

MathWorks Math Modeling Challenge 2021

King Edwards School

Team # 14679 Bath, Somerset, England

Coach: Phillip Davies

Students: Jack Garrard, Filip Oczko, Max Smith, Carl Willis



M3 Challenge HONORABLE MENTION—\$1,000 (£716) Team Prize

JUDGES' COMMENTS

Specifically for Team # 14679—Submitted at the Close of Triage Judging:

COMMENT 1: Model is presented and explained. Some valid reasoning behind model is provided. Especially in Question 2, it is innovative to use Monte Carlo to solve the problem. Question 3 is weak in model development, but overall the paper has demonstrated the analytical thinking.

COMMENT 2: Great summary that explains the methods and results. Nice! Some good assumptions and observations for Q1. I would have liked a little more justification for using your Markov model but it is an innovative idea. Looks like you may have run out of time on Q3 but a solid start to modeling overall. Great job!!

COMMENT 3: Really wonderful job throughout! Your use of Markov Chains and the Monte Carlo approach was creative, as was your approach to part 3!

Internet Costs, Needs, and Optimal Planning

27th February, 2021

Team #14679

Contents

1	Executive Summary	2
2	Question 1:	4
2.1	Restatement of the Problem	4
2.2	Assumptions	4
2.2.1	The (negative) increase rate of monthly price is the same in the UK and the US	4
2.2.2	The Internet package with a mean price has a mean speed	4
2.2.3	Upfront costs only occur when switching broadband providers	4
2.2.4	Bandwidth in Mbps is measured using average peak download speed	4
2.3	Variables	5
2.4	Model Development	5
2.5	Results	6
2.5.1	United Kingdom	6
2.5.2	United States	7
2.6	Validity of the Model	8
2.7	Strengths and Weaknesses	8
2.7.1	Strengths	8
2.7.2	Weaknesses	9
3	Question 2:	9
3.1	Restatement of the Problem	9
3.2	Assumptions	9
3.2.1	In pre and post COVID-19 worlds, the number of people working from home is negligible	9
3.2.2	People can only use up bandwidth when they are at home and awake	9
3.2.3	Each person can only perform one activity at a time	10
3.2.4	In the Markov chain model: The chance of downtime at any minute in a day is independent of downtime for any other minute in the day	10

3.2.5	Both a week and a month are representative of a year	10
3.3	Variables	10
3.4	Markov Chain Model Development	10
3.5	Monte Carlo Model Development	12
3.6	Results	12
3.7	Validity of the Model	13
3.8	Strengths and Weaknesses	13
3.8.1	Strengths	13
3.8.2	Markov chain	13
3.8.3	Monte Carlo	13
3.8.4	Weaknesses	14
3.8.5	Markov chain	14
3.8.6	Monte Carlo	14
4	Question 3:	14
4.1	Restatement of the Problem	14
4.2	Assumptions	14
4.2.1	The population in each sub-region is evenly distributed	14
4.2.2	The percentage of households with given technological equipment is the same for all sub-regions of a region	15
4.2.3	The three different bands that a cellular tower can be placed in all have the same price	15
4.2.4	Nodes emit a signal that encompasses the area of a circle	15
4.2.5	Price of each band tier is irrelevant to the plan	15
4.2.6	Each household will have a maximum of two devices	15
4.2.7	Each household will have a maximum of one computer	15
4.2.8	No household with no devices will have at home broadband	15
4.3	Variables	16
4.4	Model Development	16
4.5	Results	17
4.6	Strengths and Weaknesses	18
4.6.1	Strengths	18
4.6.2	Weaknesses	18
5	Bibliography	18
6	Appendix A	19
7	Appendix B	25

1 Executive Summary

In recent years, the usage as well as the importance of the Internet has continuously increased. Because of the recent COVID-19 pandemic as well as technological advancements, humanity has begun to become increasingly more dependant on the Internet. With this

comes the necessity of understanding future utilisation of the Internet and the technological advancements and improvements to the Internet. To enable us to understand the future potential of the Internet and its accessibility we need to understand how accessible it is going to become to people, and one of the most efficient ways of achieving this would be by observing current Internet pricing trends as well as usage statistics in order to predict future developments.

The first problem our team decided to explore was the trend of decreasing Internet costs throughout the years. To achieve this we sought to find a relationship between the price of bandwidth per Megabits per second (Mbps) and the passage of time. Within this we considered several factors that would have affected the price of Internet such as average speed of the Internet in a given year, the upfront costs some users would have to pay which were associated with Internet as well as the general trends of package prices. While researching these areas we found that the Internet displayed exponential improvements in terms of speed over time while the general price of packages followed a negative linear trend. We calculated the price of the average package to be decreasing at a rate of £0.91 per year and the Internet speed to be increasing by 22.7% and 15.8% (to 3 significant figures) per year in the US and the UK, respectively. With this information we created a mathematical model that predicted the average price of 1 Mbps of bandwidth after a given number of years based on the initial cost of Internet, the decreasing price of Internet, the upfront costs certain people would need to pay when switching Internet providers, as well as the increase in speed. After inputting known values into our equation we found that Internet prices may fall as low as £0.04 per Mbps in the UK and £0.03 per Mbps in the US by 2031.

For Question 2, we found that there were two useful approaches to model bandwidth usage per year. These two approaches both required technical computing. For the first model, we used Markov chains which produced slightly accurate but imprecise results, all in the interval 0.9 Mbps to 10.5 Mbps. As a result, we decided to create a second model, which used Monte Carlo simulation to randomly select the Internet activities of a variety of families to stochastically determine the average “downtime” in a month or year. It showed that during the COVID-19 crisis, substantially more bandwidth is required in almost all cases, which draws attention to the need for strong access to Internet to support our economy and education infrastructure. It found that school aged children were particularly affected because of their need to be constantly video streaming their face in a virtual classroom.

For the third problem we were tasked with producing an optimum plan for spacial and number distribution for cellular network nodes in 3 given regions. We approached this by looking to find the bandwidth requirements for each of the regions to decide which bands would be the most appropriate for each. For region A, we got values for the max common bandwidth of each sub-region by using the provided data for laptop and mobile phone ownership for each sub-region, respectively. This gave a max common bandwidth of 716 Mbps for the whole of region A. This decided that using 2 mid band towers would be the most appropriate for full coverage and meeting the bandwidth requirement as that gave the area a maximum of 900 Mbps. The geographical placement of these two towers is then worked out by splitting the region into two halves, and the geometric center point of each of these halves is where the node will be placed.

2 Question 1:

2.1 Restatement of the Problem

We were asked to model the trend in cost per Mbps over the next 10 years. We have created a model to predict the average cost per month per Mbps in both the US and the UK.

2.2 Assumptions

2.2.1 The (negative) increase rate of monthly price is the same in the UK and the US

Justification. Due to both a lack of data and their relatively similar markets, we considered it safe to assume that the rate of monthly price change would be the same in the two countries.

2.2.2 The Internet package with a mean price has a mean speed

Justification. The Internet market is extremely complicated and is an amalgamation of many different factors. In sum, these factors are likely to create a normal distribution on the price and speed such that the mean and median are approximately equal.

2.2.3 Upfront costs only occur when switching broadband providers

Justification. Installation and activation fees are by far the largest upfront costs [5], and these are only applied when switching broadband providers as no activation or installation is needed when staying with the same provider.

2.2.4 Bandwidth in Mbps is measured using average peak download speed

Justification. This is because broadband is defined as “the maximum amount of data transmitted over an Internet connection in a given amount of time” in source [6], meaning that average peak download speed is the best data set to use.

2.3 Variables

Variable name	Explanation	Units
S_0	Initial mean Internet speed in 2021	Mbps
S_t	Mean Internet speed after t years	Mbps
P_0	Initial mean broadband monthly price in 2021	£
P_t	Mean broadband monthly price after t years	£
R_t	Cost per month per Mbps after t years	£/Mbps
D	The (negative) increase rate of monthly price	£/year
J	The exponential increase rate of speed	...
U	The mean upfront costs per person if they change provider	£
M	The mean time before the customer switches broadband provider	months
t	Time passed since base year (2021)	years

2.4 Model Development

We observed that the mean Internet speed, regardless of the country, was going to increase exponentially by Nielsen's law [4]. We actually found that in the data sets we were using [1], the exponential relationship was much weaker, possibly because other exponential factors cancel out the high rate predicted by Nielsen. Nevertheless, there was still an exponential relationship such that we could form the following set of parametric equations, where $S_0 = 172.32$ and $J = 0.2053$ in the US, and $S_0 = 92.392$ and $J = 0.1471$ in the UK:

$$S_t = S_0 \times e^{Jt} \quad (1)$$

Using source [1], we found that the rate of monthly price change in the UK was showing a linear trend downwards. Using assumption 2.2.1, we assumed this rate of change to be the same in the US. We therefore formed this second component of the parametric equation, where $P_0 = \$49.55$ in the US, and $P_0 = \$24.63$ in the UK, and $D = -\$0.91$ for both countries:

$$P_t = P_0 + Dt \quad (2)$$

However, we realised the importance of including upfront costs. These are fixed costs which influence a customer's decision and so should be included in output of the model. We decided to amortise these upfront costs over the expected lifetime of the contract, because upfront costs are only paid if the customer switches broadband provider, as outlined in assumption 2.2.3, and although they're paid in one go they can be thought of as being paid monthly. We estimated that customers change broadband providers after 198 months, making $M = 198$. This is because the average contract lasts 18 months [2] and customers switch on average after 11 contractual cycles [3]. Using source [5], we found $U = \$69.47$ in the US and $U = \$54.28$ in the UK. This then changes our average cost per month for broadband into this equation:

$$P_t = P_0 + Dt + \frac{U}{M} \quad (3)$$

These two parametric equations, defined with parameter t , can be divided by each other to get the cost per Mbps. This quotient is a real cost per Mbps by assumption 2.2.2. This gives us the final equation:

$$R_t = \frac{\text{cost}}{\text{speed}} = \frac{P_0 + Dt + \frac{U}{M}}{S_0 \times e^{Jt}} \quad (4)$$

2.5 Results

We computed results for both the United States and the United Kingdom. All of the values not already mentioned above were generated from the M3 data set [1], by using spreadsheet software to perform exponential and linear regressions.

2.5.1 United Kingdom

We initialised our model with these values for the parameters:

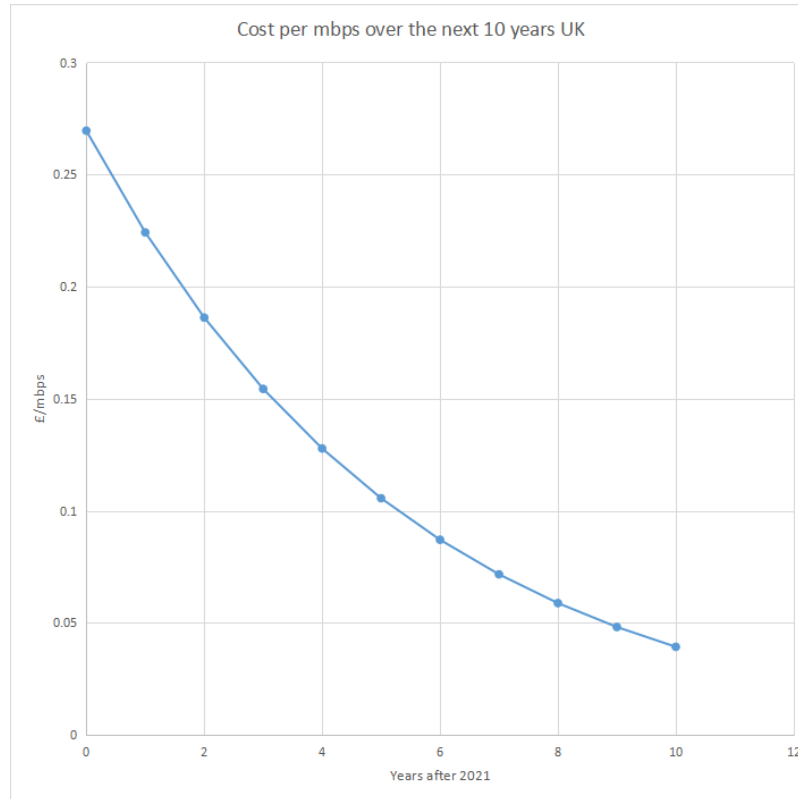
Variable	Value
P_0	24.63
D	-0.91
U	54.28
M	198
S_0	92.392
j	0.1471

Table 1: UK-specific model parameters

We then computed all the values from $t = 0$ through $t = 10$ using spreadsheet software. A chart showing the relationship between R_t and R is provided in Figure 1.

Years from 2021	Cost per Mbps rounded to the nearest pence (£/Mbps)
0	0.27
1	0.22
2	0.19
3	0.15
4	0.13
5	0.11
6	0.09
7	0.07
8	0.06
9	0.05
10	0.04

Table 2: UK-specific results

Figure 1: Relationship between R_t and t

2.5.2 United States

Years from 2021	Cost per Mbps rounded to the nearest pence (£/Mbps)
0	0.29
1	0.23
2	0.19
3	0.15
4	0.12
5	0.09
6	0.08
7	0.06
8	0.05
9	0.04
10	0.03

Table 3: US-specific results

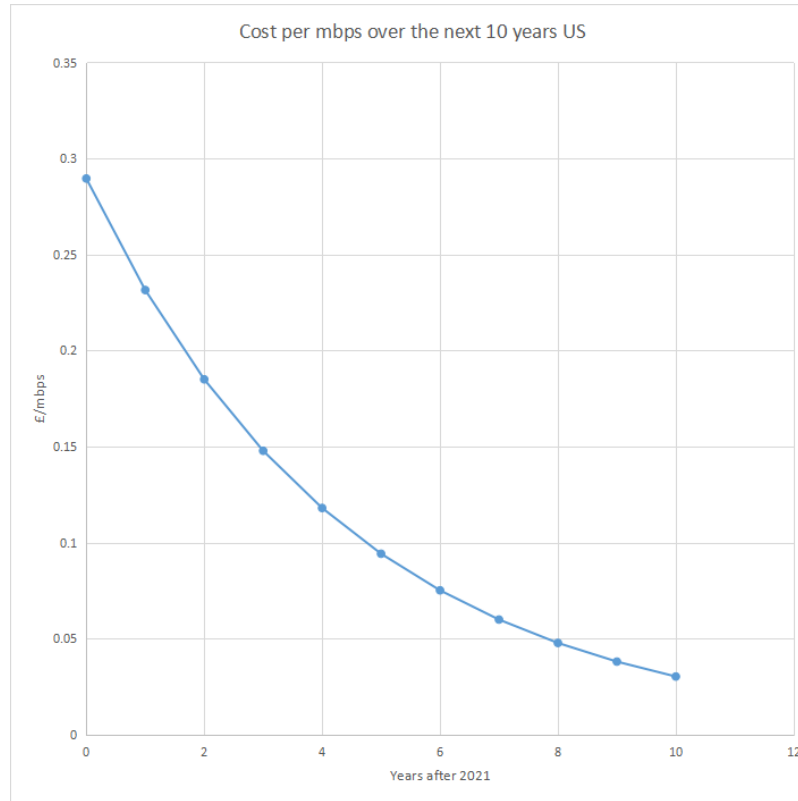


Figure 2: Relationship between R_t and t

2.6 Validity of the Model

Since we have modeled the trend in cost per Mbps with an exponential, the graphs above show a decreasing curve at a decreasing rate. As the trend isn't linear and instead begins to level out, we don't come into the problem of reaching negative value and the leveling out of the line gives a much more sensible trend. The cost per Mbps will always be a cost so even with technological advancements the price will not likely reduce any further than the extent shown in our graphs, and this is especially true when looking at the 10-year period.

2.7 Strengths and Weaknesses

2.7.1 Strengths

The model is in closed form, and so it is simple to evaluate it and understand its correctness. Modelling Internet speed advancements exponentially is a well-established methodology that has been tested on data from as early as 1998. [4]

The model also accounts for upfront costs which makes it more useful as a measure for consumers as otherwise broadband companies could hide all their costs in the form of fixed costs to make them invisible.

Our model can also be used for any country, not just the US and the UK, making it very

flexible and fit for use outside this paper.

2.7.2 Weaknesses

The model is incapable of predicting rapid technological advancements, such as a fundamental change to the medium used in Internet technology. For example, a change from copper wiring or fibre-optic cables to radio telegraphy through the air, or a change from 4G to 5G, is unlikely to be something our model could capture — but these sorts of changes are extremely difficult to predict anyway because they are probabilistic, depend on random human ingenuity, and have high uncertainty.

Additionally, our model fails to account for the occasional variability caused by different ISPs having slightly different deals, and although it is capable of handling the different types of Internet technology, the input to the model must already be an average figure that represents all the different types of broadband technology.

The model's results are only accurate for reasonably short periods of time, on the order of 10 years. This is because eventually, the model will predict a negative mean price and therefore a negative R_t value. Although a low cost per Mbps matches our intuitions that technological advancements will make the Internet more accessible, some kind of flattening out will eventually need to be introduced. A parallel can be drawn to Moore's law (which is extremely similar to Nielsen's law) — scientists currently believe that quantum effects and other limiting factors will soon prevent Moore's law from holding. It is difficult to predict what will happen after this happens to Nielsen's law, because it will provide immense pressure for technical innovation in violation of assumption 2.2.5.

3 Question 2:

3.1 Restatement of the Problem

3.2 Assumptions

3.2.1 In pre and post COVID-19 worlds, the number of people working from home is negligible

Justification. Whilst the idea of working from home now may seem incredibly normal and part of our daily lives, before and possibly after COVID-19, there will be so few people working from home that it will have no effect on our model.

3.2.2 People can only use up bandwidth when they are at home and awake

Justification. When you are at work you cannot use the home broadband, and when you are asleep you will also not be using the home broadband.

3.2.3 Each person can only perform one activity at a time

Justification. The list of activities in use have been carefully selected such that they are disjoint from each other in most cases and it is extremely difficult to coherently do two at once.

3.2.4 In the Markov chain model: The chance of downtime at any minute in a day is independent of downtime for any other minute in the day

Justification. Although this is not quite true, the large length of the day compared to the short duration of any activity (and therefore any outage) makes this assumption acceptable.

3.2.5 Both a week and a month are representative of a year

Justification. A year in the life of the average person follows the cyclic structure of the working week. We can assume that all 52 weeks of the year will follow this structure therefore simulating a week can be representative of a year. Since a month is just several weeks, that too will be representative of a year.

3.3 Variables

Variable name	Explanation
B_i	The probability of starting activity j if you're doing nothing
E_i	The probability of discontinuing activity i

3.4 Markov Chain Model Development

Our model began with data manipulation. We know that in a pre and post COVID-19 world, the number of people working from home is negligible, as outlined in assumption 3.2.1. This means that the only time people are using up the bandwidth is when they are at home and not asleep, using assumption 3.2.2. In source [1], we are given the hours per week that different age groups spend on different electronic devices. We calculated the average amount of free time that each age group has, by looking at time spent in school [7], employment data for different age groups [8], time spent in work for average full time and part time employees [9] [10], proportion of workers in part time work [11], number of children in college [12] and average length of lectures [13]. We also looked at how long people are asleep, using sources [14] and [15].

Using this data, we calculated the average amount of time each age group would spend at home, whilst not asleep. This allowed us to calculate the average proportion of their at home time that each age group spent doing each technological activity in spreadsheet D4 of source [1]. These different activities were then labelled by the amount of bandwidth they used up, which was outlined in spreadsheet D5 of source [1]. Using these probabilities we could then create a Markov chain to find the proportion of time that the maximum bandwidth will be exceeded, given a certain maximum bandwidth. This could be varied for the different individuals outlined in the question by varying their age category to one in spreadsheet D4

of source [1] to change the probabilities of different activities, and by changing the amount of time they spent in the house, based on their occupation, and their time asleep, based on their age.

The Markov chain quantises the day into indivisible minutes. It then predicts the activity someone will be doing in 1 minute given the activity they are doing right now. It is acceptable to represent each activity as its own state in the Markov chain as a consequence of assumption 3.2.2. If an activity typically takes t minutes, then the transition of that activity's state to itself should be such that after t minutes, the probability of having left the state should be 0.5. This means that on average people will discontinue the task after exactly t minutes. Consequently, the relationship between t (the time an activity takes), and p (the transition from a state to itself) must satisfy

$$p^t = 0.5 \quad (5)$$

$$\implies \ln p^t = \ln 0.5 \quad (6)$$

$$\implies t \ln p = \ln 0.5 \quad (7)$$

$$\implies \ln p = \frac{\ln 0.5}{t} \quad (8)$$

$$\implies e^{\ln p} = e^{\frac{\ln 0.5}{t}} \quad (9)$$

$$\implies p = e^{\frac{\ln 0.5}{t}} \quad (10)$$

We therefore were able to use equation 10 to populate the main diagonal of the matrix representing our Markov chain.

We add an extra state, the “free time” state, which categorises all non-sleeping activities that require 0 Mbps. The probability of transitioning from the “free time” state into an Internet-using state could be computed directly from data about how much time that activity takes per day, using the ratio

$$\frac{\text{time to do activity}}{\text{time in day}} \quad (11)$$

$$= \frac{\text{time to do activity}}{24 \times 60} \quad (12)$$

Transitioning from one activity to a different activity is unnecessary, because the time units we have selected are so short. Instead, we always transition into the “free time” state first (as a 1 minute or more “break”), before progressing to the next state. This means that the matrix M_c representing the transitions of the Markov chain for the c th person can be computed as (where M_{cij} means the j th entry in the i th row of matrix M_c , and the matrices are indexed such that the top left cell is at $(0,0)$)

$$M_{cij} = \begin{cases} B_j & \text{if } i = 0 \\ E_i & \text{if } j = 0 \\ 1 - E_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

Then the entry M_{c00} must be specially treated as simply being equal to

$$1 - \sum_{j=1}^{\# \text{ of activities}} M_{c0j} \quad (14)$$

so that the probabilities add up to 1.

Finally, we used **technical computing** (see Appendix A) to calculate the probability of person c being in state j after time t . We did this by using the binary exponentiation algorithm (relying on the associativity of matrix multiplication) to quickly compute the result of M_c^t . This relies on these well-known properties of exponentiation:

$$M^{2t} = M^t \times M^t \quad (15)$$

$$M^{2t+1} = M \times M^{2t} \quad (16)$$

$$M^0 = I \quad (17)$$

It is a theorem [16] that M_{cij}^t is the probability of reaching state j if you started at state i , after t transitions. Therefore for each time step t we compute the probability of reaching state j if at the beginning we started at 0. We then considered the probability of every combination of states after t minutes and whether or not the state's bandwidth use exceeds some maximum bandwidth threshold q .

Since q is monotonic, we were able to perform the binary search algorithm to find the optimal value of q . These results are recorded in section 3.5.

3.5 Monte Carlo Model Development

We then decided to use the same input values as in the Markov chain model, but instead of computing the probabilities directly by exponentiating the transition matrix, we wrote a computer program (Appendix B) which simulated 1 month of each of the three people. The model was in three variables per person: S , the time each person sleeps (in minutes per day), W , the time each person works (in minutes per day), and Q , the set of days of the week in which the person is not present at the household.

Please see Appendix B for the values.

3.6 Results

Ages (works/schools outside home)	Desired uptime	Output (req. broadband in Mbps)
18-35 (no), 18-35 (yes), 2 (no)	0.90	10.5
18-35 (no), 18-35 (yes), 2 (no)	0.99	10.5
65+ (no), 12-17 (yes), 12-17 (yes)	0.99	10.5
65+ (no), 12-17 (yes), 12-17 (yes)	0.90	9.0
18-35 (*), 18-35 (*), 18-35 (*)	0.90	10.5
18-35 (*), 18-35 (*), 18-35 (*)	0.99	10.5

* = the working time was carefully calibrated based on 150 minutes per week of lectures and 7 hours of part time work. See sources above for more information.

For the Monte Carlo simulation, the results were outputted as

```
90% for couple is 3.7 Mbps
99% for couple is 7.0 Mbps
99% for COVID couple is 10.7 Mbps
90% for COVID couple is 7.9 Mbps
Grandmother covid 99% is 4.9
Grandmother covid 90% is 1.2
Grandmother non-covid is same
Students covid 90% is 5.9
Students covid 99% is 8.0
Students non-covid 99% is 7.4
Students non-covid 90% is 1.0
```

3.7 Validity of the Model

The figures between $9.0 \leq x \leq 10.5$ are common broadband figures, which adds confidence that the model is behaving correctly.

The fact that the model is showing extremely similar values for the different cases may appear suspicious, but it is because q is a step-function and so the actual results are intervals which are disjoint except that they share a common point (the one which was recorded twice in the table), and also due to floating point rounding errors.

However, the values are reasonably useful to a consumer because broadband is not sold in much less than units of 1 Mbps.

3.8 Strengths and Weaknesses

3.8.1 Strengths

3.8.2 Markov chain

The use of a Markov chain allows us to fall back on a large body of existing knowledge in the field. This means that the model can more easily be understood by experts and formally verified as being correct.

Additionally, the model takes into account several factors such as the age and occupation, and can be customised much more easily.

Specifically, the Markov chain is not being used to its full potential because many of the transitions are exactly the same. We expect that future work to the model would be to realise that the order in which people do activities influences the probability of doing the next activity. In other words, the activities are not independent of one another.

3.8.3 Monte Carlo

The Monte Carlo simulation clearly produced more accurate results. There is a distinct possibility that this is caused simply by a bug in the Markov chain code.

Additionally, the Monte Carlo simulation is much more capable of handling a variety of factors since it can be easily tweaked for all sorts of special cases whereas a Markov chain can only be expressed as transition probabilities.

Additionally, the Markov chain only runs for 7 days, whereas this simulation runs for 30, and times the sleeping and working times much more realistically.

3.8.4 Weaknesses

3.8.5 Markov chain

The model does not provide very precise results. It is only able to round to the nearest 0.5.

Additionally, the model assumes that the probability of each minute having “downtime” is independent of all the other minutes. Although this is plainly false, by assumption 3.2.4 this is still adequate.

Clearly, the second model (below) is much stronger, but it relies much more heavily on computer simulation.

3.8.6 Monte Carlo

The Monte Carlo simulation is more opaque because it requires technical knowledge about the Python programming language to understand its correctness. Even using a functional programming like Haskell would have been superior because it would have allowed formal verification of the correctness of the program.

Additionally, the Monte Carlo simulation is stochastic by definition and so may produce different values on different executions. However, this is unlikely because it runs millions of times over the course of 30 days in 1-minute increments.

4 Question 3:

4.1 Restatement of the Problem

Produce a mathematical model which can give the optimal spacial and number distribution of mobile network nodes for 3 given regions.

4.2 Assumptions

4.2.1 The population in each sub-region is evenly distributed

Justification. Due to the areas being fictional, it is compulsory to assume this.

4.2.2 The percentage of households with given technological equipment is the same for all sub-regions of a region

Justification. Again, because the regions are fictional, no further data can be found on this, so distribution of devices must be assumed to be even.

4.2.3 The three different bands that a cellular tower can be placed in all have the same price

Justification. Costs of production and installation for the three types of tower are the exact same, since they are the same structures, but are just set to a different frequency.

4.2.4 Nodes emit a signal that encompasses the area of a circle

Justification. The range at which mobile signal is emitted from a node is equal 360° about the node. Therefore, this range is a radius of a circle.

4.2.5 Price of each band tier is irrelevant to the plan

Justification. Since there is no clear price data for each of the node bands, we can assume that the optimal plan for node distribution will simply be achieved by using the least number of nodes to meet the bandwidth requirements without taking into account price efficiency.

4.2.6 Each household will have a maximum of two devices

Justification. Since the average number of both computers and smartphones per household globally is below 50%, it is safe to assume that in these areas with little at home broadband, the number of devices per household will be low enough to make this assumption.

4.2.7 Each household will have a maximum of one computer

Justification. There is no need for a household to have more than one computer, so it is very rare, meaning that it will be ignored in our model.

4.2.8 No household with no devices will have at home broadband

Justification. If a household has no devices that connect to the Internet, there is no need for broadband at home.

4.3 Variables

Variable name	Value
Total number of households	2878
Proportion of households that have a desktop or laptop computer	0.643
Proportion of households that have broadband Internet service	0.495
Proportion of households that have a smartphone	0.692
Proportion of households that have only 1 smartphone	0.692

4.4 Model Development

Using assumption 4.2.2, we extrapolated the proportion of households that have a desktop or laptop computer, the proportion of households that have broadband Internet service, the proportion of households that have a smartphone, and the proportion of households that have a smartphone and no other computing device to be the same for all sub-regions.

We used this to calculate the proportion of the population that have two devices, one of which is a phone. Using source [17], which says that 50% of mobile phone users can access the Internet using only mobile phones, we found the proportion of people who have only phones.

Using assumption 4.3.6 then allows us to find the proportion of households that have two phones, which we can then use to find the proportion of houses that have a phone and a computer, and the proportion with just a computer. We now have the proportion of households with one phone, two phones, a phone and a computer and just a computer. Due to assumption 4.2.7, we know that these are all the possible devices a household can have, except for having no devices, so the number of households with no devices can then be calculated.

We can then calculate the total number of devices, using double the proportion of households with two devices multiplied by the total number of households, plus the proportion of households with only one device multiplied by the total number of households.

Using assumption 4.6.8, we know that a household with no devices will never have at home broadband, so we can calculate the number of devices that have no at home broadband. This is done by dividing the proportion of households with one device by the proportion of households that have any devices, and multiplying this by the number of households with a device with no Internet connection. This is added onto the proportion of households with two devices divided by the proportion of households that have any devices, and multiplying this by twice the number of households with a device with no Internet connection.

Using this result, we then find the maximum common bandwidth. This is done by multiplying the chance of a device video streaming in non-working hours, which was calculated in Question 2, by the Mbps it takes to stream SD quality video.

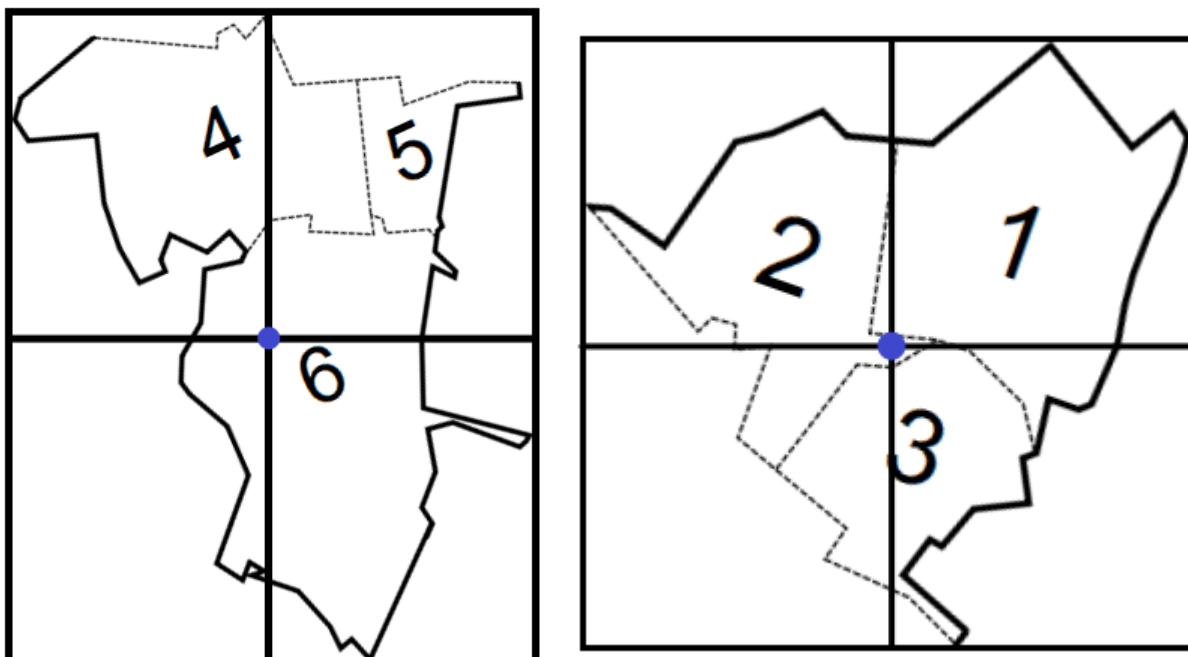
We repeat this for all the different sub-regions to find a maximum common bandwidth for all sub-regions, which alongside the size of the regions, can be used to find the number of nodes needed to both cover the entire region and have a high enough bandwidth to be above the maximum common bandwidth for all the sub-regions the node will cover.

This was done for region A to give the following results:

Variable name	Value
Proportion of households that have only phones	0.346
Proportion of households that have 2 phones	0.303
Proportion of households that have a 1 phone and 1 computer	0.346
Proportion of households that have just a computer	0.297
Proportion of households that have nothing	0.011
Total number of devices	5586
Total number of households with a device but no Internet	1378
Total number of devices without Internet	2282
Total max common bandwidth	718.1Mbps
Max common bandwidth for sub-region 1	63.4Mbps
Max common bandwidth for sub-region 2	171.3Mbps
Max common bandwidth for sub-region 3	128.4Mbps
Max common bandwidth for sub-region 4	61.9Mbps
Max common bandwidth for sub-region 5	153.1Mbps
Max common bandwidth for sub-region 6	140.6Mbps

4.5 Results

Full coverage for the bandwidth requirements of region A can be met by using two mid band nodes. The spacial distribution of each node was then provided by splitting each of the three regions into smaller roughly equal shapes. The geometric center points of these subsections can then be found as the optimum placement of each node. The center point for each subsection is found simply by encompassing it by the rectangle of smallest possible area and using the centre point of that rectangle. In the figures below, the placement of each note is denoted by a blue dot.



4.6 Strengths and Weaknesses

4.6.1 Strengths

Simplicity: Our model doesn't require usage data for households and only requires the number of devices to equate the bandwidth requirement. This coupled with the geometric method of finding the spacial distribution of the nodes gives the model adaptability for other regions and doesn't require much input data.

4.6.2 Weaknesses

Our weaknesses come from the broadness of our assumptions. Assumptions 4.2.6 and 4.2.7 are likely to be wrong, and a more accurate model would not have included them; however, they were necessary for the model to work.

5 Bibliography

References

- [1] Defeating the Digital Divide Data, *MathWorks Math Modeling Challenge 2021*, <https://m3challenge.siam.org/node/523>
- [2] Dan Howdle, *Understanding your broadband contract*, <https://www.cable.co.uk/broadband/guides/broadband-contracts/>
- [3] Jo Bailey, *Ofcom report shows decline in broadband switching*, <https://www.choose.co.uk/news/2019/ofcom-report-decline-in-broadband-switching/>
- [4] Jakob Nielsen (Nielsen Norman Group), *Nielsen's Law of Internet Bandwidth*, <https://www.nngroup.com/articles/law-of-bandwidth/>
- [5] Becky Chao and Claire Park, *The Cost of Connectivity 2020*, <https://www.newamerica.org/oti/reports/cost-connectivity-2020/>
- [6] Verizon, *Bandwidth*, <https://www.verizon.com/info/definitions/bandwidth/>
- [7] SAOTG, *How Long Should a School Day Be?*, <https://web.archive.org/web/20201023085012/https://saotg.com/how-long-should-a-school-day-be/>
- [8] Bob Watson, *Employment in the UK: October 2019*, <https://www.ons.gov.uk/employmentandlabourmarket/peopleinwork/employmentandemployeetypes/bulletins/employmentintheuk/october2019>
- [9] NiDirect, *Calculating your working time*, <https://www.nidirect.gov.uk/articles/calculating-your-working-time#:~:text=You%20have%20a%20standard%20working,your%2017%2Dweek%20reference%20period.&text=So%20you%20would%20have%20worked,within%20the%20working%20time%20limits>

- [10] Debra Leaker, *Average actual weekly hours of work for part-time workers (seasonally adjusted)*, <https://www.ons.gov.uk/employmentandlabourmarket/peopleinwork/earningsandworkinghours/timeseries/ybvb>
- [11] Karen Ho, *Part-time jobs accounted for 40% of US job growth in May*, <https://qz.com/1865486/part-time-jobs-accounted-for-40-percent-of-us-job-growth-in-may/>
- [12] U.S. Bureau of Labor Statistics, *College Enrollment and Work Activity of Recent High School and College Graduates Summary*, <https://www.bls.gov/news.release/hsgec.nr0.htm>
- [13] Kara Kots, *How Long Are College Classes Today?*, <https://www.northcentralcollege.edu/news/2020/09/17/how-long-are-college-classes-today>
- [14] Brunilda Nazario, *How Much Sleep Do I Need?*, <https://www.webmd.com/sleep-disorders/sleep-requirements#:~:text=Most%20adults%20need%207%20to%20hours%20of%20sleep%20each%20day>
- [15] The Sleep Council, *How Much Sleep Does My Child Need?*, <https://sleepcouncil.org.uk/how-much-sleep-does-my-child-need/#:~:text=While%20there%20is%20no%20hard,around%20eight%20to%20nine%20hours>
- [16] Dartmouth College, *Markov Chains*, https://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/Chapter11.pdf
- [17] Lucy Handley, *Nearly three quarters of the world will use just their smartphones to access the internet by 2025*, <https://www.cnn.com/2019/01/24/smartphones-72percent-of-people-will-use-only-mobile-for-internet-by-2025.html>
- [18] Ying Lin, *10 MOBILE USAGE STATISTICS EVERY MARKETER SHOULD KNOW IN 2021*, <https://www.oberlo.co.uk/blog/mobile-usage-statistics#:~:text=There%20are%20currently%203.5%20billion,them%20are%20users%20of%20smartphones>
- [19] Statista, *Share of households with a computer at home worldwide from 2005 to 2019*, <https://www.statista.com/statistics/748551/worldwide-households-with-computer/>

6 Appendix A

```
from itertools import product
from math import log, e
from functools import lru_cache
from copy import deepcopy

class Matrix:
    def __init__(self, rows):
        self.rows = rows
        self.num_rows = len(rows)
```

```
self.num_cols = len(rows[0])

def __mul__(self, other):
    assert self.num_cols == other.num_rows

    # Create a blank new matrix
    new_rows = [
        [
            None # Create an empty row (of Nones)
            for _ in range(other.num_cols)
        ]
        for _ in range(self.num_rows) # and then create lots of
            rows of that form
    ]

    # Now populate the values of the matrix
    for i in range(self.num_rows):
        for j in range(other.num_cols):
            # Compute the scalar product of the row from this matrix
            # with the column of the
            # other matrix, using the built in 'sum' function to
            # aggregate all the products
            new_rows[i][j] = sum(
                self.rows[i][k] * other.rows[k][j]
                for k in range(self.num_cols)
            )

    return Matrix(new_rows)

def __repr__(self):
    """
    Function for debugging purposes to
    output a string representation of the matrix
    """
    out = "{"
    for row in self.rows:
        out += ', '.join([
            str(entry) for entry in row
        ]) + '\n'
    out += "}"
    return out

@lru_cache(None)
def __pow__(self, num):
    """
```

```

    Exponentiate the matrix using the binary exponentiation
    algorithm.

    If 'num' is even, then  $M^{num} = M^{\{num/2\}} * M^{\{num/2\}}$ 
    If 'num' is odd, then  $M^{num} = M^{\{num-1\}} * M$ 
    """
    assert self.num_rows == self.num_cols
    # Construct the identity matrix
    identity_matrix = Matrix([
        [1 if i == j else 0 for j in range(self.num_cols)]
        for i in range(self.num_rows)
    ])

    # Anything to the power of 0 is the multiplicative identity
    if num == 0:
        return identity_matrix

    if num % 2 == 0:
        # Square the square root of the result of the exponentiation
        sqrt = self ** (num // 2)
        return sqrt * sqrt
    else:
        # Subtract one from the power, and then multiply by the base
        one_less = self ** (num - 1)
        return self * one_less

bandwidth_usage = [
    0, # NOT USING INTERNET ...
    1, # SURFING WEB: ...
    2, # ONLINE GAMING: ...
    2.5, # VIDEO CONFERENCING: ...
    3.5, # SD VIDEO STREAMING: ...
    50 #LARGE FILE DOWNLOADS: ...
]

MINUTES_IN_HOUR = 60
SLEEPING_HOURS = 8
HOURS_IN_DAY = 24 - SLEEPING_HOURS

# Transitions from doing nothing to doing a particular state
free_to_state = {
    "2-11": [
        0.85, 0.00, 0.04, 0.00, 0.10, 0.00
    ],

```

```
"12-17": [
    0.89,0.00,0.06,0.00,0.05,0.00
],
"18-34": [
    0.39,0.42,0.05,0.00,0.14,0.00
],
"35-49": [
    0.42,0.44,0.02,0.00,0.12,0.00
],
"50-64": [
    0.55,0.37,0.01,0.00,0.08,0.00
],
"65+": [
    0.76,0.20,0.00,0.00,0.03,0.00
]
}

# Time per activity
time_per_activity = [
    60, # NOT USING INTERNET ...
    15, # SURFING WEB: ...
    82, # ONLINE GAMING: ...
    45, # VIDEO CONFERENCING: ...
    48, # SD VIDEO STREAMING: ...
    20 #LARGE FILE DOWNLOADS: ...
]

# Transitions for continuing to do a state
state_to_state = [
    e ** (log(0.5) / time) for time in time_per_activity
]

# Binary search between these two limits
lo_limit = 0
hi_limit = 20

# Target downtime (i.e. 1 - uptime)
# This is set to 0.1 and 0.01 for 90% and 99% uptime respectively
TARGET = 0.1

DAYS_IN_WEEK = 7
WORKING_DAYS = 5

"""
Here are all the data inputs used
```

```

person_data = [
    ('18-34', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK),
    ('18-34', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK + 7 *
     MINUTES_IN_HOUR * WORKING_DAYS),
    ('2-11', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK)
]

person_data = [
    ('65+', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK),
    ('12-17', 8 * MINUTES_IN_HOUR * 2 + MINUTES_IN_HOUR *
     HOURS_IN_DAY * 5),
    ('12-17', 8 * MINUTES_IN_HOUR * 2 + MINUTES_IN_HOUR *
     HOURS_IN_DAY * 5),
]

person_data = [
    ('18-34', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK + 16 *
     MINUTES_IN_HOUR + 150 * MINUTES_IN_HOUR),
    ('18-34', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK + 16 *
     MINUTES_IN_HOUR + 150 * MINUTES_IN_HOUR),
    ('18-34', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK + 16 *
     MINUTES_IN_HOUR + 150 * MINUTES_IN_HOUR),
]
"""

while lo_limit < hi_limit:
    mid = (lo_limit + hi_limit) / 2
    bandwidth_limit = mid

    active_matrices = []

    # A person is asleep if they are not capable of using the
    # bandwidth, either because they are at a different physical
    # location or asleep
    person_data = [
        ('65+', 8 * MINUTES_IN_HOUR * DAYS_IN_WEEK),
        ('12-17', 8 * MINUTES_IN_HOUR * 2 + MINUTES_IN_HOUR *
         HOURS_IN_DAY * 5),
        ('12-17', 8 * MINUTES_IN_HOUR * 2 + MINUTES_IN_HOUR *
         HOURS_IN_DAY * 5),
    ]

    for age, asleep_time in person_data:

```



```

# Create a blank square matrix
matrix = [
    [0 for _ in range(len(bandwidth_usage))] for _ in range(len
        (bandwidth_usage))
]

for i in range(0, len(bandwidth_usage)):
    matrix[0][i] = free_to_state[age][i]
    matrix[i][i] = state_to_state[i]
    matrix[i][0] = 1 - matrix[i][i]

matrix[0][0] = 1 - sum(matrix[0][1:])
active_matrices.append(Matrix(matrix))

P_downtime = 0
avg_downtime = 0
tot_downtime = 0

current_matrices = deepcopy(active_matrices)

TIME = MINUTES_IN_HOUR * HOURS_IN_DAY * DAYS_IN_WEEK

for time in range(TIME):
    # print(list(enumerate(zip(active_matrices, current_matrices)
        )))
    current_matrices = [
        # Only tranistion if awake
        M if time < person_data[i][1] else A * M
        for i, (A, M) in enumerate(zip(active_matrices,
            current_matrices))
    ]

    # Chance of downtime in this minute
    P_downtime = 0
    #print("Chance of being in state 2 after", time, "minutes is",
        current_matrices[0].rows[0][2])
    # For every combination of states, find out how much bandwidth
        it's using
    # and the probability of it happening and multiply them
        together. Add this
    # onto the probability variable.
    #
    # Use the Cartesian product function from 'itertools' for this
        purpose
    for states in product(range(len(bandwidth_usage)), repeat=len(

```

```

    active_matrices)):
    bandwidth_of_states = 0
    P_of_this_state = 1
    #print(states)
    for i, state in enumerate(states):
        if time < person_data[i][1]:
            continue

        P_of_this_state *= (current_matrices[i] ** time).rows[0][
            state]
        bandwidth_of_states += bandwidth_usage[state]

        #print("In this state, the bandwidth used is",
            bandwidth_of_states, "and it has a chance of",
            P_of_this_state)
        if bandwidth_of_states > bandwidth_limit:
            P_downtime += P_of_this_state

    # print(time)
    tot_downtime += P_downtime

avg_downtime = tot_downtime / (MINUTES_IN_HOUR * HOURS_IN_DAY)

#if 0 <= avg_downtime <= 0.011:#break

print("Did an iteration", lo_limit, hi_limit, "mid=", mid,
    avg_downtime)

if avg_downtime < TARGET:
    hi_limit = mid
else:
    lo_limit = mid

print("Mid=_", bandwidth_limit, avg_downtime)

```

7 Appendix B

```

from itertools import product
from math import log, e
from functools import lru_cache
from copy import deepcopy
from random import choices

```

```

bandwidth_usage = [
    0, # NOT USING INTERNET ...

```

```
1, # SURFING WEB: ...
2, # ONLINE GAMING: ...
2.5, # VIDEO CONFERENCING: ...
3.5, # SD VIDEO STREAMING: ...
50 #LARGE FILE DOWNLOADS: ...
]

MINUTES_IN_HOUR = 60
SLEEPING_HOURS = 8
HOURS_IN_DAY = 24 - SLEEPING_HOURS

# Transitions from doing nothing to doing a particular state
free_to_state = {
  "2-11": [
    0.85,0.00,0.04,0.00,0.10,0.00
  ],
  "12-17": [
    0.89,0.00,0.06,0.00,0.05,0.00
  ],
  "18-34": [
    0.39,0.42,0.05,0.00,0.14,0.00
  ],
  "35-49": [
    0.42,0.44,0.02,0.00,0.12,0.00
  ],
  "50-64": [
    0.55,0.37,0.01,0.00,0.08,0.00
  ],
  "65+": [
    0.76,0.20,0.00,0.00,0.03,0.00
  ]
}

# Time per activity
time_per_activity = [
  10, # NOT USING INTERNET ...
  15, # SURFING WEB: ...
  82, # ONLINE GAMING: ...
  45, # VIDEO CONFERENCING: ...
  48, # SD VIDEO STREAMING: ...
  20 #LARGE FILE DOWNLOADS: ...
]

# Transitions for continuing to do a state
state_to_state = [
```

```

    e ** (log(0.5) / time) for time in time_per_activity
]

# Binary search between these two limits
lo_limit = 0
hi_limit = 20

DAYS_IN_WEEK = 7
WORKING_DAYS = 5

"""
Here are all the data inputs used

person_data = [
    ('18-34', 8 * MINUTES_IN_HOUR, 0, []),
    ('18-34', 8 * MINUTES_IN_HOUR, 7 * MINUTES_IN_HOUR, []),
    ('2-11', 12 * MINUTES_IN_HOUR, 0, [])
]

person_data = [
    ('65+', 8 * MINUTES_IN_HOUR, 0, []),
    ('12-17', 8 * MINUTES_IN_HOUR * 2 + MINUTES_IN_HOUR *
        HOURS_IN_DAY * 5),
    ('12-17', 8 * MINUTES_IN_HOUR * 2 + MINUTES_IN_HOUR *
        HOURS_IN_DAY * 5),
]

person_data = [
    ('18-34', 8 * MINUTES_IN_HOUR, 5 + 7 * MINUTES_IN_HOUR, []),
    ('18-34', 8 * MINUTES_IN_HOUR, 5 + 7 * MINUTES_IN_HOUR,
        [0,1,2,3,4,5]),
    ('18-34', 8 * MINUTES_IN_HOUR, 5 + 7 * MINUTES_IN_HOUR,
        [0,1,2,3,4,5])
]

"""

person_data = [
    ('18-34', 8 * MINUTES_IN_HOUR, 5 + 7 * MINUTES_IN_HOUR, []),
    ('18-34', 8 * MINUTES_IN_HOUR, 5 + 7 * MINUTES_IN_HOUR,
        [0,1,2,3,4,5]),
    ('18-34', 8 * MINUTES_IN_HOUR, 5 + 7 * MINUTES_IN_HOUR,
        [0,1,2,3,4,5])
]

```

```
person_activities = [(0, 0) for _ in person_data]

AVG.BANDWIDTHLAT.WORK = 2.4
COVID = False

# Target downtime (i.e. 1 - uptime)
# This is set to 0.1 and 0.01 for 90% and 99% uptime respectively
TARGET = 0.1

lo = 0
hi = 50
while lo < hi:
    downtime = 0
    mid = (lo + hi) / 2
    threshold = mid

    day_of_week = 0
    time_of_day = 0

    for time in range(30 * HOURS_IN_DAY * MINUTES_IN_HOUR):
        bb = 0 # broadband
        time_of_day = time % (HOURS_IN_DAY * MINUTES_IN_HOUR)
        day_of_week = (time // (HOURS_IN_DAY * MINUTES_IN_HOUR)) % 7
        #print("time of day", time_of_day)
        for person_num in range(len(person_data)):
            expiration, mbps = person_activities[person_num]

            # If not here, continue
            if day_of_week in person_data[person_num][3]:
                continue

            # If asleep, then not using Internet
            if time_of_day <= person_data[person_num][1]:
                continue

            # If at work, then depending on whether we're in COVID we're
            # either in a
            if day_of_week <= 5 and time_of_day <= person_data[
                person_num][1] + person_data[person_num][2]:
                if COVID:
                    bb += AVG.BANDWIDTHLAT.WORK
                else:
                    continue

            if expiration > 0:
```

```
    bb += mbps
    # Decrease expiration time by 1 so it will eventually
    # reach 0
    # and expire
    person_activities[person_num] = (expiration - 1, mbps)
else:
    # Pick a random activity
    selection = choices(range(len(bandwidth_usage)), weights=
        free_to_state[person_data[person_num][0]][0])

    # Insert it into the 'database'
    person_activities[person_num] = (time_per_activity[
        selection], bandwidth_usage[selection])
#print("bb", bb)
if bb > threshold:
    downtime += 1

#print("lo", "hi", lo, hi)

print("Downtime:", downtime, time, downtime / time)
print("Mid=", mid)

if (downtime/time) < TARGET:
    hi = mid
else:
    lo = mid
```