Simulation of a New Lift System

A new lift system for high buildings is tested by a simulation. The distinguishing feature of this system is that there is more than one cabin in a shaft. The cabins are driven by small electric motors and do not use any traction ropes. When the cabins are at rest in order to let people in or out and to wait for the next transport task, they are in the station outside of the shaft.

1 Introduction

Lifts are a very important invention, especially for transporting passengers in buildings. Many people can not live without them. Lifts are here to save physical energy and time. Who does not know the situation in front of a lift? You press the button and wait for the sound of the arriving lift. Nothing happens. You are waiting and you get angry. Time is a scarce resource and you do not want to waste it.

Another lift problem is typical for skyscrapers: Lifts in skyscrapers use too much space, because so many shafts are necessary. This space could be used for offices.

These problems are well known and ideas for solving are mentioned in [1]: „Elevator systems are the most important transportation method for high-rise buildings. However, in the conventional elevator, only one elevator car, suspended by ropes, occupies the whole elevator shaft. Because of this, the taller the building is, the lower the performance of the elevator becomes. With the background of progress in linear-motor technology and increasing needs for high-performance transportation systems for large-scale buildings, rope less multicar elevators (MCE) that have several cars driven by linear motors in a single elevator shaft, are attracting attention as a novel transportation system. However, considering the control of MCE, we can only apply the knowledge of existing elevator systems to MCE to a limited extent, and there is a need for new controller design methods“.

2 Aim

The aim of this work is to test a new lift system with direct travelling and multiple cabins in one

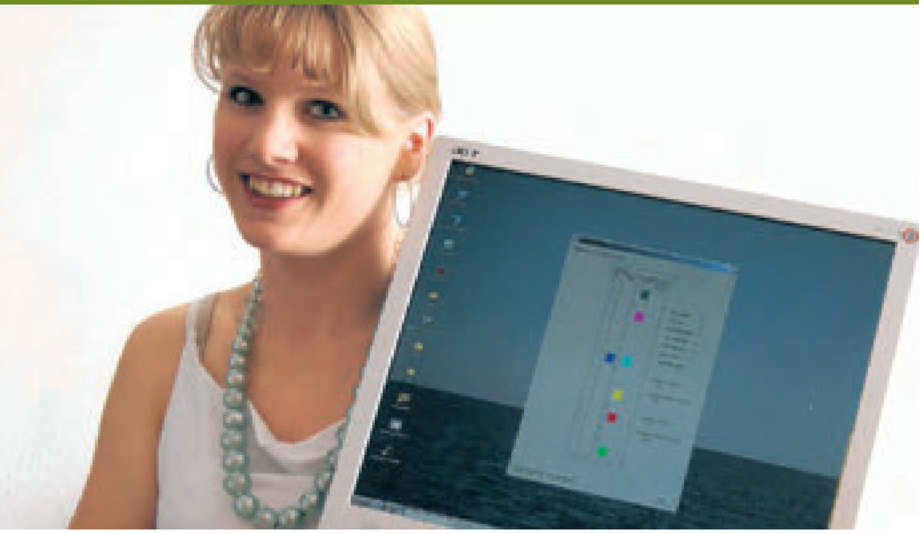
Autorin

Sara Pfenninger, *1988
Meilen

Schule:
Mathematisch-Naturwissenschaftliches Gymnasium Rämibühl, Zürich

Eingang der Arbeit:
September 2007

Zur Veröffentlichung angenommen:
Oktober 2007



shaft. The test is not done in reality, of course, but in the way of a simulation. The simulation offers the possibility to vary the different parameters. The system has to be optimised in the way that the waiting period is as short as possible. In addition, there are a lot of other questions: Is direct travelling enough powerful or are there too many cabins needed? Do the cabins block each other when circulating in the same shaft? Are there many parallel running shafts necessary? The simulation can answer these questions. The use of the software and the optimisation is beyond this work.

Grade of Service	Percentage of calls answered in		Time to answer calls	
	30 s	60 s	50 %	90 %
Excellent	>75	>98	<20	<45
Good	>70	>95	<22,5	<50
Fair	>65	>92	<25	<55
Poor/unacceptable	<65	<92	>25	>55

Table 1: Office Building Average System Response Time Performance

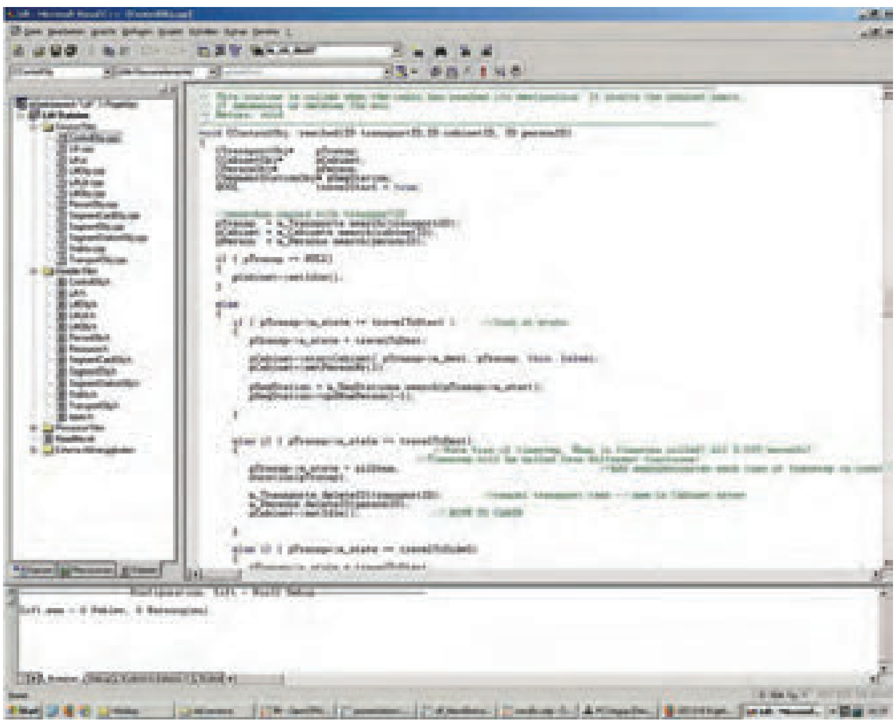


Figure 1: Program in Visual C++

3 The programming language and the simulation

In the past, high level languages, such as BASIC or FORTRAN were used to program a simulation. Nowadays the object oriented languages, such as C++, are the likely choice according to [1].

The programming language I use is Visual C++. It is based on C. The C language has steadily increased in popularity since it was created in the early 1970s. Since 1990, C++ is one of the most popular commercial languages, because it has a good performance. The software development system Visual C++ is a widely used standard package (see figure 1).

It is very difficult to test the new lift system with technical calculations and in [1] simulations are recommended: „Where the traffic design requirements are more complicated, computer simulations may need to be carried out“. For this reason, I decided to simulate the lift system. In addition, simulations are used to prove specific cases and not general designs. A simulation is just an approximation to the real conditions. It is the ideal and does not correspond to real cases in every way.

4 Lift systems

What is a good or an excellent lift system? How can you evaluate the grade of service of a lift system? To answer these questions, the table 1 helps us to ascertain whether the output, that is to say the time interval, is too long or not.

„Table 1 indicates the percentage and time values for several grades of service over one hour of peak activity in an office building. An hour of peak activity is taken in order to obtain sensible and realisable results. It should be possible to obtain the grades of service indicated in the table during the worst hour of activity ... [1]”.

4.1 Description of the new lift system

The distinguishing feature of this system is that there is more than one cabin in one shaft. The cabin is driven by small electric motors attached on it. Therefore there is no use of traction ropes in the shaft. The cabins can travel vertically up and down in the same shaft. When the cabins are at rest in order to let people in or out and to wait for the next transport task, they are outside the shaft in the station (see figure 2). The shaft is then free for the other cabins to travel. There is a mechanical device which makes it possible, as this mechanical device allows the cabins to

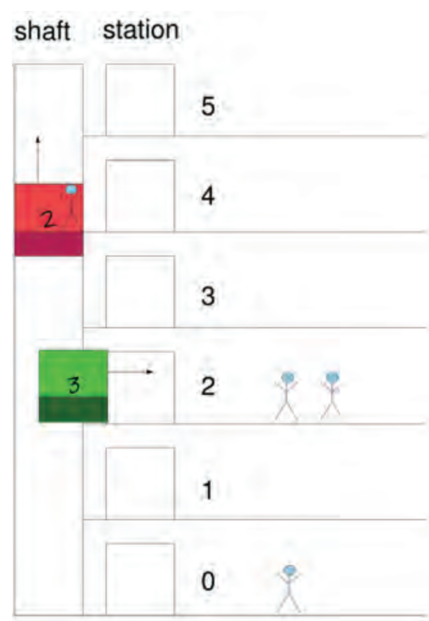


Figure 2: Overview of the new lift system

move horizontally from the shaft to the station. There, the doors of the cabin open and the passengers can exit. Now the cabin is idle and waits till it is asked to fetch other people. In figure 4 you can see the lift cabins as well as the stations. There are small cabins for a limited number of passengers who travel directly, without stopping from the start to the destination as mentioned above.

This work does not focus on the mechanical design. However, with the basic mechanical features mentioned above, there are no traction ropes for example; it is possible to realise the lift system.

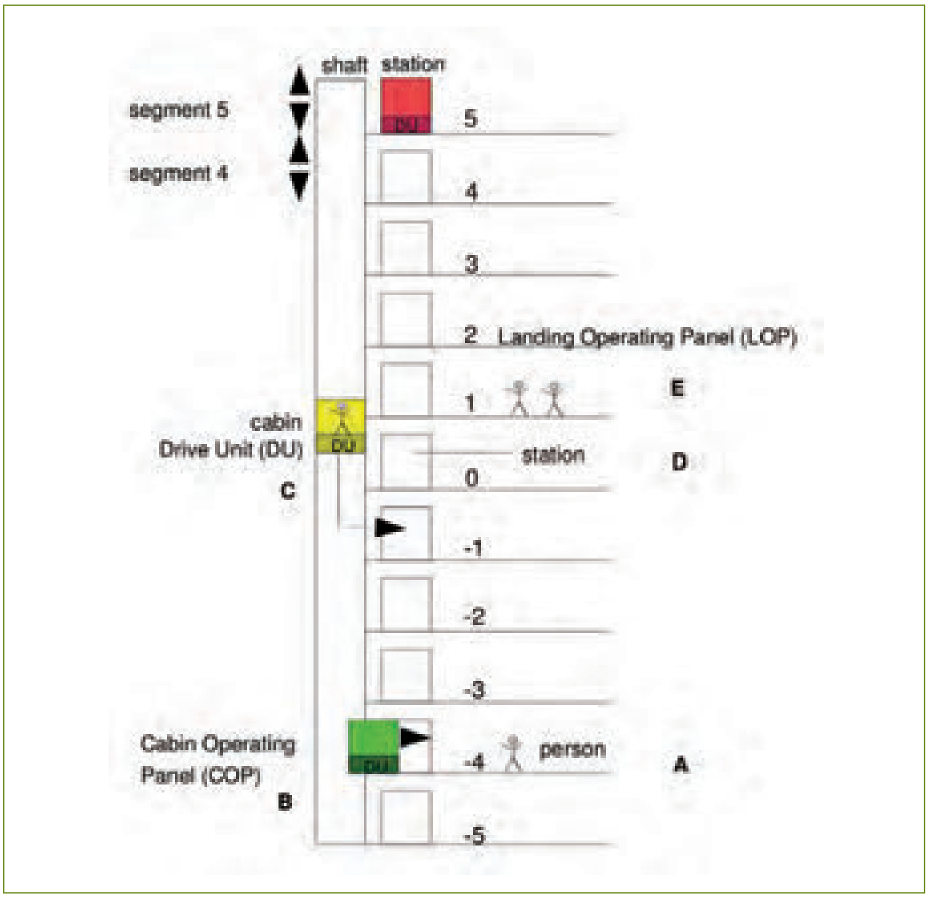


Figure 4: Overview of the objects

4.2 Other lift systems

The idea of several cabins in one common shaft was realised in the paternoster system, also known as the cyclic elevator. At one side, several cars were moving upwards without stopping, at the other side, they were travelling downwards (see figure 3). Passengers had to enter and leave it while the cabin was moving, which was quite dangerous. This was the reason why they were phased out.


Figure 3: Paternoster lift system (Source: www.joostdevree.nl Dec. 2006)

5 The idea of the simulation

The idea of the system is comparable to a motorway. The normal shaft would be the motorway. Cabins, which want to move in the normal shaft, are like cars. The computer control is like the car driver who wants to drive on the motorway and has to check that the way leading to its destination is free.

The new lift simulation program is also comparable with a taxi service. Each person must be treated as an individual. There is only one person per lift cabin in the simulation. Therefore the cabins represent different taxis. The computer control would then be the central taxi service station. The simulation software is written with the intent to vary the number of cabins, passengers and floors. The programming language Visual C++ allows a structured programming where it is possible to vary the number of objects, e. g. the number of shafts. In the simulation there is only one shaft with cabins in it. The cabins can travel in both directions. No special events such as fire alarms or defective cabin will happen.

The simulation is divided into different parts, called classes. It is very important that any simulator's user can change the programming code easily in the basic simulation to what I focused on. By using different classes, the whole simulation gets much easier. Then, it is determined that you have an overview as there is a structure in it and every user is able to see the different classes very clearly. Otherwise I would have a very complicated and long program at the end, whereas my aim is to have one program, which is as simple as possible.

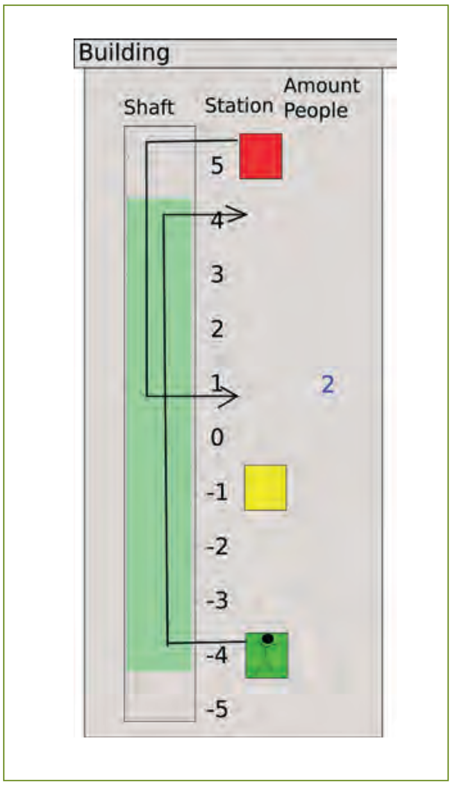


Figure 5: Avoidance of collision

For the simulation, the following assumptions are made:

- the traffic profile is ideal
- all floors are equally populated
- various lost times, such as passenger disturbance etc., are negligible.

6 The Program

6.1 Objects

In figure 4 the new lift system is shown with some of the objects. There are different objects which interact with each other. However, I will not go into further details about objects and classes.

```
if (isFree)
{
  for (ii = start; ii <= dest; ii++)
    {
      pSegment = search(ii);
      pSegment->m_state =occupied
    }
}
```

Figure 6: Simplified code for reservation

The following summary is referring to figure 4 illustrating the sequence of the different actions. It is very simplified as it should provide just an overview of the program.

Event A: A passenger at the floor -4 wants to go to the fourth floor. He presses the floor button with the arrow pointing upwards. Now the following happens: The Landing Operating Panel, called LOP, which interacts with the elevator Group Control, sends a cabin to floor -4 as fast as possible so that the passenger can enter the cabin. The Group Control focuses on the avoidance of collisions between the cabins and therefore different segments have to be reserved as they are shown in figure 5. There, the green cabin has reserved all segments which are coloured green. The red cabin has to wait until the green cabin gives the segments free. In figure 6 one can see the simplified code for the segment reservation, that is to say the reservation of the shaft. The state of each segment object is set occupied by the green cabin (m_state = occupied).

In event B in figure 4, you can see the Cabin Operating Panel, called COP. The passenger can press in the cabin the button corresponding to the floor the passenger wants to go. In this case the passenger presses button 4. Then the Cabin Control is responsible for the reservation of the segments as well as the station segment and then the cabin can move. In event C, you can see the yellow cabin travelling with a passenger in it to floor -1. In event D, a station is drawn. The passengers have to wait until the cabin is in the station and then they can leave the cabin.

In event E, you can see two passengers, waiting for a cabin. As soon as possible there will be one in the first floor. However, only one passenger can enter the cabin and the other passenger has to wait for the next one.

6.2 The Control

The control is actually the object which calls almost every function, controls and supervises everything. It is the „main office". The aim of the control is to look for the passengers and satisfy their needs with regard to the safety issue. Such a design is the best one in the view of the information technology, because it is easier to make corrections just in one class, namely the control, than to search the mistake in all different objects.

Figure 7 shows different steps of the program which are controlled by the control-object. The yellow cabin wants to move to the third floor. So the green cabin is sent away and the yellow can move to its destination.

6.3 Simulation

It is easy to add different functions to the program. By dividing the whole processing into classes, changes can be made locally and often do not affect other classes. The control does not have to do everything by itself but it gives different messages to the different classes and the classes have to execute these tasks. They are in a way smart but controlled by the control. Using this design structure makes it easier to

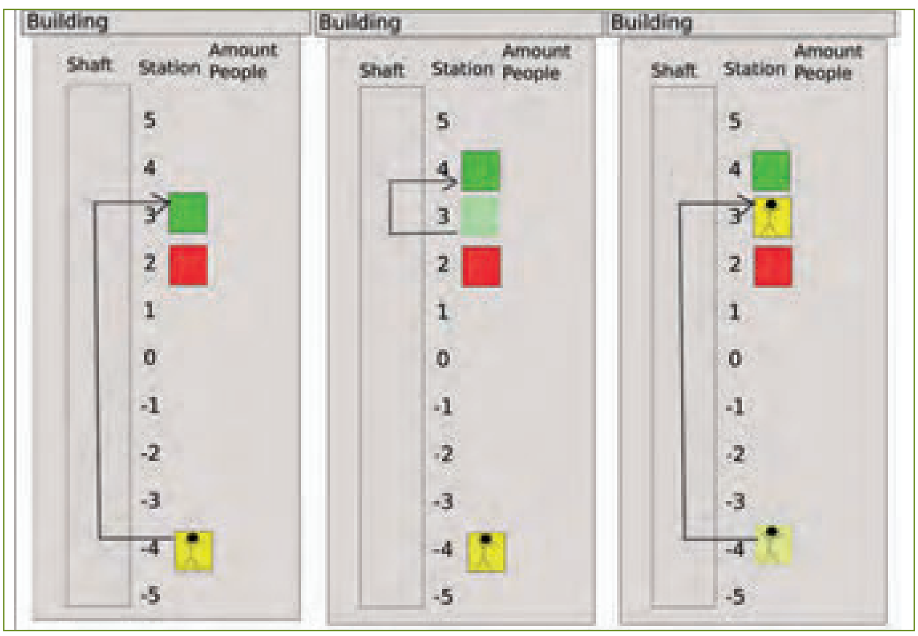


Figure 7: The control-object manages the traffic: The yellow cabin wants to move to the third floor. So the green cabin is sent away and the yellow cabin can move to its destination.

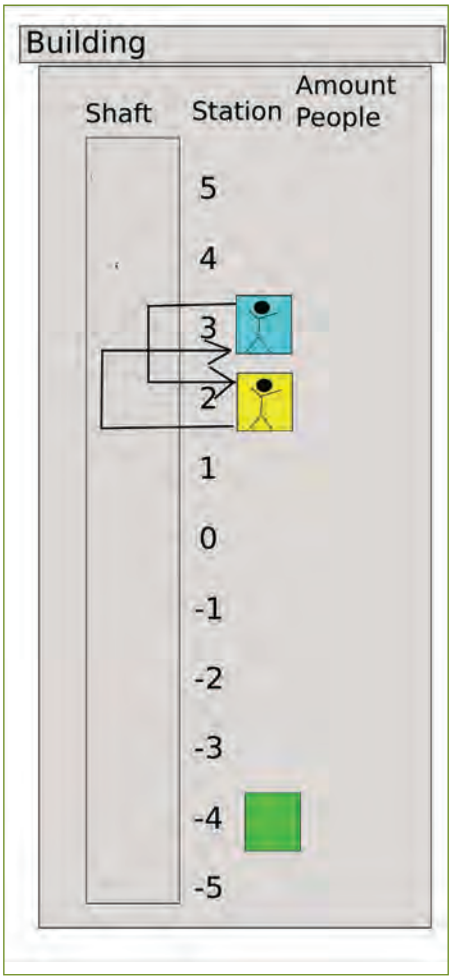


Figure 8: Situation of deadlock

change the code of the program as it is mentioned above.

6.4 The deadlock problem

A very interesting but difficult problem, which is almost completely solved now, was the deadlock one. A possible situation is shown in figure 8. The blue cabin wants to travel from the third to the second floor and the yellow cabin from the second to the third floor. They block each other which results to a deadlock of these two cabins. The solution is that the program knows whether there is a deadlock of two or more cabins. As soon as there is one it must solve it. The yellow cabin can not move the blue cabin away because the latter is not idle as it is occupied by a person. Therefore one cabin must travel away with a person in it which is not that efficient, but there is no other solution to solve such a deadlock.

6.5 The idea of the timeStep

The timeStep is a very important part of the program. A timeStep is a division of the time into small parts called steps. There are different processes going on at the same moment. A timeStep allows us to add a little time to the first one, then to the second, then to the third one and so on. Like this, a process which actually happens parallel to another one, will not be completed before the other one. Of course it can not be a one hundred per cent parallel, but they are followed by each other just by a difference of one timeStep, which is very small. It is possible to vary the duration of one timeStep. It can for example have the value 0.001 second, but it would also be possible to give it the value 1 hour, which would not be very smart in this simulation.

The simulation program therefore has a time counter. By adding a time step, it adds the same amount of time to the time counter of the program. Like this it is ensured that time passes like in reality. Processes done by the computer, for example calculating, are infinitely fast. They are not counted. In reality, this happens so fast that nobody can notice it.

The time based method, and not the event based, one ensures that the error of the time is very small, namely equal to the amount of one timeStep.

6.6 The Visualizing part of the program

To control and illustrate the program the whole shaft with its cabins in it is visualized in the graphical user interface, see figure 9. In this way it is easy to follow the way of a cabin. After each timeStep, the whole graphical user interface is updated. That is to say the cabins are redrawn on the new or on the same position.

The lift system in this visualizing part is defined by seven floors above the ground floor and seven below it, so there are 15 floors. There are up to 7 cabins, but it is possible to add a new one with minimal changes of the code of the software.

The amount of the people who are waiting for a cabin to arrive is shown in the column "amount people". You can also see straight away the destination of each cabin as well as the average time the passengers needed to reach their target floor on the right hand side.

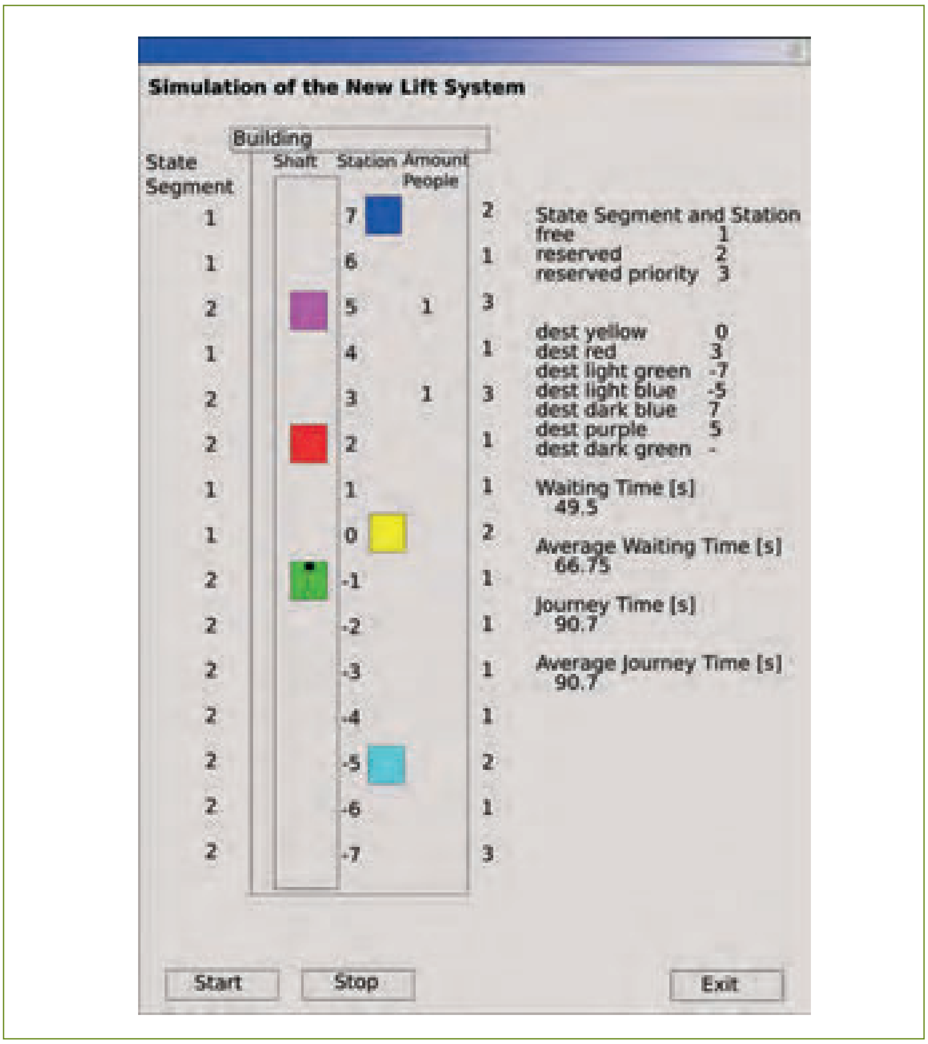


Figure 9: Screenshot of the improved program

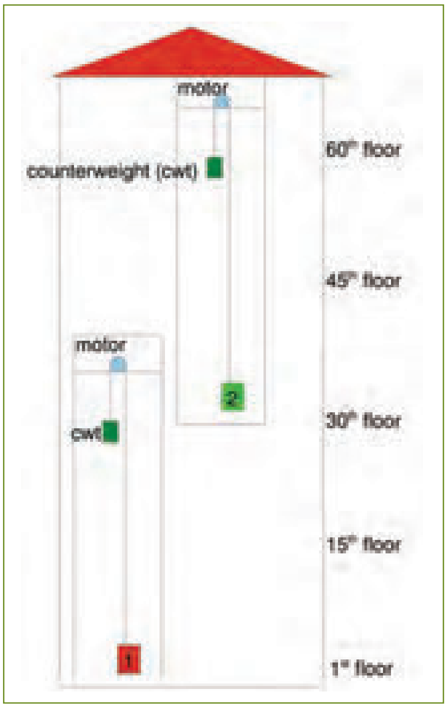


Figure 10: Elasticity of ropes

7. Discussion

7.1 Advantages of the new lift system

The new lift system has the following advantages: People can reach their destination in large buildings without changing the lift shaft as this would be the case in the conventional lift system. Currently, people move with the lift cabin for example to the 30^{th} floor and take another cabin in another shaft to get to the 60^{th} floor. The reason why there is not a direct lift to the 60^{th} floor is that the ropes would be too long and therefore are elastic and heavy. As longer the ropes are as greater is their expansion. Therefore the cabins can not stop very accurately at the floor. The weight of such a long rope is also a problem. So the length of a shaft is limited to about 30 floors. Figure 10 illustrates this. The new lift system eliminates this problem because there are no ropes.

A high comfort is guaranteed if all lifts are placed together. This is in contrast to the situation, if all lifts are distributed around a building. In the new lift system, all cabins are in one or several shafts and are therefore placed together.

A cabin moving with a person in it does not stop at any other floor to let other passengers in. It moves directly to the desired destination of the person. This ensures that the person in it reaches the fastest way its floor as soon as the cabin is moving. If more than one person of the same floor wants to go to the same destination, the amount of people in a cabin can be up to six passengers. This could be the case if it is midday and many people who stay on the same floor want to go to the cafeteria. There are several cabins moving at the same time. This is a very efficient way to handle the passengers.

7.2 Disadvantages of the new lift system

A disadvantage is that a cabin can take at a maximum six people with it, which are not many compared to the conventional lift system. If there is a large group of people all going to the same floor, several cabins are necessary to transport them.

The fact that the new lift system does not stop at any other floor to let other people in and out is an advantage for those who want to travel to a floor of great distance, but a disadvantage for those who would like to enter and travel a short distance with the cabin. Those people have to wait for some time at the station for the cabin to arrive which is reserved for them. It could be that there are a lot of almost empty cabins travelling by until the reserved empty cabin arrives.

This lift design suffers in efficiency versus cabled elevators. It must carry the heavy motor in the lift instead of it being kept stationary. There is no counterbalance; thus the elevator must be capable of lifting the car's weight (including the motor) and passengers' weight. A conventional design only needs to be able to lift the passengers' and the cable's weight. This results in using more power, and requiring a larger motor. This disadvantage has to be tested and studied.

7.3 Possible improvements

There are different aspects which could be improved. The simulation is simplified but it would be possible to add improvements to the program. Firstly, the cabins can only take one passenger at a time instead of six. Secondly, the cabin which is next in the queue is chosen and not the cabin to which the person is next to. The system would be more efficient if this was realised. Thirdly, the program can be used only for a limited time when there are more than three cabins in it. The lift system could probably be improved if there were two shafts, one for moving upwards and one for moving downwards.

7.4 Conclusion

The simulation can be used to weigh the advantages and disadvantages. However, the lift system must be optimised regarding what is possible because it is easy to change the code of the program. Finally, the new lift system represents an alternative which should be taken into account for further development of the elevator system.

Personal comment and acknowledgements

I managed to simulate a basic lift system. I made a lot of interesting experiences in programming in Visual C++. The beginning was quite hard, but now I developed an understanding, how programming in C++ works. It was not just the program itself which took me hours to work on the same function, but also the logical thinking. There are very often many ways which lead to the goal, but there is usually only one good way. I can never say that the program is finished; I always get new ideas of how the program could be improved. I still really enjoy this project and I think I will never get bored by it.

I would like to thank Mr F. Nlabu, who was the supervisor of this project of “Schweizer Jugend forscht”. He works for Schindler and provided me with different information about how lifts work. He mentioned what is good and what could be improved in this project. This motivated me a lot. I wish to thank Mr G. Schäppi, who was the supervisor of the high-school diploma project and was always ready to answer questions. Last but not least my acknowledgements are directed to all proofreaders and supporters who did a great job in helping me.

Bibliography

[1] Barney, Gina. 2003. Elevator traffic handbook. Spon Press. London.

[2] Gamse, Beryl. 1976. A Simulation of An Elevator System For a Moderate Height Building. Institute of Transportation Studies University of California. California.

[3] Markon, Sandor und Kita, Hajime et al. 2006. Control of Traffic Systems in Buildings. Springer-Verlag. London.

[4] Microsoft Corporation. 1988. C for yourself. Microsoft Corporation. Ireland.