7 Appendices

7.1 Discussion of Numerical Solutions and Unique Solutions

7.1.1 Common Edge Networks - Exponential Edge Defense Function

As stated in the text, the defender's problem is one of constrained optimization. For the exponential defense function, denoting the amount allocated to non-common edges as x, the amount allocated to the common edge as y, and also denoting α as a, his objective function with the budget constraint substituted for x is the following:

$$\left(1-e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right)\left(\left(1-e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right)e^{-\left(-\log\left(1-e^{-\frac{y}{z}}\right)\right)^{a}} + e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right) + e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}} \tag{3}$$

Differentiating with respect to y yields the following first order condition:

$$-\frac{a\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}}{4z\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)}\left(\left(\left(1-e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right)e^{-\left(-\log\left(1-e^{-\frac{y}{z}}\right)\right)^{a}} + e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right) + \frac{a\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}}{4z\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)} + \left(1-e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right)\left(-\frac{a\left(-\log\left(1-e^{-\frac{y}{z}}\right)\right)^{a}e^{-\left(-\log\left(1-e^{-\frac{y}{z}}\right)\right)^{a}}e^{-\frac{y}{z}}}{z\left(1-e^{-\frac{y}{z}}\right)\log\left(1-e^{-\frac{y}{z}}\right)}\left(1-e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}\right) + \frac{a\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}}{4z\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)} - \frac{a\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}e^{-\left(-\log\left(1-e^{-\frac{y}{z}}\right)\right)^{a}}e^{-\left(-\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\right)^{a}}e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}}{4z\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)\log\left(1-e^{-\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)}\right)}\right)\right) = 0 \tag{4}$$

Clearly it is practically infeasible to obtain a closed form solution for y in terms of B, z, and a. However, it is trivial to compute the above first order condition for a given y, B, z, and a, meaning it is feasible to numerically calculate the optimal allocations. Figure 10 presents such an analysis graphically for selected combinations of B, z, and a.

The top left graph in Figure 10 is an example of where the budget is very high, which makes the first order condition (in blue) undefined for large y due to the term $log(1-e^{\frac{B-y}{4z}})$. The inner term tends to zero as $B-y$ increases, which then implies $log(0)$ which is mathematically impossible. As the first order condition does not cross (but approaches) zero before it becomes undefined, we turn to a numerical approach that maximizes the perceived probability of overall defense (in black). For this case the solution is not unique as multiple allocations can obtain a perceived probability of what is computationally indistinguishable from one due to machine precision. We consider this an appropriate prediction as our computer is operating on levels of precision far in excess of human subjects. We find numerically that the excess budget issue occurs when $B > 37.42z$, which is independent of α as $w_p(1; \alpha) = 1 \quad \forall \alpha \in (0, 1]$. This constraint on the budget is not relevant for the parameters we used in the experiment ($B = 24$, $z = 18.2$, $z = 31.1$).

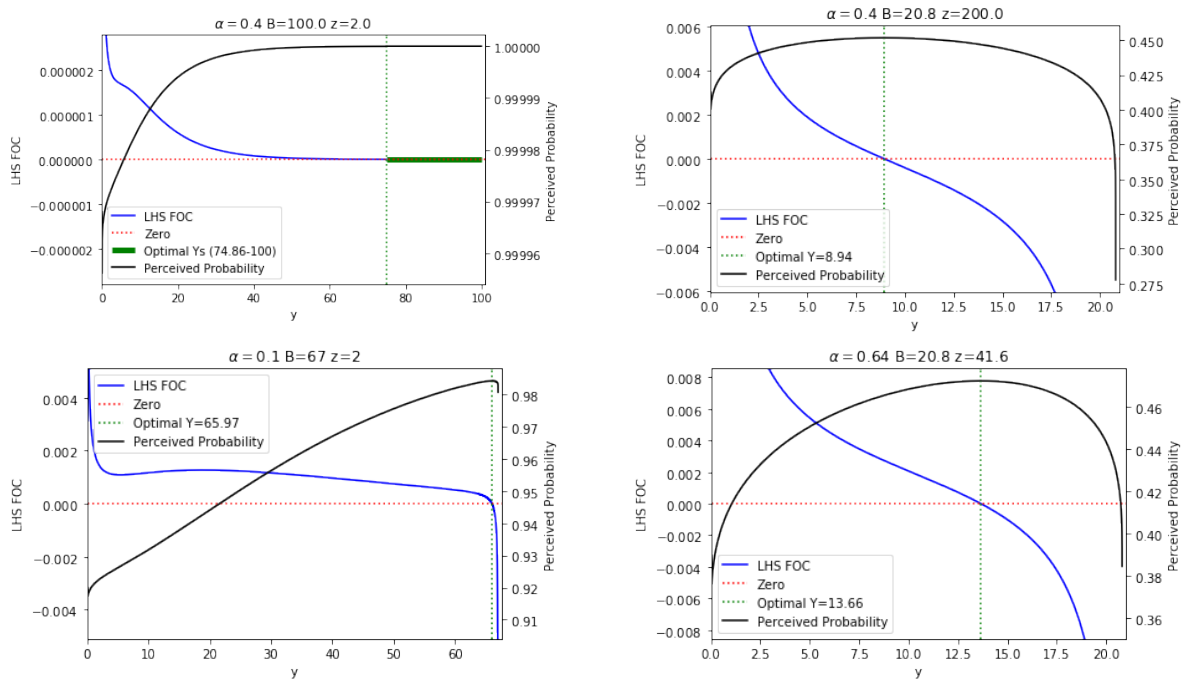


Figure 10: Selected Numerical Solutions for Common Edge and $p(x_i) = 1 - e^{\frac{-x_i}{z}}$

Several general patterns are evident from the graphs in Figure 10. Firstly, the first order condition is always positive as y tends to zero. Secondly, the first order condition is either negative or undefined as y tends to B. Finally, the first order condition is often always decreasing in y, but not always. For example, the graph in the bottom left corner shows a slight increase in the first order condition for the range about $y \in [5, 20]$. Because the first order condition is not always decreasing the optimal solution is not always uniquely defined by the first order condition since it is possible for it to cross zero more than once. However, this is uncommon for common edge networks. To investigate this we conducted a grid search of 10 equally spaced points over the parameters $\alpha \in [.2, .99]$, $B \in [1, 1000]$ and $z \in [2, 200]$, while imposing the condition $B < 37.42z$ as we have already established that if the budget is too large the solution is non-unique. This results in 7740 combinations of parameters. We confirm our first and second identified patterns, that for all of the combinations of parameters we consider the first order condition is positive when y tends to 0 and negative when y tends to B. We also find that in the vast majority (7729/7740) of our parameters the first order condition is always decreasing, and so we can be generally comfortable in assuming the solution is unique for most combinations of parameter sets. As for the 11 combinations of parameters where the first order condition is not always decreasing, they are characterized by low levels of α (typically $\alpha < .25$). For these combinations of parameters, we check their uniqueness in two ways. Firstly, we see whether the first order condition ever becomes positive again after first crossing zero. Nine of the eleven combinations pass this test, while the remaining two are a precision error, confirmed graphically as well as by requiring the subsequent observed positive first order condition to be over a small threshold. Secondly, we maximize the overall perceived probability of defense over a fine grid of y, and see if this set has more than one point in it. All eleven combinations pass this test. Based on this analysis, for the range of parameters we consider, we can be reasonably confident that the solutions are unique in the common edge network with the exponential edge defense function.

7.1.2 Common Edge Network - Alternative Edge Defense Function

We now consider the edge defense function $p(x_i) = (\frac{x_i}{z})^b$. The objective function is as follows:

$$
\begin{aligned}
&e^{-(-b\log(y)+b\log(z))^a} + 2e^{-\left(-\log\left(\left(\frac{B}{4z}-\frac{y}{4z}\right)^b\right)\right)^a} - 2e^{-\left(-\log\left(\left(\frac{B}{4z}-\frac{y}{4z}\right)^b\right)\right)^a} e^{-(-b\log(y)+b\log(z))^a} \\
&- e^{-2\left(-\log\left(\left(\frac{B}{4z}-\frac{y}{4z}\right)^b\right)\right)^a} + e^{-2\left(-\log\left(\left(\frac{B}{4z}-\frac{y}{4z}\right)^b\right)\right)^a} e^{-(-b\log(y)+b\log(z))^a}
\end{aligned}
\tag{5}
$$

With the associated first order condition:

$$
\begin{aligned}
&-\frac{ab\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}}{4\left(\frac{B}{4}-\frac{y}{4}\right)\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)}\Bigg(\Bigg(1 \\
&- e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}\Bigg) e^{-\left(-\log\left(\left(\frac{y}{z}\right)^b\right)\right)^a} + e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}\Bigg) \\
&+ \frac{ab\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}}{4\left(\frac{B}{4}-\frac{y}{4}\right)\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)} \\
&+ \left(1 - e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}\right)\left(\frac{ab\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}}{4\left(\frac{B}{4}-\frac{y}{4}\right)\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)}\right. \\
&- \frac{ab\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a e^{-\left(-\log\left(\left(\frac{y}{z}\right)^b\right)\right)^a}}{4\left(\frac{B}{4}-\frac{y}{4}\right)\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)} e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a} \\
&\left.- \frac{ab\left(-\log\left(\left(\frac{y}{z}\right)^b\right)\right)^a}{y\log\left(\left(\frac{y}{z}\right)^b\right)}\left(1 - e^{-\left(-\log\left(\left(\frac{1}{z}\left(\frac{B}{4}-\frac{y}{4}\right)\right)^b\right)\right)^a}\right) e^{-\left(-\log\left(\left(\frac{y}{z}\right)^b\right)\right)^a}\right) = 0
\end{aligned}
\tag{6}
$$

This first order condition is also analytically intractable, but very numerically tractable. We conduct the same exercise as for the exponential edge defense function, except we have an additional parameter b which we will keep between 0 and 1 as this is the particularly interesting case for this style of function (and $b = 0.4$ was actually used in the experiment). We also restrict $B < Z$, as the edge defense function is not a proper probability function otherwise ($B > Z$ would imply a probability greater than one). Figure 11 presents a graphical analysis for a selection of parameter combinations.

The patterns identified with the exponential edge defense function also hold here. The main difference is that in this environment it is possible for the first order condition to be very non-decreasing instead of just slightly, as shown in the top left graph. We consider a grid search of combinations of 20 equally spaced points of the parameters $z \in [2, 200]$, $\alpha \in [.4, .99]$, $B \in [1, 200]$ and $b \in [.1, .9]$, which yields 83600 combinations. We confirm that the first order condition is always positive as y tends to 0, but we find 27 instances of cases where the first order condition is not negative as y tends to B. These are characterized by high values of αs, Bs, zs, and bs, which are leading to an optimum value too close to B to observe the first order condition go negative before becoming undefined. For all 27 of these combinations, a numerical analysis of the maximum perceived probability confirms a unique optimum, as we cannot confirm uniqueness from the first order condition crossing zero. As expected, substantially more combinations (23456) exhibited a non-decreasing first order condition than in the exponential defense function case. For all of these combinations we confirm that the solution is unique in the same way as the exponential defense function, by observing whether the first order condition crosses the zero line more than once and by numerical maximization of perceived probability. Therefore, for the range of values we consider (which span the ones considered in the experiment), we can be confident that the solution is unique.

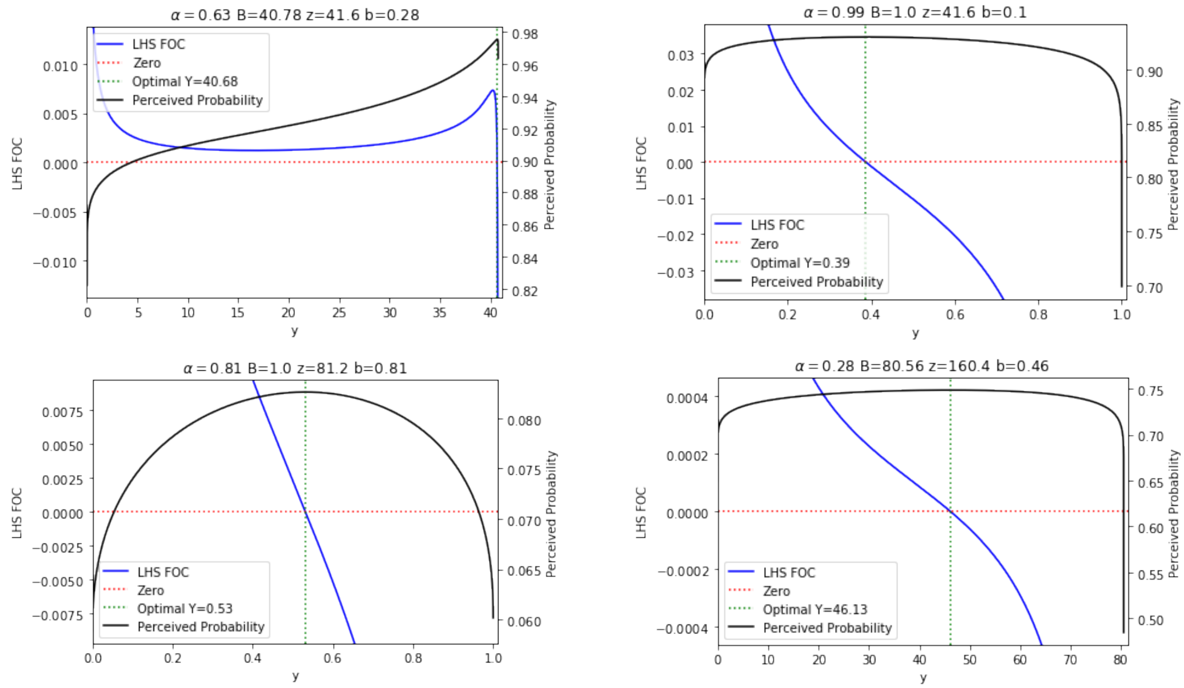


Figure 11: Selected Numerical Solutions for Common Edge and $p(x_i) = (\frac{x_i}{z})^b$

7.1.3 Network Green

We denote x as the number of units allocated to each edge in the path with 5 edges, and y as the number of units allocated to each edge in the path with 2 edges. The objective function contains a minimum operator. In order to express the objective function in a manner we can differentiate we can substitute in the constraints. The budget constraint is trivial to substitute, however there is also another implied constraint that each path should have the same perceived probability of defense. Substituting in this implied constraint would remove the need for the minimum operator, however this proves analytically intractable, as the following equation cannot be solved for x (or y for the other possible substitutable route) to obtain a substitutable closed form solution.

$$5e^{-\left(\frac{x}{z}-\log\left(e^{\frac{x}{z}}-1\right)\right)^{a}} - 10e^{-2\left(\frac{x}{z}-\log\left(e^{\frac{x}{z}}-1\right)\right)^{a}} + 10e^{-3\left(\frac{x}{z}-\log\left(e^{\frac{x}{z}}-1\right)\right)^{a}} - 5e^{-4\left(\frac{x}{z}-\log\left(e^{\frac{x}{z}}-1\right)\right)^{a}}$$
$$+ e^{-5\left(\frac{x}{z}-\log\left(e^{\frac{x}{z}}-1\right)\right)^{a}} - 2e^{-\left(-\log\left(-e^{\frac{1}{2z}(-B+5x)}+1\right)\right)^{a}} + e^{-2\left(-\log\left(-e^{\frac{1}{2z}(-B+5x)}+1\right)\right)^{a}} = 0 \quad (7)$$

An alternative route would be to form the associated Lagrangian. The two first order conditions (with respect to x and $\mathcal{L}$) can be obtained, however the same situation occurs where it is not analytically possible to substitute one of these first order conditions into the other. We can, however, conduct a similar exercise as before, except this time considering both first order conditions simultaneously. That is, we can easily calculate the value for both first order conditions for any combination of y, $\mathcal{L}$, α, B, and z, and find the combinations of y and $\mathcal{L}$ where both first order conditions are zero. For both ease of visual display as well as for a computer based minimizer we fold both first order conditions in one expression that should be minimized: $F = FOC_y^2 + FOC_{\mathcal{L}}^2$. This expression should equal zero when both first order conditions are zero, and be greater than zero otherwise. We present selected combinations of parameters in Figures 12, 13, and 14, which represent the three types of patterns that generally emerge. The top graph in each figure is a heatmap[25] of the expression representing both first order conditions, while the four following graphs are each first order condition holding one parameter fixed at the optimum value. The latter is to support the heatmap analysis, for example, the heatmap in Figure 13 looks rather flat along the $\mathcal{L}$ dimension, but the

[25]Note, the heatmap is truncated for high values of F so we can more easily focus on values where it is close to zero.

first order conditions graphs confirm that there is some movement along that dimension, and that both of the first order conditions are zero at the optimal point. It should also be noted that the perceived probability is overlaid on the heatmap, and that it is single peaked at the point where the top and bottom paths are equal in terms of perceived probability. To confirm the uniqueness of the solution for a wide range of parameters, we check whether the perceived probability is in fact singled peaked at the maximum (rather than having a non-unique flat maximum). We conduct an equally spaced grid search of size 40 along the parameter space of $\alpha \in [0.4, .99]$, $B \in [1, 1000]$, and $z \in [2, 200]$, with the restriction $2B < 37.42Z$. This yields 55400 combinations, all of which are single peaked at a maximum in terms of perceived probability. We conclude that for the range of parameters we consider, we can be confident that the solution is unique.

7.1.4 $\alpha = 1$ with Exponential Edge Defense Function

Consider a simple two path network with two edges along each edge, and no common edges (e.g. Figure 7). With an exponential defense function, and denoting the first edge as x and the second edge as y, the probability of a successful defense along a path is $1 - e^{\frac{-x}{z}} + e^{\frac{-x}{z}}(1 - e^{\frac{-x}{z}}) = 1 - e^{\frac{-x-y}{z}}$. Note that this is invariant to any x and y allocations given a fixed total T, where $x + y = T$. Therefore, any allocation that assigns a total of $\frac{B}{2}$ to each path is optimal for $\alpha = 1$ and $p(x_i) = 1 - e^{\frac{-x_i}{z}}$, and thus there is not a unique solution for this case. By the same logic, this can be shown for any number of edges along a path, the only requirement in that the total allocation along a path is split evenly. This is not the case for a network with one common edge, as with an exponential defense function an $\alpha = 1$ type would allocate all their units to the common edge and thus the solution would be unique. However, if there were two common edges, then any allocation that allocates all of their units across those two non-common edges would be optimal for $\alpha = 1$ due to the exponential defense function by the same logic as the case with the paths, and thus there would not be a unique solution.

7.1.5 $\alpha = 0$

In the Prelec probability weighting function, $w(p(x_i); \alpha) = \exp[-(-\log(p(x_i)))^{\alpha}]$, $\alpha = 0$ is a limiting case where every probability is perceived as $\frac{1}{e} = .368$. In our model, this means that the defender perceives every probability of a successful defense along an edge as $\frac{1}{e}$. The $\alpha = 0$ case is not very realistic, but solutions to any network structure and edge defense functions can be easily described. If one assumes that $w(0) = w(1) = \frac{1}{e}$, then literally any allocation is an optimal solution, so therefore there are no unique solutions. This is true for any network that could be specified. If instead one assumes that $w(0) = 0$ and $w(\epsilon) = \frac{1}{e}$, then any combination that allocates $x_i \geq \epsilon\ \forall i$ is an optimal solution, with additional possible solutions where $x_i = 0$ if i is an extraneous edge. For either assumption about the nature of $\alpha = 0$ there is no unique solution for any type of network.

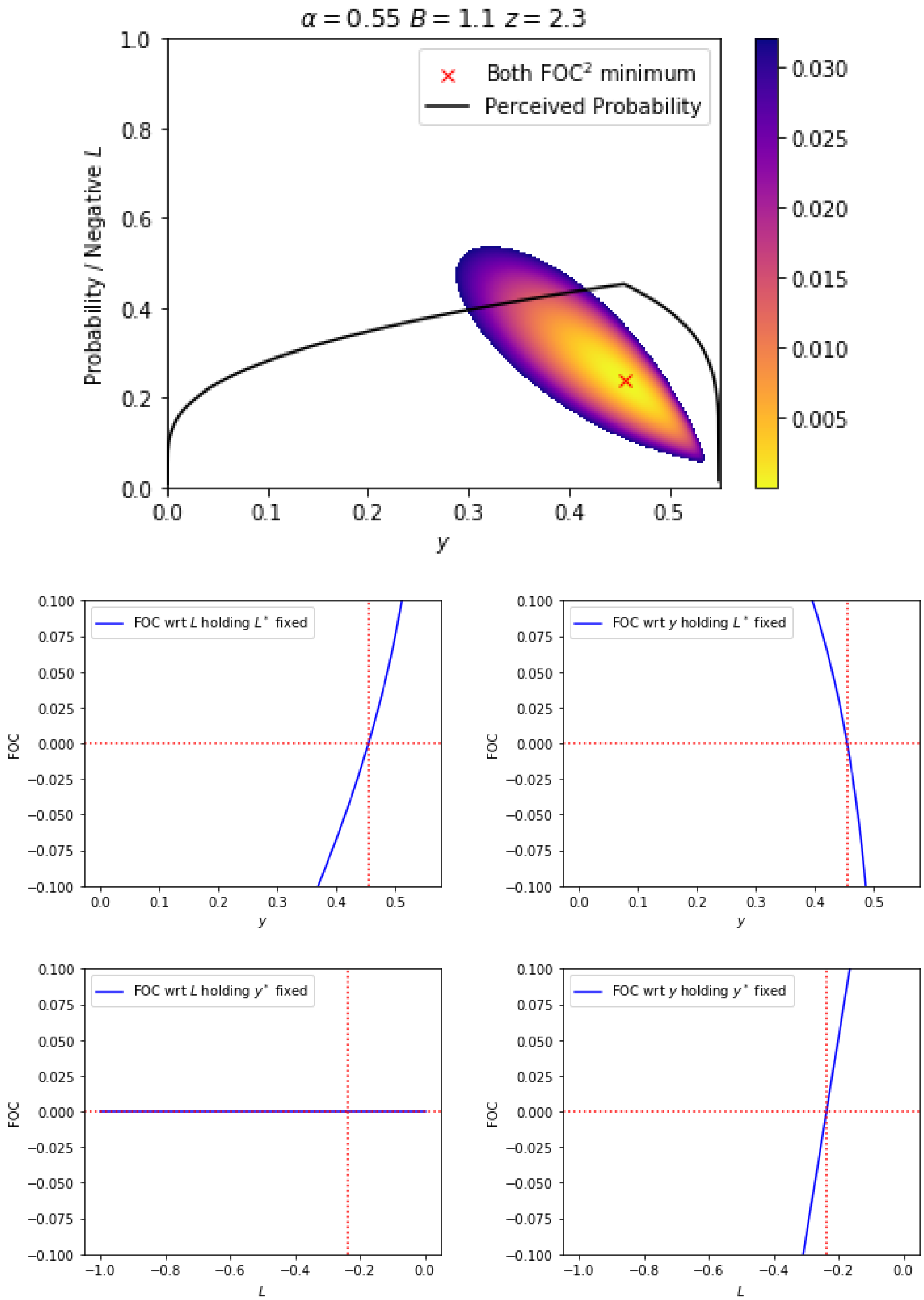


Figure 12: Network Green - $\alpha = .55$, $B = 1.1$, $z = 2.3$

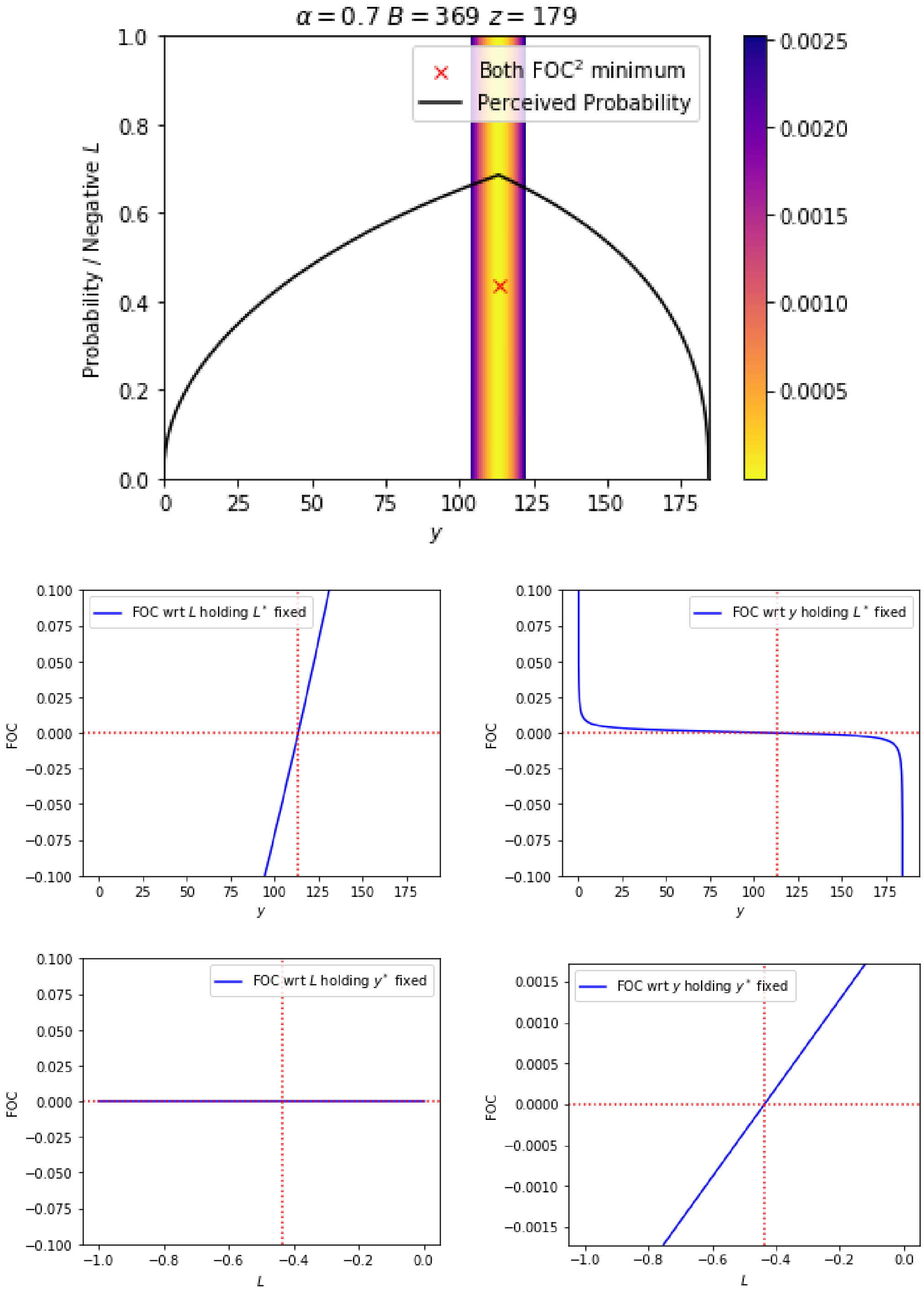


Figure 13: Network Green - $\alpha = .7$, $B = 369$, $z = 179$

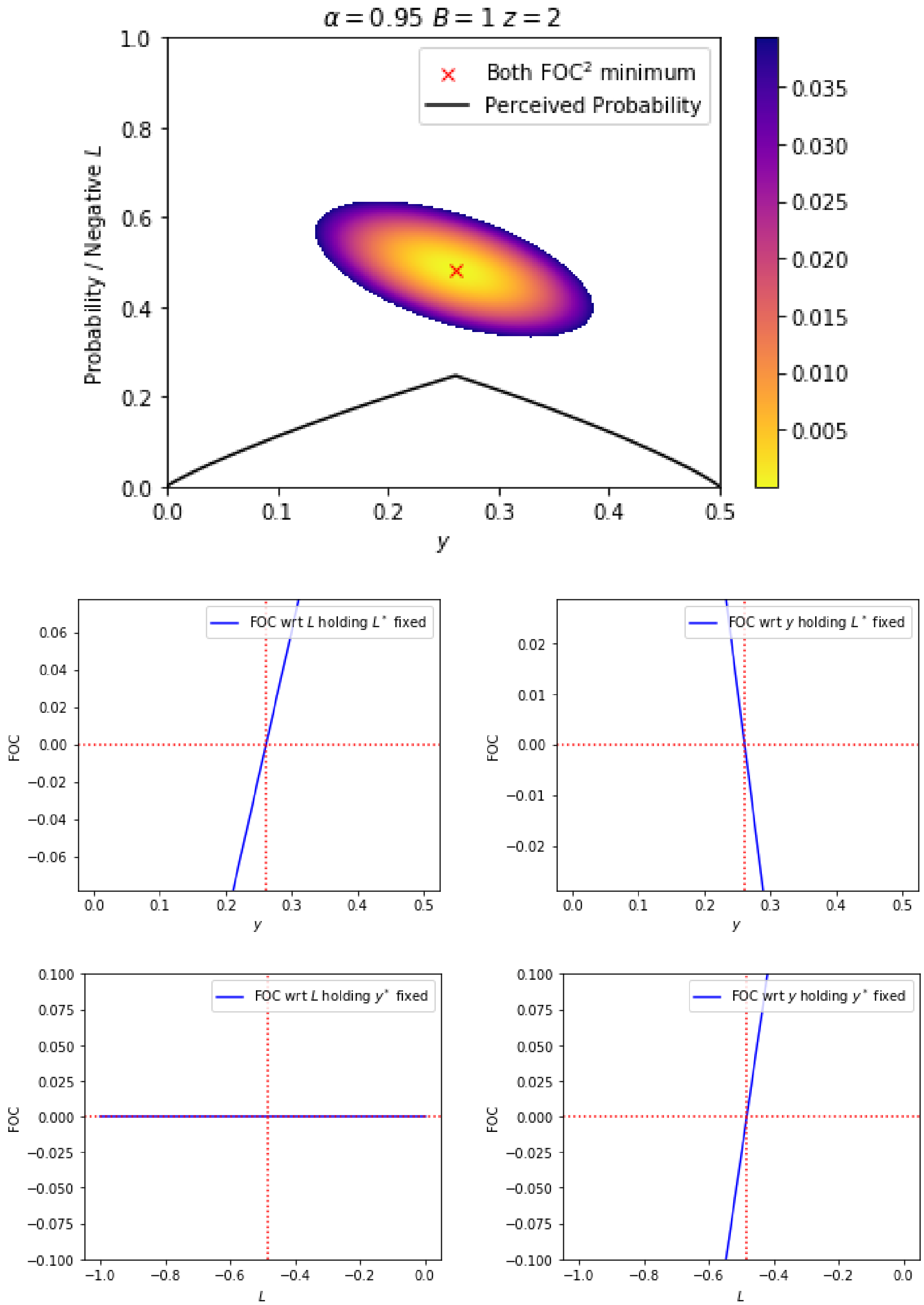


Figure 14: Network Green - $\alpha = .95$, $B = 1$, $z = 2$

7.2 DOSE Procedures

7.2.1 Network Attack Task

We build a question bank by forming two lists of probabilities. List one consists of: [0.03, 0.08, 0.12, 0.16, 0.21, 0.25, 0.29, 0.34, 0.38, 0.42, 0.47, 0.51, 0.56, 0.6, 0.64, 0.69, 0.73, 0.77, 0.82, 0.86, 0.9, 0.95, 1] and list two is: [0.06, 0.11, 0.15, 0.19, 0.24, 0.28, 0.32, 0.37, 0.41, 0.45, 0.5, 0.54, 0.59, 0.63, 0.67, 0.72, 0.76, 0.8, 0.85, 0.89, 0.93, 0.98]. These numbers were chosen in an attempt to increase the complexity of multiplying probabilities along a path. Paths were created for each possible combination of one probability from list one and one probability from list two, and then all possible combinations of 2 different paths were possible questions. This yielded a question bank of possible 255530 questions. To reduce the question bank to more relevant questions, we only chose questions where subjects with $\alpha \in \{.4, .5, .6, .7, .8, .9, .95\}$ would make different responses from a subject with $\alpha = 1$. This yields a question bank with 1397 questions.

To capture noise or errors in decision making, we assume that subjects best respond according to a logit function, so the probability of choosing the top path is: $Prob(O_T = p_1, p_2) = \frac{1}{1+exp(-\lambda(U(O_T)-U(O_B)))}$ if $U(O_T) > U(O_B)$ (and one minus this probability otherwise).

DOSE performs Bayesian updating over the likelihood of a subject being a specific type. Here, a type consists of α and λ. We consider 20 equally spaced points for $\alpha \in [.4, 1]$, and $\lambda \in [0, 100]$, and form types based on all possible combinations of these. A finer grid would be better, however due to computational limitations we use the grid space of 20.

We then begin the DOSE procedure. We assume an initial uniform prior over types, and then calculate the Kullback-Leibler (KL) divergence that results between the posterior and the prior for each response to each question in the question bank. The specific functional form we use is presented as Equation (3) in Chapman et al. (2018). The question that has the highest KL is selected to be asked (and then removed from the question bank, so that it is not asked again). In order to reduce DOSE's preference for repeatedly asking the same or similar questions (in order to get a better estimate of λ, which is of secondary interest), we remove all questions that share one of the same paths as the previously asked question. Then, for each possible response to the asked question, the prior is updated. The process then continues separately for each of the new priors, so that subjects that respond differently to the questions are asked a different sequence of questions. The order of questions are recorded in a tree-like structure, so that a software environment can ask the appropriate question given the response, without needing to calculate the KL for each individual each time. Due to computational limitations, we get a dynamic ordering of questions for 15 responses. We ask 20 questions, with the first 15 being dynamically generated and the last 5 being manually chosen by the experimenter (and the same for all subjects).

7.2.2 Binary Lottery Task

We built a question bank from the rows of the Multiple Price Lists found in Tanaka et al. (2010), Callen et al. (2014), Bruhin et al. (2010), and Holt and Laury (2002). Because these are in various currencies (and at various points of time), we normalized the highest payoff in each paper to be 1, and scaled the other payoffs accordingly. We use only Series 1 and 2 from Tanaka et al. (2010), and censor Series 1 at the 220 row, due to the large relative scale differences of the latter rows of Series 1. We then scale all payoffs up by a factor of 1472 (experimental points), to bring the average payoffs in line with our other tasks. This process gives us a question bank of 287 questions. We then remove questions with strictly dominated options (common in MPL's). We assume a utility function of an option as $U(p_1, d_1, p_2, d_2) = w_p(p_1)d_1^\sigma + (1 - w_p(p_1))d_2^\sigma$ if $d_1 > d_2$ and $U(p_1, d_1, p_2, d_2) = w_p(p_2)d_2^\sigma + (1 - w_p(p_2))d_1^\sigma$ otherwise, and iterating through perfectly responding agents with all combinations of $\alpha \in [.4, 1]$ and $\sigma \in [.2, 1.7]$ in a linespace of 12. If there are options in which none of the combinations of agents chose a particular option, then this question is deleted, leaving us with a bank of 163 questions.

To capture noise or errors in decision making, we assume that subjects best respond according to a logit function, so the probability of choosing Option A is: $Prob(O_A = p_1, d_1, p_2, d_2) = \frac{1}{1+exp(-\lambda(U(O_A)-U(O_B)))}$ if $U(O_A) > U(O_B)$ (and one minus this probability otherwise). We correct for the rescaling of payoffs relative to λ that risk aversion causes by re-normalizing the payoffs in the same manner as Goeree et al. (2003).

We then calibrate the upper bound for λ, where higher λ means subjects are getting closer to perfectly best responding. We start with a low λ, go through the logit responses of the aforementioned α and σ

types, and for each question in the bank, record the probability of choosing an option. If fewer than 80% of responses are either below 10% or above 90% (i.e. close to best responding), then λ is increased slightly. This process continues until the 80% threshold is reached. The ending λ_m is the upper bound for the DOSE procedure.

DOSE performs Bayesian updating over the likelihood of a subject being a specific type. Here, a type has its own α, σ and λ. We consider 20 equally spaced points for $\alpha \in [.4, 1]$, $\sigma \in [.2, 1.7]$, and $\lambda \in [.01, \lambda_m]$, and form types based on all possible combinations of these. A finer grid would be better, however due to computational limitations we use the grid space of 20.

We then begin the DOSE procedure. We assume an initial uniform prior over types, and then calculate the Kullback-Leibler (KL) divergence that results between the posterior and the prior for each response to each question in the question bank. The specific functional form we use is presented as Equation (3) in Chapman et al. (2018). The question that has the highest KL is selected to be asked (and then removed from the question bank, so that it is not asked again). Then, for each possible response to that question, the prior is updated. The process then continues separately for each of the new priors, so that subjects that respond differently to the questions are asked a different sequence of questions. The order of questions are recorded in a tree-like structure, so that a software environment can ask the appropriate question given the response, without needing to calculate the KL for each individual each time. Due to computational limitations, we get a dynamic ordering of questions for 16 responses. We ask 20 questions, with the first 16 being dynamically generated and the last 4 being manually chosen by the experimenter (and the same for all subjects).

7.3 Hypothesis 2 Robustness Tests

	.8	.81	.82	.83	.84	.85	.86	.87	.88	.89	.9	.91	.92	.93	.94	.95	.96	.97
Red versus Yellow if $\alpha \geq$	.029	.03	.029	.036	.021	.055	.042	.026	.018	.021	.031	.021	.013	.013	.034	.017	.143	.222
Red versus Yellow if $\alpha <$	.324	.359	.362	.434	.337	.435	.445	.488	.45	.468	.451	.493	.455	.472	.38	.416	.167	.135
Orange versus Yellow if $\alpha \geq$	.022	.027	.023	.03	.051	.055	.033	.035	.037	.029	.017	.007	.006	.009	.011	.012	.096	.176
Orange versus Yellow if $\alpha <$	.479	.425	.478	.425	.302	.26	.361	.331	.306	.373	.447	.468	.421	.438	.349	.311	.112	.098

Table 5: One-sided P-values from Wilcoxon Signed Rank Tests of Hypothesis 2 for Different α Thresholds

7.4 Binary Lottery Task Results

We assume a specific functional form for a lottery $L = (x_1, p; x_2)$ where $x_1 > x_2$ of: $U(x_1, x_2) = w(p)x_1^\sigma + (1 - w(p))x_2^\sigma$. Figures 15 and 16 present the CDFs of the estimated σ and α from this measure.

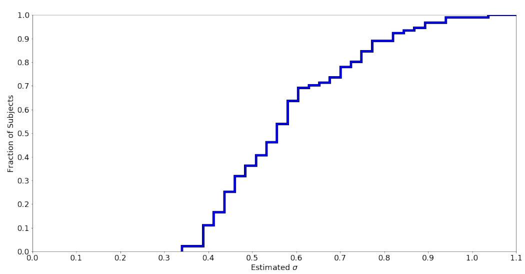


Figure 15: CDF of elicited σ

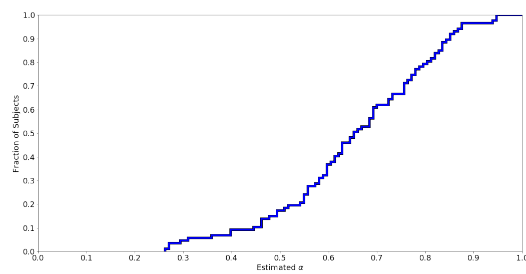


Figure 16: CDF of elicited α

Table 6 presents the Spearman's ρ for all combinations of the elicited parameters and average subject behavior. As discussed, there is a modest, marginally insignificant correlation between the two measures of α.

	α (AT)	λ (AT)	α (LT)	σ (LT)	λ (LT)	Red	Orange	Yellow	Blue
λ (AT)	**$\rho = .765$ $p < .001$**	$\rho = 1$							
α (LT)	$\rho = .166$ $p = .117$	*$\rho = .175$ $p = .098$*	$\rho = 1$						
σ (LT)	$\rho = .088$ $p = .406$	$\rho = .125$ $p = .238$	**$\rho = .381$ $p < .001$**	$\rho = 1$					
λ (LT)	$\rho = .108$ $p = .308$	$\rho = .015$ $p = .886$	$\rho = -.105$ $p = .321$	**$\rho = -.397$ $p < .001$**	$\rho = 1$				
Red	**$\rho = .289$ $p = .005$**	$\rho = .173$ $p = .101$	$\rho = -.014$ $p = .894$	$\rho = -.109$ $p = .306$	$\rho = .108$ $p = .308$	$\rho = 1$			
Orange	**$\rho = .253$ $p = .016$**	$\rho = .078$ $p = .463$	$\rho = .056$ $p = .597$	*$\rho = -.195$ $p = .064$*	$\rho = .148$ $p = .161$	**$\rho = .676$ $p < .001$**	$\rho = 1$		
Yellow	$\rho = .079$ $p = .455$	$\rho = -.041$ $p = .696$	$\rho = .007$ $p = .946$	*$\rho = -.200$ $p = .058$*	$\rho = .073$ $p = .491$	**$\rho = .601$ $p < .001$**	**$\rho = .617$ $p < .001$**	$\rho = 1$	
Blue	**$\rho = -.289$ $p = .005$**	**$\rho = -.218$ $p = .038$**	$\rho = .106$ $p = .317$	$\rho = .123$ $p = .246$	*$\rho = -.200$ $p = .058$*	**$\rho = -.380$ $p < .001$**	**$\rho = -.339$ $p = .001$**	$\rho = -.115$ $p = .276$	$\rho = 1$
Green	**$\rho = -.271$ $p = .009$**	$\rho = -.146$ $p = .168$	$\rho = -.016$ $p = .876$	$\rho = .083$ $p = .437$	$\rho = -.010$ $p = .927$	**$\rho = -.280$ $p = .007$**	**$\rho = -.322$ $p = .002$**	**$\rho = -.256$ $p = .014$**	**$\rho = .290$ $p = .005$**

Table 6: Full Spearman's ρ Table (in bold if $p < .05$, in italics if $.05 < p < .1$)

(AT=Network Attack Task, LT= Binary Lottery Task)

7.5 Individual Network Cluster Analysis

7.5.1 Network Red

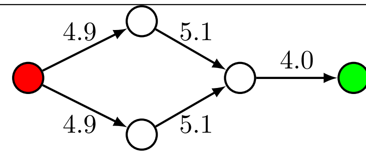
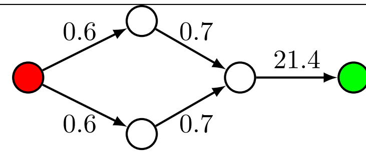
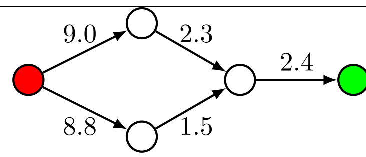
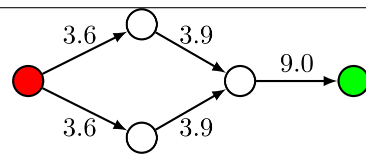
Cluster Name % of subjects in Cluster	Average Edge Allocation
Naive Diversification 14.9%	
Near Optimal $\alpha = 1$ 51.7%	
Early Revelation - Some Diversification 13.8%	
$\alpha < 1$ - Some Diversification 19.5%	

Table 7: Network Red Cluster Analysis

7.5.2 Network Orange

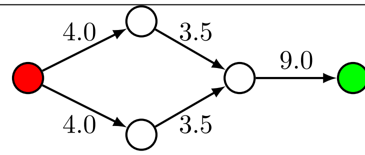
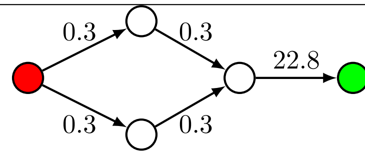
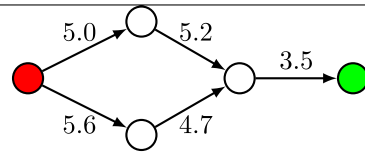
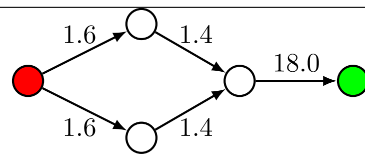
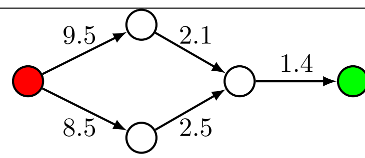
Cluster Name % of subjects in Cluster	Average Edge Allocation
$\alpha < 1$ - Some Diversification 23.0%	
Near Optimal $\alpha = 1$ 31.0%	
Naive Diversification 16.1%	
$\alpha < 1$ or Mild Diversification 23.0%	
Early Revelation - Mild Diversification 6.9%	

Table 8: Network Orange Cluster Analysis

7.5.3 Network Yellow

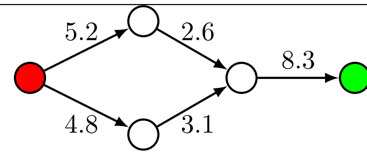
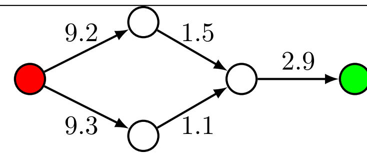
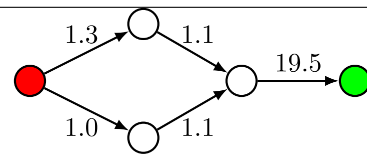
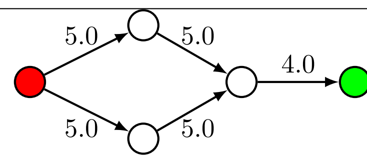
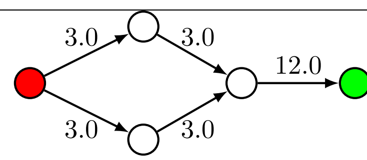
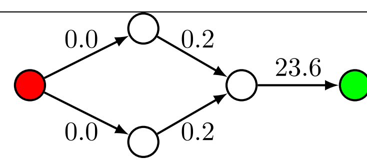
Cluster Name % of subjects in Cluster	Average Edge Allocation
$\alpha < 1$ - Some Diversification and Early Revelation 11.5%	
Early Revelation - Mild Diversification 8.0%	
Near Optimal $\alpha = 1$ - Some Late Revelation 19.5%	
Naive Diversification 24.1%	
$\alpha < 1$ - Some Diversification 16.1%	
Late Revelation 20.7%	

Table 9: Network Yellow Cluster Analysis

7.5.4 Network Blue

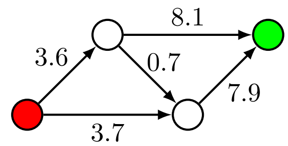
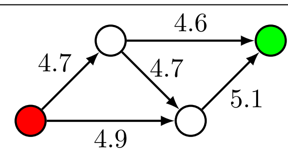
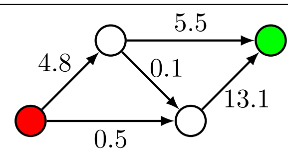
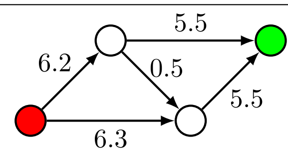
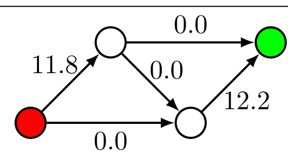
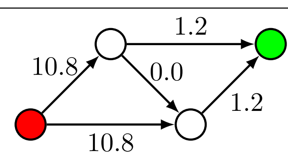
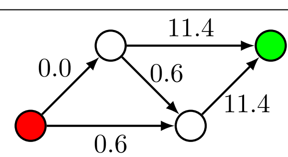
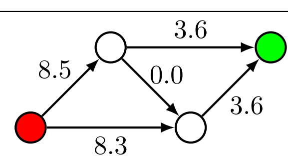
Cluster Name % of subjects in Cluster	Average Edge Allocation
Late Revelation - Mild Diversification 24.1%	
Naive Diversification 12.6%	
'False Common Edge' Heuristic 3.4%	
Mild Diversification and Mild Early Revelation 18.4%	
Near Optimal 2.3%	
Near Optimal - Early Revelation 9.2%	
Late Revelation - Very Mild Diversification 20.7%	
Near Optimal - Some Early Revelation 9.2%	

Table 10: Network Blue Cluster Analysis

7.5.5 Network Green

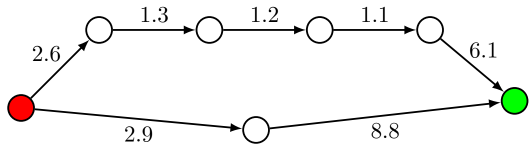
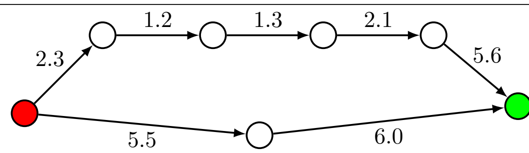
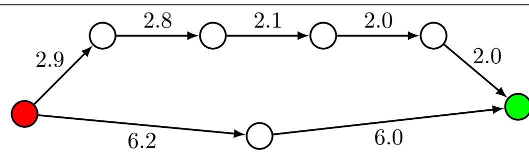
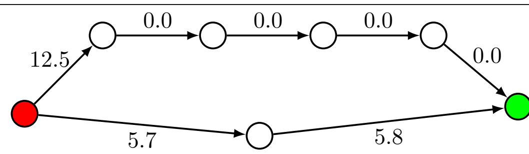
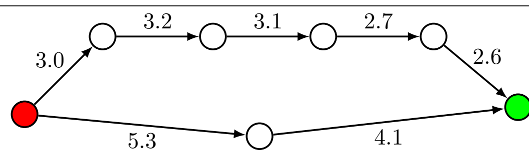
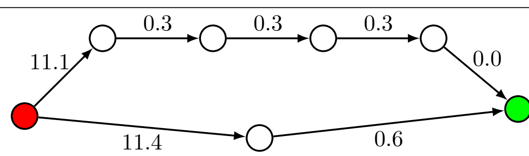
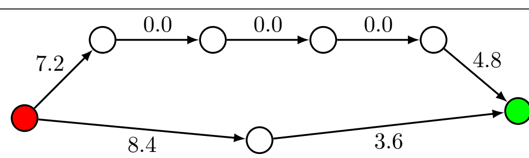
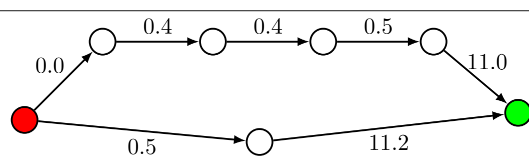
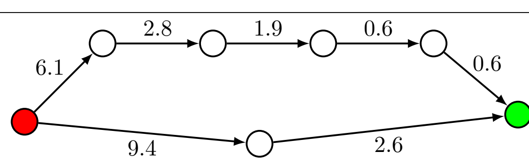
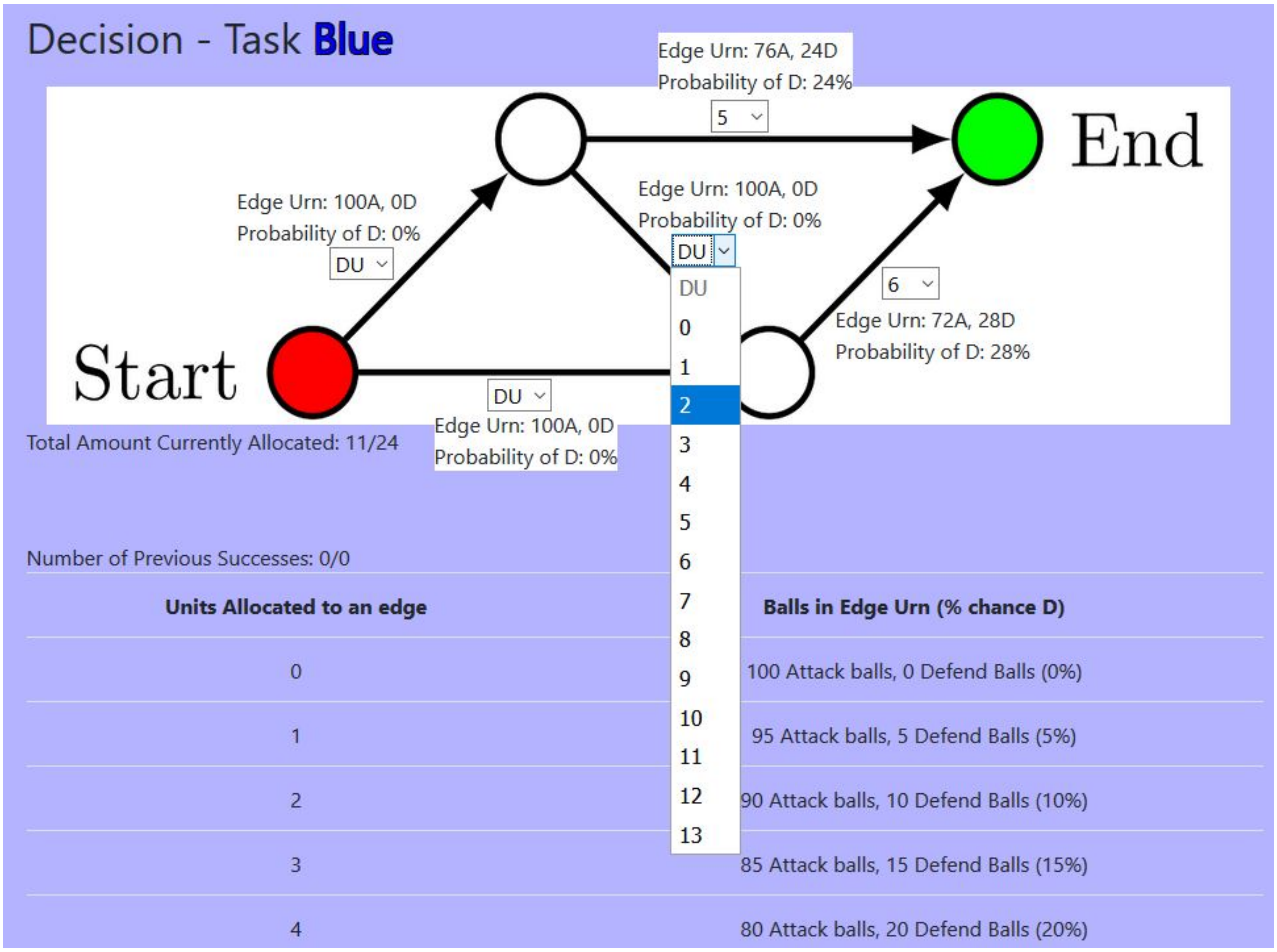
<table>
<tr><th>Cluster Name
% of subjects in Cluster</th><th>Average Edge Allocation</th></tr>
<tr><td>Some Late Revelation - Very Mild Diversification
9.2%</td><td>
</td></tr>
<tr><td>Some Diversification - Mild Late Revelation
12.6%</td><td>
</td></tr>
<tr><td>$\alpha \approx .975$ or $\alpha < .975$ with Diversification
23.0%</td><td>
</td></tr>
<tr><td>Sometimes Early Revelation
2.3%</td><td>
</td></tr>
<tr><td>Naive Diversification
11.5%</td><td>
</td></tr>
<tr><td>Near Optimal - Early Revelation
8.0%</td><td>
</td></tr>
<tr><td>Near Optimal - Some Early Revelation
6.9%</td><td>
</td></tr>
<tr><td>Late Revelation - Very Mild Diversification
11.5%</td><td>
</td></tr>
<tr><td>Near Optimal - Some Early Revelation
14.9%</td><td>
</td></tr>
</table>

Table 11: Network Green Cluster Analysis

7.6 Experiment Interface



7.7 Experiment Instructions

7.7.1 Overview

Introduction

This experiment is a study of decision making. The amount of money you earn depends partly on the decisions that you make and thus you should read the instructions carefully. The money you earn will be paid privately to you, in cash, at the end of the experiment. A research foundation has provided the funds for this study. Please put away your cell phones and other distracting devices for the duration of the experiment.

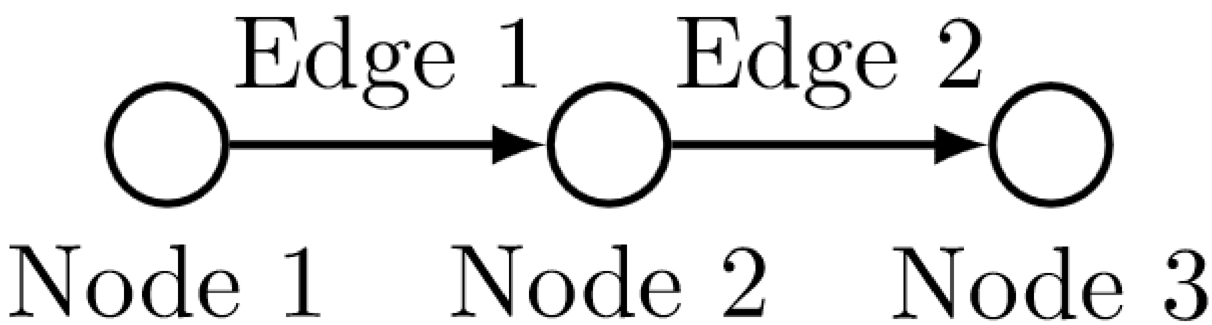
In this experiment, you will participate in Task 1, and then 6 additional colored tasks. The Task 1 instructions are given on a separate piece of paper. For the additional colored tasks, you have been given instructions printed on different colored paper. The color of the paper coincides to the name of the task. Only read the relevant instructions when the computer prompts you to do so. The tasks are independent meaning the decisions and payoffs from one do not affect the decisions and payoffs from the other. **Some of these tasks are similar, but you should take care when reading similar instructions to see what is different about the new task.**

Please do not attempt to communicate with other participants in the room during the experiment. If you have a question as you read through the instructions at any time during the experiment, please raise your hand and an experimenter will come by to answer it in private.

You cannot use a pen or a calculator until after you have completed Task White (which is the second task). If you want to use either a pen or a calculator after Task White, please raise your hand and an experimenter will bring you one.

Your earnings in this task are denominated in experimental dollars (called points in the software), which will be exchanged at a rate of 350 experimental dollars = 1 U.S. dollar at the end of the experiment.

The colored instructions use some terms which you may not be already familiar with, Nodes and Edges. The following figure is designed to illustrate these concepts:



A Node is a position, while Edges describe how you can go between these positions. A single Edge connects two Nodes. The arrow indicates the direction of the Edge, you can only go between Nodes in the direction of the Edge. In the given figure, you can go from Node 1 to Node 2 using Edge 1, and from Node 2 to Node 3 using Edge 2. Note, you cannot go from Node 2 to Node 1, or from Node 3 to Node 2, or from Node 3 to Node 1, as there are no Edges that connect those Nodes in that direction.

7.7.2 Binary Lottery Task

Task 1

Task 1 is divided into 20 decision "periods." You will be paid for this task based on your decision in one of the periods, which will be randomly selected. Each decision you make is therefore important because it has a chance of determining the amount of money you earn.

In each period, you will be asked to choose between 2 Options, Option A and Option B.

Each Option has 20 balls in an Urn. These balls are colored Red or Blue. One ball will be drawn from the Urn of the Option you choose. Each Option has a payoff in points if a Red ball is drawn, and a payoff in points if a Blue ball is drawn. You choose an Option by clicking on it, and then clicking on the Submit button.

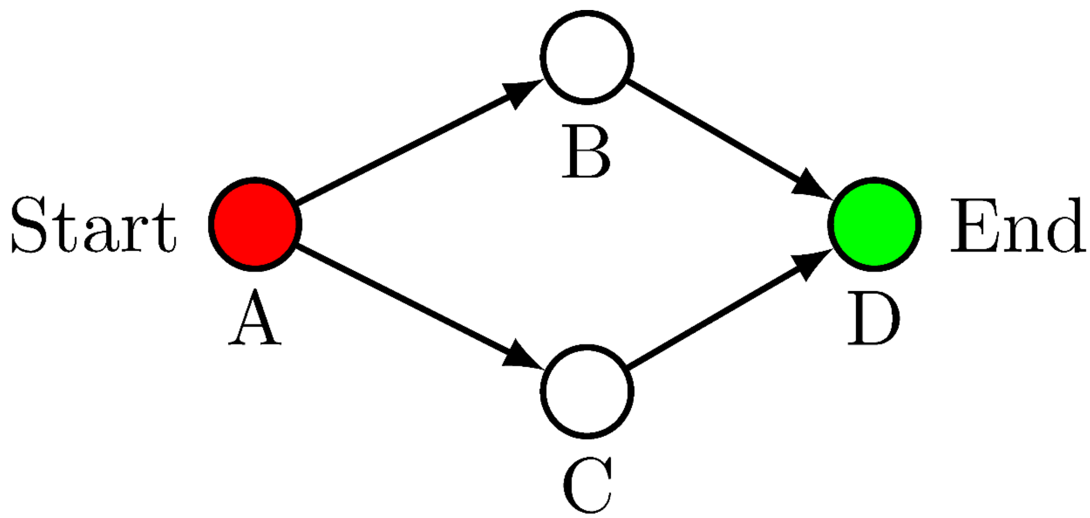
The ball for your chosen Option is not drawn until the end of the 20 decision periods. At the end of Task 1 you will be shown the randomly selected period, the Options that were available in that period, the Option you chose, the ball that was drawn from the chosen Option, and your final payoff for Task 1.

7.7.3 Network Attacker Task

Task White

Task White is divided into 20 decision "periods." You will be paid for this task based on your decision in one of the periods, which will be randomly chosen. Each decision you make is therefore important because it has a chance of determining the amount of money you earn.

There are two roles in this task, Attacker and Defender. **You will be playing in the role of the Attacker against a computerized Defender. As an Attacker, your objective is to capture the node labelled End in the figure below.**



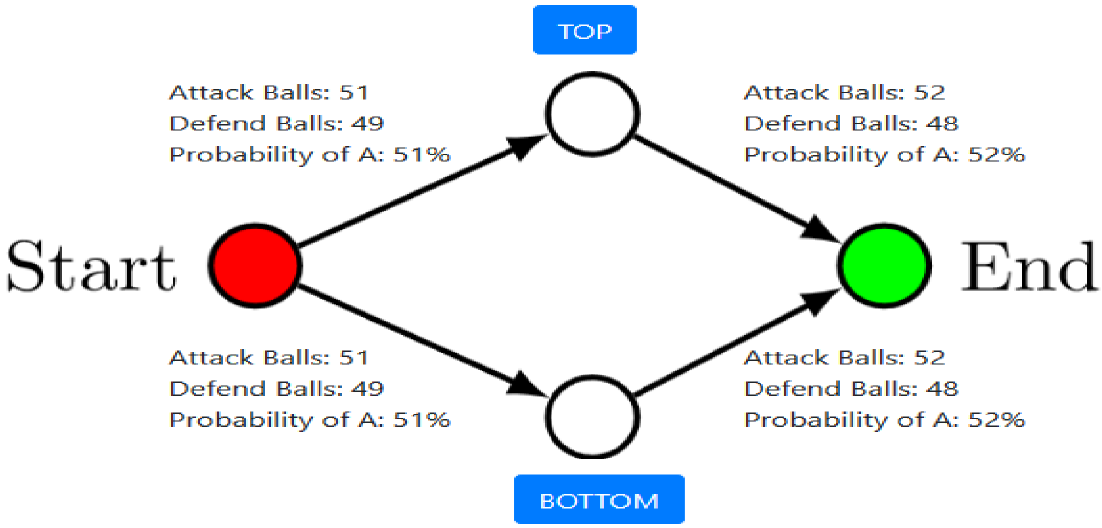
You start at Node A, labelled Start, and must decide whether to attack along the 'Top' path (from Node A to Node B, then Node B to Node D), or the 'Bottom' path (from Node A to Node C, then Node C to Node D). You can only attack along one of these paths in a period.

The probability that an attack on a Node along an Edge is successful is given by an 'Urn' attached to that Edge. This Urn contains Attack balls and Defend balls, and has 100 balls in total. When your chosen path takes you along an Edge, a ball is randomly drawn from that Edge's Urn. If this ball is an Attack ball, the attack succeeds, and you capture the subsequent node. If this ball is a Defend ball, the attack fails, and your attack for this period is over. The ball contents of the Urns on each Edge are set by the computer, and differ from period to period.

Example:

In this example the Top and Bottom paths are the same. If the Top path is selected, then a ball would be drawn from the first Edge Urn with 51 Attack and 49 Defend balls. If a Defend ball is drawn (this occurs with 49% probability), then the attack on the Top Node will fail and the attack for the period is over. If an Attack ball is drawn (occurs with 51% probability), then the attack on the Top Node will succeed and the attack for the period continues. Then, a ball would be drawn from the second Edge Urn on the top path with 52 Attack balls and 48 Defend balls. If a Defend ball is drawn (occurs with 48% probability), then the attack on the End Node will fail and the attack from the period is over. If an Attack ball is drawn (occurs with 52% probability), then the attack on the End Node will succeed, and the overall attack for this period will be successful.

Earnings



If you succeed and reach the End node, you will receive 3000 experimental dollars for that period. If you fail and do not reach the End node, you will receive 0 experimental dollars for that period. You do not receive additional payment for capturing nodes other than the End node. At the end of the experiment, one period will be randomly selected for your payment from this task.

Summary

- You are an Attacker playing against a computerized Defender.
- Your goal is to capture the End node.
- You decide whether to attack along the top path or the bottom path.
- The probability of a successful attack on a Node is determined by the Edge Urn.
  - If an Attack ball is drawn, the attack on that Node succeeds, if a Defend ball is drawn, the attack for the period fails
- If you capture the End Node you will earn 3000 for that period.
- If you do not capture the End Node, you will earn 0 for that period
  - That is, Nodes other than the End Node are all worth 0

7.7.4 Network Defense Tasks

Note: In the interests of space, the square parentheses [LIKE THIS] below indicate which parts of the instructions are common to all tasks, and which parts are unique to certain tasks. Subjects received an instructions packet with separate instructions (with both the repeated and unique parts) printed on colored paper associated with each task.

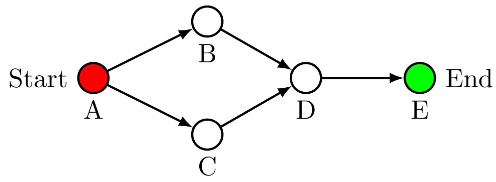
[ALL TASKS:]

Task [COLOR]

Task [COLOR] is divided into 10 decision "periods." You will be paid for this task based on your decision in one of the periods, which will be randomly chosen. Each decision you make is therefore important because it has a chance of determining the amount of money you earn.

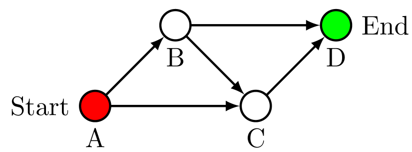
There are two roles in this task, Attacker and Defender. **You will be playing in the role of the Defender against a computerized Attacker. As a Defender, your objective is to prevent the Attacker from capturing the node labelled End in the figure below.**

[TASKS RED, ORANGE, and YELLOW:]



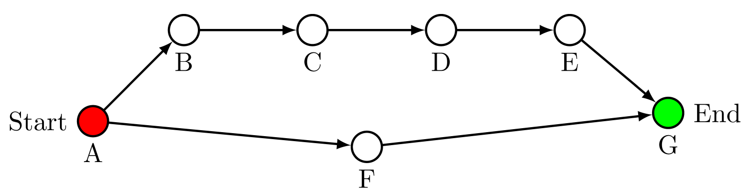
The Attacker starts at Node A, labelled Start, and can attempt to capture any other connected Node via the Edges in the direction of the arrows. For example, from Node A the attacker can attempt to capture Node B or Node C. If the Attacker captures the Node, it moves to that Node, and can then attempt to capture another Node in the direction of the arrows. For example, from Node A, if the Attacker captures Node B it will move there. Since no arrows connect Node B to Node C, and it cannot move in the opposite direction of the arrows, the attacker cannot attempt to capture Node C. If it has captured Node B, the Attacker only has one choice of Node to attack, Node D.

[TASK BLUE:]



The Attacker starts at Node A, labelled Start, and can attempt to capture any other connected Node via the Edges in the direction of the arrows. For example, from Node A the attacker can attempt to capture Node B or Node C. If the Attacker captures the Node, it moves to that Node, and can then attempt to capture another Node in the direction of the arrows. For example, from Node A, if the Attacker captures Node C it will move there. Since no arrows connect Node C to Node B, as it cannot move in the opposite direction of the arrows, the attacker cannot attempt to capture Node B. If it has captured Node C, the Attacker only has one choice of Node to attack, Node D. If the attacker captures Node B, it has two choices of Nodes to attack, Node C and Node D.

[TASK GREEN:]



The Attacker starts at Node A, labelled Start, and can attempt to capture any other connected Node via the Edges in the direction of the arrows. For example, from Node A the attacker can attempt to capture Node B or Node F. If the Attacker captures the Node, it moves to that Node, and can then attempt to capture another Node in the direction of the arrows. For example, from Node A, if the Attacker captures Node B it will move there. Since no arrows connect Node B to Node F, and it cannot move in the opposite direction of the arrows, the attacker cannot attempt to capture Node F. Node B is also not connected to any other Node except Node C. Therefore, if it has captured Node B, the Attacker only has one choice of Node to attack, Node D[26].

[ALL TASKS:]

The probability that an attack on a Node along an Edge is successful is given by an 'Urn' attached to that Edge. This Urn contains Attack balls and Defend balls, and has 100 balls in total. When the Attacker's path takes them along an Edge, a ball is randomly drawn from that Edge's Urn. If this ball is an Attack ball, the attack succeeds, and the Attacker captures the subsequent node. If this ball is a Defend ball, the

[26] This should read 'Node C' instead of 'Node D'. We present the instructions in their original form with typos included for replication purposes.

defense succeeds, and the attack for this period is over. The ball contents of the Urns on each Edge are set by you, in the role of the Defender.

You will have 24 units of defense in each period that you can allocate along each Edge to defend the Nodes. Each unit of defense allocated to an Edge increases the number of Defend balls and decreases the number of Attack balls in that Edge Urn. In other words, each unit of defense increases the probability of a successful defense if a Node is attacked along that Edge.

The table on the following page gives the probability that a Node will be successfully defended if it is attacked using an Edge with a given number of defense units.

These Edge Urns and defense probabilities may or may not be the same between different Tasks. You should re-read this table on each subsequent set of instructions you receive.

[TASKS RED, BLUE, GREEN:]

Number of Defense Units Allocated to an Edge	**Balls in Edge Urn (% chance of successful defense)**
0	100 Attack balls, 0 Defend Balls (0%)
1	95 Attack balls, 5 Defend balls (5%)
2	90 Attack balls, 10 Defend balls (10%)
3	85 Attack balls, 15 Defend balls (15%)
4	80 Attack balls, 20 Defend balls (20%)
5	76 Attack balls, 24 Defend balls (24%)
6	72 Attack balls, 28 Defend balls (28%)
7	68 Attack balls, 32 Defend balls (32%)
8	64 Attack balls, 36 Defend balls (36%)
9	61 Attack balls, 39 Defend balls (39%)
10	58 Attack balls, 42 Defend balls (42%)
11	55 Attack balls, 45 Defend balls (45%)
12	52 Attack balls, 48 Defend balls (48%)
13	49 Attack balls, 51 Defend balls (51%)
14	46 Attack balls, 54 Defend balls (54%)
15	44 Attack balls, 56 Defend balls (56%)
16	42 Attack balls, 58 Defend balls (58%)
17	39 Attack balls, 61 Defend balls (61%)
18	37 Attack balls, 63 Defend balls (63%)
19	35 Attack balls, 65 Defend balls (65%)
20	33 Attack balls, 67 Defend balls (67%)
21	32 Attack balls, 68 Defend balls (68%)
22	30 Attack balls, 70 Defend balls (70%)
23	28 Attack balls, 72 Defend balls (72%)
24	27 Attack balls, 73 Defend balls (73%)

For example, if a Node is attacked through an Edge that has 6 defense units allocated to it, the Edge Urn would have 72 Attack balls and 28 Defend balls. One ball will be drawn at random to determine the outcome, and so there is a 28 out of 100 chance (since there are 100 total balls) that the defense will succeed. If instead the Edge had 1 defense unit, then the Edge Urn would have 95 Attack balls and 5 Defend balls. In this case, the chances of a successful defense of the Node would be 5 out of 100.

[TASK ORANGE:]

Number of Defense Units Allocated to an Edge	**Balls in Edge Urn (% chance of successful defense)**
0	100 Attack balls, 0 Defend Balls (0%)
1	97 Attack balls, 3 Defend balls (3%)
2	94 Attack balls, 6 Defend balls (6%)
3	91 Attack balls, 9 Defend balls (9%)
4	88 Attack balls, 12 Defend balls (12%)
5	85 Attack balls, 15 Defend balls (15%)
6	82 Attack balls, 18 Defend balls (18%)
7	80 Attack balls, 20 Defend balls (20%)
8	77 Attack balls, 23 Defend balls (23%)
9	75 Attack balls, 25 Defend balls (25%)
10	73 Attack balls, 27 Defend balls (27%)
11	70 Attack balls, 30 Defend balls (30%)
12	68 Attack balls, 32 Defend balls (32%)
13	66 Attack balls, 34 Defend balls (34%)
14	64 Attack balls, 36 Defend balls (36%)
15	62 Attack balls, 38 Defend balls (38%)
16	60 Attack balls, 40 Defend balls (40%)
17	58 Attack balls, 42 Defend balls (42%)
18	56 Attack balls, 44 Defend balls (44%)
19	54 Attack balls, 46 Defend balls (46%)
20	53 Attack balls, 47 Defend balls (47%)
21	51 Attack balls, 49 Defend balls (49%)
22	49 Attack balls, 51 Defend balls (51%)
23	48 Attack balls, 52 Defend balls (52%)
24	46 Attack balls, 54 Defend balls (54%)

For example, if a Node is attacked through an Edge that has 6 defense units allocated to it, the Edge Urn would have 82 Attack balls and 18 Defend balls. One ball will be drawn at random to determine the outcome, and so there is an 18 out of 100 chance (since there are 100 total balls) that the defense will succeed. If instead the Edge had 1 defense unit, then the Edge Urn would have 97 Attack balls and 3 Defend balls. In this case, the chances of a successful defense of the Node would be 3 out of 100.

[TASK YELLOW:]

Number of Defense Units Allocated to an Edge	Balls in Edge Urn (% chance of successful defense)
0	100 Attack balls, 0 Defend Balls (0%)
1	82 Attack balls, 18 Defend balls (18%)
2	76 Attack balls, 24 Defend balls (24%)
3	72 Attack balls, 28 Defend balls (28%)
4	68 Attack balls, 32 Defend balls (32%)
5	65 Attack balls, 35 Defend balls (35%)
6	63 Attack balls, 37 Defend balls (37%)
7	60 Attack balls, 40 Defend balls (40%)
8	58 Attack balls, 42 Defend balls (42%)
9	56 Attack balls, 44 Defend balls (44%)
10	54 Attack balls, 46 Defend balls (46%)
11	52 Attack balls, 48 Defend balls (48%)
12	51 Attack balls, 49 Defend balls (49%)
13	49 Attack balls, 51 Defend balls (51%)
14	47 Attack balls, 53 Defend balls (53%)
15	46 Attack balls, 54 Defend balls (54%)
16	45 Attack balls, 55 Defend balls (55%)
17	43 Attack balls, 57 Defend balls (57%)
18	42 Attack balls, 58 Defend balls (58%)
19	41 Attack balls, 59 Defend balls (59%)
20	39 Attack balls, 61 Defend balls (61%)
21	38 Attack balls, 62 Defend balls (62%)
22	37 Attack balls, 63 Defend balls (63%)
23	36 Attack balls, 64 Defend balls (64%)
24	35 Attack balls, 65 Defend balls (65%)

For example, if a Node is attacked through an Edge that has 6 defense units allocated to it, the Edge Urn would have 63 Attack balls and 37 Defend balls. One ball will be drawn at random to determine the outcome, and so there is a 37 out of 100 chance (since there are 100 total balls) that the defense will succeed. If instead the Edge had 1 defense unit, then the Edge Urn would have 82 Attack balls and 18 Defend balls. In this case, the chances of a successful defense of the Node would be 18 out of 100.

[ALL TASKS:]

You can allocate your 24 defense units across the Edges in any pattern you wish. Defense units that are not allocated do not carry over to later periods. **You must choose a number of 'DU' (for Defense Units) for each Edge, even if you are allocating zero Defense Units, in order for the Next button to appear.** Once you have finished the allocation, you can finalize it by clicking on the Next button, at which point the computerized Attacker will begin.

The computerized Attacker will always attack along the path from the Start Node A to the End Node [TASKS RED, ORANGE, YELLOW]E [TASK BLUE]D [TASK GREEN]G that has the lowest probability of successful defense. If two or more paths have equally low probabilities of successful defense, then the Attacker will randomly choose one of the tied paths. Once a successful defense occurs (that is, if a Defend ball is drawn), or the Attacker captures the End node, the Attacker will stop. You will then be shown the outcome of the attack. The path the Attacker took will be represented with red arrows, and Nodes that were captured will appear red. Nodes that were not attacked at all, or a Node that was attacked but successfully defended, will appear green. If the End Node is green, then you prevented the Attacker and achieved your goal.

Earnings

If you succeed and stop the Attacker before they reach the End node, you will receive 1500 experimental dollars for that period. If you fail and the Attacker reaches the End node, you will receive 0 experimental dollars for that period. At the end of the experiment, one period will be randomly selected for your payment from this task.

Summary

- You are a Defender playing against a computerized Attacker.
- Your goal is to stop the Attacker from capturing the End node.
- You have 24 defense units in each period to allocate across Edges.
- Defense units increase the number of Defend balls and decrease the number of Attack balls in that Edge Urn.
- The Attacker can only attack in the direction of the arrows from the Start Node.
- If a Defend ball is ever drawn, the Attacker will stop attacking, and you will earn 1500 for that period.
- If the Attacker captures the End Node, you will earn 0 for that period
- The Attacker will always choose the path to the End Node that has the lowest probability of successful defense. In the case of ties, the Attacker will randomly choose one of the tied paths.