Bachelor Thesis Review

Faculty of Mathematics and Physics, Charles University

Thesis author	Krystýna Pantůčková	
Thesis title	Optimization of a circulating multi-car elevator system	
Year submitted	2018	
Study program	Computer Science	
Study branch	General Computer Science	
Review author	Filip Matzner	Reviewer
Department	Katedra softwaru a výuky informatiky	

Overall

	good	OK	poor	insufficient
Assignment difficulty		X		
Assignment fulfilled		X		
Total size *. . . text and code, overall workload*		X		

The goal of this thesis is to simulate and optimize the control of a special type of elevator system called circulating multi-car elevator (CMCE). It should be noted that CMCE is a cutting edge technology introduced in 2013 by Hitachi. The thesis text is written in a very good English and consists of 75 pages of text divided into 6 chapters (including user and developer documentation). The source code is represented by approximately 10'000 lines of C#.

The first chapter describes the CMCE principles. The next chapter introduces two possible scheduling algorithms - trivial (i.e., greedy) strategy and genetic algorithms. Afterwards, the author presents and compares the results of the aforementioned algorithms in a simulated environment. Follows a review of state-of-the-art methods for elevator control and the last two chapters contain user and developer documentations.

The implemented simulator provides a realistic environment including details such as minimal distance between elevator cars, floor height, car acceleration etc. The values for those parameters are based on officialy published parameters from various elevator manufacturers adding validity to the environment. The simulated scenario considers even morning peak, lunch time and similar real-life patterns. Furthermore, the whole simulation can be configured and visualized in a user-friendly GUI application. I encourage the author to publish the source code (e.g., on GitHub), so that other researchers can extend her work.

The genetic algorithm and its comparison to the trivial greedy algorithm has a lot of room for improvement. The genetic algorithm optimizes the schedule off-line (i.e., it knows all the passengers at once, future and past, and searches for the best assignment of the cars to the passengers). By the words of the author, such an approach cannot be used in practice but it can be used to evaluate whether some on-line scheduling algorithm can be further improved. In the case of this thesis, the evaluated on-line algorithm is the most trivial greedy algorithm, so it is quite clear that it will perform significantly worse than the schedule found by the genetic algorithm. In my opinion, a very interesting future work would be to consider the whole task to be a reinforcement learning problem. In other words, to optimize a control policy (e.g., represented by a neural network or a decision tree) that could be used as an on-line scheduling algorithm in practice.

One more note to the genetic algorithm. Replacing the worst individuals by random individuals does not seem to be a useful operation. The random individuals cannot compete (hopefully) with other individuals in the population and they should always loose in the tournament selection.

The thesis successfully fulfilled all the set out goals and I recommend it for defense.

Thesis Text

	good	OK	poor	insufficient
Form *. . . language, typography, references*	X			
Structure *. . . context, goals, analysis, design, evaluation, level of detail*	X			

Problem analysis		X		
Developer documentation	X			
User Documentation	X			
The language and typography are on very high level. The whole text is smooth and polished.				

Thesis Code

	good	OK	poor	insufficient
Design *. . . architecture, algorithms, data structures, used technologies*	X			
Implementation *. . . naming conventions, formatting, comments, testing*	X			
Stability	X			
The source code is thoroughly documented in a machine-readable format. The code structure is well designed and split into classes with small and well defined responsibilities. Big plus is the separation of the simulation library and the graphical interface, this could come in useful when running the simulation (without GUI) on Linux-based clusters. It would be nice to have a diagram of the implemented classes and their relations in the developer documentation.				

Overall grade Excellent/Very Good

Award level thesis No

Date

Signature