

# MINIMIZING THE TRAFFIC IMPACT CAUSED BY INFRASTRUCTURE MAINTENANCE USING ANT COLONY OPTIMIZATION

Katharina Lukas<sup>1\*</sup> and André Borrmann<sup>1</sup>

<sup>1</sup> *Computational Modeling and Simulation Group, Technische Universität München, Munich, Germany*

\* *Corresponding author ([lukas@bv.tum.de](mailto:lukas@bv.tum.de))*

**ABSTRACT:** Maintenance measures at infrastructural buildings such as bridges or tunnels in urban areas always have influence on the traffic flow. Finding a schedule that keeps this impact as low as possible by considering the mutual interaction between several road closures is a highly complex optimization problem. In a current research project we have developed an optimization tool to solve this problem based on ant colony optimization.

Over a number of iterations teams of ants construct maintenance schedules for the next few years based on information about the quality of the schedules found in the last iteration steps as well as general information about the condition of the different bridges. The quality of the found schedules is evaluated in the external traffic simulator VISUM. For each year of the schedule a disturbed road network (with reduced capacity at the roads containing bridges under maintenance) is created from the original network and evaluated. As traffic demand and its temporal distribution stay the same for the disturbed network as for the undisturbed, the total net number of vehicle hours during rush hour is a good measurement of the impact of the maintenance activities on traffic. The optimization goal is to minimize this number for the worst year in the schedule.

**Keywords:** *Infrastructure Maintenance, Scheduling, Optimization, Ant Colony Optimization*

## 1. INTRODUCTION

Traditionally the planning of maintenance measures in a large stock of infrastructural buildings is done manually. The building manager has to consider numerous constraints. So some of them can, if at all, only be considered marginally.

For example, the steadying of the budget flow, which will reduce the administrative effort a great deal, can not be considered in this approach. Also the impact on traffic flow that occurs due to capacity reductions on the roads at construction sites can only be considered in a very simplified way. The benefit of postponing one maintenance site to avoid the simultaneous closure of two streets can also not be evaluated.

In this paper we introduce a method to find maintenance schedules for the next few years that are able to fulfil the given constraints at the same time minimizing the impact on traffic flow.

## 2. DEFINITION OF THE PROBLEM

One major task in finding an optimal maintenance schedule is to minimize the impact on the traffic flow. This can be measured by comparing the undisturbed

road net to the net with (partially) closed links, and computing the time spent inside the net by all vehicles in both cases.

Apart from this the schedule is subject to several constraints:

First of all there is a security constraint: all bridges have to be maintained before they reach a state where security can no longer be guaranteed, i.e. the bridge is likely to collapse. This constraint is obvious as safety concerns are the reason why maintenance is done after all.

Besides this there are a number of additional constraints, though, some of them set by the public, some by the building manager himself, e.g. those resulting from monetary considerations.

So the public has an interest that a free flow of traffic is guaranteed all the time, i.e. a possible detour is not blocked by setting up a parallel work zone. Also the blocking of arterial roads and urban freeways should be minimized by doing maintenance on all buildings belonging to one of them at the same time as much as possible.

The building manager has an interest in minimizing the costs of the single work zones. This can be realized by using synergies with third parties also in need of setting up work zones, e.g. public transport firms having to maintain their streetcar net.

Another constraint set by the building manager may be to keep the amount of money spent each year as constant as possible.

In addition to these named objectives and constraints there may be several others depending on the individual customs and needs of the building manager. As not all of them can be modelled in the optimization problem, the optimization tool should provide a number of possible solutions for the manager to choose from.

### **3. APPROACH**

#### **3.1 ANT COLONY OPTIMIZATION**

As there is no known algorithm for finding an exact solution for the described problem in reasonable time it has to be solved approximately, e.g. by meta-heuristics. We use ant colony optimization (ACO) to construct the schedules, as ACO has performed very well with similar problems (e.g. [1]).

ACO is a constructive meta-heuristic developed by Dorigo [2]. Its idea is inspired by the ability of natural ants to find the shortest way between their nest and a food source. Ants deposit while walking a chemical stuff called pheromone on their way and orient themselves on traces of pheromone. This positive feedback leads over time to almost all ants using the same path (cf. [3]; [4]).

The basic principle of ACO is that each ant in a colony constructs a possible solution for the problem. These solutions are then evaluated with respect to the objective function. According to the achieved quality, pheromone is deposited on all elements belonging to this solution. The next generation of ants will now navigate using the amount of pheromone on the different elements to construct new solutions.

The first problem on which ACO was applied was the Traveling Salesman Problem (TSP, [2]). Since then several dialects of ACO have been developed and applied to many other theoretical problems, for

instance, the bi-criteria TSP [5], the Vehicle Routing Problem [6], [7],[8] and the Quadratic Assignment Problem [9]. Later it was also used in real world problems, as Timetabling Problems [10], Construction Site Planning [11], Irrigation Problems [12] and Scheduling Problems [13], [14]; [15], [16]. For an overview see [3].

#### **3.2 OBJECTIVE FUNCTION**

For an objective we use the minimization of the impact on traffic flow. To evaluate this we couple our optimization algorithm with the traffic simulator VISUM by PTV [17].

VISUM is a mesoscopic traffic simulator. The traffic demand is modelled from demographic data regarding the different behaviours of different population groups by choosing their destinations and the time of the day for their travel, In contrast to this microscopic approach the traffic itself is modelled macroscopically as a flow thus enabling one to compute a scenario in a few seconds.

For each year of each schedule a disturbed traffic network is created by reducing the capacity of those streets that contain a bridge under maintenance for this year.

As the undisturbed net is the same for all scenarios, only those networks with reduced capacities have to be evaluated. For each scenario the total vehicle time inside the network during rush hour is computed. The highest value for this indicator in all investigated years is used as the quality measure for the schedule.

The optimization goal is to minimize the total vehicle time for the schedule's worst year while satisfying the safety and budget constraints.

#### **3.3 PROBLEM MODEL**

In order to apply ACO, the investigated problem must be modelled as a way-finding problem. We map the maintenance scheduling problem to a way-finding problem by introducing a directed layered graph, where each layer corresponds to a certain year and each node in a layer represents one of the bridges (see Fig. 1). The nodes of a layer are fully connected to the following layer. An ant constructs a schedule by finding a route through the graph – a visited graph node then

corresponds to the maintenance of the respective bridge in the respective year.

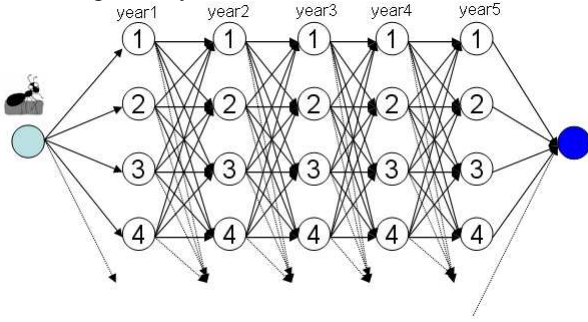


Fig. 1 Problem for schedule for five years modeled as a graph

This can be visualized by a matrix, where the columns are the considered years and the rows are all bridges under examination as shown in Fig. 2.



Fig. 2 Problem for 10 bridges and a schedule for five years modeled as a matrix

When constructing a schedule an ant will move through this matrix from left to right, for each year choosing one of the bridges.

Following Lee [1] we do not use single ants but teams of ants to model the parallel maintenance of several bridges. All the ants in one team walk in parallel through the matrix while each of them chooses its own bridge. Ants from the same team are not allowed to choose the same bridge in the same year.

### 3.4 ELITIST ANT SYSTEM

We have implemented two different dialects of ACO to test their performance in solving the maintenance scheduling problem, Elitist Ant System (EAS) and Rank Based Ant System ( $AS_{rank}$ ).

EAS was developed by Dorigo in 1992 [2] (compare also [3]) to overcome the problem, that classical Ant System (AS) does not always converge to the best solution, that was found during the run.

The basic idea of AS and EAS applied to our problem is, that the artificial ants probabilistically construct the schedule, year after year. The probability of an ant standing in year  $i-1$  choosing the bridge  $j$  in the year  $i$  is

$$P_{ij} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_l \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}} \quad (1),$$

where  $\tau_{ij}$  is the amount of pheromone deposited on bridge  $j$  for the year  $i$ ,  $\eta_{ij}$  is some heuristic information concerning this bridge and  $\alpha$  and  $\beta$  are parameters for tuning the respective influence of  $\tau$  and  $\eta$ .

In traditional AS  $\eta_{ij}$  contains information on the benefit of the choice with respect to the objective function. But as in our problem the benefit of the choice of one single bridge can not be estimated, we make different use of the heuristic information: We set the value of  $\eta_{ij}$  the higher the worse the condition of bridge  $j$  in year  $i$ . Thus we can guide our algorithm to find solutions that are more likely to fulfil the security constraint.

As we use ant teams and not single ants as in traditional AS, we have to introduce the additional rule that ants of the same team are not allowed to choose the same bridge in the same year. In all other matters the ant teams follow the same rules as single ants.

After all ant teams are finished with constructing a schedule, the quality of the found solutions is evaluated using the traffic simulator as described in Section 3.2.

Then the pheromone is updated as follows:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \sum_k \Delta\tau_{ij}^k + e\tau_{ij}^{EA} \quad (2).$$

The first term of equation (2) is to allow the algorithm to forget about bad solutions. The amount of pheromone on all bridges for all years is decreased for a factor  $\rho$  with  $0 < \rho \leq 1$  [3].

Then for all ant teams  $k$  the bridges that belong to their schedules the amount of pheromone is increased by  $\Delta\tau_{ij}^k$ . This value is computed from the quality of the solution found by team  $k$ . Following [3] we take  $\Delta\tau_{ij}^k$  as the inverse of the total vehicle travel time in the worst year of the schedule  $k$ .

The last term of (2) makes the AS an EAS. The best solution that has been found so far by the algorithm is stored in an additional ant (team), the so called Elitist Ant. This ant deposits additional pheromone, even if it is not part of the solutions found in the current generation, following the rules described above. To give the Elitist Ant even more influence, the amount of pheromone deposited by it is multiplied by a factor  $e$ . By not allowing the Elitist Ant to be an infeasible solution, we further guide the algorithm away from infeasible regions, i.e. schedules where either the safety or the budget constraint is not fulfilled.

### 3.5 RANK BASED ANT SYSTEM

The  $AS_{rank}$  is another dialect of ACO. It was developed by Bullnheimer et al. [18].

The construction of the schedules is done analogously to the EAS. The difference lies in the pheromone deposition: In  $AS_{rank}$  only the  $w$  minus 1 best ants of each generation and the Elitist Ant are allowed to deposit pheromone. The amount of pheromone deposited by ant  $k$  here also does not depend on the absolute quality of its schedule, but also on the rank  $r$  that ant  $k$  has in this generation. Thus the pheromone update follows the equation:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \sum_{r=1}^{w-1} (w - r)\Delta\tau_{ij}^r + w\Delta\tau_{ij}^{EA} \quad (3).$$

As in EAS we allow no infeasible solution to become the Elitist Ant to avoid settling the algorithm on an infeasible local optimum as final solution.

## 4. EXPERIMENTAL RESULTS

Both, EAS and  $AS_{rank}$ , have been tested on the traffic network shown in Fig. 3.

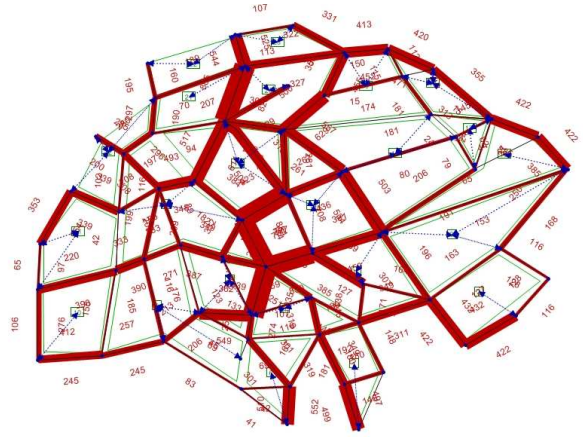


Fig. 3 Traffic network for test runs

This network consists of 100 streets, each of them is assumed to contain a bridge whose maintenance is due in the next 15 years. 10 bridges can be maintained each year. Maintenance reduces the capacity of a street by 50%. The schedules shall be designed for the next 5 years.

Fig. 4 shows the results of each 10 test runs for EAS and  $AS_{rank}$  respectively with  $\rho = 0.4, 0.5$  and  $0.6$ . The impact on traffic is measured in vehicle seconds. All runs were performed with 20 ant teams and  $\alpha=1$  and  $\beta=2$ .

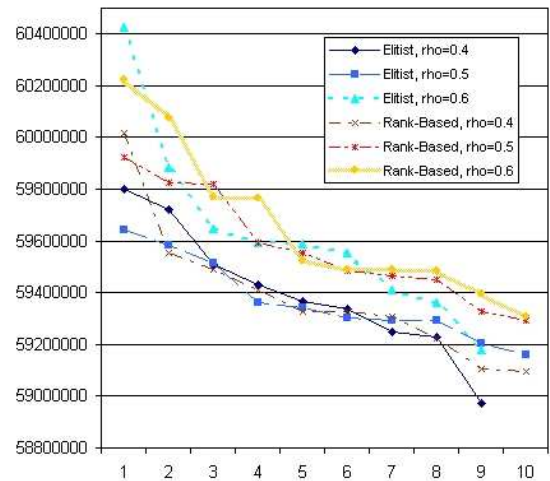


Fig. 4 Results of 10 runs at a time with different settings

It can be seen that the results of EAS and  $AS_{rank}$  are much alike. Both are able to find good results, but in some runs they get stuck in local optima, that are far from good (over 6,000,000 vehicle seconds in contrast to good solutions of about 5,900,000 vehicle seconds).

That is a difference of about 30h). This tendency can be reduced by choosing a low evaporation factor  $\rho$ .

Nevertheless it is shown that both EAS and AS<sub>rank</sub> are able to handle this highly complex, non-linear problem and can find good solutions to it. The tendency of getting stuck to far from optimal local optima can be further reduced by choosing different parameter settings, e.g. by increasing the number of ant teams or choosing different values for  $\alpha$  and  $\beta$ .

## 5. CONCLUSIONS AND FUTURE WORKS

Different dialects of ACO can be used to solve the highly complex problem of finding good schedules for the maintenance of bridges in urban networks under consideration of traffic flow.

First test runs show promising results, though further parameter studies are necessary to guarantee that the algorithm will always find good solutions or even the global optimum.

Up till now, only the impact on the traffic as an optimization goal and security and budget constraints have been considered in the algorithm. Future implementations will also try to consider additional optimization goals, e.g. trying to utilize as many third party synergies as possible, thus making the problem a multi-objective optimization problem.

## ACKNOWLEDGEMENTS

The presented project is funded by the German Federal Ministry of Economics and Technology.

## REFERENCES

[1] Lee, H.-Y., "Optimizing schedule for improving the traffic impact of work zone on roads", *Automation in Construction*, Vol. 18, pp. 1034-1044, 2009.

[2] Dorigo, M., *Optimization, learning and natural algorithms*, Politecnico di Milano, 1992.

[3] Dorigo, M. & Stützle, T., *Ant Colony Optimization*, The MIT Press, 2004.

[4] Boysen, N., *Ameisenalgorithmen*, <http://www.ameisenalgorithmus.de/downloads/ameisenalgorithmen.pdf>

[5] García-Martínez, C., Cordon, O. & Herrera, F., "An empirical analysis of Multiple Objective Ant Colony

Optimization Algorithms for the Bi-criteria TSP", *Proceedings of the 4<sup>th</sup> International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 61-72, 2004.

[6] Gambardella, L. M., Taillard, É. & Agazzi, G., "MACS\_VRPTW: A multiple colony system for vehicle routing problems with time windows", *New Ideas in Optimization*, pp. 63-76.

[7] Barán, B. & Schaerer, M., "A multiobjective ant colony system for vehicle routing problem with time windows", *Proceedings of the 21<sup>st</sup> IASTED International Conference Applied Informatics*, pp. 97-102, 2003.

[8] Reimann, M., Doerner, K. & Hartl, R. F., "D-Ants: Savings Based Ants divide and conquer the vehicle routing problem", *Computers & Operations Research*, Vol. 31, pp. 563-591, 2004.

[9] Stützle, T & Dorigo, M., "ACO algorithms for the Quadratic Assignment Problem", *New Ideas in Optimization*, pp. 33-50, 1999.

[10] Socha, K., Knowles, J. & Samples, M., "A MAX-MIN Ant System for the University Course Timetabling Problem", *Proceedings of the 3<sup>rd</sup> International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 1-13, 2002.

[11] Ning, X., Lam, K.-C. & Lam, M. C.-K., "Dynamic construction site layout planning using max-min ant system", *Automation in Construction*, Vol. 19, pp. 55-65, 2010.

[12] Mariano, C. E., Morales, E., "A multiple objective ant-q algorithm for the design of water distribution irrigation networks." *First International Workshop on Ant Colony Optimization ANTS'98*, 1998.

[13] Gagné, C., Price, W. L. & Gravel, M., *Scheduling a single machine with sequence dependent setup time using ant colony optimization*, Université Laval, 2001

[14] Merkle, D., Middendorf, M. & Schmeck, H., "Ant Colony Optimization for Resource Constraint Project Scheduling", *IEEE Trans Evolut. Comput.* 6, Vol. 4, pp.333-346, 2002.

[15] Solimanpur, M., Vrat, P. & Shankar, R., "An ant algorithm for the single row layout problem in flexible manufacturing systems", *Computers and Operations Research*, Vol. 32, pp. 583-598, 2005.

[16] Christodoulou, S., “Scheduling resource-constrained projects with ant colony optimization artificial agents”, *Journal of Computing in Civil Engineering*, Vol. 24, pp. 45-55, 2010.

[17] PTV AG, *VISUM 11.0 – User Manual*, 2009.

[18] Bullnheimer, B., Hartl, R. F., Strauss, C., „A new rank-based version of the Ant System: A computational study“, *Central European Journal for Operations Research Economics*, Vol. 7, pp. 25-38, 1999.