ELEVATOR SYSTEMS SIMULATION TECHNIQUES

Keith Jenkins
KJA Consultants Inc.
343 Principale, Suite 1
Saint-Sauveur, Québec, Canada J0R 1R0

ABSTRACT

Computer simulation techniques have in recent years largely replaced manual calculations for determining the performance of systems of elevators in building design. The rapid increase in the ratio of speed to cost for new computer hardware extends the possibilities for more advanced simulation techniques. Time-slice (as opposed to event-slice), diversified and less structured input data, extended input-output parameters, can all be integrated into low cost "user friendly" simulation programs.

1. MANUAL CALCULATIONS

In the period from 1925 to 1932 there was a major increase in high rise office building construction in the centre cores of North American cities. The elevator engineers of that era had to establish a basis for the determination of the elevator system. It was a natural application of the standard theories of probability. The basic problem can be stated as follows: given that there are P passengers embarking on an elevator at the main floor and there are N typical floors, what is the probable number of stops that the elevator will make? This is, in effect, a variation of simple high-school problems in probability. It is equivalent to saying: "If we have N boxes and P apples, and we randomly drop the apples in the boxes, how many boxes will have apples in them?".

The resulting number of probable stops S, given P passengers and N stops above the main floor is given by:

$$S=N-\frac{(N-1)^P}{N^{(P-1)}} \tag{1}$$

Once the number of probable stops S is calculated, it is possible to construct an artificial building consisting of the main floor plus S floors. The distance between each of the S floors will be equal to the travel divided by S. Using this artificial building, the time to travel from one stop to the next can be calculated based on the elevator speed and acceleration chosen. Time is added for the return express trip top to bottom. Door operation time, interlock time, passenger transfer time and dispatching time are calculated and added to the total time. All of this gives the round trip time (RTT). The designer can then derive the number, capacity and speed of the elevators to provide a desirable interval (the RTT divided by the number of elevators) and five minute handling capacity (P*300 divided by the interval).

These methods, although still used by some designers, have been largely supplanted by computer simulation techniques.

The early computer programs (circa 1965-68) duplicated the previous manual calculation methods. They simply converted the arithmetic calculations to computer calculations. Since they were fast and eliminated arithmetic errors they made the elevator system design easier. Often these programs were referred to as simulation programs although, in fact, they did not actually simulate the elevator system. The term "simulation" as used in this discussion will be taken to mean a program that duplicates the operation of an elevator and an elevator system.

2. COMPUTER SIMULATION TECHNIQUES

Until recently, actual simulation programs were implemented by only a few people, principally because of the cost of the software development. In the late sixties and early seventies, computers and computer time were expensive; development costs, since they incorporated costly computer time as well as the programmer's time, could be almost prohibitive. It therefore required a certain dedication, not to say insanity, to produce an effective simulation program.

My personal observation is that the simulation programs developed by myself at that time likely did not generate, even when employed on many projects over several years, enough revenue to offset the development cost. The principal motivation behind the writing of such programs was, of course, not money but curiosity. In this sense almost any simulation program written prior to 1980 could not be justified on a purely financial basis.

There are two basic approaches to simulating elevator systems: the time-slice and the event-slice.

The time-slice method is straightforward; each elevator is run for a discrete period of time (the time-slice) normally 0.01 seconds. This means that the simulation program does a considerable amount of work switching between elevators and for each elevator between its various functions. The net result is that the time-slice method is inherently slow and very much compute-bound. Prior to 1980, computer processing was by comparison to to-day's standards quite slow. This meant that a time-slice based program was almost a laboratory instrument rather than a production tool for analyzing various elevator systems under changing traffic patterns.

The event-slice method is more complicated; each elevator is run from the start of one 'event' until its finish. For example, if an elevator is opening its doors the simulation program allows the elevator to complete this operation before switching to another elevator. As can be seen, this method speeds the process considerably. On the other hand, care has to be taken to keep all of the elevators synchronised in time. This leads to programming complications but does not affect the speed at which the program runs. The event-slice computer simulation program was a practical production tool. Real time could be compressed by a factor of 100 to 1 even on the slower computers available in the seventies. This compression of real time is a sine qua non of any effective simulation technique since it is the only practical way of examining numbers of situations combined with numbers of different input parameters.

The great strength of computer simulation techniques lay in the versatility of the input and output parameters. It was possible to analyse, without any great effort, complex traffic patterns. Hotels with restaurants, swimming pools and multi-activity floors could be examined under varying traffic conditions for various times of the day. The effect on office buildings of major tenants with concomitant interfloor traffic could be determined with some precision. This type of analysis was virtually impossible using manual methods.

Apart from that, the simulation techniques emulated and in some cases copied the real world. Dispatching system strategies could be built into the computer programs and their effectiveness measured. This allowed the designer, if he was sufficiently aware of simulation methods and had access to or could create a simulation program, to optimize the dispatching system prior to committing it to production. Since not many elevator engineers were conversant with computer programming, there was no real attempt made to take advantage of this opportunity.

An elevator simulation program, however well written, tends to be compute-bound. There is a limited amount of input and output and a great deal of churning away of the central processor unit. This is to be expected since the program uses random number generators, has to calculate the indices for the matrices of the various values, uses real numbers as opposed to integers in many instances and does a great deal of switching from one point of the program to another. If certain of the intensively used routines are written in assembler it is possible to reduce the simulation time but no matter what is done the program running time tends to be long. The time problem is exacerbated by the introduction of the double deck simulation.

A compromise must then be made between the need to run the simulation long enough to get statistically valid results and the need to analyze a number of variations of a particular system within a limited amount of production time.

As well, for some users a simulation program is often less aesthetically satisfying than a probability calculation. A probability calculation will always yield the same result, e.g. an interval of 21.65 seconds with a handling capacity of 18.63% of the population. A simulation on the other hand gives slightly varied results each time it is run (assuming that the random number generator is seeded). There is something confidence inspiring about numbers to two or three decimal places. Unfortunately, this confidence is not well founded. In the real world, elevators operate much like the simulation program. If the data are taken on two different days the numbers will vary.

This is one of the prime advantages of the simulation technique: it gives a clearer picture of how clear our picture is of the system as it will work in practice together with an indication of the range of results that might be expected.

Typical inputs to a simulation program are as follows:

- Number of elevators
- Capacity
- Speed
- Number of stops
- Acceleration*
- Rate of change of acceleration*
- Door close time*
- Door open time*
- Traffic selection
    - Manual traffic input
        - Handling capacity
        - Traffic from to
    - Automatic traffic calculation
        - Traffic pattern
            - Up peak
            - Heavier down
            - Balanced peak
            - Heavier up
            - Down peak
        - Gross area per floor
        - Core area per floor
        - Area per person per floor
        - Percentage handling capacity per floor
- Floor height per floor
- Coincident calls*
- Forced load return*
- Headway time*
- Non-random*
- Force to main*
- Cars at main*
- Percent load dispatch*
- Percent load non stop*
- Main floor*
- Cab inside width*
- Cab inside depth*
- Single deck*
- Double deck
- Side opening doors
- Center opening doors*
- Entrance width*
- Car call dwell time*
- Hall call dwell time*
- Preference return*
- No stops for elevator X at floor Y*

The items marked with an asterisk are assigned default values (or are calculated based on other input data) by the program. The other inputs have to be entered by the user.

Typical outputs from a computer simulation program are as follows:

Average system waiting time
Maximum system waiting time
Minimum system waiting time
Average waiting time per floor
Maximum waiting time per floor
Minimum waiting time per floor
Cab 'crowding' factor
Average system time to destination
Average system trap time
Up hall calls per floor
Down hall calls per floor

The program may also output average intervals for anyone who wishes to compare the results to a manual calculation.

Outputs for such things as operating time, flight time, may optionally be given.

The programming for each elevator is identical and the programmer would normally use matrices for the particular elevator data with common coding for all elevators.

The driving force behind the elevator motion is, of course, the registration of hall calls. These hall calls can be registered 'on the fly' or can be pre-registered and recorded by an initial setup routine.

The advantage of the pre-registered calls is that the designer can, if he chooses, completely optimize the dispatching system for a particular sequence of calls. This allows him to determine the maximum theoretical efficiency of the dispatching system. The results obtained by the dispatching system operating without benefit of clairvoyance can be compared to the results of the 'perfect' system — perfect in the sense that it sees the future and functions so as to optimize not just the present but also the hereafter.

Registered hall calls (meaning ones that have been placed prior to or at the current time) are categorized as un-assigned, temporarily assigned and permanently assigned. The programmer may elect not to have un-assigned hall calls; when a hall call is registered he may immediately temporarily assign it to a particular elevator. The permanently assigned hall call results from a decision by the dispatching system that a particular elevator must answer a particular call. A permanently assigned hall call becomes, to the elevator, like a car call. The decision to make a permanent assignment may result from coincident car and hall calls or from the fact that an elevator has been committed to stop at a particular floor, i.e. that it has begun its deceleration into the floor. Normally the algorithm used to commit on the basis of coincident calls would include considerations of waiting time for that hall call compared to the average current waiting time. The temporary assignment is the normal state since at the start of each 'slice' the program will re-evaluate the assignments and re-distribute them in the optimum manner.

The determination of the optimum manner is one that has exercised the minds of dispatch system designers for some sixty years. The most prevalent current approach seems to be to calculate the time to answer the call for each elevator and to assign the call to the elevator with the shortest time.

In fact, simulation results indicate that dispatching tactics have little effect during the critical period of the day — noon time. It is apparent that this would be so since, if the elevator system is reasonably designed, it will be pressed during this period; all elevators will have a series of calls in front of them and assignments are quite straightforward.

The dispatching system even during up-peak traffic has fairly simple decisions to make. Should an elevator be detained at the main floor to allow further loading? Should headway timing be implemented at the main floor and if so, how much? Should the system revert to a split group with elevators serving a fixed selection of the total typical floors? Unless the up-peak traffic is significantly heavier than the noon traffic, an elevator system that is correctly designed for the noon traffic will have no problems handling the up-peak traffic. Since in North America over the last thirty years office building up-peaks have declined sharply, the elevator system designer, once having fixed upon an elevator system on the basis of the anticipated noon traffic, rarely has to do more than verify that the up-peak will be adequately handled.

Dispatching systems display their greatest virtues during light mixed traffic such as might obtain in mid-morning or mid-afternoon in a typical office building. At that time the elevator system is not working to full capacity and the dispatching system has more latitude in call assignments. The way in which the system handles this type of traffic is interesting but it is not the normal design point for the elevator system.

From the point of view of elevator plant design, since the difficult traffic period — normally at noon — is the design point and since the dispatching system is limited in its influence under heavy traffic, the nuances of various dispatching systems are not of signal importance to the designer of the project vertical transportation system.

Simulation programs, as with almost all elevator system analysis, measure the elevator plant's performance over a five minute period. This five minute period has its roots in the history of North American office building traffic patterns. In the early thirties, when the fundamental ideas of traffic analysis were conceived, there were severe peaks when office hours started in the morning and in the late afternoon when office hours ended. These peaks were short — somewhat more than five minutes. As a consequence, the five minute analysis became and remains the standard in the industry. It would probably be more reasonable to use a ten minute period since current traffic peaks are less pronounced and of longer duration. However, the five minute standard has become so widely accepted that it is unlikely to change.

The simulation program places hall calls and car calls for a period of time which is a multiple of the five minute period. This multiple can be set to any value; the larger it is the longer the program runs and the results are averaged to a correspondingly greater extent. A minimum reasonable multiple is 100. Thus the system is, in effect, examined for its performance during this five minute period for the equivalent of 100 days.

The simulation program can be written in any language but, given the popularity of C and Basic, and given the efficient compilers available for them, either of these languages, quite possibly combined with assembler subroutines, would be the logical choice to-day. Compilation is necessary; even on a fast computer interpretive basic is too slow to be of practical value.

3. NEW COMPUTER HARDWARE

Over the past five years there have been dramatic developments in low-cost high-speed computers — and there appears to be no end in sight. The ratio of computer speed to computer cost was over 1000 times greater in 1990 than in 1980.

This proliferation of low cost computing power allows almost any person or company to approach the writing of a simulation program without the fear of incurring massive computer costs. It is true that there is still a substantial investment in time but even this is reduced by the speed of editors, compilers and linkers. Much of the labour and delay associated with early computer programming has been eliminated. A program that in 1975 would take two or three hours to compile and link can now be put together in under a minute. A simulation that in 1975 would take ten minutes to run can now run in under 30 seconds.

Another development that is of interest to anyone running a simulation program is the evolution of low-cost network hardware. This opens up the possibility of using one computer to simulate each elevator in the group with the network transmitting the information between them and between a computer acting as the dispatcher and input/output unit.

4. TIME-SLICE RE-VISITED

To-day's computers are sufficiently fast that the earlier considerations which weighed so heavily in favour of event-slice program techniques are now no longer important.

Now that the major drawback of time-slice methods — slowness — has been eliminated by fast, cheap computing power, it is reasonable to take another look at this approach. This is particulary true if one considers that event-slice is essentially a 'trick' to minimize compute time and if we no longer need to minimize compute-time then we can dispense with this 'trick'.

Time-slice is simple, direct, corresponds more closely to reality and is easier to comprehend. These advantages mean that eventually all computer simulation programs will use this method.

An elevator operates in a number of discrete modes. It is closing its doors, opening its doors, accelerating, decelerating, running at full speed and even, on occasion, doing nothing.

Each of these modes or phases represents a portion of the computer simulation program. The program when it advances one clock 'tick' will examine each elevator in turn and prosecute the current mode for the 'tick' period of time. The time slice ('tick') may cause the particular elevator to shift from one mode to another. Thus, if it was closing its doors preparatory to starting, and the 'tick' caused this operation to be terminated then the elevator will shift to the next mode, normally acceleration.

Using fast time-slice simulation a number of extensions are possible to the earlier computer programs. First the input can be interactive in the sense that the input parameters can be varied in the course of the simulation either manually by operator intervention or automatically on the basis of feed-back from the simulation results. The user would 'fix' certain input parameters, e.g. building data, and allow the others to vary so as to optimize an elevator system for the building data. This requires that certain norms be established either by the program as a default or by the operator as an input. The operator might input the building data together with the requirement that the system have a thirty second average waiting time. The simulation then runs and varies the speed, capacity and number of elevators so as to produce the optimum system for these conditions. Similarly, the operator might fix the number of elevators and the data for the building floors but leave the system to determine the number of floors served and the elevator speed and capacity.

Another component of the extended input range is the customizing of the passenger transfer algorithm. The simulation program normally calculates passenger transfer time based on such data as cab dimensions, number of passengers in the cab, number of passengers entering, number of passengers leaving, entrance width. These algorithms are not constants for every building in every place in the world at all times of year and it is desirable that the user be able to select from a variety of algorithms or modify standard algorithms to suit the particular site conditions.

One of the desirable ancillaries to the simulation program is a good system graphic display. In the past this type of display has always been limited by the exorbitant computer time taken to run the display and the quality of the display given low screen resolutions. Recently, it has become possible to create good fast colour displays that graphically illustrate the position and state of elevators and car and hall calls. Naturally, the operator has to have control of the speed of the display, i.e. the factor by which real time is compressed, and to be able to pause the simulation so as to examine particular situations. A further extension, useful for examining dispatching operation, is the ability to playback or reverse the display. All of this has been made practicable by the advent of fast low-cost hardware.

The interactive program with optimizing feedback leads naturally to a different type of output. Instead of a printed or screen display of the results for one system under one set of conditions it is now possible to display a range of systems centering about the optimal point.

What is perhaps of more interest for the ultimate user — the project owner or developer — is a display showing the performance of the elevators for a full working day as the system is subjected to the anticipated traffic loads and patterns from morning to night. This type of output is a more useful presentation for the layman since it allows him to easily make comparisons.

5. IMAGE AND REALITY

It becomes apparent, in this review of simulation techniques present and past, that the image of the real world created in the simulation program has become remarkably like the real world. It may also be fair to say that the real world has moved closer to the world as seen by the simulation program.

The use of multiple computers (or multiple central processing units), the use of a computer for dispatching, the simulation display, even the 'bugs' that can develop within the simulation program — all of these correspond closely to an actual elevator system. In fact, it is quite possible to interchange program code between the simulation program and the elevator system.

The end result is that the modern elevator simulation program has become a remarkably effective tool for the design of new buildings, the evaluation of the effect of equipment modernization, the 'rating' of elevator systems against typical and ideal systems and the design of dispatching systems.

The increased use of simulations and the increased precision of their results means a more logical and economical application of elevators in building construction.

AUTHOR'S BIOGRAPHICAL DETAILS

Keith Jenkins is a vertical transportation consulting engineer with a number of years of experience. He started in the elevator industry as a bench fitter building machines, cabs and controllers, then subsequently moved on to become an elevator mechanic, field adjuster and engineering director. Later, in 1962, he established his own consulting practice. He has worked on the design of numerous high rise buildings, principally in North America, including such projects as 60 Wall, Times Square, Madison Square, First Canadian Place, World Financial Centre. Apart from his work in the field of elevators he has owned and operated companies in process control, bio-medics, electronics and computer software.

P102224