UDC 004.8

Andrii Kobyliuk, Artem Volokyta

METHOD OF SCHEDULING BASED ON ARTIFICIAL INTELLIGENCE

The article examines the method of planning tasks in a computer system using artificial intelligence approaches. A problem-modified genetic algorithm was used for its implementation.

Keywords: scheduling, artificial intelligence, genetic algorithm.

Fig.: 7. Tabl.: 1.

Relevance of the research topic. Because heuristic planning algorithms are specialized for a certain choice of inputs, the task of choosing the universal and most efficient one is a non-trivial one, always sacrificing in some cases to gain in others. If there are many algorithms, the choice can be made using prediction. New methods based on the field of artificial intelligence can work effectively without prior knowledge of the problem space and can be used to solve the scheduling problem.

Target setting. A genetic algorithm can be used to optimize a scheduling problem in hopes of further improving the overall performance and efficiency of a computer system.

Actual scientific researches and issues analysis. One of the first attempts was made [1] by Lawrence Davis using genetic algorithms (GA) to solve the scheduling problem. Davis noted the effectiveness of using the method stochastic search where a genetic algorithm operated on a given list which was subsequently used to form the actual planning schedule. Later, the same idea was developed by Philip Husband, who isolated [2] all the most modern, at that time, genetic planning algorithms. He noted similarity between the planning problem and other problems from the class of such problems problems such as the traveling salesman's problem, the problem of layout and packing in a backpack etc.

Uninvestigated parts of general matters defining. Despite the large number of works devoted to the application of genetic algorithms for planning, the problem of using new algorithms for this purpose remains understudied. The task of optimizing the planning problem primarily concerns the NP-class of algorithms, so a deterministic algorithm cannot be created to solve ill-posed problems.

The research objective. The purpose of this work is to research methods and models of planning, which would be based on methods of artificial intelligence, development of a software product that would implement the data model in practice.

The statement of basic materials. Steps below explain the basic flow of the algorithm, and shown in Figure 1:

1. Randomly generate initial population of individuals (first generation).

2. Evaluate the fitness of each individual according to the objective function or goals; distance and time taken

3. The following steps are repeated until termination criterion is met:

i. Reproduce the best individuals.

ii. Best individuals are put through crossover and mutation phase

iii. Birth of new offspring/individuals. iv. Evaluate the new individuals.

v. Replace least-fit individuals.

4. The termination occurs when either:

i. The goals are achieved, or

ii. The growth/improvement of the generation stops (iteration limit).

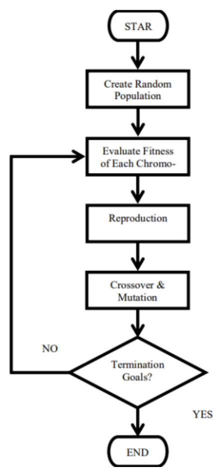

Fig. 1 - GA steps

A total of 100 chromosomes were generated randomly before further improved using genetic operators in GA. Table 1 shows the example of genes for each chromosome. The C(n) is the chromosome number for the population which was set to 100 (n=100). While, the T(i) is the task number. In this study, there are k processors utilized to complete the overall tasks.

Table 1. Chromosome structure

Chromosome	T1	T2	...	Tm
C1	P1	P2	P1	P1
C2	P2	P4	Pk-1	P1
...				
Cn	P1	Pk	P2	P3

The fitness value of every single chromosome is calculated to evaluate its quality towards the objective function. The fitness value is measured as in unit of time. The evaluation process was conducted by using the following equation:

$$F(i) = maxFT - FT(i) + \frac{1}{(maxFT - minFT + 1)}$$

where minFT and maxFT are scalar quantities denoted respectively the maximum and minimum completion time among all chromosomes in the data population. $FT(i)$ is the termination time for the ith chromosome.

The entire population of chromosomes is then sorted and the best 30 chromosomes are sorted out from the population. The selection is inspired from the natural selection in evolution. In which the best chromosome with the best fitness value has the higher probability of surviving compared to those whom have less. These selections were meant towards for a better solution.

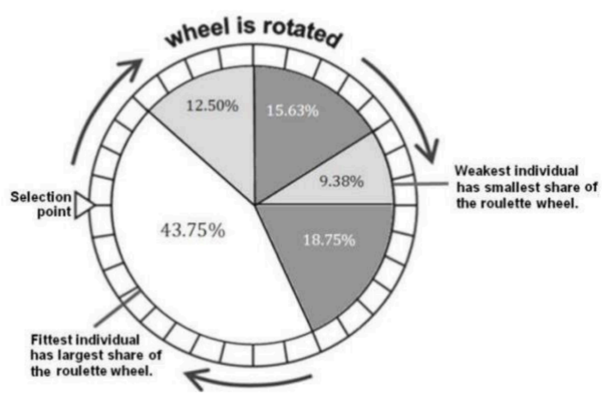

Fig. 2 - Process of selection

Each chromosome is given the proportional size of the sector, which corresponds to its value of the fitness function in the given population. The larger the value of the fitness function, the larger the areas of the sectors will occupy chromosomes and, accordingly, they are the ones with a greater fate the probabilities will be carried over to the next generation and will be selected.

The crossover is something much similar to the act of sexual reproduction of the living thing. In attempt of pushing for a better solution, the selected chromosomes are then combined and creating a new chromosome or a child. Meanwhile, the new chromosome would have both the characteristics of the parents. The idea of a crossover is both to collect and merge both qualities of the parents and increase the chances of producing a better offspring.

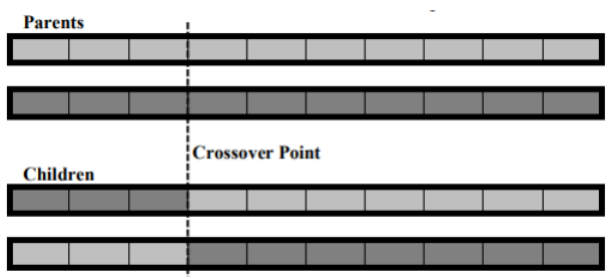

Fig. 3 - Process of crossover

Mutation is one of the genetic operations, which consists in a change one or more values of a gene in a chromosome in relation to it previous state (Fig. 4)

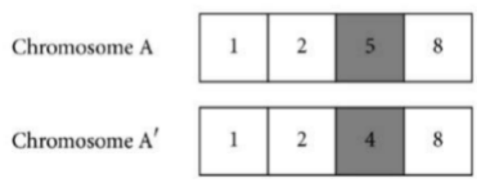

Fig. 4 - Process of mutation

After preliminary tests conducted in the beginning of this study, the best probability rate was found to be at 0.1 (10%).

After the process of crossover and mutation, the new chromosomes (offspring) will be determined for its fitness value. This population of offspring will then be sorted based on the fitness value. A new population is then created; which involves 30 child chromosomes, 30 parent chromosomes and 40 randomly generated chromosomes as shown in Figure 5. This new population will undergo the same processes as mentioned earlier until the predetermined stopping criterion is exhausted.

New population

C1	Child chromosomes
…	
C30	
C31	Parent chromosomes
…	
C60	
C61	Randomly Generated chromosomes
…	
C100	

Fig. 5 - Process of mutation

The figure 6 shows the user interface of the developed scheduler program. Through the interface, you can edit and set graphs of tasks, computer systems, and also generate data for testing

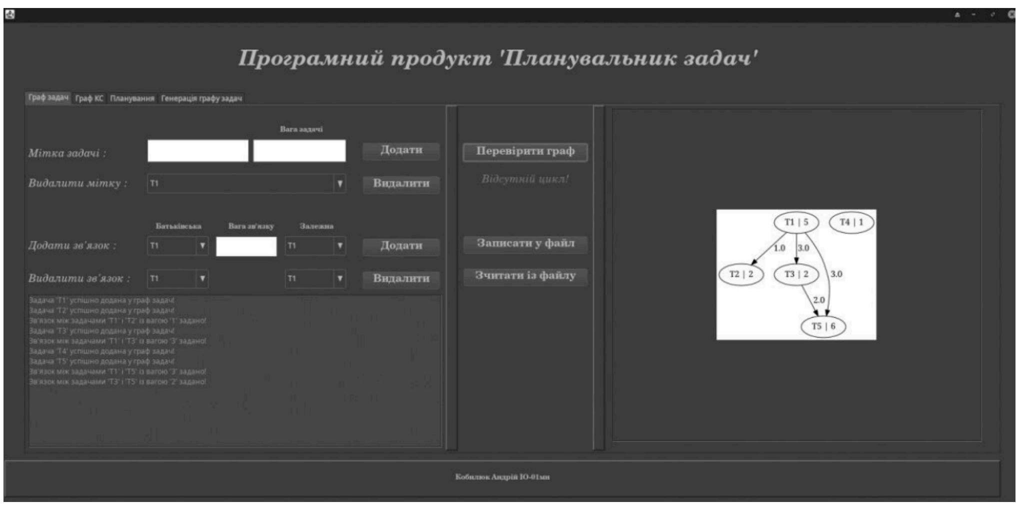

Fig. 6 - Developed GUI

As can be seen from fig. 7 significant acceleration is developed the scheduling method (orange color) does not give with a small amount vertices in the problem graph. When increasing them to 12, efficiency can be noted and the advantage of the method for the average connectivity of the problem graph. The ordinate axis is the value of the acceleration coefficient, and the abscissa axis is the value of the connectivity coefficient. If you look at the graphs below, you can see that with a small number of vertices, all methods of scheduling give practical results the same result within the margin of error in terms of acceleration factor.

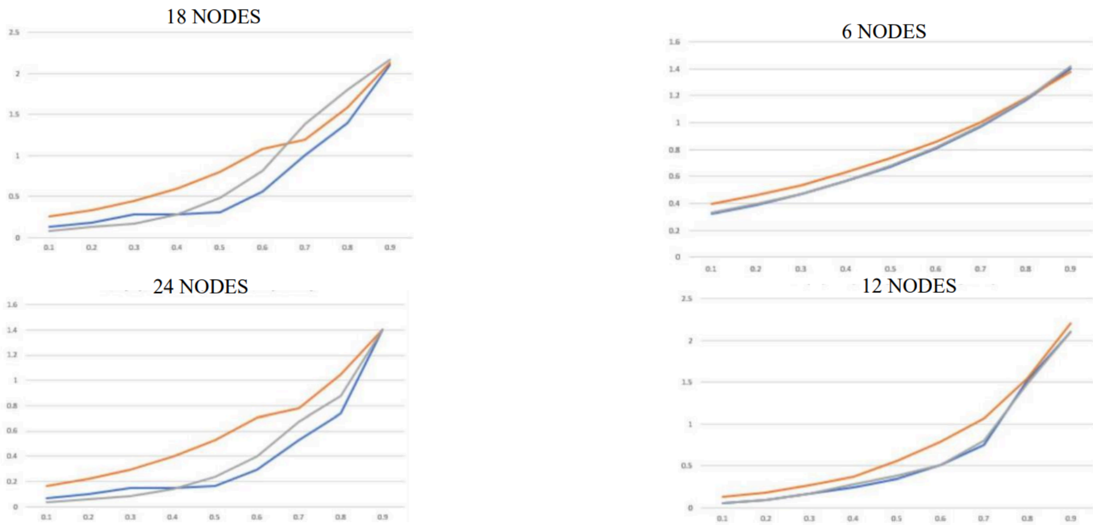

Fig. 7 - Testing of method

When the number of vertices increases, the genetic one clearly stands out planning algorithm. Taking into account the connectivity of problem graphs, it is possible note that the developed method works better with medium connectivity and degrades in efficiency to conventional methods, respectively, at low and high connectivity.

Conclusion. Theoretically, the genetic algorithm has the ability to significantly improve the overall performance of processors by fully utilizing their power. In addition, GA offers a solution for parallel execution of tasks by processors. Scheduling on a multi-core processor has several problems such as task allocation, CPU idle time, and task sequencing, but all these problems can be solved by GA. The results obtained as a result of this study clearly support the theories regarding the ability of the used method to solve the planning problem. The developed program was able to reproduce the characteristics and qualities of GA. Furthermore, the produced GA is capable of simulating and completing the specified PPJ graph. As a conclusion, the general experiment was successful. The obtained results are reliable and significant.

References

1. Davis L. Job Shop Scheduling with Genetic Algorithms / Lawrence Davis. // Proceedings of the 1st International Conference on Genetic Algorithms. – 1985. – C. 136–140
2. Hasband P. The Natural Way to Evolve Hardware / P. Hasband, I. Harvey, T. Adrian. – 1996.
3. Debanjan K., Siddhartha B. A Multi-Objective Quantum-Inspired Genetic Algorithm (Mo-QIGA) for Real-Time Tasks Scheduling in Multiprocessor Environment. Procedia Computer Science. 2018. Vol. 131. P. 591–599
4. Artificial Intelligence-based Scheduling. // The International Conference on Security, Fault Tolerance, Intelligence. – 2020. – C. 5.
5. Студентська робота [Електронний ресурс]. – 2022. – Режим доступу до ресурсу: https://ela.kpi.ua/handle/123456789/34324

Authorus

Kobyliuk Andrii – student, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.
E-mail: andrii.kobyliuk@gmail.com

Кобилюк Андрій Григорович – студент, кафедра обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».

Artem Volokyta – associate professor, PHd, Department of Computer Engineering, National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”.

Волокита Артем Миколайович – доцент, кандидат технічних наук, кафедра обчислювальної техніки, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського».