Swim, Bike, and Run

Team 7055

November 13th, 2016

1 Summary

Our team's objective was to organize a traditional open Olympic triathlon for 2000 participants of varying gender, age, and experience level for our town. The mayor asked that we minimize road closure to under 5.5 hours, and our corporate sponsor asked us to minimize congestion so that participants can proceed without hindrance during each phase of the triathlon. A successful event that satisfies both their requests would benefit both the local economy and community.

Our model divided the athletes into 18 divisions by using data from a recent triathlon to approximate the gender, age, and experience level composition of our triathlon. We then created a weighted composite metric to minimize individual congestion, road closure times, and the number of athletes who fail to finish the race. We wrote a NetLogo program to simulate the movement of triathletes in relation to these parameters. The individual speeds of all triathletes were generated probabilistically using a normal distribution (as opposed to deterministic trials) to most accurately represent the unpredictability of open registration. We then developed and evaluated fifteen race day schedules by varying the timing and athleticism of division-based "waves". Since all participants might not finish within the 5.5 hour road closing time, we tested the incorporation of cutoff times as used in most triathlons. The below schedule had, overall, the lowest congestion and the shortest road closure while allowing the most people to finish the race.

Time	Participants
8:00:00	M/F Pro, M/F Prem
8:02:00	F Ath, M Cly
8:06:00	M/F U30
8:10:00	M 50+
8:14:00	M 30-34
8:18:00	M 45-49, F 50+
8:22:00	F 30-34, F 35-39
8:26:00	M 40-44
8:30:00	M 35-39
8:34:00	F 40-44, F 45-49

Table 1: Triathlon Schedule: M=Male, F=Female, Pro=Professional, Prem=Premier, Ath=Athena, Cly=Clydesdale, U30=Under 30.

Furthermore, we observed several interesting changes in congestion and road closure time when adjusting the race distances on the best and worst start wave schedules. In our worst-case trials, our adjustments caused congestion to increase by up to a factor of 17. In our best-case trials, we observed some improvement, with an increased biking distance performing the best overall. The greatest advantage of the various length adjustments occurred when all portions of the race were shortened. However, these improvements were not proportional to the change in length. We concluded that our model was not sensitive to overall adjustments in distance, but was sensitive to individual stage adjustments.

With this model and its accompanying solution, we aim to make our town's triathlon an enjoyable, successful event that will benefit the town for years to come.

Dear Mayor,

Thank you for your gracious support of our upcoming triathlon! We're so excited to bring 2000 athletes together to benefit local youth initiatives and promote Super Tread Race Company. Our world-class event will not only boost the local economy, but also give our town nationwide publicity.

To ensure that the event is enjoyable and fruitful for all participants, we recommend that you sort all registrants into 18 divisions based on gender, age, and experience level. These divisions ensure good competition between competitors as well as adequate recognition for talented athletes.

Furthermore, we programmed a mathematical model to minimize congestion during each stage of the race. Specifically, we minimized congestion frequency and severity by adjusting the size and timing of each division-based "wave" and the predicted speeds of their athletes. In all scenarios, we closed roads for no more than 5.5 hours to prevent excessive traffic disruption. Though some athletes may not finish within this time frame, we greatly respect the requested limits and chose to prioritize our town's transit needs. Altogether, we composed a metric that weighted and combined congestion, road closure time, and number of athletes that finish the race. We compared four schedules and three timing patterns using this metric.

Ultimately, we reached a solution which minimizes congestion during each stage, closes roads for less than 5.5 hours, and maximizes the number of athletes completing the race. This will require a continuous 5.5 hour closure from 8:00AM until approximately 1:30PM on race day. We anticipate that all set up and clean up will be completed during that time.

With future races in mind, we also explored the effects of adjusting swimming, biking, and running race distances on congestion and road closure. We found that individual adjustments in any of the three sections improved our best race schedule, and could even worsen a bad scheduling system. Our model shows that holding a shorter triathlon could potentially decrease the number of non-finishing participants in a pattern of diminishing returns.

We have attached our race day schedule for your reference. We're looking forward to the triathlon! Thank you again for your help.

Sincerely,

Team 7055

Enclosure: Race Schedule

Time	Starting Divisions
8:00:00	M/F Pro, M/F Prem
8:02:00	F Ath, M Cly
8:06:00	M/F U30
8:10:00	M 50+
8:14:00	M 30-34
8:18:00	M 45-49, F 50+
8:22:00	F 30-34, F 35-39
8:26:00	M 40-44
8:30:00	M 35-39
8:34:00	F 40-44, F 45-49

Key: M=Male, F=Female, Pro=Professional, Prem=Premier, Ath=Athena, Cly=Clydesdale, U30=Under 30.

2 Introduction

Triathlons are popular sporting events that boast many male and female competitors of all ages and experience levels. However, the high volume of participants and their varied backgrounds make it difficult to organize a race. We're collaborating with our town's mayor to plan a traditional open Olympic triathlon with divisions based on gender, age, and skill level. An open Olympic triathlon consists of: 1.5 km swim, a transition, a 40 km bike ride, a second transition, and a 10 km run, in that order.

The mayor has requested that we minimize the length of time local roads are closed for the cycling and running stages of the triathlon. Super Tread Race Company, the corporate sponsor, would also like us to ensure that the race is a world-class event that will attract professional racers and serious amateurs each year. Thus, the CEO of Super Tread has asked that the race be organized in such a way that minimizes congestion in each phase of the triathlon.

Our solution proposes a method of organizing the triathlon such that both the length of time that local roads are closed and the amount of congestion in each stage of the race are minimized. Given a data set from a previous triathlon, we used a probabilistic distribution to generate open registrant speeds for our model. We also further investigate advantages of adjusting the race distances of the swimming, cycling, and running events on congestion and road closure time.

3 Assumptions

It was necessary to make several assumptions in our model:

1. The cycling, running, and transition segments of the triathlon take place on a six-meter-wide road and the swimming course is 30 meters wide.

   We assume that most of the race takes place on two-lane roads. The National Association of City Transportation Officials recommends that lanes are no wider than eleven feet; rounding down to meters gives a width of three meters per lane(3). Because the swimming portion of the triathlon takes place in open water, its bounds are less restricted than the other road-based parts of the race. We assume a broad course to accommodate the large numbers of athletes.

2. All athletes maintain constant speeds within each of the swimming, cycling, running, and transition segments, though their speeds may vary between different segments.

   The acceleration and deceleration times between each part of the race are relatively quick, especially in relation to the length of each stage. We assume that the effects of fatigue and other factors that might influence the speed of an athlete are negligible.

3. The given data set is representative of the athletes who will register for this triathlon.

   It is difficult to predict the demographics of race participants, especially since most will be open registrants. We assume that relative percentages of professional, premier, and amateur athletes for both males and females remain the same as the provided triathlon data.

4. The triathlon course never overlaps or intersects with itself.

5. Transition 1 (T1) and (T2) are both 500 m in length.

   The lengths of the two transition times have no impact on the results of the model; the model considers only the times required to move across each area. A length of 500 meters is reasonable considering that two thousand bicycles and the accompanying transition gear must be stored in each area.

6. Congestion does not occur during transition zones.

   Most athletes in the transition area are on the sidelines getting on and off their bikes. Heavy traffic is unlikely to occur on the main road.

7. Transition areas are not on roads.

   We assume that no road closures are necessary for the setup of the two transition times. Transition areas must be made available hours before the race in order to provide triathletes with enough time to prepare their areas and must be open after the last triathlete finishes. This is especially true for large races with thousands of participants. It would be infeasible to run an entire triathlon if transition area open times were included into the 5.5 hour limit.

3.1 Normality

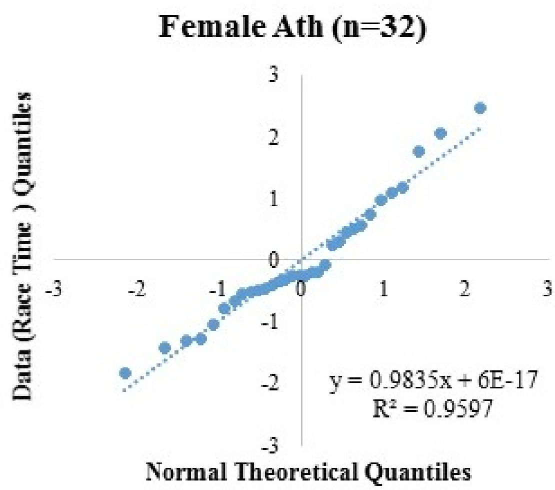


Figure 1: Example Normal Q-Q Plot for Female Athena division

To determine the theoretical distribution of the athletes' data, we separated previous participants' data into our 18 divisions and created a quantile-quantile (Q-Q) plot for each division-based data subset. Figure 1 shows the data for the Female Athena division. As the points form a line with high correlation ($R^2 = 0.96$) and random deviation, we concluded that the data can be modelled by a normal distribution. We found that the data was also normal for the 17 other divisions, as shown in Figure 1 of the appendix. This allowed us to design our program to generate probabilistic individual speeds based on a normal distribution.

3.2 Congestion

The definition of "congestion" is a essential part of our model. In conventional terms, congestion occurs whenever a triathlete is forced to travel at a slower pace because of athletes crowding the road ahead, much like a traffic jam. We determine congestion by considering the speeds of the athletes a certain distance in front of each triathlete: a triathlete may feel congested if there are more than x slower athletes within a distance of d meters in front of them. A congested athlete must match the average speed of those x people. Table 2 represents the different values of x and d for each stage of the race.

Stage of the Race	x (people)	d (meters)
Swimming	30	3
Cycling	6	3
Running	12	2

Table 2: Values of x and d for each stage of the triathlon

The values of x and d are determined by a number of factors (Subsubsection 3.2) and vary between different parts of the triathlon:

- Swimming

  On average, athletes have a width of 1 meter, thus if the swimming area has a total width of 30 meters, we can fit as many as 30 triathletes comfortably into the area in front of a triathlete before things become congested.

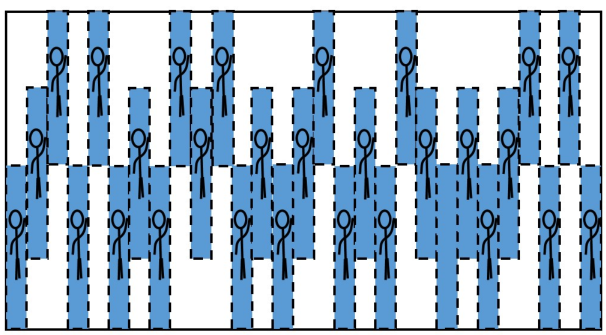


Figure 2: Example of Congested Swimming

- Cycling

  Bicycles used in triathlons must be less than 185 cm long and 50 cm wide (2). It is reasonable to see that a triathlete will need a distance of 3 meters in front of them to avoid congestion. Furthermore, considering that a triathlete and his or her bike occupies roughly 2 square meters (2 meters long and 1 meter wide), we can only fit six triathletes into the 6 meter wide and 3 meters long area in front of any triathlete as shown in the figure below:

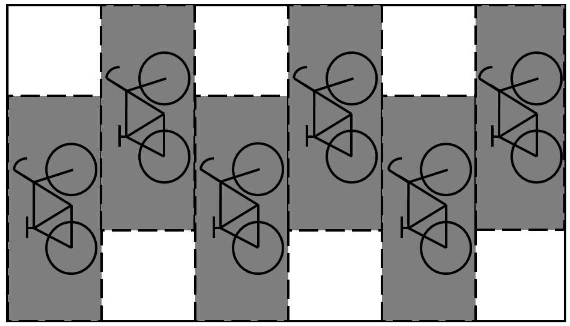


Figure 3: Example of Congested Cycling

- Running

  As previously stated, people have an average width of 1 meter, and unlike cycling, it is reasonable for a triathlete to run within a meter to the left or the right of another triathlete. This allows for a greater value of x. Furthermore, because running speeds are slower than cycling speeds, this will decrease the value of d because less distance is necessary for a triathlete to feel congested.

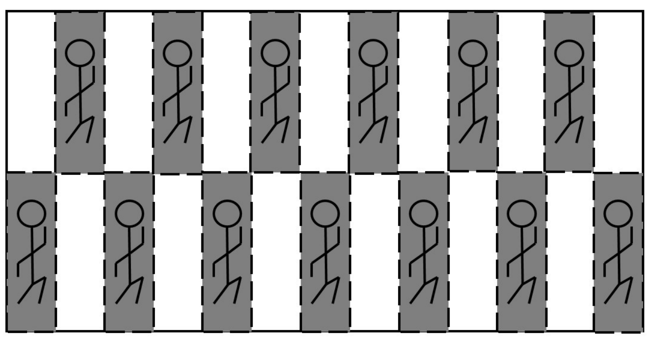


Figure 4: Example of Congested Running

If there are x slower triathletes within a distance of d meters in front of a triathlete, they may feel congested based on a "pass probability". When a triathlete is not congested, they have a pass probability of 100%, meaning that it is always possible for them to pass the triathlete in front. If the conditions for congestion are present (x slower athletes within a distance of d meters in front), the pass probability decreases to 50% because the group in front of the triathlete either let's the triathlete pass or doesn't let the triathlete pass.

4 Testing Parameters

4.1 Cutoff Schedule

We assume that roads must begin to be closed from the first wave start (i.e. the start of the triathlon) in order to ensure that they will be fully closed and set up by the time that the first athlete arrives.

Any person can participate in an open registration triathlon, regardless of speed or skill level. Therefore, it is impossible to ensure that all triathletes will be fast enough to complete the course within the hard limit of 5.5 hours set by the mayor; in fact, there were two participants in the given data set whose final total times exceeded 5.5 hours. Most triathlons implement a set of cutoff times to ensure that people do not spend excessive amounts of time on the course.

We determined cutoff times from the given data set by considering the 97-th percentile times of each of three parts of the race: swimming and T1, biking and T2, and running. The cutoff times were rounded up to the nearest quarter of an hour for convenient timekeeping.

Phase	Cutoff Time (hours)
Swimming, T1	0.75
Biking, T2	3
Running	4.5

Table 3: Cutoff times for each phase of the race (times are in hours from the last wave start)

4.2 Divisions

Division	Expected Athletes
Male Professional	4
Male Premier	31
Male Clydesdale	37
Male Under 30	175
Male 30-34	226
Male 35-39	239
Male 40-44	221
Male 45-49	176
Male Over 50	298

Table 4: Number of Expected Athletes for each Male Division

Division	Expected Athletes
Female Professional	4
Female Premier	11
Female Athena	20
Female Under 30	98
Female 30-34	130
Female 35-39	90
Female 40-44	82
Female 45-49	73
Female Over 50	85

Table 5: Number of Expected Athletes for each Female Division

We created 18 divisions for our triathlon, as shown in Tables 4 and 5. Participants were first separated by gender, as is the convention in the sporting world. They were then divided into professional, premier, and open (age and weight group) divisions based on experience level. Competing alongside athletes of similar experience helps everyone feel most comfortable. This creates a competitive yet encouraging environment that satisfies the requests of both the Mayor and Super Tread Race Company. Since open registrants are expected to make up the majority of race entrants, we further divided the open category by age and weight in accordance with the standard divisions used by USA Triathlon. The weight-based divisions were Athena for women over 165 lb, and Clydesdale for men over 220 lb. Age-based divisions were done in 5 year intervals from ages 30 through 50. In our triathlon, entrants under 30 and over 50 are each grouped into their own divisions to ensure comparable division sizes within genders.

Using the previous triathlon data, the average time and standard deviation of that time was calculated for each male and female age group.

Males (Age)		Swim	T1	Cycle	T2	Run	Final
Under 30	Average	1424	420	5178	153	3454	10629
	Standard Deviation	237	162	797	66	680	1702
30-34	Average	1457	409	4994	167	3491	10518
	Standard Deviation	244	154	705	77	682	1576
35-39	Average	1514	434	5006	182	3516	10651
	Standard Deviation	258	170	661	89	594	1503
40-44	Average	1223	533	5135	181	3372	10444
	Standard Deviation	189	184	649	94	570	1394
45-49	Average	1424	394	4952	172	3648	10590
	Standard Deviation	232	135	582	88	748	1544
Over 50	Average	1370	518	5230	210	3798	11125
	Standard Deviation	261	200	732	105	769	1669

Table 6: Average Time (sec) and Standard Deviations of Male Age Groups

Females (Age)		Swim	T1	Cycle	T2	Run	Final
Under 30	Average	1317	572	5884	208	3597	11577
	Standard Deviation	175	179	802	112	594	1596
30-34	Average	1338	588	5937	216	3647	11726
	Standard Deviation	203	208	994	124	633	1869
35-39	Average	1369	616	5956	220	3709	11868
	Standard Deviation	229	205	822	111	733	1984
40-44	Average	1380	607	5890	226	3744	11846
	Standard Deviation	205	197	822	132	701	1750
45-49	Average	1322	601	5813	216	3756	11710
	Standard Deviation	172	210	836	103	630	1739
Over 50	Average	1376	639	5871	252	4069	12207
	Standard Deviation	208	233	846	145	850	1997

Table 7: Average Time (sec) and Standard Deviations of Female Age Groups

4.3 Wave Start Schedules

There are two components of each wave start schedule:

1. Wave order

The starting wave order of each division impacts congestion, road closure, and the number of race-finishers.

2. Intermediate times

   Intermediate times are the time intervals between start waves. These vary within a wave order depending on the anticipated athleticism of the starting division(s), i.e. their ability to pull ahead far enough to not be quickly overtaken by a later wave.

4.3.1 Orders

We started with a control schedule, Schedule 0, which sends participants out in ten 200-person waves of random age, gender, and experience level composition. We then developed 4 test schedules to evaluate the effects of different wave orders on congestion, road closure, and the number of people who don't finish (Table 6). We decided not to split up divisions between start times to ensure maximal fairness. The size of each wave was limited to 300 people based on the capacity of the starting beach area of 300 m^2. This provides each swimmer at least one square meter of starting space; we saw from video footage that triathlon starting areas tend to be even more crowded.

Wave	Schedule 1		Schedule 2		Schedule 3		Schedule 4	
1	M/F Pro M/F Prem	50	M/F Pro M/F Prem	50	F Ath M Cly	57	M/F Pro M/F Prem	50
2	M/F U30	273	M U30 F Ath M Cly	232	M 50+	298	F Ath M Cly	57
3	M 30-34	226	M 30-34	226	M 45-49 F 50+	261	M/F U30	273
4	F 30-34 F 35-39	220	M 35-39	239	M 40-44	221	M 50+	298
5	M 35-39	239	M 40-44	221	F 40-44 F 45-49	155	M 30-34	226
6	F 40-44 F 45-49	155	M 45-49	176	M 35-39	239	M 45-49 F 50+	261
7	M 40-44	221	M 50+	298	F 30-34 F 35-39	220	F 30-34 F 35-39	220
8	M 45-49 F 50+	261	F U30 F 30-34	228	M 30-34	226	M 40-44	221
9	M 50+	298	F 35-39 F 40-44 F 45-49	245	M/F U30	273	M 35-39	239
10	F Ath M Cly	57	F 50+	85	M/F Pro M/F Prem	50	F 40-44 F 45-49	155

Table 8: The 4 wave start schedule orders tested using our model. M=Male, F=Female, Pro=Professional, Prem=Premier, Ath=Athena, Cly=Clydesdale, U30=Under 30. The number of athletes in each wave is shown in the shaded box to the right.

4.3.2 Times

We tested three sets of time intervals for each of the four wave start schedules, listed below. These intervals were determined based on the anticipated speed of each division. Since professionals and premiers swim faster than open participants, they need a relatively shorter amount of time to get sufficiently ahead and out of the way. We used our program to do five trials of each time and wave combination.

- 2/4: a two minute delay between the wave of professionals and premier participants and the subsequent wave of open participants and a four minute delay between waves of open participants.
- 3/5: a three minute delay between the wave of professionals and premier participants and the subsequent wave of open participants and a six minute delay between waves of open participants.
- 4/6: a four minute delay between the wave of professionals and premier participants and the subsequent wave of open participants and a six minute delay between waves of open participants.

5 Metric Design

5.1 Definitions

In order to choose the best wave start times, we quantify course congestion and road closure with a single metric for each schedule.

Definition 1. *The* ***closure time****, denoted by T, is the time in seconds that elapses from the first wave start to the end of the race, when all participants have either crossed the finish line or reached the cutoff time.*

The value of T can never exceed $19,800$ seconds (5.5 hours) because of the mayor's restriction. Therefore, we consider the **time proportion** $t = \frac{T}{19{,}800}$ in our metric. The value of t ranges from zero to one and is minimized in the "perfect" case of a race lasting for zero hours.

However, such an ideal scenario is infeasible because each triathlete requires hours of time to complete the race. In fact, there may be some athletes who are unable to even finish the race because their pace falls short of the cutoff times. Therefore, it is also important to consider the proportion of people who do not finish the race.

Definition 2. *The* ***cut proportion****, denoted by f, is the proportion of the two thousand triathletes who, because of the cutoff schedule, do not finish the race.*

Like the time proportion, the cut proportion ranges from zero to one and is minimized only in the case of every triathlete finishing.

Finally, the third and arguably most important element of the metric is the measure of congestion. It is important to quantify congestion in such a way so that not only the state of being congested, but also the severity of congestion is considered. Recall that a triathlete feels congested if he or she is unable to travel at his or her intended speed and instead is forced to move at a slower pace.

Definition 3. *At a particular second, the* ***instantaneous congestion*** *of a triathlete is equal to*

$$\frac{s_i - s_a}{s_i}, \tag{1}$$

where s_i and s_a are the triathlete's intended speed and actual speed, respectively.

The instantaneous congestion, like the time and cut proportions, ranges from zero to one and is minimized only if the triathlete experiences no congestion.

In addition, the metric should incorporate the Super Tread Race Company's prioritization of elite athletes. Therefore, we must distinguish between three groups: professional athletes, premier amateur athletes, and open registrants.

Definition 4. *The* ***congestion average*** *of a group is the average instantaneous congestion of all of its members across the closure time.*

The congestion average indicates how often the members of a group are congested and how much slower the members travel through the race because of congestion. The congestion averages of professional, premier, and open participants are denoted as c_{pro}, c_{pre}, and c_o, respectively. Then the **overall congestion score** of a wave start schedule is simply the average of the three congestion averages:

$$c = \frac{1}{3}(c_{pro} + c_{pre} + c_o). \quad (2)$$

This effectively weights the congestion of the eight participating professional triathletes more heavily than that of the 42 premier triathletes and much more heavily than the congestion of the 1950 open participants.

In this model, we determine the optimal wave start schedule such that congestion, road closure time, and proportion of non-finishers is minimized. In order to clarify comparisons between schedules, we define a metric that incorporates all three factors into a single quantity.

Definition 5. *The* ***composite rating*** *of a wave start schedule is*

$$M = w_t t + w_f f + w_c c, \quad (3)$$

where w_t*,* w_f*, and* w_c *are non-negative weights such that* $w_t = w_f = 0.25$ *and* $w_c = 0.5$*.*

This metric equals zero if the race (unrealistically) has no congestion, requires zero road closure time, and involves all participants finishing the race. Because of the probabilistic nature of the participants, we estimate the value of M by averaging the values computed through five trials of the NetLogo model (Section 6).

5.2 Weight Values

The weight values w_t, w_f, and w_c were chosen to best fulfill the specific needs of our Mayor and our sponsor, Super Tread Race Company. The Mayor wants minimal t (road closure time), f (cut proportion), and c (congestion) to ensure the race is enjoyable for everyone. Super Tread Race Company doesn't care much about t and f, but they agree with the Mayor that c, congestion, must be minimized. This is because their target sales audience of faster athletes would be greatly annoyed by performance-affecting congestion. Since both parties want low congestion, c must be given the highest weight (c_w) of the three parameters. Since only one party (the Mayor) wants low closure time and cut proportion, w_t and w_f will each be half the value of w_c. Therefore, w_c should be 0.5 while w_t and w_f are each 0.25.

While we were asked to minimize congestion time and road closure, we realized that considering the number of people who could not finish the race contributes to both goals. Therefore, we developed this composite metric to ensure that we satisfy both the Mayor and our sponsor so that this event can continue to benefit the community.

6 NetLogo Model

To evaluate our assumptions and generate data, we use an agent-based NetLogo model. By using NetLogo, we are able to visually and robustly simulate athletes in a horizontally to-scale race. We can also use NetLogo's powerful experimentation functionality to generate data.

The basis of our model is the athlete as represented by a "turtle", an icon in the grid with attributes and the ability to perform actions. Each of our turtles has `gender`, `division`, speeds for each section of the race, `deviation`, and a `congested?` attribute that indicates whether it experiences the effects of congestion.

Each athlete's divisions are deterministically determined as outlined earlier. Each simulation has the same number of athletes in each division. Each athlete has a randomly generated deviation which is used for all of its speeds. This deviation is given by NetLogo's built-in `random-normal` function with a mean of 0 and a standard deviation of 1. The athlete's speeds are then generated given mean and standard deviation based on their division. Waves are generated based on our several predetermined wave schedules, and athletes start based on our predetermined wave times. Several values can be edited before the model runs, including the wave configuration, congestion passing probability, pro-open delay, and open-open delay. These are accessed using NetLogo's dropdown boxes and sliders. Several values can be observed, including plots of number of athletes congested, total congestion averages, and average values of congestion per division.

The main world of our model displays 2000 athletes proceeding across the multicolored track at various speeds. Each color represents a different section of the race - swimming is blue, T1 is green, biking is yellow, etc. Athletes are represented as a white icon if they are not congested, and a black icon if they are. Each "tick" of the model represents a real-world second, and each "patch" represents 100 meters. (The model is not to scale vertically.)

Figure 5: This screenshot shows the "world" of our model, with athletes lined up at the start.

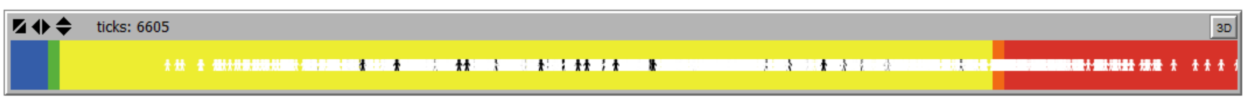


Figure 6: This screenshot shows the "world" after many ticks. Some athletes are shown to be congested.

When `setup` is called, it creates athletes, assigning them division and wave in `setup-turtles`. We then set up the patches that represent the race course in `setup-patches`. As the model is set to `go`, it first generates a list of the athletes sorted by xcor, or distance from the start. Using this, it determines which athletes are `congested?` based on a random probability and the current segment of the race they are in. Each athlete then updates its statistics to the interface and moves forward. When the number of seconds passed reaches a cutoff time, athletes still in our specified cutoff sections are asked to `die` and are no longer represented in the model. The model stops when all participants have finished, or 19800 ticks (5.5 hours) have passed.

We use NetLogo's powerful BehaviorSpace to run several batches of simulations. It allow us to set levels of variables and trials, and automatically runs as many simulations as specified. These data can be then exported as a spreadsheet for further analysis.

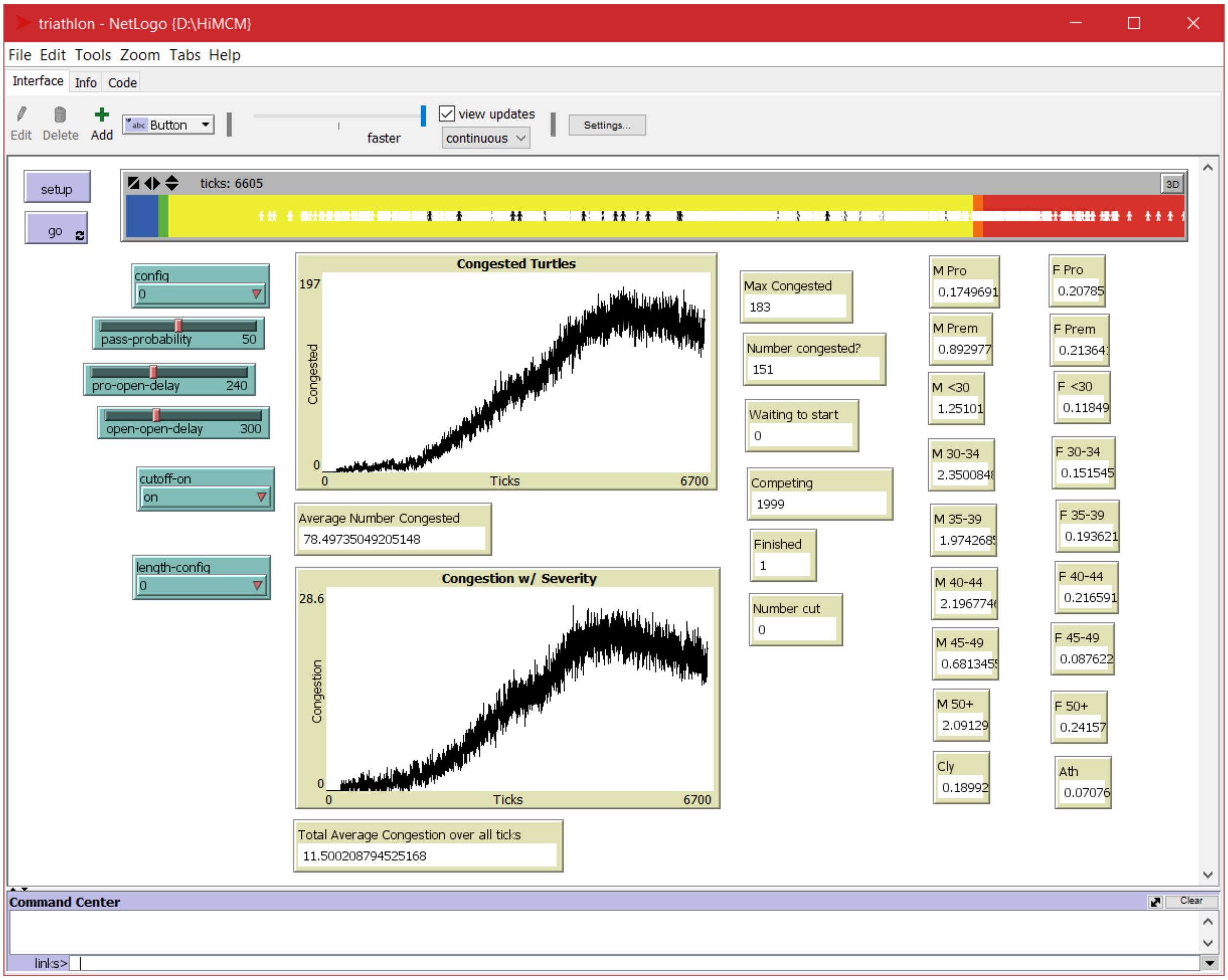


Figure 7: This full screenshot shows the entire NetLogo interface of our model in progress.

7 Results

Schedule	c	t	f	M
0	0.0178	0.9213	0.0035	0.24011
1	0.0012	0.9213	0.0063	0.23251
2	0.0011	0.9213	0.0045	0.23198
3	0.0270	0.9213	0.0025	0.24442
4	0.0013	0.9213	0.0025	0.23158

Table 9: Metric values for 2/4 Wave Times

Schedule	c	t	f	M
0	0.0210	0.9485	0.0027	0.24828
1	0.0012	0.9475	0.0042	0.23854
2	0.0011	0.9485	0.0056	0.23909
3	0.0245	0.9485	0.0018	0.24985
4	0.0013	0.9485	0.0031	0.23856

Table 10: Metric values for 3/5 Wave Times

Schedule	c	t	f	M
0	0.0228	0.9758	0.0030	0.25608
1	0.0013	0.9758	0.0042	0.24564
2	0.0011	0.9758	0.0046	0.24566
3	0.0226	0.9756	0.0009	0.25542
4	0.0013	0.9758	0.0024	0.24520

Table 11: Metric values for 4/6 Wave Times

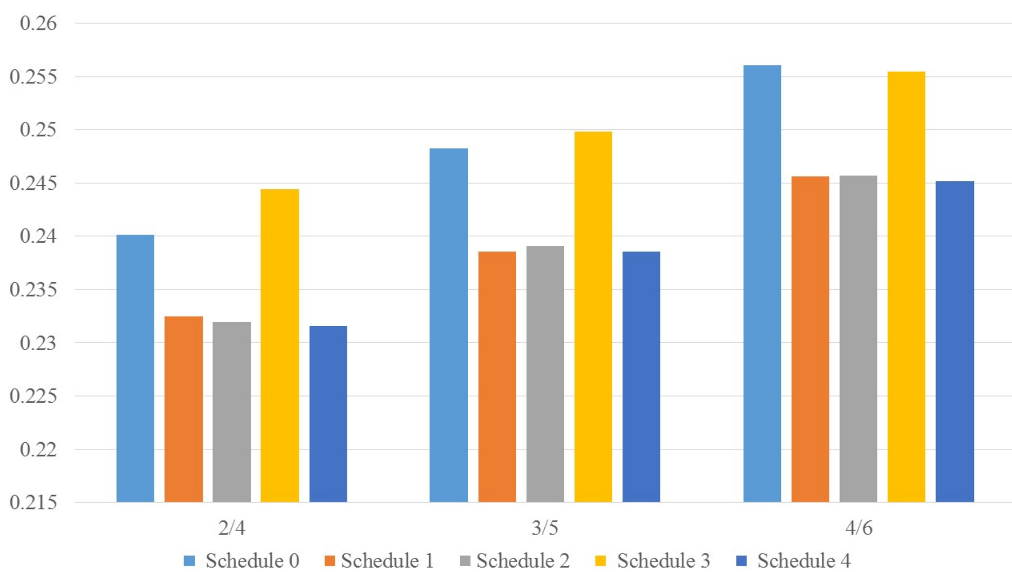


Figure 8: M values for each schedule and wave time

From our Netlogo simulations, we can see that Schedule 4 with the 2/4 interval delays produced the smallest value of M and Schedule 0 with the 4/6 interval delays produced the largest value of M.

These results follow logical reasoning. By increasing the time interval between waves, faster athletes would have to move past larger and more frequent groups of slower athletes. This would increase congestion, and allow faster athletes to proceed, lowering road closure times.

By alternating fast-slow-fast-slow-etc., schedule 4 consistently produces low values of f as slower athletes are released earlier. It also produces low values of c, as congestion is reduced when pro and premier athletes begin ahead of the race.

Schedule 3, while not the worst schedule, was consistently higher than other schedules due to its arrangement. It is problematic as it consistently releases pro and premier athletes behind several waves of open athletes.

Schedule 0, with randomly generated waves, would naturally cause the most congestion and the worst M as faster athletes could potentially start much later than slower athletes. In addition, it could also have a high f as it has a chance of releasing slower athletes near the end, preventing them from finishing.

We also note the differences in best M values based on wave times. Schedule 4 is very closely surpassed by schedule 1 in wave time 3/5, with values of 0.23856 and 0.23854 respectively (a 0.0084% difference). This can be attributed to schedule 1's low t value - it is ordered such that faster athletes tend to start earlier than slower athletes, allowing for them to finish faster. However, it has a much higher f, which is reasonable as slower athletes starting last cannot finish in time.

8 Adjusting Course Distances

The triathlon planned is a traditional open Olympic triathlon with set distances for each part of the race. However, we also examined how congestion and road closure would change if the race distances of each event were adjusted. We re-examine the best and worst wave schedules as determined previously in Section 7 and run them on four different altered courses, hoping to understand how adjustments in distances would affect the metrics of each schedule. In particular, we wish to determine what changes would alleviate congestion, decrease road closure, or decrease cut proportions. We did not increase the total race distance, as longer distances require strictly more time to traverse and the mayor's 5.5 hour limit remains constant regardless of distance.

Four new courses are created:

1. Increased swimming distance, constant total distance,
2. Increased biking distance, constant total distance,
3. Increased running distance, constant total distance,
4. Decreased overall triathlon distance.

8.1 Course 1: Increased Swimming Distance

For course 1, the race distance of the swimming event is increased by 0.5 km to 2 km and the race distance of the biking event is decreased to 39.5 km. The total distance of the triathlon is kept constant at 51.5 km.

In previous parts of the model, congestion seemed to be lowest during the first part of the race (see Figure 7), when most of the active contestants were in the swimming event. It is of interest to investigate whether this was because of the swimming parameters or simply because, due to the structure of wave starts, there are less triathletes on the course during the beginning. If the former is true, then a longer swimming distance would result in improved congestion and lower metric values.

8.2 Course 2: Increased Biking Distance

For course 2, the total distance of the triathlon is again kept constant at 51.5 km. However, the race distance of the swimming event is decreased by 0.5 km to 1 km and the race distance of the running event is halved from 10 km to 5 km. The race distance of the biking event then becomes 45.5 km.

The biking portion of a triathlon is unique because it is by far the longest in length. Most of the distance covered during a triathlon is by biking. Since each triathlete moves at a constant speed through each event, longer distances increase congestion by giving more time for faster athletes to catch up with slower athletes in front of them. As a result, we would expect the same wave start schedule to do more poorly with a longer biking distance.

However, the increase in race distance for biking has another effect when the total distance of the triathlon is kept constant. Of the three events, biking involves the fastest speed; triathletes travel much faster when biking than when running or swimming. We would expect the race to finish more quickly if more of the race distance was for biking rather than the two other slower events.

8.3 Course 3: Increased Running Distance

For course 3, the race distance of the running event is doubled to 20 km and the race distance of the biking event is decreased to 30 km. The total race distance remains constant at 51.5 km.

The length of the running event has a significant impact on the amount of congestion that occurs. We predict that if the distance is short, less congestion will occur because a greater number of slower athletes will finish the race without the faster athletes behind them ever catching up. However, a longer running distance can lead to high congestion just before the finish line. In order to confirm this hypothesis, we increase the race distance of the running event.

8.4 Course 4: Decreased Overall Distance

For course 4, we decrease the swimming distance to 1.2 km, the biking distance to 32 km, and the running distance to 8 km in order to remain proportional relative to standard Olympic lengths.

We are also interested in an overall proportional decrease of the race length. While a shorter length would decrease the overall time of the race, it could also lead to higher rates of congestion as the same number of people remain in a smaller track. Also, comparing the ratios to the proportion of decrease would reveal how sensitive the closure time of a schedule is to changes in overall length.

8.5 Results

For each course, the best schedule (Schedule 4 with 2/4 interval delays) was re-tested by running three trials in the NetLogo model and calculating average metrics. To compare the results, we computed the ratio of the new metric values to the original ones (Table 12). A ratio greater than one indicates that the schedule performed better on the original Olympic triathlon course, and a ratio less than one indicates that the schedule is improved on the adjusted course.

Course	c ratio	t ratio	f ratio	M ratio
1	0.98	0.94	3.75	0.95
2	1.02	0.84	1.32	0.84
3	1.05	0.94	35.42	1.03
4	0.88	0.89	0.00	0.89

Table 12: Ratios of the new metric values to the original metric values of the best wave start schedule

Ratios were similarly computed for the worst wave start schedule, Schedule 0 with 4/6 interval delays

(Table 13).

Course	c ratio	t ratio	f ratio	M ratio
1	17.19	0.98	1.81	1.02
2	13.55	0.87	1.11	0.90
3	15.72	0.98	24.51	1.07
4	15.44	0.94	0.00	0.98

Table 13: Ratios of the new metric values to the original metric values of the worst wave start schedule

While there were no drastic differences in the composite rating, M, there were interesting trends in the ratios of the c, t, and f metrics. The time proportions, t, decreased in each of the eight cases even when all other metrics increased. Further investigation is required to understand why this occurs. We hypothesize from the higher f value that more of the slowest runners are eliminated by the cutoff schedule, resulting in faster times for the athletes that remain. It is also interesting to note how greatly the congestion increased from the original value under the worst schedule.

The increase in swimming difference did decrease congestion and composite rating under the best schedule slightly, but the change was incremental. Increasing the biking distance did, in fact, both increase congestion and decrease time as expected. The change in running distance had a drastically negative effect on the cut proportions under both schedules and increased the values of all metrics except for t, indicating that increases in running distance give very little improvement.

As expected, course 4 resulted in reduced cut proportions and composite ratings. In fact, the cut proportion for both schedules was zero. The cutoff schedule remained unchanged for course 4 because it depends only on the mayor's limit of 5.5 hours. However, the decreased overall distance caused closure times to be lower, even when congestion increased markedly under the worst schedule.

Overall, the change that improved congestion and road closure the most was course 4 (proportionately decreased distances), which is not surprising. However, if the total distance of the triathlon is fixed, the most advantageous change is to increase the biking distance; course 1 decreased the composite rating by 16% from the original under the best schedule and increased congestion only slightly.

9 Analysis

9.1 Removing Cutoffs

Like many open triathlons, this model used a cutoff schedule to ensure that the race would end by the mayor's 5.5 hour limit. However, the cutoff schedule results in some less experienced or slower runners to be removed from the triathlon early on, sometimes even before they are able to complete the swimming portion. As this is a new triathlon, we wish to encourage participants of all levels. We therefore consider running the triathlon without the complete cutoff system. In this scenario, the only cap is a limit of 5.5 hours from the start of the race to its end. Participants then have a chance to experience more of the triathlon, even if they are ultimately unable to complete the race in its entirety.

Each of the wave start schedules was run again with three trials and without the cutoff schedule. Overall, the removal of the cutoff schedule increased all metrics except for the cut proportion, f (Table 14). The minimum value of f decreased by 78% from 0.0009 to 0.002. This is to be expected; without the cutoff schedule, slow triathletes stay in the race for as long as possible and cause more congestion. The optimal wave start schedule was different from the original (Section 7): Schedule 2 with 2/4 interval delays.

Cutoff	Schedule	c	t	f	M
Off	2	0.00099	0.99917	0.0006	0.05088
On	4	0.00127	0.92126	0.0025	0.04733

Table 14: Optimal schedules for the triathlon with and without a cutoff schedule

9.2 Robustness of Composite Weighting

We assessed the robustness of the solutions yielded in Part I by adjusting the relative sizes of the weighting values w_t, w_f, and w_c. We set 0.05 as the minimum value of each parameter as all three must be represented. As minimizing congestion is a concern of both the Mayor and Super Tread Race Company, we also decided that w_c could not go below 0.5. As illustrated in Tables 15-17, our original solution of Schedule 2 remained the best solution through changes of +0.3 in w_c and up to +0.2 in w_t. Since all three add up to one, changes in these two parameters imply changes in the third, w_f. Therefore, the specific weighting of our composite rating can sustain wide variation before a change in output. This further demonstrates the high robustness and easily adapted nature of our probability-based model.

2/4		w_t				
		-0.2	**-0.1**	**0**	**+0.1**	**+0.2**
w_c	**0**	Y	Y	Y	Y	Y
	+0.1	Y	Y	Y	N	
	+0.2	Y	Y	N		
	+0.3	Y	N			
	+0.4	N				

Table 15: Weighting adjustment for 2/4 time interval set. Y=Yes, Schedule 4 remains the best. N=N, Schedule 4 is no longer the best. w_c is the weight value of congestion and w_t is the weight value of road closure.

3/5		w_t				
		-0.2	**-0.1**	**0**	**+0.1**	**+0.2**
w_c	**0**	Y	Y	Y	Y	Y
	+0.1	Y	Y	Y	Y	
	+0.2	Y	Y	N		
	+0.3	Y	N			
	+0.4	N				

Table 16: Weighting adjustment for 3/5 time interval set. Y=Yes, Schedule 4 remains the best. N=N, Schedule 4 is no longer the best. w_c is the weight value of congestion and w_t is the weight value of road closure.

4/6		w_t				
		-0.2	**-0.1**	**0**	**+0.1**	**+0.2**
w_c	**0**	Y	Y	Y	Y	Y
	+0.1	Y	Y	Y	Y	
	+0.2	Y	Y	N		
	+0.3	Y	N			
	+0.4	N				

Table 17: Weighting adjustment for 4/6 time interval set. Y=Yes, Schedule 4 remains the best. N=N, Schedule 4 is no longer the best. w_c is the weight value of congestion and w_t is the weight value of road closure.

9.3 Strengths and Weaknesses

This model had a number of strengths and weaknesses:

- **Strength:** Our model uses probabilistic methods to calculate data.

  Because we use previous data but add random variance, we are able to simulate realistic scenarios that would arise. A deterministic model would not be able to factor the significant changes that occur in the real world.

- **Strength:** Our model simulates and analyzes a large number of schedules, wave spacing, and deviations.

  By simulating several combinations of conditions, we can more effectively assess the each combination and analyze their resulting data.

- **Strength:** Our model is flexible and readily adaptable.

  By using NetLogo, we can very quickly and easily program schedules and wave timings, while easily switching between configurations. The experimentation capabilities of NetLogo allow us to automatically compile large quantities of data in spreadsheets. Our many inputs allow a wide variety of schedules to be fine-tuned.

- **Strength:** Our solution presents an improvement in the given 5.5 hour limit with little congestion.

  Our optimal wave start schedule is under the mayor's requested limit, and the Super Tread Race Company will be pleased to know that the wave schedule that we found to be optimal has an overall congestion score of less than one seventh of a percent.

- **Weakness:** Our model cannot assess all potential combinations of schedules and wave timings.

  The ideal, but though unreasonable, solution would test each of the thousands of possible wave start schedules. Instead, our model compares a large set of schedules to determine the best one. Because our model can only compare schedules, it might not find the most optimal schedule.

- **Weakness:** Our model does not test the significance or sensitivity of the "pass probability" factor.

  This "pass probability" factor has a potentially large effect on our model. Even small changes in pass probability may decrease or increase congestion and its effects by a significant amount.

- **Weakness:** Our model has a limited number of trials for each combination of factors.

  Though we are able to analyze data with our 5 trials each, due to the probabilistic nature of the model, these trials may not accurately reflect the real-world system. More trials would allow us to more accurately analyze our model.

10 References

1. “Event Organizer’s Guide.” Triathlon Scotland. Triathlon Scotland, n.d. Web. 13 Nov. 2016. http://www.triathlonscotland.org/files/Event_organisers_guide.pdf.

2. Garrett, William E.; Kirkendall, Donald T. (2000). Exercise and sport science. Lippincott Williams & Wilkins. p. 919. ISBN 978-0-683-03421-9.

3. “ITU Competition Rules” (PDF). International Triathlon Union. 14 December 2015. Retrieved 13 November 2016

4. ”ITU Event Organizer’s Manual.” International Triathlon Union. International Triathlon Union, 11 Dec. 2015. Web. 13 Nov. 2016. https://s3.amazonaws.com/triathlon-images/uploads/docs/EVENT%20 ORGANISERS%20MANUAL%202015%20ALL%20SECTIONS.pdf.

5. ”Lane Width.” National Association of City Transportation Officials. National Association of City Transportation Officials, n.d. Web. 13 Nov. 2016.

6. “USA Triathlon Competitive Rules”. USA Triathlon. January 2013. Retrieved November 13, 2016.

7. Wilensky, U. (1999). NetLogo. http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

A Q-Q Normality Plots

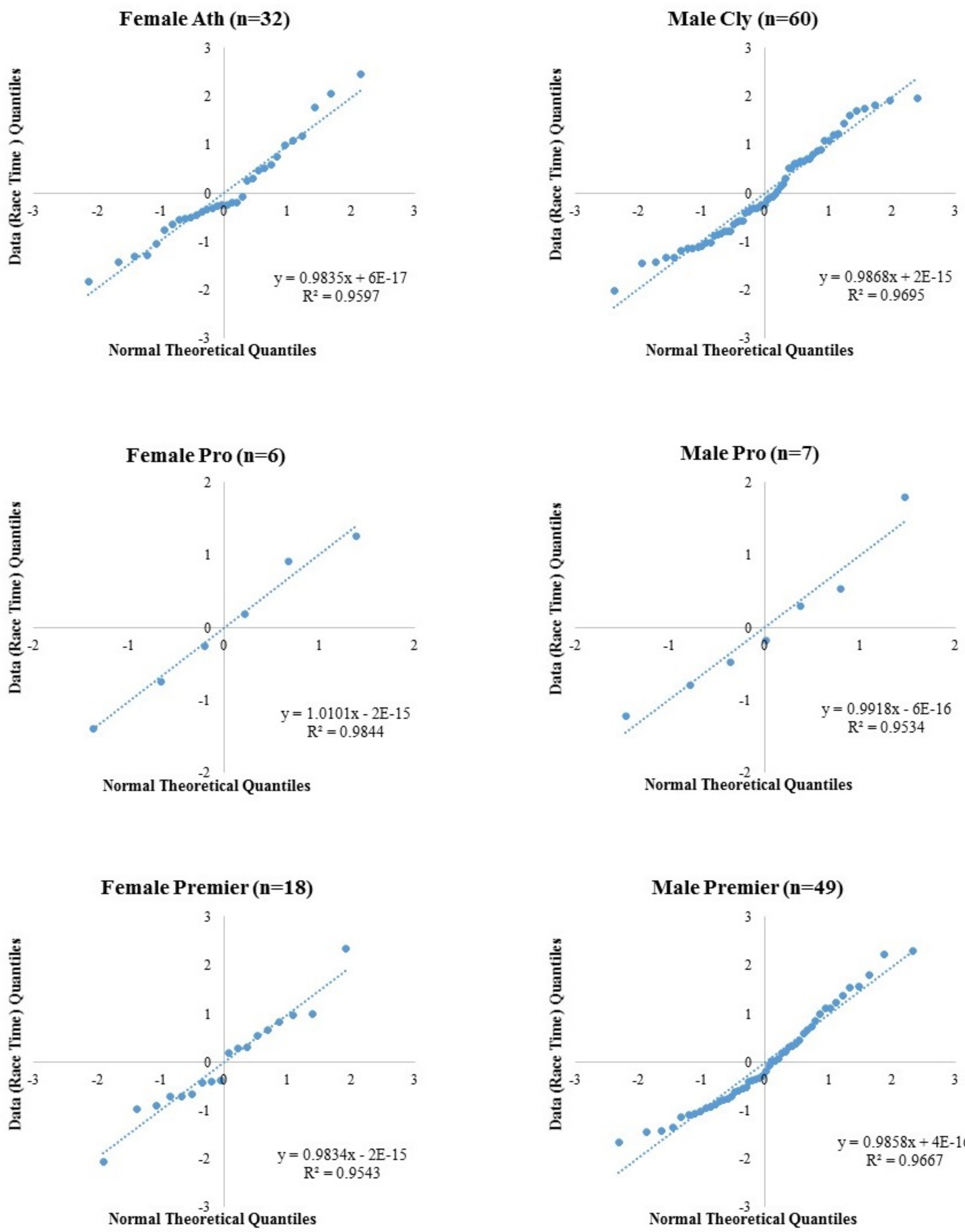
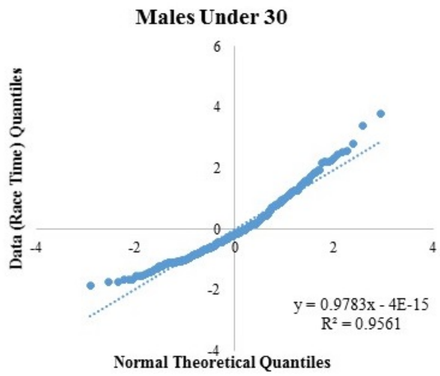
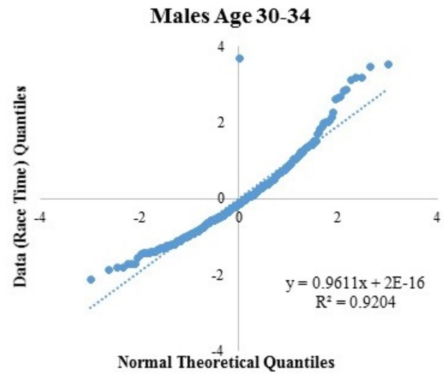
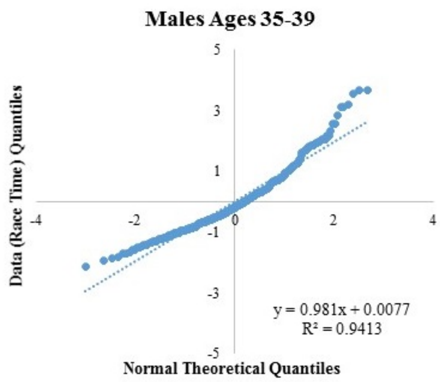
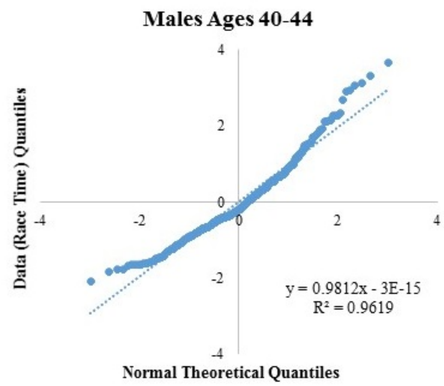


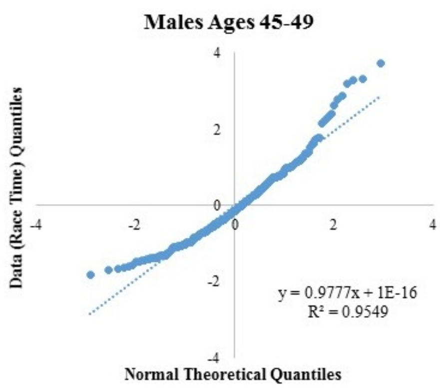
Figure 9: Q-Q Plots of Normality for Female Athena, Male Clydesdale, Female Pro, Male Pro, Female Premier, and Male Premier.











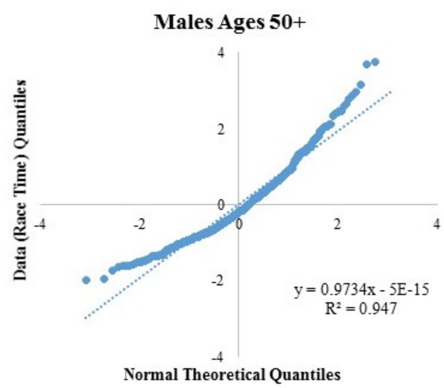


Figure 10: Q-Q Plots of Normality for Male Open Age Divisions.

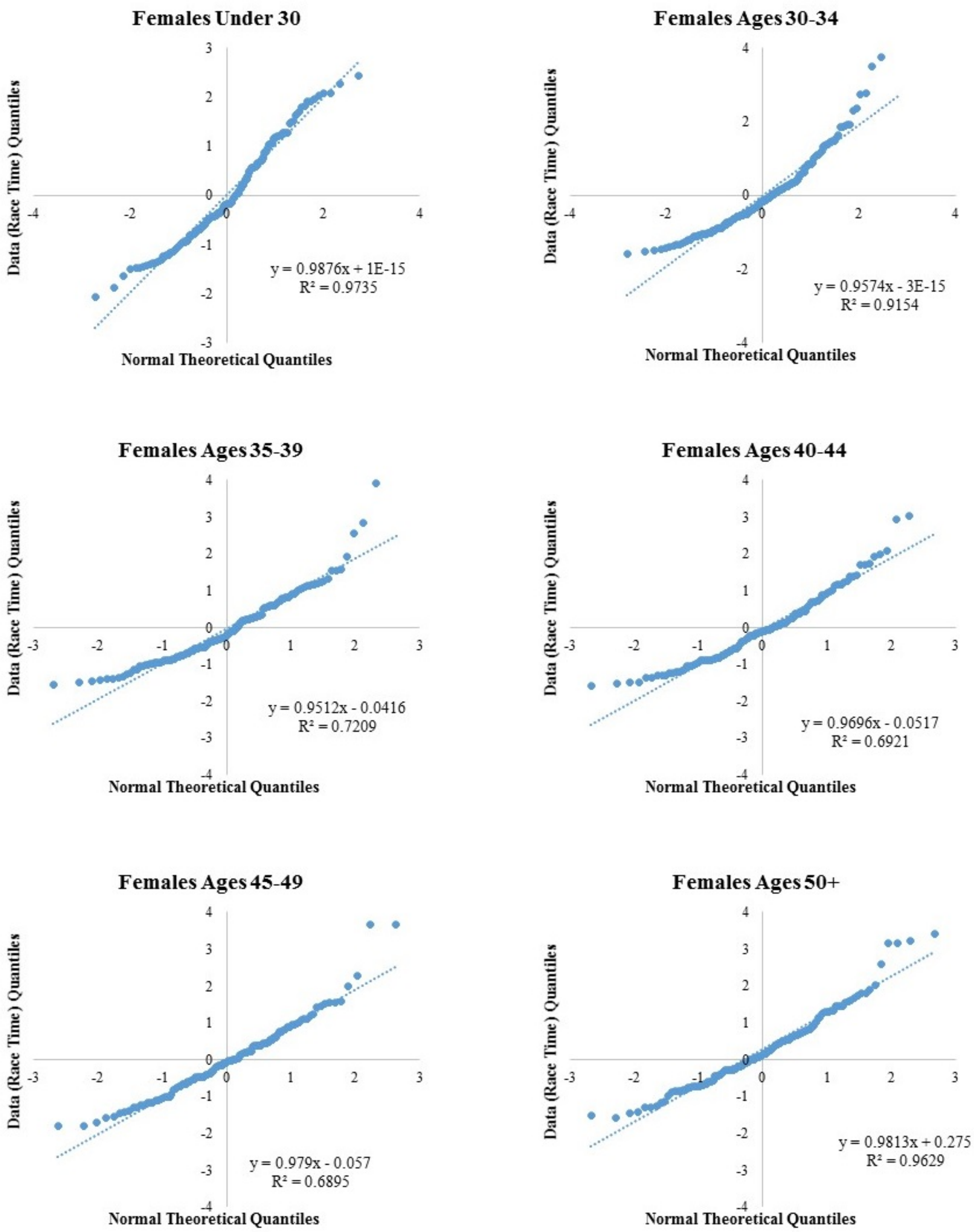


Figure 11: Q-Q Plots of Normality for Female Open Age Divisions.

B Code

The following is the code for the NetLogo model used in this paper:

```
turtles-own [
```

```
 wave
 gender
 division
 age
 swim-speed
 run-speed
 bike-speed
 t1-speed
 t2-speed
 curr-speed
 cong-speed
 dev
 congested?
 d
 x
 delay
 congestedness
]

globals [
  sorted-turtles
  ripcounter
  max-congested
  total-total-congestion
  curr-total-congestion
  total-num-congestion
  all-turtles
  num-turtles
  cong-mpro
  cong-fpro
  cong-mprem
  cong-fprem
  cong-mopenu30
  cong-mopenu35
  cong-mopenu40
  cong-mopenu45
  cong-mopenu50
  cong-mopen50plus
  cong-fopenu30
  cong-fopenu35
  cong-fopenu40
  cong-fopenu45
  cong-fopenu50
  cong-fopen50plus
  cong-cly
  cong-ath
  nonfinishers
  swimlength
  bikelength
  runlength
]

;; Main setup function
to setup
```

```
  clear-all
  reset-ticks
  reset-timer
  set num-turtles 2000
  crt num-turtles
  setup-turtles
  setup-length
  setup-patches
end

;; Initializes variables for turtles, assigns division, speed, and wave
to setup-turtles
  ask turtles[
    set congested? false
    set division "open"
    set size 5
    set shape "person"
    setxy 0 max-pycor / 2
    facexy 1 ycor
    ifelse gender = "male" [set color sky][set color magenta]
    set color white
    setup-division
    while[swim-speed <= 0 or t1-speed <= 0 or bike-speed <= 0 or t2-speed <= 0 or run-speed <= 0][
      setup-speed
    ]
    set all-turtles (list turtles)
    set all-turtles shuffle all-turtles
    setup-waves
  ]
end

;; Sets up race lengths based on input
to setup-length
  if length-config = 0 [
    set swimlength 1.5
    set bikelength 40
    set runlength 10
  ]
  if length-config = 1 [
    set swimlength 2
    set bikelength 39.5
    set runlength 10
  ]
  if length-config = 2 [
    set swimlength 1
    set bikelength 45.5
    set runlength 5
  ]
  if length-config = 3 [
    set swimlength 1.5
    set bikelength 30
    set runlength 20
  ]
  if length-config = 4 [
```

```
    set swimlength 1.2
    set bikelength 32
    set runlength 8
  ]
end

;; Sets pcolor for each patch based on pxcor
to setup-patches
  ask patches[
    set pcolor blue
    if pxcor > swimlength * 10 [set pcolor green]
    if pxcor > swimlength * 10 + 5 [set pcolor yellow]
    if pxcor > bikelength * 10 + swimlength * 10 + 5 [set pcolor orange]
    if pxcor > bikelength * 10 + swimlength * 10 + 10 [set pcolor red]
  ]
end

;; Determines whether a turtle should be cut off
to cutoff
  let t ticks - (pro-open-delay + 8 * open-open-delay)
  if not(t = 2700 or t = 10800 or t = 16200) [stop]
  ask turtles[
  ifelse t = 2700 and xcor <= swimlength * 10[
    set num-turtles num-turtles - 1
    set nonfinishers nonfinishers + 1
    die
  ][
  ifelse t = 10800 and xcor <= bikelength * 10 + swimlength * 10 + 5[
    set num-turtles num-turtles - 1
    set nonfinishers nonfinishers + 1
    die
  ][
  if t = 16200 and xcor <= max-pxcor[
    set num-turtles num-turtles - 1
    set nonfinishers nonfinishers + 1
    die
  ]]]
  ]

end

;; Sets up wave
to setup-waves
  if config = 0 [
    set wave random 10
  ]
  if config = 1 [
    ifelse division = "premier" or division = "pro"[
      set wave 0
    ][
    ifelse age = "<30"[
      set wave 1
    ][
    ifelse gender = "male" and age = "30-34"[
```

```
    set wave 2
  ][
  ifelse gender = "female" and (age = "30-34" or age = "35-59") [
    set wave 3
  ][
  ifelse gender = "male" and age = "35-39"[
    set wave 4
  ][
  ifelse gender = "female" and (age = "40-44" or age = "45-49") [
    set wave 5
  ][
  ifelse gender = "male" and age = "40-44"[
    set wave 6
  ][
  ifelse gender = "female" and age = "50+"[
    set wave 7
  ][
  ifelse gender = "male" and age = "50+"[
    set wave 8
  ][ ;; cly and ath
    set wave 9
  ]]]]]]]]
]
if config = 2 [
  ifelse division = "premier" or division = "pro"[
    set wave 0
  ][
  ifelse (gender = "male" and age = "<30") or division = "ath" or division = "cly"[
    set wave 1
  ][
  ifelse gender = "male" and age = "30-34"[
    set wave 2
  ][
  ifelse gender = "male" and age = "35-39" [
    set wave 3
  ][
  ifelse gender = "male" and age = "40-44"[
    set wave 4
  ][
  ifelse gender = "male" and age = "45-49" [
    set wave 5
  ][
  ifelse gender = "male" and age = "50+"[
    set wave 6
  ][
  ifelse gender = "female" and (age = "<30" or age = "30-34") [
    set wave 7
  ][
  ifelse gender = "female" and (age != "50+") [
    set wave 8
  ][
    set wave 9
  ]]]]]]]]]
]
```

```
if config = 3 [
  ifelse division = "premier" or division = "pro"[
    set wave 9
  ][
  ifelse age = "<30"[
    set wave 8
  ][
  ifelse gender = "male" and age = "30-34"[
    set wave 7
  ][
  ifelse gender = "female" and (age = "30-34" or age = "35-59") [
    set wave 6
  ][
  ifelse gender = "male" and age = "35-39"[
    set wave 5
  ][
  ifelse gender = "female" and (age = "40-44" or age = "45-49") [
    set wave 4
  ][
  ifelse gender = "male" and age = "40-44"[
    set wave 3
  ][
  ifelse gender = "female" and age = "50+"[
    set wave 2
  ][
  ifelse gender = "male" and age = "50+"[
    set wave 1
  ][ ;; cly and ath
    set wave 0
  ]]]]]]]]]
]
if config = 4 [
  ifelse division = "premier" or division = "pro"[
    set wave 0
  ][
  ifelse division = "ath" or division = "cly"[
    set wave 1
  ][
  ifelse age = "<30"[
    set wave 2
  ][
  ifelse gender = "male" and age = "50+"[
    set wave 3
  ][
  ifelse gender = "male" and age = "30-34+"[
    set wave 4
  ][
  ifelse gender = "female" and age = "50+" or (gender = "male" and age = "45-49") [
    set wave 5
  ][
  ifelse gender = "female" and (age = "30-34" or age = "35-39")[
    set wave 6
  ][
  ifelse gender = "male" and age = "40-44"[
```

```
      set wave 7
    ][
    ifelse gender = "male" and age = "35-39"[
      set wave 8
    ][
      set wave 9
    ]]]]]]]]
  ]
  ifelse wave = 0[
    set delay 0
  ][
  set delay pro-open-delay + ((wave - 1) * open-open-delay)
  ]
end

;; Sets up division based on "who", or id
to setup-division
  ifelse who < 4[
    set gender "male"
    set division "pro"
  ][ifelse who < 8[
    set gender "female"
    set division "pro"
  ][ifelse who < 39[
    set gender "male"
    set division "premier"
  ][ifelse who < 50[
    set gender "female"
    set division "premier"
  ][ifelse who < 225[
    set gender "male"
    set division "open"
    set age "<30"
  ][ifelse who < 451[
    set gender "male"
    set division "open"
    set age "30-34"
  ][ifelse who < 690[
    set gender "male"
    set division "open"
    set age "35-39"
  ][ifelse who < 991[
    set gender "male"
    set division "open"
    set age "40-44"
  ][ifelse who < 1087[
    set gender "male"
    set division "open"
    set age "45-49"
  ][ifelse who < 1385[
    set gender "male"
    set division "open"
    set age "50+"
  ][ifelse who < 1483[
```

```
    set gender "female"
    set division "open"
    set age "<30"
  ][ifelse who < 1613[
    set gender "female"
    set division "open"
    set age "30-34"
  ][ifelse who < 1703[
    set gender "female"
    set division "open"
    set age "35-39"
  ][ifelse who < 1785[
    set gender "female"
    set division "open"
    set age "40-44"
  ][ifelse who < 1858[
    set gender "female"
    set division "open"
    set age "45-49"
  ][ifelse who < 1943[
    set gender "female"
    set division "open"
    set age "50+"
  ][ifelse who < 1980[
    set gender "male"
    set division "cly"
  ][if who < 2000[
    set gender "female"
    set division "ath"
  ]]]]]]]]]]]]]]]]]]
end

;; Sets up speeds based on division and devation
to setup-speed
  set dev random-normal 0 1
  ifelse gender = "male"[
    ifelse division = "open" [
        ifelse age = "<30"[
          set swim-speed 1424 + (237 * dev)
          set t1-speed 420 + (162 * dev)
          set bike-speed 5178 + (797 * dev)
          set t2-speed 153 + (66 * dev)
          set run-speed 3454 + (680 * dev)
        ][
        ifelse age = "30-34"[
          set swim-speed 1457 + (244 * dev)
          set t1-speed 409 + (154 * dev)
          set bike-speed 4994 + (705 * dev)
          set t2-speed 167 + (77 * dev)
          set run-speed 3491 + (682 * dev)
        ][
        ifelse age = "35-39"[
          set swim-speed 1514 + (258 * dev)
          set t1-speed 434 + (170 * dev)
```

```
        set bike-speed 5006 + (661 * dev)
        set t2-speed 182 + (89 * dev)
        set run-speed 3516 + (594 * dev)
      ][
      ifelse age = "40-44"[
        set swim-speed 1223 + (189 * dev)
        set t1-speed 533 + (184 * dev)
        set bike-speed 5135 + (649 * dev)
        set t2-speed 181 + (94 * dev)
        set run-speed 3372 + (570 * dev)
      ][
      ifelse age = "45-49"[
        set swim-speed 1424 + (232 * dev)
        set t1-speed 394 + (135 * dev)
        set bike-speed 4952 + (582 * dev)
        set t2-speed 172 + (88 * dev)
        set run-speed 3648 + (748 * dev)
      ][
      if age = "50+"[
        set swim-speed 1370 + (261 * dev)
        set t1-speed 518 + (200 * dev)
        set bike-speed 5230 + (732 * dev)
        set t2-speed 210 + (105 * dev)
        set run-speed 3798 + (769 * dev)
      ]]]]]]
  ][
  ifelse division = "pro"[
    set swim-speed 788 + (73 * dev)
    set t1-speed 196 + (20 * dev)
    set bike-speed 3730 + (175 * dev)
    set t2-speed 51 + (5 * dev)
    set run-speed 2120 + (112 * dev)
  ][
  ifelse division = "premier"[
    set swim-speed 1000 + (86 * dev)
    set t1-speed 264 + (34 * dev)
    set bike-speed 4118 + (283 * dev)
    set t2-speed 75 + (19 * dev)
    set run-speed 2385 + (212 * dev)
  ][
  if division = "cly"[
    set swim-speed 1538 + (226 * dev)
    set t1-speed 483 + (191 * dev)
    set bike-speed 5296 + (756 * dev)
    set t2-speed 208 + (127 * dev)
    set run-speed 4281 + (809 * dev)
  ]]]]
][
  ifelse division = "open"[
    ifelse age = "<30"[
      set swim-speed 1317 + (175 * dev)
      set t1-speed 572 + (179 * dev)
      set bike-speed 5884 + (802 * dev)
      set t2-speed 208 + (112 * dev)
```

```
    set run-speed 3597 + (594 * dev)
  ][
  ifelse age = "30-34"[
    set swim-speed 1338 + (203 * dev)
    set t1-speed 588 + (208 * dev)
    set bike-speed 5937 + (994 * dev)
    set t2-speed 216 + (124 * dev)
    set run-speed 3647 + (633 * dev)
  ][
  ifelse age = "35-39"[
    set swim-speed 1369 + (229 * dev)
    set t1-speed 616 + (205 * dev)
    set bike-speed 5956 + (986 * dev)
    set t2-speed 220 + (111 * dev)
    set run-speed 3709 + (733 * dev)
  ][
  ifelse age = "40-44"[
    set swim-speed 1380 + (205 * dev)
    set t1-speed 607 + (197 * dev)
    set bike-speed 5890 + (822 * dev)
    set t2-speed 226 + (132 * dev)
    set run-speed 3744 + (701 * dev)
  ][
  ifelse age = "45-49"[
    set swim-speed 1322 + (172 * dev)
    set t1-speed 601 + (210 * dev)
    set bike-speed 5813 + (836 * dev)
    set t2-speed 216 + (103 * dev)
    set run-speed 3756 + (630 * dev)
  ][
  if age = "50+"[
    set swim-speed 1376 + (208 * dev)
    set t1-speed 639 + (233 * dev)
    set bike-speed 5871 + (846 * dev)
    set t2-speed 252 + (145 * dev)
    set run-speed 4069 + (850 * dev)
  ]
  ]]]]]
][
ifelse division = "pro"[
  set swim-speed 778 + (21 * dev)
  set t1-speed 218 + (10 * dev)
  set bike-speed 4058 + (146 * dev)
  set t2-speed 55 + (9 * dev)
  set run-speed 2299 + (119 * dev)
][
ifelse division = "premier"[
  set swim-speed 1022 + (84 * dev)
  set t1-speed 297 + (38 * dev)
  set bike-speed 4576 + (276 * dev)
  set t2-speed 86 + (29 * dev)
  set run-speed 2648 + (225 * dev)
][
if division = "ath"[
```

```
      set swim-speed 1405 + (213 * dev)
      set t1-speed 698 + (219 * dev)
      set bike-speed 6539 + (1162 * dev)
      set t2-speed 271 + (112 * dev)
      set run-speed 4646 + (837 * dev)
    ]]]]
  ]

end

;; Determine how much a turtle moves and moves it
to move
  if pcolor = blue[set curr-speed 1.5 / swim-speed]
  if pcolor = green[set curr-speed 0.5 / t1-speed]
  if pcolor = yellow[set curr-speed 40 / bike-speed]
  if pcolor = orange[set curr-speed 0.5 / t2-speed]
  if pcolor = red[set curr-speed 10 / run-speed]
  if ticks > delay[
    ifelse congested?[
      fd cong-speed * 10
    ][
    fd curr-speed * 10
    ]
  ]
end

;; Track congestion for the interface
to update-monitors
  ifelse gender = "male"[
    ifelse division = "open" [
      ifelse age = "<30"[
        set cong-mopenu30 cong-mopenu30 + congestedness
      ][
      ifelse age = "30-34"[
        set cong-mopenu35 cong-mopenu35 + congestedness
      ][
      ifelse age = "35-39"[
        set cong-mopenu40 cong-mopenu40 + congestedness
      ][
      ifelse age = "40-44"[
        set cong-mopenu45 cong-mopenu45 + congestedness
      ][
      ifelse age = "45-49"[
        set cong-mopenu50 cong-mopenu50 + congestedness
      ][
      if age = "50+"[
        set cong-mopen50plus cong-mopen50plus + congestedness
      ]]]]]]
    ][
    ifelse division = "pro"[
      set cong-mpro cong-mpro + congestedness
    ][
    ifelse division = "premier"[
      set cong-mprem cong-mprem + congestedness
```

```
    ][
    if division = "cly"[
      set cong-cly cong-cly + congestedness
    ]]]]
  ][
  ifelse division = "open" [
    ifelse age = "<30"[
      set cong-fopenu30 cong-fopenu30 + congestedness
    ][
    ifelse age = "30-34"[
      set cong-fopenu35 cong-fopenu35 + congestedness
    ][
    ifelse age = "35-39"[
      set cong-fopenu40 cong-fopenu40 + congestedness
    ][
    ifelse age = "40-44"[
      set cong-fopenu45 cong-fopenu45 + congestedness
    ][
    ifelse age = "45-49"[
      set cong-fopenu50 cong-fopenu50 + congestedness
    ][
    if age = "50+"[
      set cong-fopen50plus cong-fopen50plus + congestedness
    ]]]]]]
  ][
  ifelse division = "pro"[
    set cong-fpro cong-fpro + congestedness
  ][
  ifelse division = "premier"[
    set cong-fprem cong-fprem + congestedness
  ][
  if division = "ath"[
    set cong-ath cong-ath + congestedness
  ]]]]
  ]
end

;; Determine if each turtle is congested
to eval-congested
  let i 0
  foreach sorted-turtles [
    ask ?[
    if i >= num-turtles - 1[
      set congested? false
      stop
    ]
    ifelse xcor = 0 or xcor >= max-pxcor or pcolor = green or pcolor = orange[
      set congested? false
    ][
    ifelse pcolor = yellow[
      set x 6
      set d 3
    ][
    ifelse pcolor = red [
```

```
      set x 12
      set d 2
    ][
    if pcolor = blue [
      set x 30
      set d 3
    ]]]
    let j i + 1
    let num-in-front 0
    let num-slower 0
    let total-slower 0
    let other-turtle (item j sorted-turtles)
    while[j < num-turtles and
      [xcor] of other-turtle <= xcor + d and
      num-in-front < x
    ][
      if [xcor] of other-turtle >= max-pxcor[stop]
      let s [curr-speed] of other-turtle
      if s < curr-speed[
        set num-slower num-slower + 1
        set total-slower total-slower + s
      ]
      set j (j + 1)
      set num-in-front (num-in-front + 1)
      if j < num-turtles[set other-turtle (item j sorted-turtles)]
    ]
    set congested? (num-slower >= x and num-slower > 0 and total-slower / num-slower <= curr-speed)
        and random-float 1 > pass-probability / 100
    if not(congested?)[
      set congestedness 0
    ]
    if congested? and num-slower > 0[
      set congestedness (curr-speed - total-slower / num-slower) / curr-speed
      if not(division = "pro" or division = "premier")[
        set curr-total-congestion curr-total-congestion +
            (curr-speed - total-slower / num-slower) / curr-speed
      ]
      set cong-speed total-slower / num-slower
    ]
    ]
    ]
    set i (i + 1)
  ]
end

;; Main loop
to go
  ifelse any? turtles with [xcor < max-pxcor] and ticks <= 19800[
  set sorted-turtles sort-on [xcor] turtles
  set curr-total-congestion 0
  eval-congested
  set total-total-congestion total-total-congestion + curr-total-congestion
  ask turtles[
    ifelse congested? [
```

```
      set color black set total-num-congestion total-num-congestion + 1
    ][
      set color white
    ]
    update-monitors
    move
  ]
  if cutoff-on = "on" [cutoff]
  if count turtles with [congested?] > max-congested[
    set max-congested count turtles with [congested?]
  ]
  tick
  ][
  show timer
  stop
  ]
end
```