INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI®
TRUCKING SIMULATION

USING GENETIC ALGORITHMS

QIXIA DENG

A THESIS

IN

THE DEPARTMENT

OF

COMPUTER SCIENCE

PRESENTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE AT
CONCORDIA UNIVERSITY
MONTREAL, QUEBEC, CANADA

MARCH 2003

© QIXIA DENG, 2003
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-77709-X
ABSTRACT

Trucking Simulation Using Genetic Algorithms

Qixia Deng

Genetic Algorithms (GAs) are stochastic search and optimization methods inspired by the mechanisms of natural adaptation. In the last two decades they have been researched and applied in a variety of areas. Currently GAs are used extensively in solving complex optimization problems with large but finite search space. This thesis studies two genetic algorithms applied to a trucking simulation problem where trucks travel among dealers in a country and transport commodities from producers to retailers and from retailers to consumers. Both trucks and retailers attempt to survive and make the most individual profits. Trucks and retailers evolve simultaneously in the simulation. Their evolution progress in two economy types is examined. The results show different effectiveness of these two algorithms in the two economy types.
ACKNOWLEDGEMENTS

I would like to use this opportunity to give grateful thanks to my supervisor, Dr. Peter Grogono, for his support and guidance throughout the project.

I would also like to express thanks to my husband, Weining Zhong, for his support and patience.
Table of Contents

LIST OF FIGURES ........................................................................................................... viii

1 INTRODUCTION ........................................................................................................... 1

2 AN OVERVIEW OF OPTIMIZATION AND GENETIC ALGORITHMS ............ 3
   2.1 Categories of Optimization ............................................................................... 3
   2.2 Minimum Seeking Algorithms .......................................................................... 5
   2.3 Natural Optimization Methods .......................................................................... 6
   2.4 Genetic Algorithms ............................................................................................ 7
      2.4.1 Components of A Genetic Algorithm ......................................................... 8
      2.4.1.1 Encoding .................................................................................................. 8
      2.4.1.2 Basic GA Operators .............................................................................. 8
      2.4.1.3 Convergence and The Termination Conditions ..................................... 10
      2.4.2 GAs vs. Other Traditional Optimization Methods .................................... 10
      2.4.3 Applications of Genetic Algorithms ......................................................... 11

3 THE TRUCKING PROJECT ......................................................................................... 13
   3.1 Elements .............................................................................................................. 13
      3.1.1 The Country .................................................................................................. 13
      3.1.2 Commodities ............................................................................................... 15
      3.1.3 Producers ..................................................................................................... 16
3.1.4 Consumers .................................................................................. 16
3.1.5 Retailers ..................................................................................... 17
3.1.6 Gas stations ................................................................................. 18
3.1.7 Trucks ......................................................................................... 19

3.2 Economy Types and Numerical Analysis .............................................. 20

3.3 The Simulation .................................................................................. 24

4 THE GENETIC ALGORITHMS FOR TRUCKING ......................... 27

4.1 The Genetic Algorithm for Trucks ....................................................... 27
  4.1.1 Encoding .................................................................................. 27
  4.1.2 Fitness ..................................................................................... 36
  4.1.3 Selection .................................................................................. 36
  4.1.4 Crossover ............................................................................... 37
  4.1.5 Mutation ................................................................................ 38

4.2 The Genetic Algorithm for Retailers .................................................... 39
  4.2.1 Encoding ................................................................................ 39
  4.2.2 Fitness ................................................................................... 41
  4.2.3 Selection ................................................................................. 42
  4.2.4 Crossover .............................................................................. 43
  4.2.5 Mutation ................................................................................. 43

5 SIMULATION PARAMETERS ............................................................. 44

6 SIMULATION RESULTS AND DISCUSSION ................................. 50

  6.1 Results of Experiment 1: Supply-driven Economy ............................. 51
LIST OF FIGURES

Figure 3-1 The country........................................................................................................... 14
Figure 3-2 Basic simulation parameters............................................................................... 21
Figure 4-1 Summary of truck’s genes .................................................................................. 35
Figure 4-2 Summary of retailer’s genes .............................................................................. 41
Figure 5-1 Locations of producers and consumers .............................................................. 45
Figure 5-2 Summary of parameters to be inputted................................................................. 49
Figure 6-1 TPT and NT in a supply-driven economy ............................................................. 52
Figure 6-2 APT and NT in a supply-driven economy ............................................................. 53
Figure 6-3 HRT and NT in a supply-driven economy ............................................................. 54
Figure 6-4 NR in a supply-driven economy ....................................................................... 56
Figure 6-5 TPR in a supply-driven economy ...................................................................... 57
Figure 6-6 APR in a supply-driven economy .................................................................... 58
Figure 6-7 HRR in a supply-driven economy .................................................................... 59
Figure 6-8 Genes of the top 10 capital-gaining trucks in a supply-driven economy ....... 61
Figure 6-9 Genes of the top 5 capital-gaining retailers in a supply-driven economy ...... 61
Figure 6-10 TPT and NT in a demand-driven economy ....................................................... 62
Figure 6-11 APT and NT in a demand-driven economy ....................................................... 63
Figure 6-12 HRT and NT in a demand-driven economy ...................................................... 64
Figure 6-13 NR in a demand-driven economy ................................................................... 65
Figure 6-14 TPR in a demand-driven economy ................................................................ 66
Figure 6-15 APR in a demand-driven economy ............................................................... 66
Figure 6-16 HRR in a demand-driven economy .................................................. 67

Figure 6-17 Genes of the top 10 capital-gaining trucks in a demand-driven economy .... 68

Figure 6-18 Genes of the top 5 capital-gaining retailers in a demand-driven economy ... 68
1 INTRODUCTION

Trucking is a platform for researching the application of evolutionary computation methods. It simulates a small country where trucks travel among dealers and transport commodities from producers to retailers and from retailers to consumers. Both trucks and retailers attempt to survive and make the most individual profit. Trucking originated from a game invented by Mark Stefik and others at Xerox Palo Alto Research Center (PARC) [16] during the eighties as part of their research into expert systems.

Genetic Algorithms (GAs) are a form of optimization and search technique. They solve problems by simulating the natural phenomenon of adaptation by evolution. Genetic algorithms have been widely used to solve complex optimization problems with a large but finite search space.

In this thesis, we use Genetic Algorithms to evolve trucks and retailers so as to find the strategies that enable them to achieve the best performances. There is an earlier version of Trucking done by Jeff Edelstein [4] and Debbie Papoulis [14] that also used the genetic algorithm to solve this problem. However, an insufficiency of their simulation was lack of the evolution of retailers. In addition, the search space of trucks was quite small. The purpose of this thesis is to make improvements to this topic by designing more strategies of trucks to increase their search space and by adding the evolution of retailers. Besides, we also added two forms of limitation to resources to increase the competition.
of trucks and retailers. On the other hand, we also want to find out if the genetic algorithm is a good solution to the Trucking problem.

In the next section, an overview of optimization and genetic algorithms is described. The details of Trucking are introduced in Section 3. Section 4 explains the two genetic algorithms used in Trucking. Some simulation parameters are described in Section 5. The simulation results and a discussion are presented in Section 6. The last section draws a conclusion.
2 AN OVERVIEW OF OPTIMIZATION AND GENETIC ALGORITHMS

"Optimization is the process of adjusting the inputs to or characteristics of a device, mathematical process, or experiment to find the minimum or maximum output or result." [9] Optimization problems exist everywhere. For example, what is the fastest route from home to the school? In what order should we arrange the things so that we can finish them in the shortest time period? Optimization is also an important task in many fields of scientific research and engineering. Many methods have already been developed to solve optimization problems in different areas.

2.1 Categories of Optimization

R. L. Haupt and S. E. Haupt break optimization algorithms into six categories. [9] We give a brief introduction to these categories below.

1. Trial-and-error and mathematical function optimization

In trial-and-error optimization, the process that produces the output is like a black box. Different input parameters are fed to one side of the box, and the results are seen on another side. How the results are changed is invisible. In mathematical function optimization, the process is like a white box. The actions inside the box,
which are assumed to be represented as a mathematical formula, are known and can be controlled. Filling the box with different formula produces different results.

2. One-dimensional and multi-dimensional optimization
   
   One-dimensional optimization handles the problems that have only one parameter.
   Multi-dimensional optimization deals with the problems with more than one parameter.

3. Static and dynamic optimization
   
   Dynamic optimization means time is one of the factors that affect the optimum solution. Static optimization means that the solution of the problem is not related with time.

4. Discrete and continuous parameter optimization
   
   Discrete parameter optimization solves problems whose parameters have a finite number of possible values. Continuous parameter optimization solves problems whose parameters have an infinite number of possible values.

5. Constrained and unconstrained optimization
   
   Constrained optimization takes the limitations to the parameters into consideration.
   Parameters of unconstrained optimization do not have limitations. They can take any value.

6. Minimum seeking and random optimization
   
   Since the output from the process is usually seen as the cost that needs to be minimized, optimization becomes minimization. The process is also called the cost function. Minimum seeking optimization is to find the optimum solution by starting from an initial set of parameter values and then search in a determinant direction.
This kind of optimization methods tends to be fast but is easily stuck in local optima. Random optimization also starts from an initial set of parameter values, but it guides itself by using the previous results combined with random choice. Random optimization tends to be slower but has greater chance of finding the global optimum. [9]

### 2.2 Minimum Seeking Algorithms

Minimum seeking algorithms are traditional optimization methods. We introduce two of them: exhaustive search and analytical optimization.

**Exhaustive Search** – Exhaustive search algorithm finds the global minimum by sampling the cost function as sufficiently as necessary. Since the evaluation of the cost function required is extremely large, exhaustive search methods are only suitable to problems with small number of parameters and in a limited search space.

**Analytical Optimization** – Analytical optimization uses calculus to find the minimum of the cost function. It can quickly find a minimum by computing the derivatives of the cost function with a single parameter. To functions with two or more parameters, gradient, Laplacian of the functions are calculated to find the extrema. It is difficult to find all the extrema if there are too many parameters. In addition, a search scheme needs to be applied to all the extrema to find the global minimum. Furthermore, “Continuous functions with analytical derivatives are necessary (unless derivatives are taken
numerically, which results in even more function evaluations plus a loss of accuracy).” [9] These shortfalls make it difficult for analytical optimization to solve practical problems.

2.3 Natural Optimization Methods

Natural optimization methods model processes in nature. Like minimum seeking methods, natural optimization methods also start from an arbitrary point or points. They explore new points in the search space by analyzing the results that have got, combined with applying stochastic factors at the same time.

Simulated Annealing – Simulated annealing was introduced by Kirkpatrick and coworkers [12] in the early 1980s. This method models the process of annealing. It uses a control parameter, which is analogous to the annealing temperature, to control the speed of converging to the optimum.

2.4 Genetic Algorithms

In genetic algorithms, each candidate solution to a problem is represented as a genome, which includes one or more chromosomes (most applications of genetic algorithms use a single chromosome). A chromosome is a set of genes, each of which determines a particular aspect of the solution. An initial population of candidate solutions is then created. After that, a series of genetic operators are applied, driving the population to evolve so as to find the best solution(s).

The typical process of genetic algorithms works as follows:

a. An initial population is created randomly.

b. The individuals of the population are evaluated based on the fitness function that is used to measure how well the individuals perform. The better an individual performs, the higher its fitness score.

c. Two individuals are selected for reproduction. The higher an individual’s fitness score, the higher its chance of being selected.

d. The two selected individuals reproduce, yielding one or more offspring(s). The usual technique of reproduction is to perform crossover.

e. With probability p, the newly created offspring(s) are randomly mutated.

f. Replace the old population with the new one, then go to step b if the termination conditions have not been met.
Each iteration of the above process is called a *generation*.

### 2.4.1 Components of A Genetic Algorithm

#### 2.4.1.1 Encoding

To use a genetic algorithm, the first thing to do is to represent each solution as a set of genes. Each gene encodes a particular aspect of the solution. The most common encoding method is binary encodings, which encode solutions into binary strings (strings of 1s and 0s). Other methods include many-character and real-valued encodings, tree encodings, etc. Many-character and real-valued encodings define the value of each gene (or say, parameter) as a character from an alphabet of many characters or a real number, and thus are more natural for many problems. Which encoding works best? Some researchers (e.g., Janikow and Michalewicz [11]; Wright [17]; Haupt and Haupt [9]) indicate that real-valued encodings perform better than binary encodings. However, as Mitchell [13] says, “The performance depends very much on the problem and the details of the GA being used, and at present there are no rigorous guidelines for predicting which encoding will work best.”

#### 2.4.1.2 Basic GA Operators

The basic GA operators include selection, crossover and mutation.
**Selection** - This operator chooses individuals in the population for reproduction based on their fitness. Individuals with higher fitness are more likely to be selected, thus they are more likely to have offspring. “Selection has to be balanced with variation from crossover and mutation (the ‘exploitation/exploration balance’): too-strong selection means that suboptimal highly fit individuals will take over the population, reducing the diversity needed for further change and progress; too-weak selection will result in too-slow evolution.” [13] There are many selection methods. The most common ones include Roulette Wheel selection [6], Stochastic Universal sampling [2], Sigma Scaling [5], Elitism [3], Rank selection [1], Tournament selection [7], etc.

**Crossover** - Crossover is the feature that distinguishes GA from other optimization techniques. There are different approaches or different implementation details for crossover in different contexts of encodings. The simplest method is the *single-point crossover*. In the context of binary encodings, a crossover point is randomly chosen and the corresponding binary strings of the parents are exchanged up to this point to form new offspring. The difference in the context of many-character and real-valued encodings is the unit of parts being swapped is a whole gene (parameter), rather than a binary bit. Two-point and multi-point crossovers are extensions of the single-point crossover. Another common approach is *uniform crossover*, in which each bit/gene of the offspring is selected randomly from the corresponding bit/gene of the parents.
**Mutation** - Mutation is to randomly alter a small percentage of bits/genes of the offspring. The purpose of mutation is the same as crossover; that is to achieve variation. The mutation probability is usually very low, e.g. 0.01.

### 2.4.1.3 Convergence and The Termination Conditions

A genetic algorithm converges when all or most of the genes of the population are identical. John Holland proposed the schema theorem [10] for convergence analysis of genetic algorithms. A global convergence proof through a Markov chain model for genetic algorithms using elitism was presented by G. Rudolph. [15] An algorithm must comply strictly with the assumptions of this proof in order to converge. In practice, we usually apply some termination conditions to stop the algorithm. Some common examples are: a satisfactory answer has been reached; no improvement in the chromosomes; a predetermined number of generations or time has elapsed.

### 2.4.2 GAs vs. Other Traditional Optimization Methods

As an optimization technique, genetic algorithms are relatively new. However, they have achieved remarkable success during the last two decades. Compared with traditional optimization methods, some of GAs advantages include: [9]
• GAs search from a wide range of the solution space simultaneously instead of
starting from a single solution to search through the space, thus they have more
chance to reach the global optimum.
• GAs work well in solving complex problems with large but finite search space.
• GAs encode the parameters rather than search with the actual values of the solutions.
• GAs only use the fitness value of a solution to guide themselves through the search
space. They do not require derivative information.
• GAs are easy to be implemented in parallel.
• GAs can handle optimization problems with a large number of parameters.

However, genetic algorithms have their disadvantages. A problem is that genes from
comparatively highly fit (but not optimal) individuals may take over the population, thus
making the algorithm converge on a local optimum. Since GAs need to sample and
evaluate a large population of potential solutions, they are usually slower than other
methods if they run on a serial computer. Moreover, genetic algorithms are not suitable
for every optimization problem. Some simple problems can be solved easily and quickly
by using traditional methods. [9]

2.4.3 Applications of Genetic Algorithms

“Genetic algorithms have been used in two ways: as techniques for solving technological
problems, and as simplified scientific models that can answer questions about nature.”
[13]
Some examples of the application in problem solving include: evolving computer programs, data analysis and prediction, evolving neural networks, scheduling, signal processing, etc.

Some examples of the application in scientific models include: modeling interactions between learning and evolution, modeling sexual selection, modeling ecosystems, measuring evolutionary activity, etc.
3 THE TRUCKING PROJECT

In this section, we describe the details of the Trucking project in three parts: its composite elements, economy types and numerical analysis, and the simulation.

3.1 Elements

The elements of the Trucking project include a small country, four kinds of dealers (producers, consumers, retailers, and gas stations), trucks, and three kinds of commodities (crates, items, and gas). One of the differences between the current version and the previous one is that there are no controllers and managers in this version. Controllers and managers acted as information providers and mediators to trucks and retailers in the previous version to control their behaviors. They are not necessary any more since Trucking is not seen as a competition between programmers in this version.

3.1.1 The Country

The country is a square. It is made up of a grid of roads: 10 streets running from West to East and 10 avenues running from South to North (Figure 3-1). Grid points are 10 kilometers apart. Each intersection may be either empty or occupied by at most one dealer. We use (street number, avenue number) to indicate the intersection formed by the
crossing of these two numbered roads. Then as shown in Figure 3-1, (1, 2) is the
intersection of street No.1 and avenue No.2. As mentioned above, if it is not empty at (1, 2), there must be a producer, a consumer, a retailer, or a gas station there. Whatever it is, there must be only one.

Figure 3-1 The country

Trucks run along the streets and avenues. Traffics on all roads are in both directions.
Since the country is a square, not a torus, its north edge does not connect to the south edge, and the east does not connect to the west. A truck moving in a direction must change to another direction once it has arrived at the edge of the country and it wants to continue to move.
Traffic congestion is taken into consideration in this version of Trucking. A maximum number is added to limit the total number of trucks at a place at the same time, no matter traveling through or stopping there. This maximum number is predetermined as a simulation parameter. Once the maximum number is reached, that place becomes congested. No more trucks can move in. Only trucks that have already been at that place can move out. Congestion disappears after at least one truck has moved out.

3.1.2 Commodities

There are three kinds of commodities flowing in the country: crates, items, and gas.

The flow of gas is quite simple. Gas can only be sold by gas stations to trucks. Gas stations supply whatever amount of gas that trucks request.

A crate is actually a package of a number of items (100 items in this simulation). Crates are manufactured by producers. Trucks buy crates from producers and then sell them to retailers. After unpacking the crates, retailers get items and then sell items to trucks, which then sell the items to consumers. Consumers consume the items.

We can see that the flow of crates starts from producers, and then passes to trucks when trucks buy crates, and to retailers when retailers buy crates, where it turns into the flow of items. With trucks purchasing items from retailers and selling them to consumers, the
flow of items moves to trucks from retailers, and finally ends at consumers. Therefore the number of crates in the flow determines the number of items in the flow. There is a limitation of the number of crates and items in the flow. This limitation is discussed in detail in Section 3.2 Economy Types and Numerical Analysis.

3.1.3 Producers

Producers manufacture crates and sell them to trucks. However, they do not sell actively. Only when trucks want to buy can producers sell crates. All producers sell crates at a fixed price. In this simulation, the price is $60 per crate. They may keep a stock of crates. A producer’s warehouse is assumed with unlimited capacity, but its supply of crates is not guaranteed to be unlimited. The total number of crates that producers produce depends on which economy type the simulation is running. The economy types and their limitations to the number of crates that producers produce are discussed in Section 3.2 Economy Types and Numerical Analysis.

3.1.4 Consumers

Consumers buy items from trucks and consume the items. Like producers, they do not consume actively. Only when trucks want to sell can consumers buy items. Consumers buy items at a fixed price. In this simulation, the price is $1 per item. Consumers may maintain a stock of items. A consumer’s warehouse is also assumed with unlimited
capacity, but its capacity of consuming is not guaranteed to be unlimited. The total number of items that consumers can consume depends on which economy type the simulation is running.

3.1.5 Retailers

Retailers buy crates from trucks, unpack the crates to yield items, and sell the items back to trucks. They are passive too; that is they always wait for trucks to trade with them. They gain profit by selling items in a higher price than the price in which they buy a crate divided by the number of items yielded from the crate, just like dealers in real life. At the same time, since retailers need to rent space for keeping their stock, they must pay rent. The amount of rent depends on the size of the warehouse. For instance, a retailer pays $500 every 1000 simulation time units for a warehouse of 8 crates capacity. To keep the simulation simple, all retailers have the same size of warehouse. A retailer needs to ensure that its stock does not exceed the storage limit.

Each retailer has its buying and selling prices. These prices are given when the retailer is created. They are fixed during the retailer’s lifetime. One major difference of this version from the previous one is that retailers and trucks may negotiate their trading prices in this version. How is negotiation performed? Let’s look at the following example.

Truck: My selling price for a crate is $70.
Retailer: My buying price is $68.
Truck: Sorry, no deal.

Or:

Truck: My selling price for a crate is $69.
Retailer: My buying price is $73.
Truck: Fine! Let's trade at $0.5 \times (69+73) = $71.

In other words, the buying price must be higher than or equal to the selling price if they want to get a deal. Their actual trading price is a half of the sum of their announcing prices. However, this price is only the price for the current deal. Their original announcing prices are not changed.

Retailers become bankrupt by going into debt. Retailers that are bankrupt are removed from the simulation.

3.1.6 Gas stations

The only business of gas stations is to sell gas to trucks. We assume their supply of gas is unlimited. Therefore a truck’s request of buying gas is always satisfied. To make things simple, all gas stations sell gas at a same price. In this simulation, it is $1 per litre. This price is fixed during the simulation.
3.1.7 Trucks

Trucks travel across the country transporting goods between dealers, which are distributed across the country. A truck may be: a PR truck moving crates from producers to retailers; an RC truck moving items from retailers to consumers; or both. A truck makes profit by keeping the selling price higher than the buying price of the same commodity. As mentioned above, producers sell crates at $60 and consumers buy items at $1, thus if a truck does not want to lose money, it must sell crates at a price higher than $60 and buy items at a price lower than $1. A truck negotiates a trading price with the retailer that it tries to make a deal with (see Section 3.1.5 Retailers).

We assume trucks move at a steady speed. In this simulation, their average speed is 60 km/hour; that is 1 km/minute. They consume gas in the course of moving. Their gas-consuming rate is fixed at 0.1 litres/km. When a truck is low on gas, it needs to find a gas station and purchase gas. A truck becomes inactive when it has run out of gas and is not able to buy gas immediately. Inactive trucks are removed from the simulation.

Each truck has a carrying capacity that is determined when the truck is created. Trucks must ensure their stock does not exceed the storage limits.

A truck may use a cell phone to call intersections of the country. If the called intersection has a dealer, the truck gets the dealer’s information such as the type of the dealer, price, etc. If no dealers exist at that intersection, the truck knows the intersection is empty. No
matter whether the intersection has dealer or not, the truck must pay a certain amount of money for the call.

Trucks move along the streets and avenues, and make deals at intersections. Sometimes a truck can encounter traffic congestions. It cannot move into the congested sites, but must wait for the congestions to disappear or go away.

3.2 Economy Types and Numerical Analysis

In this version of Trucking, the concept of economy types is introduced. Two economy types have been defined: supply-driven and demand-driven.

- “In a supply-driven economy, producers manufacture creates at a constant rate and consumers buy as many items as they can.” [8]

- “In a demand-driven economy, consumers buy items at a constant rate and producers provide as many crates as necessary to keep up with the demand.” [8]

Therefore, in a supply-driven economy consumers do not need to consider how many items they can consume. They simply consume as many as provided by trucks. That means a truck will never be refused when it wants to sell items to consumers. On the other hand, it is possible that the truck is turned down when it requests to buy crates from a producer because the producer’s turnout is limited by the rate. Similarly, in a demand-driven economy producers’ turnouts are not limited. They simply manufacture as many
as requested by trucks. That means a truck will never be refused when it asks to buy crates from producers, but it can be turned down when it tries to sell items to a consumer.

The concept of economy types embodies the prevalent reality of limited resources. Both of these two types are provided as options in the simulation. Their rates will be computed later in this section.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_r$</td>
<td>10</td>
<td></td>
<td>Number of roads in the country</td>
</tr>
<tr>
<td>$d$</td>
<td>10</td>
<td>km</td>
<td>Distance between grid points</td>
</tr>
<tr>
<td>$N_i$</td>
<td>100</td>
<td></td>
<td>Number of items in a crate</td>
</tr>
<tr>
<td>$v$</td>
<td>1</td>
<td>km/min</td>
<td>Average speed of a truck</td>
</tr>
<tr>
<td>$k$</td>
<td>0.1</td>
<td>litres/km</td>
<td>Gas-consuming rate of trucks</td>
</tr>
<tr>
<td>$C_g$</td>
<td>1</td>
<td>$/litre</td>
<td>Cost of gas</td>
</tr>
<tr>
<td>$\pi$</td>
<td>0.4</td>
<td></td>
<td>Profitability rate</td>
</tr>
<tr>
<td>$C_i$</td>
<td>1</td>
<td>$</td>
<td>Cost of one item</td>
</tr>
<tr>
<td>$N_t$</td>
<td></td>
<td></td>
<td>Number of trucks</td>
</tr>
<tr>
<td>$N_c$</td>
<td></td>
<td></td>
<td>Average carrying capacity of trucks</td>
</tr>
<tr>
<td>$\mu$</td>
<td></td>
<td></td>
<td>Estimated factor of the length of a round trip</td>
</tr>
<tr>
<td>$\lambda$</td>
<td></td>
<td></td>
<td>Estimated proportion of the total time spent on traveling</td>
</tr>
</tbody>
</table>

*Figure 3-2 Basic simulation parameters*
Let the simulation time unit be 1 minute. Figure 3-2 lists some of the simulation parameters.

The values of the first eight parameters in the table are built into the simulation. They have already been mentioned in the previous sections except \( \pi \). \( N_t \) and \( N_c \) may vary in the process of the simulation. At last, \( \mu \) and \( \lambda \) are two estimated values that are required to be inputted at the beginning of the simulation. They are only used to predict the performance of the simulation. \( \pi, \mu \) and \( \lambda \) are explained below.

\( \pi \) is the profitability rate that indicates the profit space of trucks and retailers. \( \pi \) satisfies \( 0 < \pi < 1 \). According to Figure 3-2, the cost of buying one crate from a producer is \( (1 - \pi)N_tC_t \); the total profit made from a crate, which is the difference between the money earned by selling the items to consumers and the money spent on buying the crate, is \( \pi N_tC_t \). Given \( \pi = 0.4 \) as shown in Figure 3-2, we get the cost of a crate is \( (1 - 0.4) \times 100 \times 60 \), as mentioned in Section 3.1.3 Producers, and the profit trucks and retailers can get from a crate is \( 0.4 \times 100 \times 60 = 40 \).

Before explaining parameter \( \mu \), the concept of "round trip" needs to be introduced first. A round trip is the journey within which a truck acts as a PR truck followed by as an RC truck. In other words, during a round trip, a truck performs the following actions in order: go to a producer to buy crates, go from the producer to a retailer (R1) and sell the crates, go from R1 to a retailer (R2) (R2 and R1 may be the same retailer) to buy items, go from R2 to a consumer to sell the items, go to a producer. Trucks are actually not necessary to
move in round trips. This concept is only used to estimate the performance of trucks. The average length of a round trip is measured by multiplying the length of a road by factor $\mu$. In our simulation, typically $1 < \mu < 4$. For instance, let $\mu = 3$, the road length is $N_r \times d = 10 \times 10 = 100$ km according to Figure 3-2, then the average length of a round trip would be $\mu N_r d = 3 \times 10 \times 10 = 300$ km.

A truck spends time on traveling, trading crates or items, and buying gas during its lifetime. Parameter $\lambda$ is the estimated proportion of the total time spent traveling. It satisfies $0 < \lambda < 1$. For example, if $\lambda = 0.6$, that means in average a truck spends 60% of its time traveling and the remaining 40% trading crates or items, and buying gas.

To make the simulation meaningful, "the flow of goods through the system must be sufficient to ensure profitability and the carrying capacity of trucks must be sufficient to support this flow." [8] How many crates (or items) are sufficient? The analysis is shown below.

The following calculations apply to a single truck making a round trip.

Distance traveled: $\mu N_r d$

Cost of gas: $\mu N_r d k C_g$

Time spent traveling: $\mu N_r d / v$

Total time for the round trip: $\mu N_r d / \lambda v$

Profit made from one crate: $\pi N_i C_i$
Profit made from \( N_c \) crates: \( N_c \pi N_i C_i \)

In order for a truck to make a profit, the money earned during a round trip must be more than the money spent on gas for the trip; that is \( N_c \pi N_i C_i > \mu N_i d k C_g \). [8] According to Figure 3-2:

- The profit made from \( N_c \) crate is: \( N_c \pi N_i C_i = N_c * 0.4 * 100 * 1 = 40N_c \) ($)
- The cost of gas spent on the round trip is: \( \mu N_i d k C_g = 3 * 10 * 10 * 0.1 * 1 = 30 \) ($)

Obviously, \( 40N_c > 30 \) \( (N_c \geq 1) \). So it is possible that the truck can make a profit.

One truck needs \( \mu N_i d / \lambda v \) minutes to sell \( N_c \) crates. There are \( N_t \) trucks running simultaneously in the country, so in \( \mu N_i d / \lambda v \) minutes \( N_t \) trucks can sell \( N_c N_t \) crates. In another word, \( N_t \) trucks can sell \( \lambda v N_c N_t / \mu N_i d \) crates in one minute. Therefore, a supply-driven simulation must ensure that the manufacture rate of producers is at least \( \lambda v N_c N_t / \mu N_i d \) crates per time unit, and a demand-driven simulation must ensure that the consumption capability of consumers is not lower than \( \lambda v N_c N_t / \mu N_i d \) items per time unit. [8] \( \lambda v N_c N_t / \mu N_i d \) crates per time unit and \( \lambda v N_c N_t / \mu N_i d \) items per time unit are set as the manufacture rate of producers and the consumption capability of consumers respectively in this simulation.

### 3.3 The Simulation

The control program first reads parameters from the user, then creates and initializes the country, all dealers (producers, consumers, retailers, and gas stations), and trucks. After
that, trucks begin to run. So does the evolutionary process. Here, several problems need to be addressed.

The first is the positions of dealers and trucks. In the simulation, the positions of producers and consumers are predetermined, not randomly created. They are given by the user at the time of initialization. “The idea is that producers are in ‘rural districts’ near the boundaries of the country and consumers are in ‘urban districts’ near the centre of the country.” [8] Retailers and gas stations are positioned randomly, but not on top of the existing dealers. A dealer is fixed once it is positioned. If a retailer is removed from the simulation, the intersection that it occupied is released. Trucks are initialized at random locations.

Secondly, trucks and retailers are allowed to run for a period of time before they are evaluated. The length of this period is determined by the user as a simulation parameter. At the end of each period, inactive trucks and retailers are removed and all active trucks and retailers are evaluated. After that the selected individuals perform adaptation. A major difference from the previous version is that all active trucks and retailers will remain in the simulation. They make up of next generation’s population along with the newborn individuals. Moreover, in the following generation they will resume their final statuses and activities of the previous generation.

Next, different population policies are applied to retailers and trucks. Retailers have a constant population. An initial population of retailers of size $P_r$ is created at the beginning
of the simulation. In the process of evolution a new retailer can be born only when a 
retailer has been removed. The actual population size of retailers may be less than \( P_r \), but 
it is not allowed to exceed \( P_r \). In contrast to retailers, trucks have a variable population. 
Trucks who are dead are removed from the simulation. New trucks are born whenever 
there are enough healthy trucks at the time of adaptation.

Finally, there are three conditions, any one of which leads to the termination of the 
simulation. They are: there are no active trucks left; there are no retailers left; a 
predetermined number (given by the user) of simulation time units have elapsed.
4 THE GENETIC ALGORITHMS FOR TRUCKING

As described above, Trucking is a complex search problem. To find out the most profitable trucks and retailers, we need to find out the strategies they use. Two genetic algorithms that are used in the present Trucking simulation are described below.

4.1 The Genetic Algorithm for Trucks

4.1.1 Encoding

A truck has many activities when it is active. It moves around the country, finds deals, buys and sells crates, buys and sells items, buys gas, and makes phone calls. Sixteen areas of strategies are developed to represent a truck’s solution. We encode these areas of strategies into a single chromosome consisting of 16 genes. They are described below.

1. $G_g$ - Determine the lowest gas limit. We divide the gas tank into 20 levels. Suppose that the truck has $\gamma$ litres of gas and its gas tank capacity is $\Gamma$. The truck will search for gas if

$$\frac{\gamma}{\Gamma} < \frac{G_g}{20}$$

Here, $0 \leq G_g \leq 20$. We have the following situations:
- If $G_g = 0$, the lowest gas limit is 0; that means the truck will never look for gas. It will eventually run out of gas and become inactive.

- If $G_g = 20$, the lowest gas limit is 20; that means the truck will always search for gas unless it has full tank of gas ($\gamma = \Gamma$). This kind of trucks has little chance to do other trading.

- If $0 < G_g < 20$, the truck will search for gas whenever the amount of gas in its tank ($\gamma$) falls below

$$\frac{G_g}{20} * \Gamma$$

2. $G_s$ - Determine the selling price of crates. Let $T_s$ be the price at which the truck sells crates to retailers, then $T_s$ is calculated as follows:

$$T_s = N_i C_i \left( 1 - \pi + \pi \frac{G_i}{255} \right)$$

Here, $0 \leq G_s \leq 255$. We have the following situations:

- If $G_s = 0$, $T_s = N_i C_i \left( 1 - \pi \right)$, which is the price at which producers sell crates to trucks. That means such trucks do not earn money in the PR trips.

- If $G_s = 255$, $T_s = N_i C_i$. The truck will sell crates at the highest price. If the truck succeeds in selling one crate, it will earn all profits created from a crate only in the PR trip and leave no profit margin for the crate’s journey there after.

- If $0 < G_s < 255$, the truck will earn part of the profits created from crates in the PR trips. The remaining profit margin will flow to the remaining journey of the crates.

3. $G_b$ - Determine the buying price of items. Let $T_b$ be the price at which the truck buys items from retailers. $T_b$ is calculated as follows:
\[ T_b = C_i \frac{G_b}{255} \]

Here, \( 0 \leq G_b \leq 255 \). We have the following situations:

- If \( G_b = 0 \), \( T_b = 0 \). The truck will not get any item unless a retailer gives to it for free.

- If \( G_b = 255 \), \( T_b = C_i \). The truck will buy items at the highest price.

- If \( 0 < G_b < 255 \), the truck will buy items at a price between 0 and \( C_i \).

4. \( G_a \) - Determine the proportion of capital that will be transferred to the child truck. Let \( \alpha \) be the proportion of capital that will be transferred to the child truck during reproduction. We define

\[ \alpha = \frac{G_a}{100} \]

Here \( 0 \leq G_a \leq 100 \). Let \( c \) be the capital of a truck. In the case of reproduction, suppose \( T_1 \) and \( T_2 \) are the parent trucks, and \( T \) is the new created child truck. We use "object oriented" notation, so that \( T_1.c \) is the capital of truck \( T_1 \), etc. Then the transfers of capital are:

\[ T_1.c \leftarrow (1 - T_1.\alpha) \times T_1.c \]
\[ T_2.c \leftarrow (1 - T_2.\alpha) \times T_2.c \]
\[ T.c \leftarrow T_1.\alpha \times T_1.c + T_2.\alpha \times T_2.c \]

- If \( G_a = 0 \), the truck does not transfer money to its child.
- If \( G_a = 100 \), the truck transfers all of its capital to the child.
- If \( 0 < G_a < 100 \), the truck transfers part of its capital to the child.

5. \( G_{\text{capacity}} \) - The truck’s carrying capacity. Its unit is crates. It satisfies \( 1 \leq G_{\text{capacity}} \leq 10 \).
6. $G_{\text{tank}}$ - Capacity of the truck's gas tank. Its unit is litres. It satisfies $1 \leq G_{\text{tank}} \leq 60$.

7. $G_{\text{reserve}}$ - The strategy that determines the amount of money that the truck reserves for buying gas. It has three values.

   0: Reserve $0$.

   1: Reserve just enough money for buying a full tank of gas.

   2: Reserve money for buying enough gas to travel the longest distance across the country. According to Figure 3-1, the longest distance across the country is the distance between (0,0) and (9,9) or between (0, 9) and (9,0), which is calculated as: $2N_d \cdot d = 2 \times 10 \times 10 = 200$ (km). The cost of gas for traveling this distance will be $2N_d \cdot d \cdot k \cdot C_g = 200 \times 0.1 \times 1 = 20$ ($).

8. $G_{\text{corner}}$ - The strategy that determines whether the truck needs to move to the nearest corner right after creation. It has two values. Each value and the corresponding meaning are described as follows.

   0: Do not go to the nearest corner, but stay at the initial position.

   1: Go to the nearest corner.

9. $G_{\text{scan}}$ - The strategy that determines whether the truck scans the country after the strategy $G_{\text{corner}}$, but before looking for deals. It also has two values.

   0: The truck scans the whole country to mark down the locations of dealers before looking for any deal.

   1: The truck does not scan the country.
10. $G_{\text{deal}}$ - The strategy that determines how the truck looks for deals. It has three values.

0: The truck usually looks for a deal after it has completed one. However, in the course of trying to complete the current deal (say d1), if the truck discovers that it can make another deal (say d2) with the dealer at its present location, it will make the deal (d2) and then continue to complete the holding deal (d1) if the holding deal is still valid (e.g. still have room available for at least one crate if d1 is to buy crates).

1: The truck always looks for a deal and then goes to complete this deal. The difference with the above scheme is that it never makes other deals before the completion of the current deal even if it is at the right intersection where a dealer offers a good deal of other kinds.

2: The truck does not look for deals beforehand. It only keeps traveling across the country. It will search for gas when it finds itself low on gas. When the truck arrives an intersection where a dealer is located, it will check if it can make a deal with the dealer. If it can, it makes the deal; otherwise it just passes the dealer and goes to the next intersection. The truck does not memorize any information of dealers except the locations of gas stations. The truck will ignore $G_{\text{scan}}$, $G_{\text{priority}}$, $G_{\text{phone}}$ and $G_{\text{badbuy}}$ with this scheme.

11. $G_{\text{priority}}$ - The strategy that determines the priority of deals. It has three values as well.

0: In descending order: buy gas when the amount of gas in the tank falls below the lowest gas limit; sell crates if any; sell items if any; buy crates and buy items in turn.
1: In descending order: buy gas when the amount of gas in the tank falls below the lowest gas limit; sell crates if any; sell items if any; buy crates or buy items depending on which kind of dealer is closer.

2: The truck chooses a deal following these steps: buy gas when the amount of gas in the tank falls below the lowest gas limit or the truck is right at a gas station, otherwise check its status to see what kinds of deal it may make (for example, if the truck’s status is: has at least one crate in stock; has no items; has capital that is enough for buying a crate; has space available for at least one crate, the deals it may make are: sell crates; buy crates; buy items), then choose the closest deal. $G_{best}$ is useless with this scheme.

12. $G_{best}$ - The strategy that determines the criteria of the best retailer. It has three values.

0: The one that offers the best price.

1: The one that is at the nearest location with a good price. What is a good price? When the truck’s current deal is selling crates, a good buying price is the one that is higher than or equal to the truck’s selling price. When the truck’s current deal is buying items, any selling price that is lower than or equal to the truck’s buying price is a good selling price.

2: When the deal is selling crates, in addition to calculating the money the truck will gain on selling crates, also calculate the cost on the gas consumed in order to complete the deal, then the best retailer is the one that the truck can gain the biggest profit from.

When the deal is buying items, in addition to calculating the money the truck will spend on buying items, also calculate the cost on the gas consumed in order to complete the deal, then the best retailer is the one buying items from which the truck will cost the least.
13. $G_{\text{phone}}$ - The strategy that determines whether the truck uses the cell phone and how it uses the cell phone. It has three values.

0: The truck does not use the cell phone. With this scheme the truck only uses the information kept in its memory, some of which could be out of date.

1: Use the cell phone to verify the dealers' information, or when the truck has not enough information for making a decision, use the cell phone to get new information of intersections chosen randomly. Maximum is five calls in the latter situation.

2: This scheme is almost the same with the above except that the truck does not randomly choose intersections to get new information, but starts from the nearest intersections and extends to the father ones gradually. The maximum is five calls again.

14. $G_{\text{badsell}}$ - The strategy that determines whether the truck permits unprofitable sells. It has two values.

0: The truck sells crates only at a good price; that is at a price higher than or equal to its selling price.

1: The truck sells crates at a good price if possible. However, after it has failed continuously for at least three times in selling the crates, the truck sells the crates even at a bad price that may lead to a capital loss.

15. $G_{\text{badbuy}}$ - The strategy that determines whether the truck permits unprofitable buys. It has two values as well.

0: The truck buys items only at a good price; that is at a price lower than or equal to its buying price.
1: The truck buys items at a good price if possible. However, after it has failed continuously for at least three times in buying items, the truck buys even at a bad price that may lead to a capital loss.

16. $G_{\text{congestion}}$ - The strategy that determines how the truck responds to traffic congestions. We assume a truck always has a destination to go when it is moving. $G_{\text{congestion}}$ has three values.

0: Stay still until the congestion disappears. Then resume the journey.

1: If the congestion happens at the truck's destination, stand still until the congestion disappears; otherwise go back to the previous intersection and move along another route.

2: Cancel the current destination and look for a new one.

<table>
<thead>
<tr>
<th>Gene number</th>
<th>Symbol</th>
<th>Strategy</th>
<th>Value range</th>
<th>Number of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$G_s$</td>
<td>Lowest gas limit</td>
<td>[0, 20]</td>
<td>21</td>
</tr>
<tr>
<td>2</td>
<td>$G_s$</td>
<td>Selling price of crates</td>
<td>[0, 255]</td>
<td>256</td>
</tr>
<tr>
<td>3</td>
<td>$G_b$</td>
<td>Buying price of items</td>
<td>[0, 255]</td>
<td>256</td>
</tr>
<tr>
<td>4</td>
<td>$G_a$</td>
<td>Capital transfer proportion</td>
<td>[0, 100]</td>
<td>101</td>
</tr>
<tr>
<td>5</td>
<td>$G_{\text{capacity}}$</td>
<td>Carrying capacity</td>
<td>[1, 10]</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>$G_{\text{tank}}$</td>
<td>Gas tank capacity</td>
<td>[1, 60]</td>
<td>60</td>
</tr>
<tr>
<td>7</td>
<td>$G_{\text{reserve}}$</td>
<td>The amount of money reserved for buying gas</td>
<td>[0, 2]</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>$G_{\text{corner}}$</td>
<td>Move to the nearest corner or not</td>
<td>[0, 1]</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>$G_{\text{scan}}$</td>
<td>Scan the country or not</td>
<td>[0, 1]</td>
<td>2</td>
</tr>
<tr>
<td>---</td>
<td>-----------------</td>
<td>------------------------</td>
<td>--------</td>
<td>---</td>
</tr>
<tr>
<td>10</td>
<td>$G_{\text{deal}}$</td>
<td>How to look for deals</td>
<td>[0, 2]</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>$G_{\text{priority}}$</td>
<td>Priority of the deals</td>
<td>[0, 2]</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>$G_{\text{best}}$</td>
<td>The criteria of the best retailer</td>
<td>[0, 2]</td>
<td>3</td>
</tr>
<tr>
<td>13</td>
<td>$G_{\text{phone}}$</td>
<td>How to use the cell phone</td>
<td>[0, 2]</td>
<td>3</td>
</tr>
<tr>
<td>14</td>
<td>$G_{\text{badsell}}$</td>
<td>Permit unprofitable sells or not</td>
<td>[0, 1]</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>$G_{\text{badbuy}}$</td>
<td>Permit unprofitable buys or not</td>
<td>[0, 1]</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>$G_{\text{congestion}}$</td>
<td>How to respond to traffic congestions</td>
<td>[0, 2]</td>
<td>3</td>
</tr>
</tbody>
</table>

**Figure 4-1 Summary of truck's genes**

The 16 genes are summarized in Figure 4-1. The total number of trucks can be explored in the search space is approximately:

$$21 \times 256 \times 256 \times 101 \times 10 \times 60 \times 3 \times 2 \times 2 \times 3 \times 3 \times 3 \times 2 \times 2 \times 3$$

$$= 972,790,589,030,400$$

The actual number will be a little smaller because changing a gene does not always change the behaviour of a truck. Take for example, two trucks that both have '2' for the gene $G_{\text{deal}}$ and their values of corresponding genes are the same except $G_{\text{scan}}, G_{\text{priority}}, G_{\text{phone}}$ and $G_{\text{badbuy}}$. These two trucks will have identical behaviour even if the values of their $G_{\text{scan}}, G_{\text{priority}}, G_{\text{phone}}$ and $G_{\text{badbuy}}$ genes are different. This is a huge space. Genetic algorithms are well known to solve such complex optimization problems with finite but large search space.
Real-valued encoding is used. The chromosome is composed of 16 fixed-length and fixed-order genes. Each gene is represented by an unsigned short integer.

### 4.1.2 Fitness

There is not a formula served as the fitness function of trucks in this simulation. The fitness of a truck is measured by the profit earned during the evaluated generation. In detail, we say the value of a truck is the sum of the following four quantities: its capital, the value of the crates in stock (VC), the value of the items in stock (VI), and the value of the gas in the tank (VG). VC is computed by multiplying the selling price of crates by the number of crates in stock. In the similar way, VI is obtained by multiplying the selling price of items by the number of items in stock. VG is got by multiplying the price of gas by the amount of gas in the tank. The profit a truck earned during a generation is the difference of the truck's value between the time of evaluation and the beginning of this generation. If the difference is positive, that means the truck made a profit in this generation. If the difference is zero or negative, the truck did not make a profit in this generation. The truck that made the largest profit has the highest fitness.

### 4.1.3 Selection
A truck will not be discarded unless it has become inactive. Each time when the simulation control program performs selection, the first step is to remove all inactive trucks out of the simulation.

Although all active trucks are kept for the next generation, not all of them can reproduce. The remaining trucks are divided into two parts: healthy trucks and unhealthy trucks. Healthy trucks are the trucks that have made profit in the current generation, and the ones that did not gain profit are unhealthy. In other words, healthy trucks have positive fitness values and unhealthy trucks have zero or negative fitness values. Though the unhealthy trucks are not discarded, they are not allowed to reproduce. Reproduction only happens to healthy trucks.

We apply a reproduction probability $p$ to the remaining healthy trucks to yield the number of trucks that can reproduce. The number of reproducing trucks, $N$, is obtained by multiplying the number of healthy trucks by $p$. If $N$ is odd, we subtract one from it so that $N$ is always even. Then each pair of parent trucks is selected randomly from the healthy trucks for reproduction.

### 4.1.4 Crossover

After the trucks are selected and paired, they can reproduce. Each pair of parent trucks will perform crossover and yield a new child truck. The crossover method used here is uniform crossover. Half of the genes of the child truck are picked randomly from one
parent and the remaining half are taken from the other. Unlike some other methods in
which it is possible that the genes of the offspring are all taken from only one parent, this
method makes the genes of every child truck taken from both parents evenly (crossover
rate is 1), thus this method introduces more new chromosomes. Since the parents will
remain in the new population, this method can explore the search space faster without
destroying the existing solutions.

4.1.5 Mutation

Crossover can bring new chromosomes into the population. However, with the crossover
method described above, the new chromosomes are only the recombination of the
existing genes. No new gene values are introduced. So it is necessary to introduce
mutation to child trucks to increase the variation. Mutation is performed to each gene of a
new child truck with a very low probability known as mutation rate. A gene is mutated by
replacing its value with a new random value.

A new truck will be created with the new chromosome at a random location, with initial
capital given by its parents and with empty stock. There are two ways to have gas in its
tank. The first one is that the simulation control program simply gives it full tank of gas
to start. This way the new truck need not pay for the starting gas. Another solution is to
deduct the cost of gas from the new truck's capital. If it has not enough money to pay for
a full tank of gas, the new truck only can get its starting gas that costs half of its initial
capital. The user can choose which solution to use in the simulation.
4.2 The Genetic Algorithm for Retailers

4.2.1 Encoding

In contrast to trucks, retailers are passive. They always wait for things to be done to them. A retailer, for example, can have something to do only when a truck approaches it and wants to make a deal with it. A retailer only responds to requests from a truck. So the behaviour of a retailer is simple: it buys crates when a truck sells to it, and it sells items when a truck buys from it. The only factors that a retailer has to control its profitability are its buying and selling prices. Retailers also need a strategy for reproducing. Therefore we encode these areas of strategies into a single chromosome consisting of three genes. They are described below.

1. \( G_b \) - Determine the buying price of crates. Let \( R_b \) be the price at which the retailer buys crates from trucks. \( R_b \) is calculated as follows:

\[
R_b = N \cdot C_i \left( 1 - \pi + \pi \cdot \frac{G_b}{255} \right)
\]

Here, \( 0 \leq G_b \leq 255 \). We have the following situations:

- If \( G_b = 0 \), \( R_b = N \cdot C_i \left( 1 - \pi \right) \), which is the price at which producers sell crates to trucks.

In this case, the retailer buys crates at the lowest price. It can be satisfied only when a truck sells crates not requiring a profit.
- If $G_b = 255$, $R_b = N_iC_i$. The retailer will buy crates at the highest price.

- If $0 < G_b < 255$, the retailer will buy crates at a price between $N_iC_i (1 - \pi)$ and $N_iC_i$.

2. $G_s$ - Determine the selling price of items. Let $R_s$ be the price at which the retailer sells items to trucks, then $R_s$ is calculated as follows:

$$R_s = C_i \cdot \frac{G_s}{255}$$

Here, $0 \leq G_s \leq 255$. We have the following situations:

- If $G_s = 0$, $R_s = 0$. The retailer gives out items for free. Such retailers will surely become bankrupt in a short time.

- If $G_s = 255$, $R_s = C_i$. Which is the price at which consumers buy items. That is the highest selling price that a retailer can reach.

- If $0 < G_s < 255$, the retailer will sell items at a price between 0 and $C_i$.

3. $G_a$ - Determine the proportion of capital that will be transferred to the child retailer. Let $\alpha$ be the proportion of capital that will be transferred to the child retailer during reproduction. We define

$$\alpha = \frac{G_a}{255}$$

Here $0 \leq G_a \leq 255$. The way in which the capital is transferred is much like that of trucks. Let $c$ be the capital of a retailer. In the case of reproduction, suppose $R1$ and $R2$ are the parent retailers, and $R$ is the new created child retailer. Then the transfers of capital are:

$$R1.c \leftarrow (1 - R1.\alpha) * R1.c$$
\[ R2.c \leftarrow (1 - R2.a) \times R2.c \]
\[ R.c \leftarrow R1.a \times R1.c + R2.a \times R2.c \]

- If \( G_a = 0 \), the retailer does not transfer money to its child.

- If \( G_a = 255 \), the retailer transfers all of its capital to the child.

- If \( 0 < G_a < 100 \), the retailer transfers part of its capital to the child.

The three genes are summarized in Figure 4-2. The total number of retailers can be explored in the search space is

\[ 256 \times 256 \times 256 = 2^{24} = 16777216 \]

<table>
<thead>
<tr>
<th>Gene number</th>
<th>Symbol</th>
<th>Strategy</th>
<th>Value range</th>
<th>Number of values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( G_b )</td>
<td>Buying price of crates</td>
<td>[0, 255]</td>
<td>256</td>
</tr>
<tr>
<td>2</td>
<td>( G_x )</td>
<td>Selling price of items</td>
<td>[0, 255]</td>
<td>256</td>
</tr>
<tr>
<td>3</td>
<td>( G_a )</td>
<td>Capital transfer proportion</td>
<td>[0, 255]</td>
<td>256</td>
</tr>
</tbody>
</table>

**Figure 4-2 Summary of retailer's genes**

We use binary encoding to represent a retailer's solution. The chromosome is a 24-bit binary string. Each gene occupies 8 bits and is interpreted as an unsigned binary number in the range \([0, 255]\).

### 4.2.2 Fitness
Just like trucks, retailers do not have a formulated fitness function either. The fitness of a retailer is the profit it has made during the evaluated generation. How to calculate the profit a retailer has gained in a period of time? We know that a retailer unpacks the crates it has into items. So we assume the stock of a retailer only consists of items. Then the value of a retailer is the sum of its capital and the value of the items in stock (VI). VI is obtained by multiplying the selling price of items by the number of items in stock. The profit is the difference of the retailer’s value between the end and the beginning of the period. If the difference is positive, that means the retailer made a profit in this period. If the difference is zero or negative, the retailer did not earn a profit in this period. The retailer that makes the largest profit has the highest fitness.

4.2.3 Selection

A retailer will be removed from the simulation when it has become bankrupt. All retailers that are not bankrupt are kept for the next generation. Since retailers have a constant population, the reproduction of retailers is strictly controlled.

We also divide the remaining retailers into two parts: healthy retailers and unhealthy retailers. Healthy retailers are the retailers that have made profit in the current generation, and the ones that did not make profit are unhealthy. In other words, healthy retailers have positive fitness values and unhealthy retailers have zero or negative fitness values. Unhealthy retailers are not allowed to reproduce.
A quota is set up to control the number of new retailers can be created. Subtracting the allowed maximum number of retailers by the number of existing retailers yields the quota's value. A new retailer can be born only when the quota's value is greater than or equal to one.

When there is a need to create a new retailer, two parents will be selected randomly from the healthy retailers.

4.2.4 Crossover

Each pair of parent retailers performs crossover and yields a new child retailer. Here we use single-point crossover. A split point is randomly picked, then the first parent retailer copies its bits on the left of and including the split point to the child retailer, and the second parent retailer copies its bits on the right of the split point to the child retailer.

4.2.5 Mutation

Since the population of retailers is small, it is necessary to introduce mutation to achieve diversity. Mutation is performed to each bit of the new chromosome with probability $p$, where $p$ is the mutation rate. A bit is mutated by inverting its value.
5 SIMULATION PARAMETERS

Some parameters are required to be inputted by the user prior to the simulation. They are classified into three categories. These three categories of parameters and their values used in the experiments discussed in Section 6 are discussed below.

The first is the initialization parameters. These parameters are used to initialize the simulation. They include the number and locations of producers, the number and locations of consumers, the number of gas stations, the number of retailers, the storage capacity of retailers, the initial capital and stocks given to each retailer, the initial number of trucks, and the initial capital and stocks given to each truck. We know from Section 3.3 that once created the positions of producers and consumers are fixed during the simulation.

- The number of producers does not influence the amount of goods flowing in the country. Neither does the number of consumers. However, the number and locations of producers and consumers may have impact on the traffic of the country. For example, if the number of producers is small and the producers are located closely, that can easily cause heavy traffic congestions because trucks wanting to buy crates all move to that small area where the producers are located. Increasing the number of producers can enlarge that area. Similarly, distributing the producers across the country widely can effectively prevent the congestion problem when the number of trucks is large. Four producers were established in the “rural districts” near the four corners of the country in
the experiments. There were forty consumers. The consumers near the centre of the country were more than those located near the boundaries, as shown in Figure 5-1.

Figure 5-1 Locations of producers and consumers

- There were eight gas stations in the country.

- From the discussion above, producers, consumers and gas stations have already occupied $4 + 40 + 8 = 52$ intersections. So retailers should be no more than 48 (there are total 100 intersections). The number of retailers in the experiments was set to thirty. All retailers have the same storage capacity that was 8 crates. They started with $3000$ initial capital. No stock was given.
In order to explore more search space, the population size of trucks must be large. However, the country is so small that it actually has become a limitation to the number of trucks that can be developed. In addition, trucks have a variable population. Its size changes with the death of inactive trucks and born of new trucks. Only the initial population size needs to be set. This initial size will be used to calculate the amount of goods flowing through the country per time unit; that is $\lambda v N_c N_t / \mu N_t d$ crates in a supply-driven economy and $\lambda v N_c N_t N_i / \mu N_i d$ items in a demand-driven economy (see Section 3.2). Here, $N_t$ is the initial number of trucks. In the experiments $N_t$ was set to 400. Each truck started with $600$ initial capital and full tank of gas. No stock was given.

The second category is the adaptation parameters. They include how long the adaptation interval will be, the reproduction rate and mutation rate of trucks, the mutation rate of retailers, and whether a new child truck needs to deduct the cost of initial gas from its capital (as described in Section 4.1).

- The adaptation interval should be long enough to let trucks and retailers fully apply their strategies and develop their business. In the experiments the adaptation interval was 4000 simulation time units.
- To control the population size of trucks, the reproduction rate should be kept low to prevent a rapid shooting up. It was 0.2 in the experiments.
- Mutation rates are usually low. 0.05 and 0.04 were used respectively as the mutation rates of trucks and retailers in the experiments. In other words, in average, the
probability of a truck’s chromosome being mutated was $0.05 \times 16 = 0.8$, and the probability of a retailer’s chromosome being mutated was $0.04 \times 24 = 0.96$.

- Child trucks are required to deduct the cost of initial gas from their capital in the experiments.

The third category includes all other parameters: the pay rent interval of retailers and the rent, the average cost of making a phone call, the estimated factor of the length of a round trip $\mu$ (see Section 3.2), the estimated proportion of the total time spent on traveling $\lambda$ (see Section 3.2), the economy type of the simulation and the simulation time.

- Retailers pay rent every a period of time. The rent and interval in the experiments were $\$500$ for every 1000 simulation time units.
- The average cost for a truck making a phone call was $\$2$.
- The estimated factor of the length of a round trip was $3$.
- The estimated proportion of the total time spent on traveling was $0.6$.
- Both of supply-driven and demand-driven economies were experimented with different lengths of simulation time. A length of 400,000 simulation time units (100 generations) will be used as an example for the discussion of the simulation results.

A summary of the parameters that need to be inputted by users is listed in Figure 5-2.

<table>
<thead>
<tr>
<th>Category</th>
<th>Name</th>
<th>Value in the experiments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Number of producers</td>
<td>4</td>
</tr>
<tr>
<td><strong>Initialization parameters</strong></td>
<td><strong>Locations of producers</strong></td>
<td>See Figure 5-1</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-----------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Number of consumers</td>
<td></td>
<td>40</td>
</tr>
<tr>
<td>Locations of consumers</td>
<td></td>
<td>See Figure 5-1</td>
</tr>
<tr>
<td>Number of gas stations</td>
<td></td>
<td>8</td>
</tr>
<tr>
<td>Number of retailers</td>
<td></td>
<td>30</td>
</tr>
<tr>
<td>Storage capacity of each retailer</td>
<td></td>
<td>8 crates</td>
</tr>
<tr>
<td>Initial capital of each retailer</td>
<td></td>
<td>$3000</td>
</tr>
<tr>
<td>Initial stocks given to each retailer</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Initial number of trucks</td>
<td></td>
<td>400</td>
</tr>
<tr>
<td>Initial capital of each truck</td>
<td></td>
<td>$600</td>
</tr>
<tr>
<td>Initial stocks given to each truck</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td><strong>Adaptation parameters</strong></td>
<td>Adaptation interval</td>
<td>4000 time units</td>
</tr>
<tr>
<td>Truck’s reproduction rate</td>
<td>Truck’s reproduction rate</td>
<td>0.2</td>
</tr>
<tr>
<td>Truck’s mutation rate</td>
<td>Truck’s mutation rate</td>
<td>0.05</td>
</tr>
<tr>
<td>Retailer’s mutation rate</td>
<td>Retailer’s mutation rate</td>
<td>0.04</td>
</tr>
<tr>
<td>Does a new child truck deduct the cost of initial gas from its capital?</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>Other</strong></td>
<td>Pay rent interval of retailers</td>
<td>1000 time units</td>
</tr>
<tr>
<td>The rent a retailer pays every pay rent interval</td>
<td></td>
<td>$500</td>
</tr>
<tr>
<td>Average cost of making a phone call</td>
<td></td>
<td>$2</td>
</tr>
<tr>
<td>Estimated factor of a round trip ($\mu$)</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>parameters</td>
<td>Estimated proportion of the total time spent on traveling (λ)</td>
<td>0.6</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------------------------------</td>
<td>-----</td>
</tr>
<tr>
<td>Economy type</td>
<td>Supply-driven / demand-driven</td>
<td></td>
</tr>
<tr>
<td>Simulation time</td>
<td>400,000 time units</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(100 generations)</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 5-2 Summary of parameters to be inputted**

Two measures are adopted to improve the performance of the simulation. One is to set a maximum congestion time. The maximum number of trucks at a same place at the same time is set to 5 in the simulation. When the number of trucks increases, congestion occurs frequently. The worst scenario is that too many congested trucks on the roads will lead to the result that no trucks can move and finally the system will crash. To prevent this situation from happening, a maximum congestion time is used to limit the length of time a truck being congested. If the length of time a truck being congested reaches the maximum limit, this truck will be cleared out of traffic and be removed from the simulation. The maximum congestion time is set to 2000 simulation time units. Another measure is that a truck or a retailer will be removed from the simulation if it has been unprofitable for consecutive 3 generations.
6 SIMULATION RESULTS AND DISCUSSION

To validate the algorithms described in Section 4, the simulation was carried out using a program developed in C++ on Windows platform. To achieve better performance, various parameter values were experimented. The simulation results talked in the remaining part of this section were got by running the simulation using the parameter values of Figure 5-2.

We will first examine the results in a supply-driven economy and then the results in a demand-driven economy. Four kinds of information were recorded at each generation to reflect the evolution of trucks: number of trucks (NT), total profit of trucks (TPT), average profit of trucks (APT) and healthy rate of trucks (HRT). NT is recorded at the beginning of each generation. TPT, APT and HRT are got at the end of the generation.

- NT is the total number of active trucks at the beginning of the generation, which is the number of trucks that survived the previous generation plus the number of new trucks created in the previous adaptation. These trucks will compete to survive and reproduce in the current generation.
- TPT is the sum of each truck’s profit earned in the current generation.
- APT is TPT divided by NT.
- HRT is the ratio of the number of healthy trucks to NT.
Correspondingly, four kinds of information were recorded at each generation to reflect
the evolution of retailers: number of retailers (NR), total profit of retailers (TPR), average
profit of retailers (APR) and healthy rate of retailers (HRR).

6.1 Results of Experiment 1: Supply-driven Economy

We first examine the results of trucks in a run of supply-driven economy. Figure 6-1, 6-2
and 6-3 show the graphs of NT, TPT, APT and HRT.

Let's look at NT first. Its graph is given in all three figures. The number of trucks
dropped at first due to the randomly created initial genomes. With the improvement of
the overall fitness, more and more trucks could survive and more and more new trucks
were born at the same time, pushing the population size going up. It was at a peak at the
35th generation, with 577 trucks. That was very crowded in a country with only 100
intersections. It caused severe traffic congestions. Since trucks that have been congested
for longer than 2000 time units need to be cleared out, part of the trucks was removed for
this reason after a period of time. Moreover, congestions brought poor performance, thus
further decreased the number of surviving trucks and newborn trucks. That is a kind of
over population. So the number of trucks fell straightly. After adjustment, NT went up
again, and down due to the congestions, and oscillated during the remaining part of the
simulation.

51
The graph of TPT is shown in Figure 6-1. Like NT, the total profit fell at first due to the randomly created initial genomes. After that it went up quickly following the rise of NT and reached a peak at the 33rd generation. But TPT dropped sharply when the number of trucks was approaching its highest point. As analyzed in the previous paragraph, a large number of trucks caused heavy traffic congestions and then brought the poor performance. That caused the drop. After that, TPT swung following the changes of NT. When observing carefully, we found that most upward curves of TPT were corresponding to downward curves of NT; most downward curves of TPT were corresponding to upward curves of NT. That means when the number of trucks increases the total profit of trucks is likely to decrease. On the other hand, when the number of trucks decreases the total profit
is likely to increase. From the analysis of NT, the increase of NT brings poor performance of trucks, which causes the shrink of TPT; with the decrease of NT, the effect of traffic congestions lessens, and trucks can fully apply their strategies to make a profit, so TPT is likely to increase.

![Graph showing the relationship between average profit of trucks and number of trucks over generations.](image)

**Figure 6-2 APT and NT in a supply-driven economy**

Figure 6-2 shows the graph of APT. Its trend is similar with the TPT's. The average profit of trucks also decreased at first due to the randomly created initial genomes. It went up from the third generation and continued to rise following the increasing of NT. When there were too many trucks in the country, APT fell as the result of poor performance caused by traffic congestions. After that, like TPT, most upward-downward curves of APT were opposite to those of NT. If NT increases, both the competition between trucks
and the chances of trucks being congested increase, thus the average profit trucks can make is likely to fall. If NT decreases, it lessens the competition and the chances of being congested, thus trucks are likely to be able to make more money.

Figure 6-3 HRT and NT in a supply-driven economy

The graph of HRT is shown in Figure 6-3. As we discussed above, with the improvement of the overall fitness of trucks during the first 33 generations, the healthy rate of trucks improved too, from the lowest value of 0.29 up to the highest value of 0.97. More than 90 per cent of trucks successfully made profit from the 11th generation to the 34th generation. But HRT dropped every time when NT was at a peak. Obviously, poor performance made many of the trucks unprofitable. HRT oscillated after the 36th generation. When the
number of trucks increased, the healthy rate decreased; the healthy rate increased when the number of trucks decreased.

From the three figures above, we know that trucks were performing very well in the first 33 generations. TPT rose nearly constantly. Although there were some downward sloping curves, the overall trends of APT and HRT were going up too. Trucks were becoming more and more profitable. The genetic algorithm really worked during this period. However, the performance of trucks turned unsteady after the over population. The profitability varied following the changes of the population size of trucks. This trend will probably last out the remaining of the simulation.

In addition, there are other factors affecting the performance of trucks: locations and prices of retailers. With some retailers turning into bankruptcy and some others being born, the population of retailers may change. A truck could earn a surprisingly high profit in a generation with many unsmart retailers whose buying prices are very high and whose selling prices are very low. However, the same truck may be unprofitable or even get a negative profit due to the changed environment in another generation. For example, truck t112 made $2579.15 in the 1st generation, but only got $1.89 in the 4th generation, and even worse it lost $164.92 in the 5th generation according to the result data.

Next we examine the results of retailers. The graphs of NR, TPR, APR and HRR in a supply-driven economy are shown in Figure 6-4, 6-5, 6-6 and 6-7.
NR was 30 initially. Most of the retailers went into bankruptcy at the first generation due to the randomly created initial genomes. The population size came back to 30 eventually at the 16th generation. The number of retailers remained at 30 thereafter, since we set an upper limit to the population size of retailers.

TPR and APR developed slowly in the first 17 generations. After the retailer's population size jumped back to 30 and kept at this level, they went up quickly. That is because with the increase of trucks, the amount of deals that retailers got increased too (A condition that the flow of crates and items had not reached the maximum limit applied). Upon over population, most of the trucks were congested on the roads. That led to the starving of the
retailers for not being able to trade to gain profit. Therefore TPR and APR fell sharply. After that, the development of TPR and APR were affected by the changes of NT and thus became unstable. Their figures swung up and down following the changes of the number of trucks.

![Graph of Total profit of retailers vs Generation](image)

**Figure 6-5 TPR in a supply-driven economy**

During the first 34 generations, retailers made more and more money. That suggested that retailers were getting healthier and healthier. Their healthy rate reached 1.0 at the 27th generation and from the 29th to the 34th generation. However, the over population of trucks made many retailers starving and caused the deep drop of HRR. Like TPR and APR, HRR oscillated in the remaining part of the simulation.
By comparing the graphs of total profit, average profit and healthy rate of trucks and retailers, one thing was found that most of the times when a graph of trucks arrived a peak (or a lull), the corresponding graph of retailers also arrived a peak (or a lull) exactly at that generation or at a very close generation. Take for example the total profit. The peak points were at:

TPT: 33, 42, 49, 56, 67, 84, 86, 97, 100

TPR: 32, 42, 49, 56, 67, 84, 86, 96, 100

The lull points were at:

TPT: 37, 45, 52, 64, 78, 85, 94, 99
Finally, to discover the gene structures of the most successful individuals, capital gains in the 100 generations of all remaining trucks and retailers are calculated by subtracting their final capitals by the capitals they got when they were created. The genes of the top 10 capital-gaining trucks are listed in Figure 6-8. The genes of the top 5 capital-gaining retailers are listed in Figure 6-9.

From Figure 6-8, $G_{\text{badbuy}}$ had eight values of ‘1’. Seven out of ten values of $G_{\text{badsell}}$, $G_{\text{reserve}}$ and $G_{\text{corner}}$ were the same, which were ‘1’, ‘2’ and ‘0’ respectively. Value of ‘1’
was excluded from $G_{congestion}$. Genes that have a large value range had more differences. However, they did have common if compared carefully. For example, nine out of ten values of $G_{task}$ were between 50 and 60. Even the last one, 47, was close to 50. Seven of $G_5$’s values were between 73 and 85. Eight of $G_6$’s values were over 220. Eight out of ten values of $G_a$ were less than 10.

<table>
<thead>
<tr>
<th>Genes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_g(0\sim20)$</td>
<td>11</td>
<td>13</td>
<td>17</td>
<td>18</td>
<td>14</td>
<td>12</td>
<td>8</td>
<td>11</td>
<td>8</td>
<td>18</td>
</tr>
<tr>
<td>$G_5(0\sim255)$</td>
<td>73</td>
<td>85</td>
<td>73</td>
<td>73</td>
<td>218</td>
<td>17</td>
<td>75</td>
<td>34</td>
<td>85</td>
<td>73</td>
</tr>
<tr>
<td>$G_6(0\sim255)$</td>
<td>229</td>
<td>224</td>
<td>230</td>
<td>247</td>
<td>252</td>
<td>230</td>
<td>247</td>
<td>229</td>
<td>206</td>
<td>185</td>
</tr>
<tr>
<td>$G_a(0\sim100)$</td>
<td>8</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>13</td>
<td>5</td>
<td>5</td>
<td>8</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>$G_{capacity}(1\sim10)$</td>
<td>10</td>
<td>7</td>
<td>5</td>
<td>8</td>
<td>4</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>$G_{tank}(1\sim60)$</td>
<td>57</td>
<td>55</td>
<td>58</td>
<td>58</td>
<td>47</td>
<td>51</td>
<td>50</td>
<td>58</td>
<td>50</td>
<td>53</td>
</tr>
<tr>
<td>$G_{reserve}(0\sim2)$</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>$G_{corner}(0\sim1)$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$G_{scan}(0\sim1)$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$G_{deal}(0\sim2)$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$G_{priority}(0\sim2)$</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>$G_{best}(0\sim2)$</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$G_{phone}(0\sim2)$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$G_{badself}(0\sim1)$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$G_{badbuy}(0\sim1)$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$G_{congestion}(0\sim2)$</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>
Figure 6-8 Genes of the top 10 capital-gaining trucks in a supply-driven economy

The genes of the top 5 capital-gaining retailers have the similar situation. Among all five of Gb’s values, four were between 80 and 95, with the remaining one just a little less than 80. Four out of five values of Gs were close to each other.

<table>
<thead>
<tr>
<th>Genes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gb(0~255)</td>
<td>90</td>
<td>79</td>
<td>90</td>
<td>82</td>
<td>94</td>
</tr>
<tr>
<td>Gs(0~255)</td>
<td>242</td>
<td>246</td>
<td>242</td>
<td>203</td>
<td>251</td>
</tr>
<tr>
<td>Go(0~255)</td>
<td>21</td>
<td>175</td>
<td>20</td>
<td>16</td>
<td>175</td>
</tr>
</tbody>
</table>

Figure 6-9 Genes of the top 5 capital-gaining retailers in a supply-driven economy

6.2 Results of Experiment 2: Demand-driven Economy

Figure 6-10, 6-11 and 6-12 give the graphs of NT, TPT, APT and HRT in a run of demand-driven economy.

Compared with the graphs in the supply-driven run, the varying range of the truck’s population size in the demand-driven run was larger, which was from 106 to 664. The NT graphs also indicate that on average DDE can support more trucks running in the country.
Figure 6-10 TPT and NT in a demand-driven economy

than SDE. In a supply-driven economy, the number of crates each time a truck can buy from a producer is limited by the manufacture rate. A truck can sell all of the items it carries to a consumer quickly without worrying about the consumer's capability. So trucks in a supply-driven economy likely need to go to producers often. In a demand-driven economy, since the number of items a consumer can consume is limited by the consumption capability, it is usually difficult for a truck to find a consumer who can accept all the items the truck carries at one time. Sometimes a truck needs to visit several consumers in order to sell its items out. On the other hand, a truck can buy whatever amount of crates it can afford from a producer. So trucks in a demand-driven economy likely need to travel a lot among consumers. In our experiments, there were only 4 producers, whereas the number of consumers was 40. That means trucks likely moved

62
towards the 4 producers often in the supply-driven economy, while they likely moved around the 40 consumers often in the demand-driven economy. The chance of causing traffic congestions is much higher in the former situation than in the latter. This is the reason why the highest value of NT in the SDE was only 577 and it reached 664 in the DDE.

![Graph](attachment:image.png)

**Figure 6-11 APT and NT in a demand-driven economy**

More trucks mean more competition. With limited resources and more competitors, trucks in the demand-driven economy did not perform as well as in the supply-driven economy, as seen in Figure 6-10 and 6-11. The highest total profit was only $465,684 at the 31st generation, compared with $935,819 at the 33rd generation in the supply-driven economy.
economy. APT was not able to fully develop following the upward trend of NT, but began to decline from the 17th generation.

![Graph showing Healthy rate of trucks and Number of trucks over generations](image)

**Figure 6-12 HRT and NT in a demand-driven economy**

The results of retailers are shown in Figure 6-13, 6-14, 6-15 and 6-16.

Like in the supply-driven economy, NR was 30 initially. With the bankruptcy of the weak retailers composed of randomly created initial genes, the population size dropped at first. It came back to 30 at the 13th generation.
Although the average amount of deals each truck could make reduced because of the harder competition, the amount of deals that retailers could get did not decrease due to the increased number of trucks. From the figures, we can see that, on average, the profits retailers earned in the demand-driven economy were not less than the profits they earned in the supply-driven economy.

All of the TPR, APR and HRR look better than those in the supply-driven economy. Both of TPR and APR arrived a peak at the 31st generation. HRR remained at 1.0 from the 22nd to 31st generation. Retailers were doing well in the first 31 generations. Although TPR and APR swung up and down repeatedly during the later part of the simulation, their
Figure 6-14 TPR in a demand-driven economy

Figure 6-15 APR in a demand-driven economy
Figure 6-16 HRR in a demand-driven economy

overall trends were still going up. Over 80 per cent of retailers were healthy after the 35th generation.

Just like in the supply-driven economy, the genes of the top 10 capital-gaining trucks are given in Figure 6-17, and the genes of the top 5 capital-gaining retailers in Figure 6-18.

<table>
<thead>
<tr>
<th>Genes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_b(0\sim 20)$</td>
<td>10</td>
<td>14</td>
<td>10</td>
<td>11</td>
<td>19</td>
<td>17</td>
<td>14</td>
<td>19</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td>$G_s(0\sim 255)$</td>
<td>57</td>
<td>44</td>
<td>5</td>
<td>73</td>
<td>20</td>
<td>69</td>
<td>57</td>
<td>20</td>
<td>55</td>
<td>13</td>
</tr>
<tr>
<td>$G_b(0\sim 255)$</td>
<td>229</td>
<td>179</td>
<td>133</td>
<td>179</td>
<td>171</td>
<td>213</td>
<td>243</td>
<td>213</td>
<td>113</td>
<td>179</td>
</tr>
<tr>
<td>Genes</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_a(0\sim100)$</td>
<td>19</td>
<td>4</td>
<td>3</td>
<td>21</td>
<td>3</td>
<td>17</td>
<td>18</td>
<td>4</td>
<td>18</td>
<td>61</td>
</tr>
<tr>
<td>$G_{capacity}(1\sim10)$</td>
<td>6</td>
<td>8</td>
<td>8</td>
<td>10</td>
<td>6</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>$G_{tank}(1\sim60)$</td>
<td>48</td>
<td>60</td>
<td>48</td>
<td>50</td>
<td>35</td>
<td>60</td>
<td>60</td>
<td>54</td>
<td>36</td>
<td>60</td>
</tr>
<tr>
<td>$G_{reserve}(0\sim2)$</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>$G_{corner}(0\sim1)$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$G_{scana}(0\sim1)$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$G_{deal}(0\sim2)$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$G_{priority}(0\sim2)$</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$G_{best}(0\sim2)$</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$G_{phone}(0\sim2)$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$G_{badself}(0\sim1)$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$G_{badbuy}(0\sim1)$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$G_{congestion}(0\sim2)$</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

Figure 6-17 Genes of the top 10 capital-gaining trucks in a demand-driven economy

<table>
<thead>
<tr>
<th>Genes</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_b(0\sim255)$</td>
<td>120</td>
<td>104</td>
<td>158</td>
<td>104</td>
<td>96</td>
</tr>
<tr>
<td>$G_s(0\sim255)$</td>
<td>228</td>
<td>197</td>
<td>226</td>
<td>228</td>
<td>228</td>
</tr>
<tr>
<td>$G_d(0\sim255)$</td>
<td>44</td>
<td>44</td>
<td>177</td>
<td>44</td>
<td>177</td>
</tr>
</tbody>
</table>

Figure 6-18 Genes of the top 5 capital-gaining retailers in a demand-driven economy
The most frequent occurring gene values here are not the same as those observed in the supply-driven economy. Take for example, $G_{\text{priority}}$. Among the ten values in the supply-driven economy, '1' occurred five times, and then '2' had four occurrences. While in the demand-driven economy, six out of ten values were '2' and '1' only occurred twice. Look at $G_b$ of retailers. Its values were mainly between 80 and 90 in the supply-driven economy. However, none of the $G_b$'s values in the demand-driven economy was in this range. This suggests that in different economy types trucks and retailers need to use different strategies to survive and to make the most profit.

Since there are two kinds of agents (trucks and retailers) evolving at the same time and they interact with each other, their evolutions can be complicated. Whether the algorithms used in this thesis can converge has not been proved. Subsequent work will need to be done in this area.
7 CONCLUSION AND FUTURE WORK

In this thesis two genetic algorithms were used to evolve trucks and retailers in a small country. Two economy types were simulated as different forms of limitation to resources. The results indicate that in a supply-driven economy both trucks and retailers were evolving towards better solutions before over population of trucks. While the results of trucks in a demand-driven economy are not as optimistic as in a supply-driven economy, they did have upward slope curves indicating trucks were getting fitter during some periods. The results of retailers in the demand-driven economy reflect the algorithm works too. The results also disclose the significant influence of truck’s population size on the performance of trucks and retailers during the evolution process.

In this project, a real-valued encoding method is used to encode the genome of trucks, and a binary encoding method is used to encode the genome of retailers. All of the three genes of retailers have values ranging from 0 to 255. Each gene can easily be represented as an 8-bit binary string. So binary encoding is suitable for retailers. On the other hand, among the 16 genes of trucks, some have values ranging from 0 to 255, some have values from 1 to 60, and some only have two values. The irregularity makes the real-valued encoding a more suitable method for trucks. Based on these two encoding methods, the selection, crossover, and mutation of trucks and retailers performed well in our simulation. As a result, the genes of trucks and retailers were improving, and individuals became more profitable when the population size of trucks was under control. This indicates that genetic algorithms are suitable for the Trucking problem.
As discussed in the Section 6, over population of trucks inhibited further evolving of both trucks and retailers. So solving this problem will be a major part of future work. A possible solution is to make the "birth rate" match the "death rate" by reducing the number of new trucks born in each generation and/or by increasing the number of trucks that die in each generation. To reduce the number of newborn trucks, we may set the reproduction rate lower. Another way is to add extra requirements to make it more difficult for a truck to be healthy. For example, trucks may be required to pay tax in each generation. We may even improve the standard of being a healthy truck by requiring that a healthy truck must be a truck that has earned at least a certain amount of money in the evaluated generation. Less healthy trucks mean less reproducing parents, so the number of newborn trucks will be less. A way of increasing the number of trucks that die in each generation is to let the trucks that did not make enough money die. Another approach to solve the over population problem is to set an upper limit to the number of active trucks in each generation. Another possible improvement to this problem may be to let trucks smarter by checking the road conditions beforehand.

To improve the results, other improvements to the project include developing better strategies for trucks, developing new diverse strategies for retailers, and varying the adaptation control parameters to find out their most suitable values. Some ideas of retailer's strategies are: making the size of the warehouse a gene of retailers; a retailer may choose whether it needs to reserve money for paying rent and the amount of the reservation.
REFERENCES


APPENDIX A: RESULTS OF TRUCKS IN A RUN UNDER SUPPLY-DRIVEN ECONOMY

The following data are the results of trucks collected from the experiment discussed in Section 6.1.

Description of columns:

A: Generation sequence
B: Average profit of trucks
C: Number of trucks
D: Number of surviving trucks
E: Number of healthy trucks
F: Number of newborn trucks
G: Healthy rate of trucks
H: Total profit of trucks

Results:

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>391.309</td>
<td>400</td>
<td>203</td>
<td>138</td>
<td>13</td>
<td>0.345</td>
<td>156524</td>
</tr>
<tr>
<td>2</td>
<td>156.47</td>
<td>216</td>
<td>154</td>
<td>62</td>
<td>6</td>
<td>0.287037</td>
<td>33797.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>------</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>298.103</td>
<td>160</td>
<td>116</td>
<td>71</td>
<td>7</td>
<td>0.44375</td>
<td>47696.4</td>
</tr>
<tr>
<td>4</td>
<td>456.804</td>
<td>123</td>
<td>82</td>
<td>71</td>
<td>7</td>
<td>0.577236</td>
<td>56186.9</td>
</tr>
<tr>
<td>5</td>
<td>765.162</td>
<td>89</td>
<td>82</td>
<td>72</td>
<td>7</td>
<td>0.808989</td>
<td>68099.4</td>
</tr>
<tr>
<td>6</td>
<td>717.162</td>
<td>89</td>
<td>85</td>
<td>72</td>
<td>7</td>
<td>0.808989</td>
<td>63827.4</td>
</tr>
<tr>
<td>7</td>
<td>1096.58</td>
<td>92</td>
<td>89</td>
<td>81</td>
<td>8</td>
<td>0.880435</td>
<td>100885</td>
</tr>
<tr>
<td>8</td>
<td>1011.02</td>
<td>97</td>
<td>95</td>
<td>87</td>
<td>8</td>
<td>0.896907</td>
<td>98068.8</td>
</tr>
<tr>
<td>9</td>
<td>1244.59</td>
<td>103</td>
<td>98</td>
<td>95</td>
<td>9</td>
<td>0.92233</td>
<td>128193</td>
</tr>
<tr>
<td>10</td>
<td>1028.95</td>
<td>107</td>
<td>101</td>
<td>94</td>
<td>9</td>
<td>0.878505</td>
<td>110098</td>
</tr>
<tr>
<td>11</td>
<td>1224.02</td>
<td>110</td>
<td>109</td>
<td>99</td>
<td>9</td>
<td>0.9</td>
<td>134642</td>
</tr>
<tr>
<td>12</td>
<td>1422.74</td>
<td>118</td>
<td>112</td>
<td>107</td>
<td>10</td>
<td>0.90678</td>
<td>167883</td>
</tr>
<tr>
<td>13</td>
<td>1607.12</td>
<td>122</td>
<td>118</td>
<td>114</td>
<td>11</td>
<td>0.934426</td>
<td>196069</td>
</tr>
<tr>
<td>14</td>
<td>1779.33</td>
<td>129</td>
<td>126</td>
<td>122</td>
<td>12</td>
<td>0.945736</td>
<td>229534</td>
</tr>
<tr>
<td>15</td>
<td>1766.82</td>
<td>138</td>
<td>137</td>
<td>132</td>
<td>13</td>
<td>0.956522</td>
<td>243821</td>
</tr>
<tr>
<td>16</td>
<td>1778.41</td>
<td>150</td>
<td>148</td>
<td>144</td>
<td>14</td>
<td>0.96</td>
<td>266762</td>
</tr>
<tr>
<td>17</td>
<td>1807.61</td>
<td>162</td>
<td>157</td>
<td>154</td>
<td>15</td>
<td>0.950617</td>
<td>292833</td>
</tr>
<tr>
<td>18</td>
<td>1768.07</td>
<td>172</td>
<td>165</td>
<td>164</td>
<td>16</td>
<td>0.953488</td>
<td>304108</td>
</tr>
<tr>
<td>19</td>
<td>1841.38</td>
<td>181</td>
<td>176</td>
<td>17</td>
<td>0.972376</td>
<td>333290</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1896.34</td>
<td>198</td>
<td>194</td>
<td>190</td>
<td>19</td>
<td>0.959596</td>
<td>375476</td>
</tr>
<tr>
<td>21</td>
<td>1893.77</td>
<td>213</td>
<td>208</td>
<td>199</td>
<td>19</td>
<td>0.934272</td>
<td>403373</td>
</tr>
<tr>
<td>22</td>
<td>1957.18</td>
<td>227</td>
<td>220</td>
<td>217</td>
<td>21</td>
<td>0.955947</td>
<td>444280</td>
</tr>
<tr>
<td>23</td>
<td>1868.99</td>
<td>241</td>
<td>236</td>
<td>231</td>
<td>23</td>
<td>0.958506</td>
<td>450426</td>
</tr>
<tr>
<td>24</td>
<td>1856.74</td>
<td>259</td>
<td>255</td>
<td>245</td>
<td>24</td>
<td>0.945946</td>
<td>480895</td>
</tr>
<tr>
<td>25</td>
<td>1919.28</td>
<td>279</td>
<td>276</td>
<td>268</td>
<td>26</td>
<td>0.960573</td>
<td>535480</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>26</td>
<td>1924.76</td>
<td>302</td>
<td>299</td>
<td>291</td>
<td>29</td>
<td>0.963576</td>
<td>581276</td>
</tr>
<tr>
<td>27</td>
<td>1913.94</td>
<td>328</td>
<td>320</td>
<td>309</td>
<td>30</td>
<td>0.942073</td>
<td>627773</td>
</tr>
<tr>
<td>28</td>
<td>1895.38</td>
<td>350</td>
<td>343</td>
<td>334</td>
<td>33</td>
<td>0.954286</td>
<td>663382</td>
</tr>
<tr>
<td>29</td>
<td>1946.9</td>
<td>376</td>
<td>371</td>
<td>360</td>
<td>36</td>
<td>0.957447</td>
<td>732035</td>
</tr>
<tr>
<td>30</td>
<td>1940.34</td>
<td>407</td>
<td>395</td>
<td>387</td>
<td>38</td>
<td>0.95086</td>
<td>789717</td>
</tr>
<tr>
<td>31</td>
<td>1948.71</td>
<td>433</td>
<td>429</td>
<td>418</td>
<td>41</td>
<td>0.965358</td>
<td>843790</td>
</tr>
<tr>
<td>32</td>
<td>1924.7</td>
<td>470</td>
<td>459</td>
<td>449</td>
<td>44</td>
<td>0.955319</td>
<td>904608</td>
</tr>
<tr>
<td>33</td>
<td>1860.47</td>
<td>503</td>
<td>493</td>
<td>477</td>
<td>47</td>
<td>0.94831</td>
<td>935819</td>
</tr>
<tr>
<td>34</td>
<td>1442.94</td>
<td>540</td>
<td>528</td>
<td>499</td>
<td>49</td>
<td>0.924074</td>
<td>779187</td>
</tr>
<tr>
<td>35</td>
<td>697.764</td>
<td>577</td>
<td>344</td>
<td>317</td>
<td>31</td>
<td>0.549393</td>
<td>402610</td>
</tr>
<tr>
<td>36</td>
<td>965.248</td>
<td>375</td>
<td>289</td>
<td>196</td>
<td>19</td>
<td>0.522667</td>
<td>361968</td>
</tr>
<tr>
<td>37</td>
<td>1126.58</td>
<td>308</td>
<td>255</td>
<td>196</td>
<td>19</td>
<td>0.636364</td>
<td>346985</td>
</tr>
<tr>
<td>38</td>
<td>1410.24</td>
<td>274</td>
<td>216</td>
<td>194</td>
<td>19</td>
<td>0.708029</td>
<td>386405</td>
</tr>
<tr>
<td>39</td>
<td>1882.5</td>
<td>235</td>
<td>215</td>
<td>211</td>
<td>21</td>
<td>0.897872</td>
<td>442388</td>
</tr>
<tr>
<td>40</td>
<td>2391.77</td>
<td>236</td>
<td>233</td>
<td>230</td>
<td>23</td>
<td>0.974576</td>
<td>564458</td>
</tr>
<tr>
<td>41</td>
<td>2414.13</td>
<td>256</td>
<td>255</td>
<td>253</td>
<td>25</td>
<td>0.988281</td>
<td>618018</td>
</tr>
<tr>
<td>42</td>
<td>2412.94</td>
<td>280</td>
<td>278</td>
<td>277</td>
<td>27</td>
<td>0.989286</td>
<td>675622</td>
</tr>
<tr>
<td>43</td>
<td>1853.65</td>
<td>305</td>
<td>249</td>
<td>245</td>
<td>24</td>
<td>0.803279</td>
<td>565362</td>
</tr>
<tr>
<td>44</td>
<td>1571.09</td>
<td>273</td>
<td>215</td>
<td>202</td>
<td>20</td>
<td>0.739927</td>
<td>428909</td>
</tr>
<tr>
<td>45</td>
<td>1649.99</td>
<td>235</td>
<td>215</td>
<td>181</td>
<td>18</td>
<td>0.770213</td>
<td>387747</td>
</tr>
<tr>
<td>46</td>
<td>1992.95</td>
<td>233</td>
<td>222</td>
<td>217</td>
<td>21</td>
<td>0.93133</td>
<td>464356</td>
</tr>
<tr>
<td>47</td>
<td>1882.71</td>
<td>243</td>
<td>242</td>
<td>237</td>
<td>23</td>
<td>0.975309</td>
<td>457499</td>
</tr>
<tr>
<td>48</td>
<td>1868.03</td>
<td>265</td>
<td>264</td>
<td>256</td>
<td>25</td>
<td>0.966038</td>
<td>495029</td>
</tr>
<tr>
<td></td>
<td>1762.63</td>
<td>289</td>
<td>282</td>
<td>277</td>
<td>27</td>
<td>0.958478</td>
<td>509399</td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>----</td>
<td>-----------</td>
<td>--------</td>
</tr>
<tr>
<td>50</td>
<td>1351.8</td>
<td>309</td>
<td>300</td>
<td>290</td>
<td>29</td>
<td>0.938511</td>
<td>417707</td>
</tr>
<tr>
<td>51</td>
<td>858.162</td>
<td>329</td>
<td>255</td>
<td>244</td>
<td>24</td>
<td>0.741641</td>
<td>282335</td>
</tr>
<tr>
<td>52</td>
<td>1010.76</td>
<td>279</td>
<td>239</td>
<td>197</td>
<td>19</td>
<td>0.706093</td>
<td>282002</td>
</tr>
<tr>
<td>53</td>
<td>1214.44</td>
<td>258</td>
<td>229</td>
<td>216</td>
<td>21</td>
<td>0.837209</td>
<td>313325</td>
</tr>
<tr>
<td>54</td>
<td>1727.69</td>
<td>250</td>
<td>224</td>
<td>219</td>
<td>21</td>
<td>0.876</td>
<td>431922</td>
</tr>
<tr>
<td>55</td>
<td>2141.75</td>
<td>245</td>
<td>243</td>
<td>238</td>
<td>23</td>
<td>0.971429</td>
<td>524730</td>
</tr>
<tr>
<td>56</td>
<td>2056.4</td>
<td>266</td>
<td>260</td>
<td>251</td>
<td>25</td>
<td>0.943609</td>
<td>547002</td>
</tr>
<tr>
<td>57</td>
<td>1430.54</td>
<td>285</td>
<td>278</td>
<td>268</td>
<td>26</td>
<td>0.940351</td>
<td>407705</td>
</tr>
<tr>
<td>58</td>
<td>1179.19</td>
<td>304</td>
<td>295</td>
<td>277</td>
<td>27</td>
<td>0.911184</td>
<td>358474</td>
</tr>
<tr>
<td>59</td>
<td>1110.47</td>
<td>322</td>
<td>313</td>
<td>294</td>
<td>29</td>
<td>0.913043</td>
<td>357573</td>
</tr>
<tr>
<td>60</td>
<td>1032.28</td>
<td>342</td>
<td>331</td>
<td>310</td>
<td>31</td>
<td>0.906433</td>
<td>353041</td>
</tr>
<tr>
<td>61</td>
<td>955.299</td>
<td>362</td>
<td>347</td>
<td>319</td>
<td>31</td>
<td>0.881215</td>
<td>345818</td>
</tr>
<tr>
<td>62</td>
<td>897.583</td>
<td>378</td>
<td>358</td>
<td>333</td>
<td>33</td>
<td>0.880952</td>
<td>339286</td>
</tr>
<tr>
<td>63</td>
<td>860.386</td>
<td>391</td>
<td>362</td>
<td>332</td>
<td>33</td>
<td>0.849105</td>
<td>336411</td>
</tr>
<tr>
<td>64</td>
<td>412.804</td>
<td>395</td>
<td>268</td>
<td>227</td>
<td>22</td>
<td>0.574684</td>
<td>163058</td>
</tr>
<tr>
<td>65</td>
<td>636.201</td>
<td>290</td>
<td>197</td>
<td>177</td>
<td>17</td>
<td>0.610345</td>
<td>184498</td>
</tr>
<tr>
<td>66</td>
<td>1947.74</td>
<td>214</td>
<td>193</td>
<td>188</td>
<td>18</td>
<td>0.878505</td>
<td>416817</td>
</tr>
<tr>
<td>67</td>
<td>3208.08</td>
<td>211</td>
<td>211</td>
<td>209</td>
<td>20</td>
<td>0.990521</td>
<td>676905</td>
</tr>
<tr>
<td>68</td>
<td>2171</td>
<td>231</td>
<td>230</td>
<td>227</td>
<td>22</td>
<td>0.982684</td>
<td>501501</td>
</tr>
<tr>
<td>69</td>
<td>1483.92</td>
<td>252</td>
<td>251</td>
<td>244</td>
<td>24</td>
<td>0.968254</td>
<td>373948</td>
</tr>
<tr>
<td>70</td>
<td>1342.45</td>
<td>275</td>
<td>273</td>
<td>263</td>
<td>26</td>
<td>0.956364</td>
<td>369175</td>
</tr>
<tr>
<td>71</td>
<td>1214.9</td>
<td>299</td>
<td>291</td>
<td>274</td>
<td>27</td>
<td>0.916388</td>
<td>363254</td>
</tr>
<tr>
<td></td>
<td>1067.19</td>
<td>318</td>
<td>308</td>
<td>293</td>
<td>29</td>
<td>0.921384</td>
<td>339367</td>
</tr>
<tr>
<td>-----</td>
<td>---------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>----------</td>
<td>---------</td>
</tr>
<tr>
<td>73</td>
<td>960.379</td>
<td>337</td>
<td>309</td>
<td>291</td>
<td>29</td>
<td>0.863501</td>
<td>323648</td>
</tr>
<tr>
<td>74</td>
<td>1016.49</td>
<td>338</td>
<td>284</td>
<td>273</td>
<td>27</td>
<td>0.807692</td>
<td>343574</td>
</tr>
<tr>
<td>75</td>
<td>1030.03</td>
<td>311</td>
<td>307</td>
<td>292</td>
<td>29</td>
<td>0.938907</td>
<td>320340</td>
</tr>
<tr>
<td>76</td>
<td>885.308</td>
<td>336</td>
<td>328</td>
<td>305</td>
<td>30</td>
<td>0.907738</td>
<td>297464</td>
</tr>
<tr>
<td>77</td>
<td>675.72</td>
<td>358</td>
<td>315</td>
<td>287</td>
<td>28</td>
<td>0.801676</td>
<td>241908</td>
</tr>
<tr>
<td>78</td>
<td>596.493</td>
<td>343</td>
<td>287</td>
<td>211</td>
<td>21</td>
<td>0.61516</td>
<td>204597</td>
</tr>
<tr>
<td>79</td>
<td>708.886</td>
<td>308</td>
<td>268</td>
<td>205</td>
<td>20</td>
<td>0.665584</td>
<td>218337</td>
</tr>
<tr>
<td>80</td>
<td>756.688</td>
<td>288</td>
<td>229</td>
<td>186</td>
<td>18</td>
<td>0.645833</td>
<td>217926</td>
</tr>
<tr>
<td>81</td>
<td>912.69</td>
<td>247</td>
<td>222</td>
<td>204</td>
<td>20</td>
<td>0.825911</td>
<td>225434</td>
</tr>
<tr>
<td>82</td>
<td>1308.12</td>
<td>242</td>
<td>233</td>
<td>227</td>
<td>22</td>
<td>0.938017</td>
<td>316565</td>
</tr>
<tr>
<td>83</td>
<td>1351.52</td>
<td>255</td>
<td>254</td>
<td>248</td>
<td>24</td>
<td>0.972549</td>
<td>344637</td>
</tr>
<tr>
<td>84</td>
<td>1302.82</td>
<td>278</td>
<td>273</td>
<td>266</td>
<td>26</td>
<td>0.956835</td>
<td>362183</td>
</tr>
<tr>
<td>85</td>
<td>1035.81</td>
<td>299</td>
<td>257</td>
<td>246</td>
<td>24</td>
<td>0.822742</td>
<td>309707</td>
</tr>
<tr>
<td>86</td>
<td>1822.49</td>
<td>281</td>
<td>257</td>
<td>250</td>
<td>25</td>
<td>0.88968</td>
<td>512121</td>
</tr>
<tr>
<td>87</td>
<td>1451.1</td>
<td>282</td>
<td>280</td>
<td>270</td>
<td>27</td>
<td>0.957447</td>
<td>409211</td>
</tr>
<tr>
<td>88</td>
<td>1229.05</td>
<td>307</td>
<td>302</td>
<td>291</td>
<td>29</td>
<td>0.947883</td>
<td>377319</td>
</tr>
<tr>
<td>89</td>
<td>813.652</td>
<td>331</td>
<td>324</td>
<td>298</td>
<td>29</td>
<td>0.900302</td>
<td>269319</td>
</tr>
<tr>
<td>90</td>
<td>725.265</td>
<td>353</td>
<td>343</td>
<td>313</td>
<td>31</td>
<td>0.886686</td>
<td>256019</td>
</tr>
<tr>
<td>91</td>
<td>675.687</td>
<td>374</td>
<td>358</td>
<td>321</td>
<td>32</td>
<td>0.858289</td>
<td>252707</td>
</tr>
<tr>
<td>92</td>
<td>650.202</td>
<td>390</td>
<td>373</td>
<td>344</td>
<td>34</td>
<td>0.882051</td>
<td>253579</td>
</tr>
<tr>
<td>93</td>
<td>477.997</td>
<td>407</td>
<td>347</td>
<td>297</td>
<td>29</td>
<td>0.72973</td>
<td>194545</td>
</tr>
<tr>
<td>94</td>
<td>507.82</td>
<td>376</td>
<td>324</td>
<td>258</td>
<td>25</td>
<td>0.68617</td>
<td>190940</td>
</tr>
<tr>
<td>Year</td>
<td>Value</td>
<td>Year</td>
<td>Value</td>
<td>Year</td>
<td>Value</td>
<td>Year</td>
<td>Value</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>---------</td>
<td>------</td>
<td>---------</td>
</tr>
<tr>
<td>95</td>
<td>577.093</td>
<td>349</td>
<td>292</td>
<td>238</td>
<td>23</td>
<td>0.681948</td>
<td>201405</td>
</tr>
<tr>
<td>96</td>
<td>1166.58</td>
<td>315</td>
<td>266</td>
<td>261</td>
<td>26</td>
<td>0.828571</td>
<td>367471</td>
</tr>
<tr>
<td>97</td>
<td>1807.3</td>
<td>292</td>
<td>257</td>
<td>212</td>
<td>21</td>
<td>0.726027</td>
<td>527733</td>
</tr>
<tr>
<td>98</td>
<td>1125.54</td>
<td>278</td>
<td>247</td>
<td>203</td>
<td>20</td>
<td>0.730216</td>
<td>312900</td>
</tr>
<tr>
<td>99</td>
<td>1104.79</td>
<td>267</td>
<td>248</td>
<td>238</td>
<td>23</td>
<td>0.891386</td>
<td>294979</td>
</tr>
<tr>
<td>100</td>
<td>1567.43</td>
<td>271</td>
<td>268</td>
<td>264</td>
<td>26</td>
<td>0.97417</td>
<td>424774</td>
</tr>
</tbody>
</table>
APPENDIX B: RESULTS OF RETAILERS IN A RUN
UNDER SUPPLY-DRIVEN ECONOMY

The following data are the results of retailers collected from the experiment discussed in Section 6.1.

Description of columns:

A: Generation sequence
B: Average profit of retailers
C: Number of retailers
D: Number of surviving retailers
E: Number of healthy retailers
F: Number of newborn retailers
G: Healthy rate of retailers
H: Total profit of retailers

Results:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>204.41</td>
<td>30</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td>0.0666667</td>
<td>6132.31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2674.03</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>0.571429</td>
<td>18718.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td>3</td>
<td>2730.01</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>2</td>
<td>0.714286</td>
<td>19110.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1698.78</td>
<td>9</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>0.555556</td>
<td>15289</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1097.79</td>
<td>10</td>
<td>6</td>
<td>5</td>
<td>2</td>
<td>0.5</td>
<td>10977.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2547.51</td>
<td>8</td>
<td>7</td>
<td>5</td>
<td>2</td>
<td>0.625</td>
<td>20380.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1914.24</td>
<td>9</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>0.555556</td>
<td>17228.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>2593.04</td>
<td>10</td>
<td>8</td>
<td>6</td>
<td>3</td>
<td>0.6</td>
<td>25930.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1688.89</td>
<td>11</td>
<td>9</td>
<td>6</td>
<td>3</td>
<td>0.545455</td>
<td>18577.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2310.36</td>
<td>12</td>
<td>11</td>
<td>8</td>
<td>4</td>
<td>0.666667</td>
<td>27724.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>2080.96</td>
<td>15</td>
<td>12</td>
<td>9</td>
<td>4</td>
<td>0.6</td>
<td>31214.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>2369.71</td>
<td>16</td>
<td>14</td>
<td>10</td>
<td>5</td>
<td>0.625</td>
<td>37915.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2421.3</td>
<td>19</td>
<td>17</td>
<td>12</td>
<td>6</td>
<td>0.631579</td>
<td>46004.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1785.59</td>
<td>23</td>
<td>20</td>
<td>15</td>
<td>7</td>
<td>0.652174</td>
<td>41068.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1885.97</td>
<td>27</td>
<td>23</td>
<td>18</td>
<td>7</td>
<td>0.666667</td>
<td>50921.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1859.93</td>
<td>30</td>
<td>28</td>
<td>19</td>
<td>2</td>
<td>0.633333</td>
<td>55798</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>1765</td>
<td>30</td>
<td>28</td>
<td>18</td>
<td>2</td>
<td>0.6</td>
<td>52950</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2609.56</td>
<td>30</td>
<td>24</td>
<td>21</td>
<td>6</td>
<td>0.7</td>
<td>78286.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>3201.55</td>
<td>30</td>
<td>26</td>
<td>23</td>
<td>4</td>
<td>0.766667</td>
<td>96046.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>3504.95</td>
<td>30</td>
<td>30</td>
<td>24</td>
<td>0</td>
<td>0.8</td>
<td>105148</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>3480</td>
<td>30</td>
<td>28</td>
<td>26</td>
<td>2</td>
<td>0.866667</td>
<td>104400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>4137.96</td>
<td>30</td>
<td>28</td>
<td>27</td>
<td>2</td>
<td>0.9</td>
<td>124139</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>5179.57</td>
<td>30</td>
<td>30</td>
<td>28</td>
<td>0</td>
<td>0.933333</td>
<td>155387</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>5848.36</td>
<td>30</td>
<td>29</td>
<td>28</td>
<td>1</td>
<td>0.933333</td>
<td>175451</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>6627.02</td>
<td>30</td>
<td>29</td>
<td>29</td>
<td>1</td>
<td>0.966667</td>
<td>198811</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
</tr>
<tr>
<td>---</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td>6986.79</td>
<td>8264.14</td>
<td>8753.56</td>
<td>9916.46</td>
<td>11109.3</td>
<td>12200.3</td>
<td>13195.6</td>
<td>13104.9</td>
<td>11993.7</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>30</td>
<td>29</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0.966667</td>
<td>1</td>
<td>0.966667</td>
<td>1</td>
<td>0.966667</td>
<td>1</td>
<td>0.966667</td>
<td>1</td>
<td>0.966667</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>209604</td>
<td>247924</td>
<td>262607</td>
<td>297494</td>
<td>333278</td>
<td>366008</td>
<td>395868</td>
<td>393147</td>
<td>359810</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>4822.9</td>
<td>30</td>
<td>29</td>
<td>25</td>
<td>1</td>
<td>0.833333</td>
<td>144687</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>3844.76</td>
<td>30</td>
<td>29</td>
<td>22</td>
<td>1</td>
<td>0.733333</td>
<td>115343</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>2793.96</td>
<td>30</td>
<td>27</td>
<td>18</td>
<td>3</td>
<td>0.6</td>
<td>83818.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>2711.61</td>
<td>30</td>
<td>26</td>
<td>17</td>
<td>4</td>
<td>0.566667</td>
<td>81348.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>3255.19</td>
<td>30</td>
<td>27</td>
<td>20</td>
<td>3</td>
<td>0.666667</td>
<td>97655.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>4407.47</td>
<td>30</td>
<td>27</td>
<td>20</td>
<td>3</td>
<td>0.666667</td>
<td>132224</td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>5158.13</td>
<td>30</td>
<td>26</td>
<td>20</td>
<td>4</td>
<td>0.666667</td>
<td>154744</td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>5745.81</td>
<td>30</td>
<td>28</td>
<td>21</td>
<td>2</td>
<td>0.7</td>
<td>172374</td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>4659.15</td>
<td>30</td>
<td>28</td>
<td>21</td>
<td>2</td>
<td>0.7</td>
<td>139774</td>
<td></td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>4495.51</td>
<td>30</td>
<td>26</td>
<td>22</td>
<td>4</td>
<td>0.733333</td>
<td>134865</td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>4205.34</td>
<td>30</td>
<td>28</td>
<td>21</td>
<td>2</td>
<td>0.7</td>
<td>126160</td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>4270.85</td>
<td>30</td>
<td>28</td>
<td>21</td>
<td>2</td>
<td>0.7</td>
<td>128125</td>
<td></td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>4111.94</td>
<td>30</td>
<td>26</td>
<td>21</td>
<td>4</td>
<td>0.7</td>
<td>123358</td>
<td></td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>4110.71</td>
<td>30</td>
<td>27</td>
<td>21</td>
<td>3</td>
<td>0.7</td>
<td>123321</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>4206.02</td>
<td>30</td>
<td>28</td>
<td>22</td>
<td>2</td>
<td>0.733333</td>
<td>126180</td>
<td></td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>2780.71</td>
<td>30</td>
<td>26</td>
<td>16</td>
<td>4</td>
<td>0.533333</td>
<td>83421.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>2392.98</td>
<td>30</td>
<td>28</td>
<td>14</td>
<td>2</td>
<td>0.466667</td>
<td>71789.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>5158.46</td>
<td>30</td>
<td>27</td>
<td>23</td>
<td>3</td>
<td>0.766667</td>
<td>154754</td>
<td></td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>8109.29</td>
<td>30</td>
<td>28</td>
<td>25</td>
<td>2</td>
<td>0.833333</td>
<td>243279</td>
<td></td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>6032.08</td>
<td>30</td>
<td>29</td>
<td>23</td>
<td>1</td>
<td>0.766667</td>
<td>180962</td>
<td></td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>4826.69</td>
<td>30</td>
<td>28</td>
<td>22</td>
<td>2</td>
<td>0.733333</td>
<td>144801</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>4565.87</td>
<td>30</td>
<td>26</td>
<td>19</td>
<td>4</td>
<td>0.633333</td>
<td>136976</td>
<td></td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>4292.83</td>
<td>30</td>
<td>28</td>
<td>20</td>
<td>2</td>
<td>0.666667</td>
<td>128785</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

83
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>4867.93</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>5</td>
</tr>
<tr>
<td>73</td>
<td>5660.77</td>
<td>30</td>
<td>27</td>
<td>20</td>
<td>3</td>
</tr>
<tr>
<td>74</td>
<td>5714.22</td>
<td>30</td>
<td>29</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>75</td>
<td>5504.04</td>
<td>30</td>
<td>25</td>
<td>21</td>
<td>5</td>
</tr>
<tr>
<td>76</td>
<td>5873.74</td>
<td>30</td>
<td>28</td>
<td>22</td>
<td>2</td>
</tr>
<tr>
<td>77</td>
<td>5448.71</td>
<td>30</td>
<td>29</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>78</td>
<td>4204.75</td>
<td>30</td>
<td>25</td>
<td>18</td>
<td>5</td>
</tr>
<tr>
<td>79</td>
<td>4357.42</td>
<td>30</td>
<td>28</td>
<td>18</td>
<td>2</td>
</tr>
<tr>
<td>80</td>
<td>4388.2</td>
<td>30</td>
<td>26</td>
<td>20</td>
<td>4</td>
</tr>
<tr>
<td>81</td>
<td>4522.18</td>
<td>30</td>
<td>26</td>
<td>19</td>
<td>4</td>
</tr>
<tr>
<td>82</td>
<td>7019.09</td>
<td>30</td>
<td>28</td>
<td>23</td>
<td>2</td>
</tr>
<tr>
<td>83</td>
<td>7726.72</td>
<td>30</td>
<td>27</td>
<td>24</td>
<td>3</td>
</tr>
<tr>
<td>84</td>
<td>8465.51</td>
<td>30</td>
<td>27</td>
<td>25</td>
<td>3</td>
</tr>
<tr>
<td>85</td>
<td>6347.48</td>
<td>30</td>
<td>30</td>
<td>21</td>
<td>0</td>
</tr>
<tr>
<td>86</td>
<td>9038.32</td>
<td>30</td>
<td>28</td>
<td>26</td>
<td>2</td>
</tr>
<tr>
<td>87</td>
<td>8581.19</td>
<td>30</td>
<td>28</td>
<td>26</td>
<td>2</td>
</tr>
<tr>
<td>88</td>
<td>8502.64</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
</tr>
<tr>
<td>89</td>
<td>6750.76</td>
<td>30</td>
<td>30</td>
<td>25</td>
<td>0</td>
</tr>
<tr>
<td>90</td>
<td>6724.2</td>
<td>30</td>
<td>29</td>
<td>26</td>
<td>1</td>
</tr>
<tr>
<td>91</td>
<td>6720.56</td>
<td>30</td>
<td>28</td>
<td>25</td>
<td>2</td>
</tr>
<tr>
<td>92</td>
<td>6620.68</td>
<td>30</td>
<td>30</td>
<td>27</td>
<td>0</td>
</tr>
<tr>
<td>93</td>
<td>4606.49</td>
<td>30</td>
<td>27</td>
<td>19</td>
<td>3</td>
</tr>
<tr>
<td>94</td>
<td>4442.07</td>
<td>30</td>
<td>30</td>
<td>21</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>--------</td>
</tr>
<tr>
<td>95</td>
<td>4434.2</td>
<td>30</td>
<td>24</td>
<td>20</td>
<td>6</td>
</tr>
<tr>
<td>96</td>
<td>5488.38</td>
<td>30</td>
<td>28</td>
<td>20</td>
<td>2</td>
</tr>
<tr>
<td>97</td>
<td>3755.24</td>
<td>30</td>
<td>29</td>
<td>22</td>
<td>1</td>
</tr>
<tr>
<td>98</td>
<td>3817.34</td>
<td>30</td>
<td>24</td>
<td>17</td>
<td>6</td>
</tr>
<tr>
<td>99</td>
<td>5301.11</td>
<td>30</td>
<td>29</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>100</td>
<td>9172.85</td>
<td>30</td>
<td>28</td>
<td>26</td>
<td>2</td>
</tr>
</tbody>
</table>
APPENDIX C: RESULTS OF TRUCKS IN A RUN UNDER DEMAND-DRIVEN ECONOMY

The following data are the results of trucks collected from the experiment discussed in Section 6.2.

Description of columns:

A: Generation sequence
B: Average profit of trucks
C: Number of trucks
D: Number of surviving trucks
E: Number of healthy trucks
F: Number of newborn trucks
G: Healthy rate of trucks
H: Total profit of trucks

Results:

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>346.889</td>
<td>400</td>
<td>206</td>
<td>139</td>
<td>13</td>
<td>0.3475</td>
<td></td>
<td>138756</td>
</tr>
<tr>
<td>2</td>
<td>68.07</td>
<td>219</td>
<td>154</td>
<td>61</td>
<td>6</td>
<td>0.278539</td>
<td></td>
<td>14907.3</td>
</tr>
<tr>
<td></td>
<td>180.942</td>
<td>160</td>
<td>124</td>
<td>67</td>
<td>6</td>
<td>0.41875</td>
<td>28950.7</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>-----</td>
<td>----------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>424.623</td>
<td>130</td>
<td>98</td>
<td>83</td>
<td>8</td>
<td>0.638462</td>
<td>55200.9</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>833.673</td>
<td>106</td>
<td>98</td>
<td>92</td>
<td>9</td>
<td>0.867925</td>
<td>88369.4</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1123.12</td>
<td>107</td>
<td>106</td>
<td>100</td>
<td>10</td>
<td>0.934579</td>
<td>120174</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1023.5</td>
<td>116</td>
<td>109</td>
<td>105</td>
<td>10</td>
<td>0.905172</td>
<td>118726</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>864.109</td>
<td>119</td>
<td>115</td>
<td>97</td>
<td>9</td>
<td>0.815126</td>
<td>102829</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>792.194</td>
<td>124</td>
<td>119</td>
<td>96</td>
<td>9</td>
<td>0.774194</td>
<td>98232</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1008.62</td>
<td>128</td>
<td>119</td>
<td>104</td>
<td>10</td>
<td>0.8125</td>
<td>129104</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>1592.88</td>
<td>129</td>
<td>126</td>
<td>125</td>
<td>12</td>
<td>0.968992</td>
<td>205481</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1808.22</td>
<td>138</td>
<td>138</td>
<td>136</td>
<td>13</td>
<td>0.985507</td>
<td>249535</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>1547.07</td>
<td>151</td>
<td>149</td>
<td>140</td>
<td>14</td>
<td>0.927152</td>
<td>233608</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>1825.19</td>
<td>163</td>
<td>162</td>
<td>153</td>
<td>15</td>
<td>0.93865</td>
<td>297505</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>1756.61</td>
<td>177</td>
<td>171</td>
<td>168</td>
<td>16</td>
<td>0.949153</td>
<td>310919</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>1778.07</td>
<td>187</td>
<td>187</td>
<td>181</td>
<td>18</td>
<td>0.967914</td>
<td>332500</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>1410.12</td>
<td>205</td>
<td>203</td>
<td>192</td>
<td>19</td>
<td>0.936585</td>
<td>289074</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>1430.71</td>
<td>222</td>
<td>218</td>
<td>212</td>
<td>21</td>
<td>0.954955</td>
<td>317617</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1301.72</td>
<td>239</td>
<td>235</td>
<td>221</td>
<td>22</td>
<td>0.924686</td>
<td>311110</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>1322.57</td>
<td>257</td>
<td>254</td>
<td>243</td>
<td>24</td>
<td>0.945525</td>
<td>339899</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>1197.45</td>
<td>278</td>
<td>269</td>
<td>255</td>
<td>25</td>
<td>0.917266</td>
<td>332891</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>1241.17</td>
<td>294</td>
<td>287</td>
<td>277</td>
<td>27</td>
<td>0.942177</td>
<td>364903</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>1173.06</td>
<td>314</td>
<td>306</td>
<td>297</td>
<td>29</td>
<td>0.94586</td>
<td>368342</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>1151.29</td>
<td>335</td>
<td>331</td>
<td>318</td>
<td>31</td>
<td>0.949254</td>
<td>385683</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1171.81</td>
<td>362</td>
<td>357</td>
<td>341</td>
<td>34</td>
<td>0.941989</td>
<td>424197</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>----------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>1124.16</td>
<td>391</td>
<td>385</td>
<td>371</td>
<td>37</td>
<td>0.948849</td>
<td>439546</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>1013.82</td>
<td>422</td>
<td>409</td>
<td>383</td>
<td>38</td>
<td>0.907583</td>
<td>427830</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>983.598</td>
<td>447</td>
<td>440</td>
<td>427</td>
<td>42</td>
<td>0.953257</td>
<td>439668</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>937.422</td>
<td>482</td>
<td>467</td>
<td>441</td>
<td>44</td>
<td>0.914938</td>
<td>451837</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>908.778</td>
<td>511</td>
<td>503</td>
<td>468</td>
<td>46</td>
<td>0.915851</td>
<td>464386</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>848.24</td>
<td>549</td>
<td>517</td>
<td>496</td>
<td>49</td>
<td>0.903461</td>
<td>465684</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>260.5</td>
<td>566</td>
<td>195</td>
<td>112</td>
<td>11</td>
<td>0.19788</td>
<td>147443</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>787.888</td>
<td>206</td>
<td>158</td>
<td>101</td>
<td>10</td>
<td>0.490291</td>
<td>162305</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>951.854</td>
<td>168</td>
<td>124</td>
<td>100</td>
<td>10</td>
<td>0.595238</td>
<td>159912</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>1709.35</td>
<td>134</td>
<td>130</td>
<td>129</td>
<td>12</td>
<td>0.962687</td>
<td>229053</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>1720.57</td>
<td>142</td>
<td>141</td>
<td>139</td>
<td>13</td>
<td>0.978873</td>
<td>244321</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>1718.56</td>
<td>154</td>
<td>152</td>
<td>151</td>
<td>15</td>
<td>0.980519</td>
<td>264658</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>1740.08</td>
<td>167</td>
<td>165</td>
<td>163</td>
<td>16</td>
<td>0.976048</td>
<td>290594</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>1739.5</td>
<td>181</td>
<td>179</td>
<td>177</td>
<td>17</td>
<td>0.977901</td>
<td>314849</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>1602.1</td>
<td>196</td>
<td>196</td>
<td>192</td>
<td>19</td>
<td>0.979592</td>
<td>314011</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>1635.68</td>
<td>215</td>
<td>212</td>
<td>210</td>
<td>21</td>
<td>0.976744</td>
<td>351670</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>1601.41</td>
<td>233</td>
<td>231</td>
<td>228</td>
<td>22</td>
<td>0.978541</td>
<td>373130</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>1559.39</td>
<td>253</td>
<td>250</td>
<td>248</td>
<td>24</td>
<td>0.980237</td>
<td>394526</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>1525.76</td>
<td>274</td>
<td>270</td>
<td>267</td>
<td>26</td>
<td>0.974453</td>
<td>418059</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>1504.92</td>
<td>296</td>
<td>295</td>
<td>291</td>
<td>29</td>
<td>0.983108</td>
<td>445457</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>1430.32</td>
<td>324</td>
<td>319</td>
<td>308</td>
<td>30</td>
<td>0.950617</td>
<td>463424</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>1331.13</td>
<td>349</td>
<td>346</td>
<td>338</td>
<td>33</td>
<td>0.968481</td>
<td>464564</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>1198.98</td>
<td>379</td>
<td>371</td>
<td>361</td>
<td>36</td>
<td>0.952507</td>
<td>454414</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>1125.86</td>
<td>407</td>
<td>401</td>
<td>373</td>
<td>37</td>
<td>0.916462</td>
<td>458225</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>1011.83</td>
<td>438</td>
<td>429</td>
<td>404</td>
<td>40</td>
<td>0.922374</td>
<td>443182</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>954.891</td>
<td>469</td>
<td>455</td>
<td>440</td>
<td>44</td>
<td>0.938166</td>
<td>447844</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>913.754</td>
<td>499</td>
<td>486</td>
<td>461</td>
<td>46</td>
<td>0.923848</td>
<td>455963</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>852.274</td>
<td>532</td>
<td>520</td>
<td>487</td>
<td>48</td>
<td>0.915414</td>
<td>453410</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>811.511</td>
<td>568</td>
<td>553</td>
<td>514</td>
<td>51</td>
<td>0.90493</td>
<td>460938</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>770.924</td>
<td>604</td>
<td>584</td>
<td>552</td>
<td>55</td>
<td>0.913907</td>
<td>465638</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>678.707</td>
<td>639</td>
<td>609</td>
<td>559</td>
<td>55</td>
<td>0.874804</td>
<td>433694</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>400.913</td>
<td>664</td>
<td>454</td>
<td>421</td>
<td>42</td>
<td>0.634036</td>
<td>266206</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>493.317</td>
<td>496</td>
<td>402</td>
<td>237</td>
<td>23</td>
<td>0.477823</td>
<td>244685</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>606.124</td>
<td>425</td>
<td>370</td>
<td>243</td>
<td>24</td>
<td>0.571765</td>
<td>257603</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>675.114</td>
<td>394</td>
<td>287</td>
<td>244</td>
<td>24</td>
<td>0.619289</td>
<td>265995</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>1038.81</td>
<td>311</td>
<td>304</td>
<td>295</td>
<td>29</td>
<td>0.948553</td>
<td>323069</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>994.539</td>
<td>333</td>
<td>331</td>
<td>318</td>
<td>31</td>
<td>0.954955</td>
<td>331181</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>957.501</td>
<td>362</td>
<td>355</td>
<td>343</td>
<td>34</td>
<td>0.947514</td>
<td>346615</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>875.754</td>
<td>389</td>
<td>378</td>
<td>361</td>
<td>36</td>
<td>0.928021</td>
<td>340668</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>862.443</td>
<td>414</td>
<td>405</td>
<td>391</td>
<td>39</td>
<td>0.944444</td>
<td>357051</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>759.231</td>
<td>444</td>
<td>432</td>
<td>412</td>
<td>41</td>
<td>0.927928</td>
<td>337099</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>651.141</td>
<td>473</td>
<td>465</td>
<td>438</td>
<td>43</td>
<td>0.926004</td>
<td>307990</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>497.787</td>
<td>508</td>
<td>410</td>
<td>384</td>
<td>38</td>
<td>0.755906</td>
<td>252876</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>624.707</td>
<td>448</td>
<td>369</td>
<td>284</td>
<td>28</td>
<td>0.633929</td>
<td>279869</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>714.444</td>
<td>397</td>
<td>351</td>
<td>300</td>
<td>30</td>
<td>0.755668</td>
<td>283634</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>761.201</td>
<td>381</td>
<td>331</td>
<td>318</td>
<td>31</td>
<td>0.834646</td>
<td>290018</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>867.526</td>
<td>362</td>
<td>349</td>
<td>344</td>
<td>34</td>
<td>0.950276</td>
<td>314044</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>762.042</td>
<td>383</td>
<td>376</td>
<td>356</td>
<td>35</td>
<td>0.929504</td>
<td>291862</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>573.885</td>
<td>411</td>
<td>402</td>
<td>372</td>
<td>37</td>
<td>0.905109</td>
<td>235867</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td>503.839</td>
<td>439</td>
<td>425</td>
<td>396</td>
<td>39</td>
<td>0.90205</td>
<td>221185</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>447.458</td>
<td>464</td>
<td>382</td>
<td>335</td>
<td>33</td>
<td>0.721983</td>
<td>207621</td>
<td></td>
</tr>
<tr>
<td>77</td>
<td>517.172</td>
<td>415</td>
<td>363</td>
<td>270</td>
<td>27</td>
<td>0.650602</td>
<td>214626</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td>546.911</td>
<td>390</td>
<td>341</td>
<td>278</td>
<td>27</td>
<td>0.712821</td>
<td>213295</td>
<td></td>
</tr>
<tr>
<td>79</td>
<td>628.858</td>
<td>368</td>
<td>315</td>
<td>277</td>
<td>27</td>
<td>0.752717</td>
<td>231420</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>680.329</td>
<td>342</td>
<td>311</td>
<td>283</td>
<td>28</td>
<td>0.827485</td>
<td>232672</td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>708.012</td>
<td>339</td>
<td>330</td>
<td>315</td>
<td>31</td>
<td>0.929204</td>
<td>240016</td>
<td></td>
</tr>
<tr>
<td>82</td>
<td>662.667</td>
<td>361</td>
<td>359</td>
<td>336</td>
<td>33</td>
<td>0.930748</td>
<td>239223</td>
<td></td>
</tr>
<tr>
<td>83</td>
<td>539.688</td>
<td>392</td>
<td>283</td>
<td>271</td>
<td>27</td>
<td>0.691327</td>
<td>211558</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td>755.023</td>
<td>310</td>
<td>255</td>
<td>195</td>
<td>19</td>
<td>0.629032</td>
<td>234057</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>823.896</td>
<td>274</td>
<td>246</td>
<td>197</td>
<td>19</td>
<td>0.718978</td>
<td>225747</td>
<td></td>
</tr>
<tr>
<td>86</td>
<td>1009.85</td>
<td>265</td>
<td>239</td>
<td>234</td>
<td>23</td>
<td>0.883019</td>
<td>267611</td>
<td></td>
</tr>
<tr>
<td>87</td>
<td>1191.42</td>
<td>262</td>
<td>260</td>
<td>252</td>
<td>25</td>
<td>0.961832</td>
<td>312153</td>
<td></td>
</tr>
<tr>
<td>88</td>
<td>975.667</td>
<td>285</td>
<td>282</td>
<td>273</td>
<td>27</td>
<td>0.957895</td>
<td>278065</td>
<td></td>
</tr>
<tr>
<td>89</td>
<td>860.146</td>
<td>309</td>
<td>304</td>
<td>295</td>
<td>29</td>
<td>0.954693</td>
<td>265785</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>804.752</td>
<td>333</td>
<td>330</td>
<td>324</td>
<td>32</td>
<td>0.972973</td>
<td>267982</td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>743.495</td>
<td>362</td>
<td>360</td>
<td>343</td>
<td>34</td>
<td>0.947514</td>
<td>269145</td>
<td></td>
</tr>
<tr>
<td>92</td>
<td>607.926</td>
<td>394</td>
<td>386</td>
<td>370</td>
<td>37</td>
<td>0.939086</td>
<td>239523</td>
<td></td>
</tr>
<tr>
<td>93</td>
<td>491.288</td>
<td>423</td>
<td>417</td>
<td>385</td>
<td>38</td>
<td>0.910165</td>
<td>207815</td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>432.39</td>
<td>455</td>
<td>447</td>
<td>416</td>
<td>41</td>
<td>0.914286</td>
<td>196737</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>388.524</td>
<td>488</td>
<td>471</td>
<td>434</td>
<td>43</td>
<td>0.889344</td>
<td>189600</td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>354.181</td>
<td>514</td>
<td>503</td>
<td>451</td>
<td>45</td>
<td>0.877432</td>
<td>182049</td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>320.334</td>
<td>548</td>
<td>533</td>
<td>475</td>
<td>47</td>
<td>0.866788</td>
<td>175543</td>
<td></td>
</tr>
<tr>
<td>98</td>
<td>282.284</td>
<td>580</td>
<td>565</td>
<td>494</td>
<td>49</td>
<td>0.851724</td>
<td>163725</td>
<td></td>
</tr>
<tr>
<td>99</td>
<td>264.098</td>
<td>614</td>
<td>589</td>
<td>524</td>
<td>52</td>
<td>0.85342</td>
<td>162156</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>286.811</td>
<td>641</td>
<td>517</td>
<td>431</td>
<td>43</td>
<td>0.672387</td>
<td>183846</td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX D: RESULTS OF RETAILERS IN A RUN

UNDER DEMAND-DRIVEN ECONOMY

The following data are the results of retailers collected from the experiment discussed in Section 6.2.

Description of columns:

A: Generation sequence  
B: Average profit of retailers  
C: Number of retailers  
D: Number of surviving retailers  
E: Number of healthy retailers  
F: Number of newborn retailers  
G: Healthy rate of retailers  
H: Total profit of retailers

Results:

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
<td>F</td>
<td>G</td>
<td>H</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>----</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>----</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-48.0627</td>
<td>30</td>
<td>9</td>
<td>3</td>
<td>1</td>
<td>0.1</td>
<td></td>
<td>-1441.88</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>829.685</td>
<td>10</td>
<td>6</td>
<td>6</td>
<td>3</td>
<td>0.6</td>
<td></td>
<td>8296.85</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>-----------</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>240.395</td>
<td>9</td>
<td>8</td>
<td>3</td>
<td>1</td>
<td>0.333333</td>
<td>2163.56</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>990.331</td>
<td>9</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>0.555556</td>
<td>8912.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1086.81</td>
<td>10</td>
<td>9</td>
<td>7</td>
<td>3</td>
<td>0.7</td>
<td>10868.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1588.21</td>
<td>12</td>
<td>11</td>
<td>5</td>
<td>2</td>
<td>0.416667</td>
<td>19058.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1251.31</td>
<td>13</td>
<td>12</td>
<td>7</td>
<td>3</td>
<td>0.538462</td>
<td>16267</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1841.31</td>
<td>15</td>
<td>11</td>
<td>8</td>
<td>4</td>
<td>0.533333</td>
<td>27619.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1912.89</td>
<td>15</td>
<td>11</td>
<td>10</td>
<td>5</td>
<td>0.666666</td>
<td>28693.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2309.8</td>
<td>16</td>
<td>15</td>
<td>10</td>
<td>5</td>
<td>0.625</td>
<td>36956.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>2866.17</td>
<td>20</td>
<td>19</td>
<td>14</td>
<td>7</td>
<td>0.7</td>
<td>57323.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>2986.95</td>
<td>26</td>
<td>22</td>
<td>17</td>
<td>8</td>
<td>0.653846</td>
<td>77660.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2750.65</td>
<td>30</td>
<td>28</td>
<td>21</td>
<td>2</td>
<td>0.7</td>
<td>82519.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2659.71</td>
<td>30</td>
<td>30</td>
<td>22</td>
<td>0</td>
<td>0.733333</td>
<td>79791.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>3005.93</td>
<td>30</td>
<td>24</td>
<td>21</td>
<td>6</td>
<td>0.7</td>
<td>90177.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>2548.01</td>
<td>30</td>
<td>29</td>
<td>23</td>
<td>1</td>
<td>0.766666</td>
<td>76440.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>3458.45</td>
<td>30</td>
<td>30</td>
<td>25</td>
<td>0</td>
<td>0.833333</td>
<td>103754</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>3958.92</td>
<td>30</td>
<td>26</td>
<td>26</td>
<td>4</td>
<td>0.866666</td>
<td>118768</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>4423.7</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966666</td>
<td>132711</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>5062.45</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966666</td>
<td>151874</td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>5518.26</td>
<td>30</td>
<td>29</td>
<td>29</td>
<td>1</td>
<td>0.966666</td>
<td>165548</td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>5972.5</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>179175</td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>6314.68</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>189440</td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>6631.07</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>198932</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>7994.04</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>239821</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>-----</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>8467.31</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>254019</td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>8371.26</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>251138</td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>8679.86</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>260396</td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>8928.23</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>267847</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>9444.43</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>283333</td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>10014.2</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>300425</td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>2942.76</td>
<td>30</td>
<td>30</td>
<td>22</td>
<td>0</td>
<td>0.733333</td>
<td>88282.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>2834.09</td>
<td>30</td>
<td>30</td>
<td>19</td>
<td>0</td>
<td>0.633333</td>
<td>85022.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>2842.7</td>
<td>30</td>
<td>23</td>
<td>19</td>
<td>7</td>
<td>0.633333</td>
<td>85281</td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>4577.35</td>
<td>30</td>
<td>30</td>
<td>24</td>
<td>0</td>
<td>0.8</td>
<td>137321</td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>4342.37</td>
<td>30</td>
<td>29</td>
<td>25</td>
<td>1</td>
<td>0.833333</td>
<td>130271</td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>4670.37</td>
<td>30</td>
<td>27</td>
<td>26</td>
<td>3</td>
<td>0.866667</td>
<td>140111</td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>5288.27</td>
<td>30</td>
<td>30</td>
<td>26</td>
<td>0</td>
<td>0.866667</td>
<td>158648</td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>5808.87</td>
<td>30</td>
<td>30</td>
<td>28</td>
<td>0</td>
<td>0.933333</td>
<td>174266</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>5927.76</td>
<td>30</td>
<td>28</td>
<td>27</td>
<td>2</td>
<td>0.9</td>
<td>177833</td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>6894.34</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>206830</td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>7528.21</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>225846</td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>8158.85</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>244765</td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>8836.74</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>265102</td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>9804.44</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>294133</td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>10662.6</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>319877</td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>10818.7</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>324561</td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>10770.1</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>323104</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10778.4</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>323353</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>10823.9</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>324717</td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>11210.2</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>336306</td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>11586.9</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>347608</td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>11559.5</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>346785</td>
<td></td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>11774.3</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>353228</td>
<td></td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>12002.4</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>360073</td>
<td></td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>11201.1</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>336033</td>
<td></td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>7486.48</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>224594</td>
<td></td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>6354.62</td>
<td>30</td>
<td>30</td>
<td>27</td>
<td>0</td>
<td>0.9</td>
<td>190639</td>
<td></td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>6629.09</td>
<td>30</td>
<td>29</td>
<td>28</td>
<td>1</td>
<td>0.933333</td>
<td>198873</td>
<td></td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>6843.46</td>
<td>30</td>
<td>29</td>
<td>28</td>
<td>1</td>
<td>0.933333</td>
<td>205304</td>
<td></td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>7946.36</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>238391</td>
<td></td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>8483.25</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>254497</td>
<td></td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>9063.51</td>
<td>30</td>
<td>29</td>
<td>29</td>
<td>1</td>
<td>0.966667</td>
<td>271905</td>
<td></td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>8977.63</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>269329</td>
<td></td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>9598.26</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>287948</td>
<td></td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>9505</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>285150</td>
<td></td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>9190.17</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>275705</td>
<td></td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>8255.14</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>247654</td>
<td></td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>8109.46</td>
<td>30</td>
<td>30</td>
<td>28</td>
<td>0</td>
<td>0.933333</td>
<td>243284</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>8209.05</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>246271</td>
<td></td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>8658.43</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>259753</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

95
<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>9298.43</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>278953</td>
</tr>
<tr>
<td>73</td>
<td>8461.21</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>253836</td>
</tr>
<tr>
<td>74</td>
<td>7426.65</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>222800</td>
</tr>
<tr>
<td>75</td>
<td>7490.98</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>224729</td>
</tr>
<tr>
<td>76</td>
<td>6893.89</td>
<td>30</td>
<td>30</td>
<td>28</td>
<td>0</td>
<td>0.933333</td>
<td>206817</td>
</tr>
<tr>
<td>77</td>
<td>6379.9</td>
<td>30</td>
<td>30</td>
<td>28</td>
<td>0</td>
<td>0.933333</td>
<td>191397</td>
</tr>
<tr>
<td>78</td>
<td>6568.05</td>
<td>30</td>
<td>29</td>
<td>29</td>
<td>1</td>
<td>0.966667</td>
<td>197042</td>
</tr>
<tr>
<td>79</td>
<td>7060.48</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>211815</td>
</tr>
<tr>
<td>80</td>
<td>7125.94</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>213778</td>
</tr>
<tr>
<td>81</td>
<td>8328.7</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>249861</td>
</tr>
<tr>
<td>82</td>
<td>8455.04</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>253651</td>
</tr>
<tr>
<td>83</td>
<td>8063.85</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>241915</td>
</tr>
<tr>
<td>84</td>
<td>8358.04</td>
<td>30</td>
<td>30</td>
<td>27</td>
<td>0</td>
<td>0.9</td>
<td>250741</td>
</tr>
<tr>
<td>85</td>
<td>7938.22</td>
<td>30</td>
<td>30</td>
<td>25</td>
<td>0</td>
<td>0.833333</td>
<td>238147</td>
</tr>
<tr>
<td>86</td>
<td>9571.37</td>
<td>30</td>
<td>29</td>
<td>28</td>
<td>1</td>
<td>0.933333</td>
<td>287141</td>
</tr>
<tr>
<td>87</td>
<td>10787.6</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>323629</td>
</tr>
<tr>
<td>88</td>
<td>9827.26</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>294818</td>
</tr>
<tr>
<td>89</td>
<td>9615.93</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>288478</td>
</tr>
<tr>
<td>90</td>
<td>10097.1</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>302914</td>
</tr>
<tr>
<td>91</td>
<td>10259.2</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>307776</td>
</tr>
<tr>
<td>92</td>
<td>9670.89</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>290127</td>
</tr>
<tr>
<td>93</td>
<td>8495.72</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>254872</td>
</tr>
<tr>
<td>94</td>
<td>8465.88</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>253976</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---------</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td>8301.12</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>249034</td>
</tr>
<tr>
<td>96</td>
<td>8244.39</td>
<td>30</td>
<td>30</td>
<td>28</td>
<td>0</td>
<td>0.933333</td>
<td>247332</td>
</tr>
<tr>
<td>97</td>
<td>8147.05</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>244411</td>
</tr>
<tr>
<td>98</td>
<td>8058.54</td>
<td>30</td>
<td>30</td>
<td>29</td>
<td>0</td>
<td>0.966667</td>
<td>241756</td>
</tr>
<tr>
<td>99</td>
<td>8023.53</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>1</td>
<td>240706</td>
</tr>
<tr>
<td>100</td>
<td>7914.73</td>
<td>30</td>
<td>30</td>
<td>27</td>
<td>0</td>
<td>0.9</td>
<td>237442</td>
</tr>
</tbody>
</table>