Intelligent Information Systems VIII
Proceedings of the Workshop held in
Ustroń, Poland, June 14-18, 1999

Timetabling of the city tram service using a genetic algorithm

Halina Kwaśnicka, Bogusław Molecki

Department of Computer Science, Wroclaw University of Technology

E-mail: kwasnicka@ci.pwr.wroc.pl, molecki@alice.ci.pwr.wroc.pl

Abstract

The paper presents designing a timetable for the city tram service using a genetic algorithm (**GA**). Definition of the task in the terms of **GA**, i.e., representation of individuals, the coding schema and the evaluation function is described. A case studies of two real tram services – in Wrocław and Poznań are presented.

Key words: genetic algorithm, optimisation, city tram service

1. The tram timetable problem

In such cities as Wrocław and Poznań the tram timetable for all tramlines is based on an assumption of the constant time between succeeding runs. It means that trams run with the same cycle in all lines. Assuring the well-fitting service with the demand in all passages causes problems. Usually it is done by differentiation passenger cars coursing in particular lines, or by doubling the interval between runs. Often the cycle is stated as 6, 7.5, 10, 12 or 15 minutes, these values are suitable because passengers can easily remember the departure time. Such organisation of tram service causes that we can analyse only the first shift for each line – arrivals and departures of trams in particular points in the city are calculated by shifting time in relation to the first shift. Let us explain it using the simple example shown in Figure 1. Trams in both lines run with the same cycle (e.g., equal to 20 minutes). The departure time for *line_1* from the initial tram stop s_0 is equal to *full hour + shift_1*, *full hour + shift_1+cycle*, *full hour + shift_1+ 2·cycle*, and so on, where *full hour* means each full hour, e.g., 7^{00}, 8^{00}, 9^{00}, ..., *shift_1* is a time (in minutes) from the range of 0 to cycle. The departure time from the first stop s_1 is equal to: *full hour* + *shift_1*+t_1, *full hour* + *shift_1*+*cycle*+t_1, *full hour* + *shift_1*+ 2·*cycle*+t_1, ..., where t_1 is a time of tram travel between stops s_0 and s_1. Similarly, the departure time for *line_2* from the first tram stop s'_0 is equal to *full hour* + *shift_2*, *full hour* + *shift_2*+*cycle*, *full hour* + *shift_2*+ 2·*cycle*, and so on.

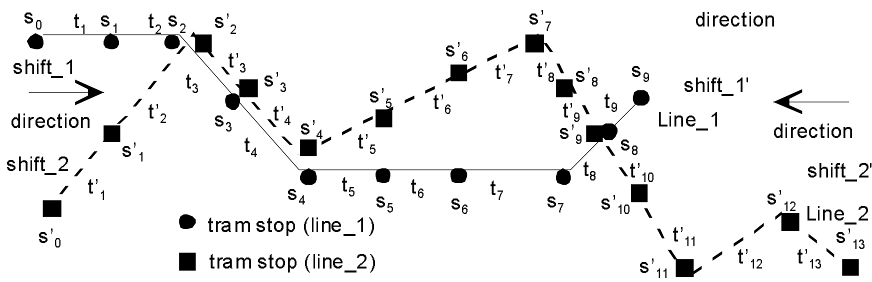

Figure 1. The two lines tram service

Generaly, on the l^{th} tram stop (s_l) of i^{th} line, the departure time can be calculated as:

$$departure\ \ time = \ full\ hour + shift\_i + k \cdot cycle + \sum_{j=1}^{l} t_j, \ \ k = 0, 1, 2, \ldots .$$

Usually each section of tram networks is used by a number of tramlines. The optimisation task is to find *shift_i* for each tram line i = 1, 2, ..., n (n – a number of tram lines in the city) in such a way that during a single cycle the tram traffic is uniform in all tram stops in the city. If the cycle for the two lines in Figure 1 is equal to 20 minutes, the interval between two trams on the double stops (s_2-s'_2, s_3-s'_3, s_4-s'_4, s_8-s'_9) should be equal to 10 minutes. It is not comfortable for the passengers if one tram goes 3 minutes after other one and during the next 17 minutes no tram arrives to this stop. We should take into consideration the two features significant from the passengers' point of view:

1. Long sections being the route between the suburbs and the city centre are more important than sections in the centre.
2. In morning the directions from the suburbs to the centre, factories, and universities are more important, but in afternoon the opposite directions are significant.

Developed method should fulfil the next three requirements to be useful for the real problems:

1. Possibilities of developing a timetable only for a part of tramlines (for example, a number of lines have to be modified because of repairs).
2. Assuring free from the possibility of collision traffic in single-track sections.
3. Generation of a timetable for different cycles accordingly to the time of day.

Usually only a few single-track sections are in each city, they can be easy considered. We can state the shifts between departures of trams coursing on each single-track as known constant values and exclude these sections from the optimisation task. The problem arises if the tram service contains large number of such single-track tramline sections, as it is for example in Silesian Industrial District, where single-track lines share is 44% of all lines.

The problem of developing timetable only for a part of tramlines we can overcome by forcing the shifts for the rest of lines and excluding them from the optimisation task. The third problem – different timetable for particular daytime can be seen as separate task for our method.

Concluding, our task is: to find all *shift_i*, $i = 1, 2, ..., n$, assuring that for each tram stop, all intervals between two successively trams departures (arriving) are similar ($\Delta t_1 \approx \Delta t_2 \approx \Delta t_3 \approx \Delta t_4$ in Figure 2).

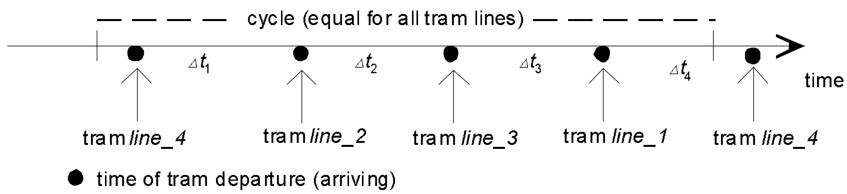

Figure 2. One cycle in an exemplary tram stop with four lines

2. Genetic algorithms

Genetic Algorithms (**GAs**) are the technique based on biological analogy [1], [2]. They imitate the biological evolution according to the Neo-Darwinian paradigm. It considers, that the four statistic processes operating within populations and species can explain history of life: reproduction, mutation, competition and selection (Figure 3). *Reproduction* is the necessary process for species to survive, but potential abilities of species to reproduce are so large, that the size of a population would exponentially increase if all individuals could reproduce with success. *Mutation* guarantees the positive entropy of a biological system. Because of the limited environment of an evolving population, there is a *competition* between individuals. The outcome of *selection* is elimination of some individuals due to competition: only a part of individuals can survive and have offsprings. Phenotype diversity is a consequence of

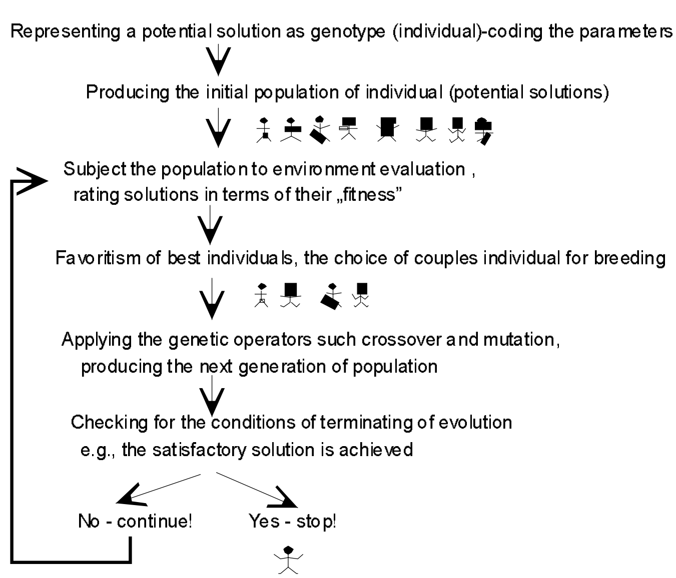

Figure 3. A general schema of a GA performing

recombination and errors in genome transcription. The selection operates on diversified phenotypes.

GAs use a vocabulary borrowed from the genetics. In some environment and at time t there exists a population of individuals of the same species. Some elements are better and therefore they survive and give offsprings in greater quantity. Some of the worse individuals will also survive if they are lucky. In consequence this process gives a mixture of genetic material of individuals (some worse individuals breed with better ones and new genotypes rise by *crossover* process). Some of the genetic material is changed by *mutation* process – it is a random change of a gene. The resulting “babies” population will be, on average, better then the “parents” population. Parents individuals are selected for reproduction according to their *fitness* values – the best individuals are preferred. Fitness depends directly on a phenotype of individual, but – if the individual is represented also at a genotype level – indirectly depends on its genotype. Phenes of individuals represent the parameters of optimized function (fitness). In some problems, the parameters of fitness function have to be constrained. In such cases, the common attempt is to add some value of a penalty to the fitness function of individual if his parameters do not fulfill the constraints.

3. A genetic algorithm as a method of tram timetable design

We propose to use a genetic algorithm for optimisation of timetable for tram service. The chromosome of individual (genotype) contains two genes for each tramline without single-track sections (the first shift after full hour in initial stops in both directions e.g., *shift_1* and *shift_1'* for *line_1*, and shift_*2'*, *shift_2'* for line_2 in Figure 1), and one gene for the lines containing single-track sections (only shift for one direction, for the second direction shift is calculated to assure free from collision traffic).

In our implementation we assumed fictitious cycle equal to 256 – it is comfortable from the computation point of view. Each gene is an integer value from the range of [0, 255]. It requires pre-processing input data – duration of driving from the initial to considered nodes dt_i (for each line and each node) must be recalculated as follow: $new\_dt_i = dt_i \cdot 256/cycle$. An evolving population consists of 1024 individuals. An initial population is created randomly. Individuals are selected for reproduction using elitist method: in each generation 50% better individuals survive without any change, worse individuals are replaced by applying crossover on the better ones - the one point crossover is applied. Crossed over offsprings are mutated: during mutation one random selected gene can be changed, maximal change is equal to ± 12,5 %. The genotypes are normalised – genes are decreased about value of 0^{th} gene because the differences between genes are significant

All chromosomes from the population are decoded into phenotypes. Phenotypes represent times of trams arriving into observed nodes (tram stops) during one cycle. These times are calculated taking into account initial shift *shift_i* for each tramline proceeding by the considered node. The partial fitness value is calculated on the bases of uniformity of trams arriving into the considered node. A number of tramlines in particular nodes is different, let us assume that through i^{th} node pass N_i tramlines. The time of tram arriving into i^{th} node is calculated and it is equal to a_k^i, $k = 1, 2, ..., N_i$, a_k^i are sorted increasingly.
Partial fitness value for j^{th} individual in the i^{th} node is calculated as follow:

$$p\_fitness_i = \sum_{j=1}^{N_i} \left(\frac{256}{N_i} - \left| \frac{256}{N_i} - (a_{j+1} - a_j) \right| \right),$$

where $a_{Ni+1} = a_1 + 256$.

A global fitness function of individual is calculated by weighting sum of all partial fitness values (from all nodes). The weights depend on the importance of the nodes (e.g., nodes on long sections in the suburbs are more important than in the city centre).

As an effect of many tests, we assume that our genetic algorithm acts in three stages: the first 30 generations evolution goes without mutation, during the next 30 generations probability of mutation $p_m = 0{,}01$ and during last 210 generations $p_m = 0{,}1$.

4. Obtained results

We have used the proposed approach for two cities: Wrocław and Poznań. Recently, in Wrocław there exist 23 tramlines with three cycles for different day times: 12, 15 and 20 minutes [3]. In Poznań 15 tramlines coursing with cycle 10, 15, and 20 minutes [4].

For Wrocław a chromosome consists of 45 genes (23 lines, but one – number 17 – has single-track section, and one – number 0 – is circular). Fitness function consists of 44 partial fitness values: 26 for 13 important nodes (we consider both directions) and 18 for supplementary nodes – short sections in the centre city. The weights for the important nodes are equal to 2, for the additionals are equal to 1. Obtained results are satisfactory. The most important is to assure the uniform frequency of trams in the important nodes. Only for one of them, node *Pilczyce* – which is far from the city centre – distribution of trams arriving is far from being uniform. For all other nodes frequency of tram arriving is good or satisfying. We have modified a weight for the node *Pilczyce* from 2 to 4, and it allows to obtain a good result for all important nodes. For some of the supplementary nodes the results are a little worse, but still satisfactory.

Similar experiments have been made for Poznań, for working days with cycle 10 minutes, Saturdays with cycle 15 minutes and Sundays with cycle 20 minutes [4]. A chromosome consists of 30 genes (15 tramlines) for working days and 28 genes for holidays (tram D does not go in holidays). Fitness function consists of 40 partial fitness values (18 for important nodes and 22 for the rest nodes). The weights are stated to 3 for the two important nodes (*Piątkowska* and *Kórnicka* street), to 2 for the rest of important nodes and to 1 for the additional nodes. For all cycles obtained results are good or satisfactory. Only for some of the supplementary nodes (*Junikowo*, *Ogrody*, *Połabska*) the results are not good enough. Analysing the map of tram service in Poznań one can observe that probably it is the result of decreasing a number of lines in Poznań few years ago, and increasing the length of lines. It resulted in decreasing the degree of freedom of the optimised system and it is difficult or impossible to obtain uniform distribution of trams arriving to all nodes.

Summarising obtained results, we can conclude that our experiments proof the applicability of presented methodology. Assumed fitness function seems to be suitable for that problem. Recently, the experimentation with of optimal choice of weights to match expectations of city tram service users is undertaken.

References

[1] Goldberg D.E., (1995), *Algorytmy genetyczne i ich zastosowania*, WNT, Warszawa.
[2] Michalewicz Z. (1996) *Algorytmy genetyczne + struktury danych = programy ewolucyjne*, WNT , Warszawa.,
[3] The map and timetable of tram service in Wrocław, Municipal Traffic Service (MPK), Wrocław, 1999.
[4] The map and timetable of tram service in Poznań, Municipal Traffic Service (MPK), Poznań, 1999.