. For more details about structure evolution, see Sendhoff (1998).

The following tasks have to be solved in order to implement such an optimization:

- Find an adequate problem encoding. Structure and parameters have to be represented in such a way that they are accessible for the optimization algorithm. The design of such a representation or encoding is an important task, because it determines the properties of the whole search process.
- Choose operators for the search in structure and parameter space. Evolutionary algorithms enhance the population's average fitness step by step by creating new individuals in the neighbourhood of the last generation and selecting the fittest of them as next generation. The operators for creating these new individuals determine the way through the solution space and therefore determine which solutions can be found and how fast they are reached. Since in our case we have to do a time consuming simulation run for each individual (i.e. for each proposed solution) we are interested in a very fast search.
- Couple structure and parameter optimization in a way that guarantees a stable search.
- Find an adequate fitness function. Here the same arguments as for the encoding apply: The representation of the problem determines the characteristics of the search process.

THE OPTIMIZATION PROCESS

As described in the previous section, the process of optimization consists of several parts. Roughly we distingush between

- a system that generates a signal operation. This contains the evolutionary algorithm with the structure and parameter encoding, the search operators and the coupling of parameter adaptation and structure evolution,
- and a system that tests the proposed signal operation. This contains the environment model that yields the fitness function.

Since we want to be able to use different traffic models as well as different generating algorithms, we demand a modular structure for the optimization process. Thus our process gets a third, passive element: the intersection container. The optimization process gets the following structure:

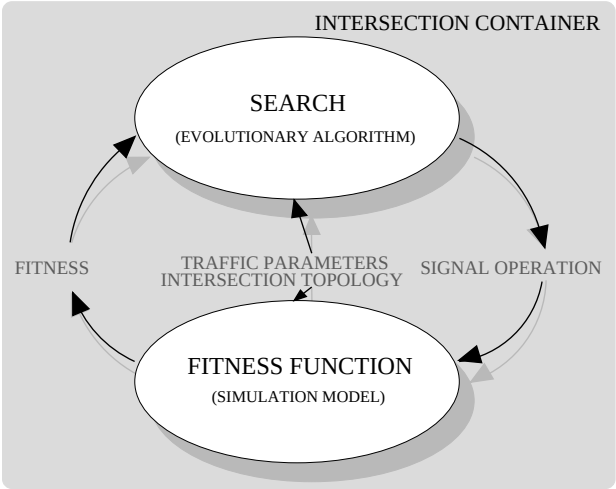


Figure 2: Schematic diagram of the optimization process.

- To remain as independent of special models as possible, we implement a container for the intersection data. This allows an easy changing of the fitness function as well as the optimizing procedure.
- The search algorithm extracts intersection data such as traffic densities, former signal operations and their quality from the intersection container. Based on this information it generates new signal operations and writes them back into the container.
- The traffic model extracts intersection data (including a new signal operation) from the intersection container. Based on this information it evaluates the solution's quality and writes it back into the container.

THE INTERSECTION CONTAINER.

The task of the intersection container is to guarantee the independence of optimization and evaluation. It contains the fixed conditions under which traffic in this intersection takes place: The number of lanes per direction, maximum traffic densities, information about the installed traffic lights etc. Additionally, the actual traffic situation, which has a main influence on the optimization task, is filed in the intersection container.
We decided to implement it as a group of vectors of fixed size. Each vector element is assigned to one traffic movement: The first three elements (0 to 2) represent traffic coming from north and turning left, straight on or right. Elements 3 to 5 describe traffic from west, 6 to 8 cars from south, and elements 9 to 11 the ones coming from east. Three vectors contain the information about the number of lanes per direction (including 0), the actual and maximum traffic density. Possible extension are e.g. a traffic density array instead of the vector to encode more than one traffic situation or model specific parameters.

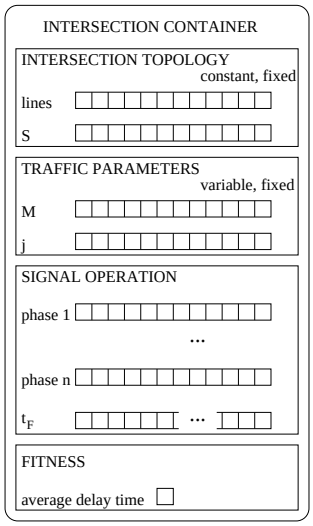


Figure 3: Schematic diagram of The intersection container

THE FITNESS FUNCTION: A TRAFFIC MODEL

Results of optimization depend in manifold ways on the fitness function. Obviously the optimization result cannot be better than the description of reality the model provides. On the other hand, the fitness function can bias the optimization process. An other problem is the time consumption of fitness evaluations. Evolutionary optimization is based on systematically proposing and evaluating a large amount of possible solutions. When fitness evaluations are very time consuming the search has to be restrained and abbreviated. The better and faster the fitness function is, the less elaborated and more exhaustive the search can be.

THE SEARCH ALGORITHM

As previously described, an efficient search requires a problem adequate task encoding and a set of efficient operators.

The Chromosomes - Encoding of the Signal Program

In evolutionary algorithms a solution is represented by an individual. The solution's properties considered as necessary for the optimization are represented in chromosomes. We decided to encode structure and parameters directly. This is done in the following way:

- The first chromosomes contain the phase operation. Each phase is represented by a single chromosome. The phase chromosomes have the same size as the container vectors. If a direction is part of the phase, the corresponding allel (i.e. the vector element) is marked with a 1, otherwise is set to zero. The number of phases is variable. The minimum is 2, the maximum corresponds to the number of directions, when each movement has its own phase. To speed up search, the numper of phases can be restricted.
- The next chromosome contains the order of phases.
- The green time chromosome contains one allel per phase.
- Further chromosomes contain information for traffic responsive signal operation, detector parameters for example.

For some applications it is useful to have a self adaption of the evolutionary process. In this case, each individual can contain further chromosomes with data about the search process.

The Operators

The operators generate new solutions out of the so far best ones. An established way for parameter search is for example to add a normal distributed random number to each parameter. The software packet SHARK, developed at the Institut für Neuroinformatik[1] contains several mutation operators not only for continuous valued problems but also for discrete optimization tasks. Depending on the task and its error surface, more special operators are needed. In case of structure evolution for example it is known that small mutations lead to a more efficient search in the state space than random mutation. (Rechenberg (1994))

Operators are (besides the fitness function) the point where expert knowledge can be included in the search. Instead of spreading new solutions randomly, the search can be more directed.

FIRST EXPERIMENTAL RESULTS

Our experiments are based on a model intersection with four arms. Each arm contains three lines: one line for turn-left-movements, one for straight on traffic and one for turn-right-movements. The main road traffic density is twice as high as on the crossing road. To obtain significant conflicts between straight-on and turn-left traffic we choose an asymmetric arrangement of the traffic density per movement: while the main direction from east and south is straight on, from west and north it is turn left. For a first demonstration of the feasibility of our approach, we restricted our first experiments on non traffic responsive intersections.

PARAMETER ADAPTATION

In our first experiments, we tested the efficiency of different parameter adaptation strategies. We implemented a $(1+1)$-strategie, also known as hillclimbing algorithm. This means that our population consists of one single parent producing one single offspring. Selection takes place among these two individuals. The hillclimber is known to be the most efficient parameter adaptation algorithm in evolutionary optimization(see e.g. Rechenberg (1994)) Comparisions with other population sizes and mutation schemes confirm these results.

As mutation operator, we decided to use a a binomial distribution with zero expectation. We experimented with different variances and different initializations, i.e. starting points for the parameter adaptation. Other experiments dealt with different evaluation schemes: Since the fitness function is noisy, it could proove usefull to evaluate the parent's fitness each time it competes with a new offspring.

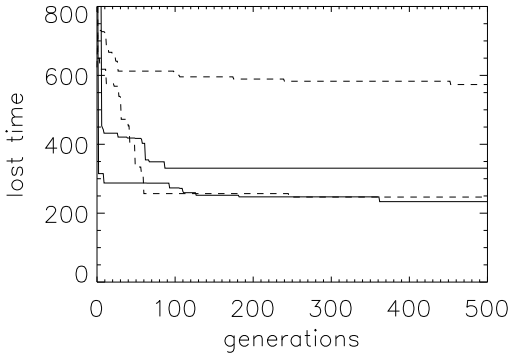
As fitness function we used the event based simulation model SIMVAS++, the simulation system of the Technische Universität Dresden, Fakultät für Verkehrswissenschaften (Ringel (1995)). Immanent to this model (as well as to reality) is a certain fluctuation of the simulation results. Therefore we have to deal with noise in the adaptation process.

Results

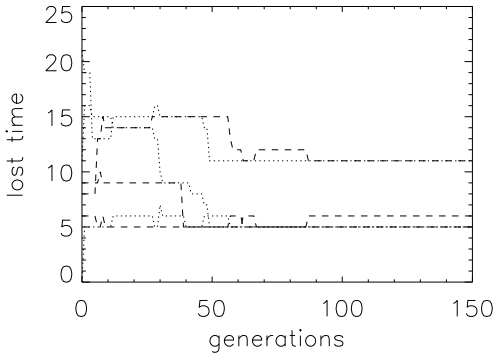
In the following, we describe our results for parameter adaptation under different conditions: We tested the influence of initialization, adaptation step size and the fitness function's inherent fluctuations on the adaptation process and result. Each of the accompaning figures contains the results of one experiment. The left graphic depicts the fitness developement during the adaptation, for each experiment a typical good run and the worst one. The fitness is represented by the sum over the average lost times per lane. Similar for all runs is the fast decay at the beginning of the optimization: The attracting optimum is found very fast. The second graphic contains the developement of the parameters, i.e. the green times per phase. Most runs, even with different fitness values, reach the same or at least very similar parameter sets.

<u>Initialization</u>: A first result is, that the gain of using expert knowledge for initialization is minimal: Started from small random values, the algorithm reaches a good parameter set very fast (see figure 4. In cases of high traffic densities, traditional heuristic green time estimation methods overestimate the cycle length significantly and lead to bad fitness values. Our small random initialization may even be worse at the beginning. But since the decay towards the optimum for growing cycle lengths is much steeper than for shrinking cycle lengths, particular at the start of the adaptation, the hillclimber is significantly faster. Furthermore, the results are less varying: while from five heuristically initialized runs only one reached a reasonable fitness, the randomly initialized runs nearly all ended up in the same minimum.

[1]The software packet SHARK is freely accessible. For further informations please contact dietrich@neuroinformatik.ruhr-uni-bochum.de



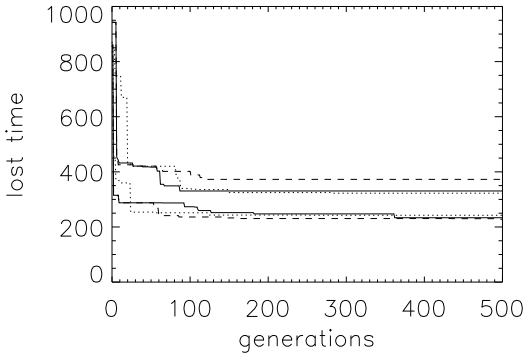
(a) parameter adaptation with different initializations



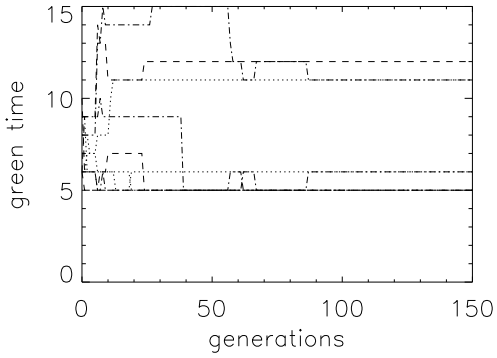
(b) evaluated parameter sets (solid: random init, dashed: heuristic init)

Figure 4: First experiment: parameter adaptation with different initializations

Step size: We had to choose our mutation step size problem adequate. In the literatur about evolutionary algoriths exist a wide range of methods to deal with this task automatically. In these methods, step sizes become part of the optimization process. All these methods are oversized for our task. We compared two non-adaptive strategies (small and large step size) with a simple simulated annealing, where the step size is systematically reduced during the adaptation process (see figure 5. In some experiments we could observe a small speed gain for simulated annealing versus large step sizes. Small step sizes can result in a slow adaptation when starting far from the optimum. The step size does not seem to be a critical parameter.



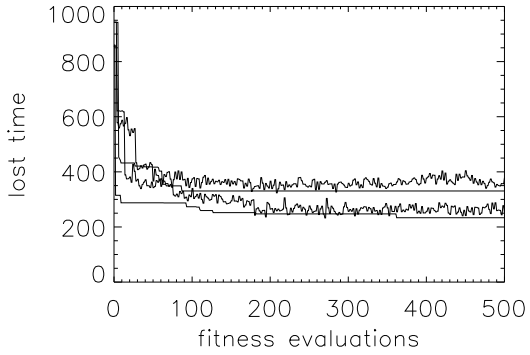
(a) parameter adaptation with different step sizes (solid: large, dashed: simulated annealing, dotted: small)



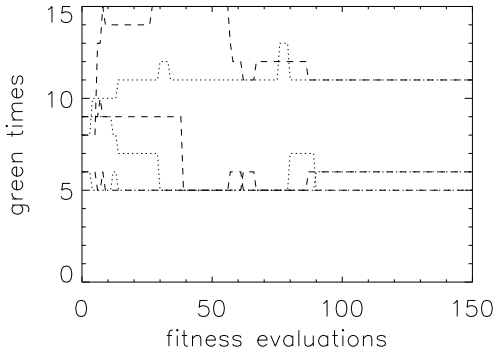
(b) evaluated parameter sets (dash-dotted: large, dashed: simulated annealing, dotted: small)

Figure 5: Second experiment: parameter adaptation with different step sizes

Fluctuations of the fitness function: Our fitness function is inherently noisy, i.e. several evaluations with the same parameter set lead to slightly different results. These fluctuations stem from the random number generator of SIMVAS++. There exist some methods for coping with noisy fitness functions. They are based on an ensemble of evaluations instead of just one evaluation. Therefore, they are very time consuming. Our experiments show, that the gain of such methods is negligible for our task. As it can be seen in figure 6, the qualitative results are the same with and without multiple fitness evaluations. Compared over generations, the fitness developement with noise is nearly the same as without noise. When we compare the run time, taking the noise into account becomes a drawback: Instead of one fitness evaluation (i.e. one simulation run) per generation we need two and thus double the run time of our parameter adaptation.



(a) parameter adaptation with and without noise (large step-size)



(b) evaluated parameter sets (dashed: without noise, dotted: with noise)

Figure 6: Third experiment: parameter adaptation with noisy fitness function

Conclusions: Parameter adaptation is a task that can be easily solved with evolutionary methods. To optimize the parameter set for a given structure, it might prove useful to use simulated annealing to reduce the mutation step size during the adaptation. All the proposed methods lead to similar results: differences in the fitness values are more often caused by the fluctuations of the fitness function than by ending up with different parameter sets.

STRUCTURE EVOLUTION

Combined with structure evolution, the aim of parameter adaptation is not to find the optimal parameter set for each structure. As described in Keesing and Stokes (1991), a too exhaustive parameter adaptation reduces the performance of the complete optimization process. Thus it is uncritical wether we coose a constant parameter step size or simulated annealing or whatever clever strategy. Besides the mixture of parameter and structure steps, good structure mutation operators are crucial for structure optimization. While a rarely changed structure benefits from the so far reached parameter adaptation, a structure that is changed too much in a single mutation looses all these benefits. Other points to keep in mind in structure optimization are the population sizes and the selection scheme.

the evaluation of the structure optimization is subject of current work. Our first results show a positive tendency. Starting from randomly selected four phase operation, the algorithm reduces either conflicts between traffic movement or reduces the number of phases.

CONCLUSIONS

The adaption of signal operation systems to special intersections and traffic situations is a high dimensional optimization problem that consists of combinatorial as well as continuous valued problems.A homogeneous approach to such problems are evolutionary algorithms. In this paper we presented a decomposition of the optimization process for using different signal operation systems and traffic models as well as a variety of optimization tools. Since the problem encoding and the optimization operators determine mainly the optimization results, they have to be choosen very carefully.

In our first experiments, our approach shows very promising results. The parameter optimization process seems to be rather robust against different operators and initialization and reliably finds good solutions. Structure optimization tends to reduce complexity in reducing the number of phases. In future experiments, we would like to investigate whether this effect is caused by a maybe too small number of parameter adaptation steps or if it mirrors properties of the fitness function. The modell will be extended to traffic responsive signal operations and to coupled intersections.

References

Bücher, T. (2000). Ein videobasiertes System zur Verkehrsdatenerfassung an Lichtsignalanlagen. Technical report, Institut für Neuroinformatik, Ruhr-Universität Bochum.

Bergener, T., C. Bruckhoff, and C. Igel (1999). Evolutionary parameter optimization for visual obstacle detection. In *Proceedings of the first International Symposium on Advanced Concepts for Intelligent Vision Systems, pp. 99-104*. The International Institute for Advanced Studies in Systems Research and Cybernetics.

Hüsken, M., C. Goerick, and A. Vogel (2000). Fast adaptation of the solution of differential equations to changing constraints. In *Proceedings of the Second International ICSC Symposium on Neural Computation (NC'2000)*. ICSC Academic Press.

Keesing, R. and D. G. Stokes (1991). Evolution and learning in neural networks: The number and distribution of learning trails affect the rate of evolution. In *Neural Information Processing Systems 3 (NIPS'1991)*.

Rechenberg, I. (1994). *Evolutionsstrategie '94*. Stuttgart: Friedrich Frommann Holzboog Verlag.

Ringel, R. (1995). Schaffung eines Tools zur Modellierung und Simulation von Prozessen des Straßenverkehrs. Master's thesis, Technische Universität Dresden.

Schnabel, W. and D. Lohse (1997). *Grundlagen der Straßenverkehrstechnik und der Verkehrsplanung*, Volume 1. Verlag für das Bauwesen.

Sendhoff, B. (1998). *Evolution of Structures*. Ph. D. thesis, Ruhr-Universität Bochum.