MGT 721 - Modeling Operational Processes
Spring 2025

Diffusion and Queueing Approximations Problem Set: Solutions

These questions are meant to review and extend the concepts introduced in class. You may use any computational assistant you like, including for example Wolfram Alpha's tools for symbolic integration or summation, or even chatGPT, but you are responsible for any errors that might result (especially with chatGPT). You can complete all of these questions without such computational assistants, but nonetheless the option is there.

1. An NCAA Basketball Win Probability Graph

In class, we discussed a simple win probability model for basketball. To summarize that model, let random variable $X(t)$ represent the net number of points gained by the favorite (i.e. points scored by the favorite minus underdog points) over t minutes of play in the game. The model holds that $X(t)$ follows a Brownian motion

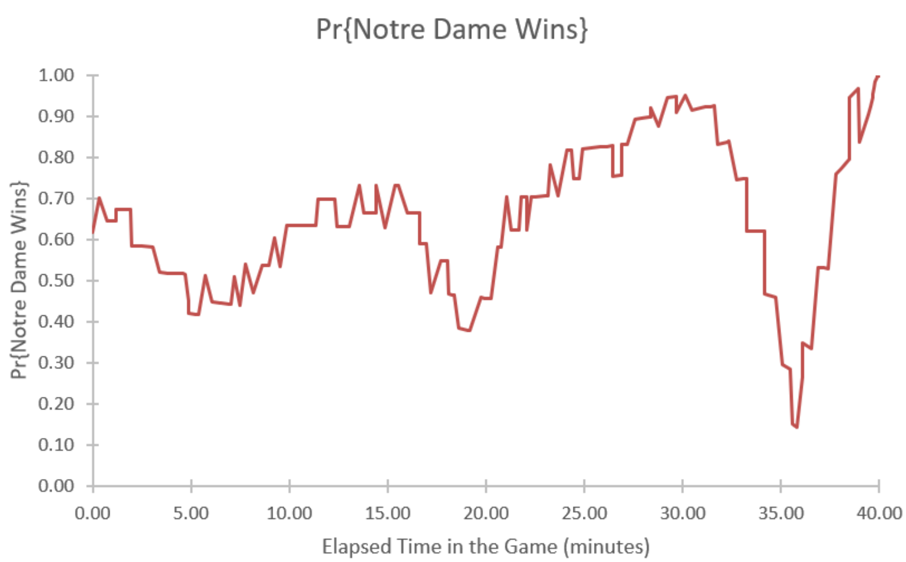
$$dX(t) = (\lambda - \mu)dt + \sqrt{\lambda + \mu}dZ$$

where λ and μ are the scoring rates per unit time for the favorite and underdog respectively. The probability that the favorite wins given the current state of the game (which is the favorite up by x_t after t minutes of play) is given by

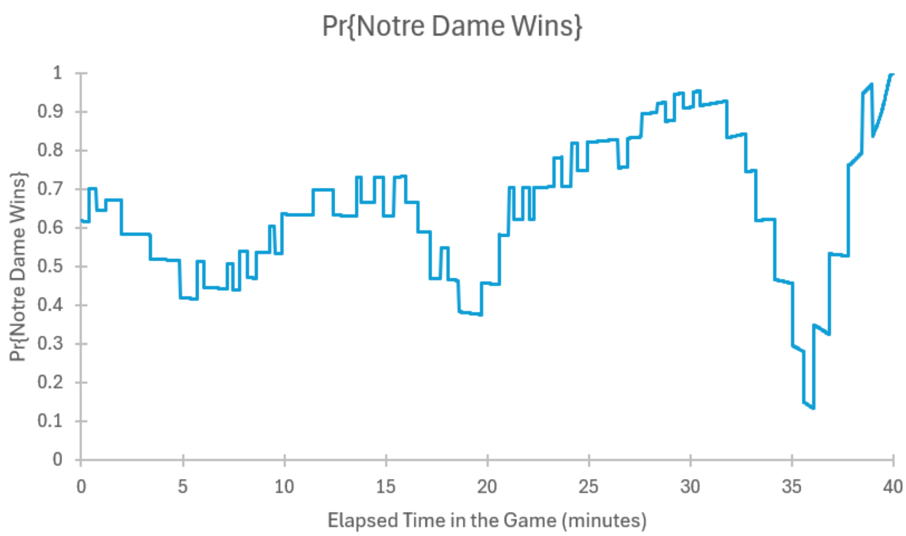
$$\Pr\{\text{Win}\} = \Pr\{X(40-t) + x_t > 0\} = \Pr\{X(40-t) > -x_t\}.$$

(a) The file UConn_Notre_Dame_2011 on our course website contains a complete play-by-play account of the University of Connecticut's final home game of the 2010-11 season, which took place on March 5, 2011 against Notre Dame, which was the favorite in this game. Suppose you knew then what you know now about modeling win probability. Produce your own "win probability graph" for this game by applying the model above to the data from the UConn-Notre Dame game. Your graph should show the probability that Notre Dame wins the game as a function of the elapsed time in the game Suppose that the closing betting odds prior to the 40 minute game (when t =0 minutes) had a spread of 4 points favoring Notre Dame ($\lambda - \mu = 4$), and an over/under (points total) line of 176 ($\lambda + \mu = 176$).

This is ridiculously easy! Since the game file tells us how many minutes are left along with the score at any moment, it is easy to calculate the required probability using the normal distribution exactly as described above (note that minutes left = $40 - t$). Since the game is timed in minutes while the opening spread/point total (i.e. drift/variance) was given for the 40 minute game duration, it is convenient to simply divide by 40 to obtain a per minute drift of 0.1 points and a per minute variance of 4.4 points^2. To plot the resulting win probability against time elapsed (as opposed to time remaining), we use the fact that *Time Elapsed = 40 − Time Remaining* (duh....). Here's what I get:



Note that this graph was created on the cheap, as I only computed probabilities at the event times reported in the data file. A smoother picture can be created by computing probabilities every 0.01 minutes, for example, as the win probability changes even when the score does not. Here is what that graph looks like:



Note how the second graph shows movement in the win probability even when the score does not change (that is, when there are no jumps in the graph). Compare to the basketball win probability graphs you see on ESPN or other sports sites. Cool, right?

(b) With 9.85 minutes left in the game, Notre Dame was leading UConn by the score of 60-50. At that moment, what was the probability that UConn would win the game?

Just look at the probabilities you created for the win probability graph above. With 9.85 minutes left in the game, the probability that Notre Dame would win was 0.95, which means the probability that UConn would win was 0.05. Not so great!

(c) With 4.22 minutes remaining, UConn led Notre Dame by the score of 65-60. At that moment, what was the probability that UConn would win the game?

Same idea – with 4.22 minutes left, the probability that Notre Dame would win was equal to 0.14, which means that UConn had an 86% chance of winning the game! But, as the game graph shows, that 5 point lead was UConn's high water mark, as Notre Dame went on a run to win the game by a final score of 70-67,

2. Price Discovery in Waiting Lists (after Ashlagi *et al*, 2022)

In a paper with the same name as this problem, Ashlagi *et al* stated the following example:

Example 1. *A single kind of item is allocated by queue. Items arrive according to a Poisson process with rate* 1. *Agents arrive according to a Poisson process with rate* 2. *Agents have heterogeneous values for the item distributed* $v \sim U[0,1]$ *and a cost of waiting* $c(w) = 0.02 \cdot w$. *Arriving agents choose whether to join the queue or to leave unassigned after observing the length of the queue.*

(a) Build a birth-death process model for the situation above, and solve for the steady state distribution of the number of agents in queue. What is the mean and variance of the steady state number of agents in queue?

Let x denote the number of agents waiting in queue. If a new agent arrives when there are already x agents in queue, that agent will only join the queue if the value V of receiving the item (V is distributed uniform (0, 1)) exceeds the cost of waiting (which includes the processing for the prior x agents plus additional waiting cost for the newly arriving agent. Let $f_{W|x}(w)$ denote the probability density for the waiting time to process all x agents already in queue plus the newly arriving agent (in this example $f_{W|x}(w)$ will be the Erlang distribution that results from the sum of $x+1$ exponentials). In particular, since the item interarrival time is exponentially distributed with mean 1, we have that $E(W|x) = x+1$. Since the waiting cost is $0.02w$ when the waiting time to process all $x+1$ agents equals w, and since the new agent will only join if the item valuation exceeds this waiting cost, we have

$$\begin{aligned} \Pr\{\text{New agent joins queue with } x \text{ agents}|\text{waiting time } w\} &= \Pr\{V > 0.02w\} \\ &= 1 - 0.02w. \end{aligned}$$

Unconditioning over the waiting time distribution yields

$$\begin{aligned} \Pr\{\text{New agent joins queue with } x \text{ agents}\} &= E_{W|x}(1 - 0.02W|x) \\ &= 1 - 0.02(x+1) \end{aligned}$$

Now, given that the agent arrival rate is equal to 2 agents per unit time, but that agents only join the queue with probability $1 - 0.02(x+1)$ when x agents are already present, we see that the joining rate of new agents to the queue when x are already present, call this the "birth rate" $\lambda(x)$, is given by

$$\lambda(x) = 2 \times (1 - 0.02(x+1)).$$

The departure rate of agents who have joined the queue is simply equal to the item arrival rate of 1 per unit time, which means that the "death rate" $\mu(x)$ in this model equals 1.

So, we have a birth-death process with $\lambda(x) = 2\times(1-0.02(x+1))$ and $\mu(x) = 1$. Note that there will never be more than 49 agents waiting, as $\lambda(49) = 0$. We can solve for the steady state probabilities $p(x)$ that there are x agents waiting for $x = 0, 1, 2, ..., 49$ using the standard birth-death equations

$$\lambda(x)p(x) = \mu(x+1)p(x+1) \text{ for } x = 0, 1, ...48$$

$$\sum_{x=0}^{49} p(x) = 1.$$

Since $\mu(x) = 1$, the solution is given by

$$p(x) = \frac{\prod_{j=0}^{x-1} \lambda(j)}{\sum_{k=0}^{49} \prod_{j=0}^{k-1} \lambda(j)}, \; x = 0, 1, 2, ..., 49.$$

I did this in Excel, and discovered to two decimal places that the steady state expected number of agents in queue $E(X) = 24$ while the variance is equal to 25.

(b) Using the methods covered in class, construct an Ornstein Uhlenbeck diffusion model for this same situation, and obtain the limiting distribution of the number of agents in queue. What is the mean and variance?

Let's start with the fluid model for $x(t)$, the (deterministic) number of agents in queue at time t. Working directly with the birth-death model above we get

$$\begin{aligned}
\frac{dx(t)}{dt} &= \lambda(x) - \mu(x) \\
&= 2 \times (1 - 0.02(x+1)) - 1 \\
&= 0.96 - 0.04x \\
&= -0.04 \times (x - \frac{0.96}{0.04}) \\
&= -0.04 \times (x - 24).
\end{aligned}$$

This is exactly in the form that we want, and we see that x^*, the deterministic equilibrium number of agents in the queue equals 24.

We approximate the infinitesimal variance σ^2 as

$$\begin{aligned}\sigma^2 &= \lambda(x^*) + \mu(x^*) \\ &= 2 \times (1 - 0.02(24+1)) + 1 \\ &= 2\end{aligned}$$

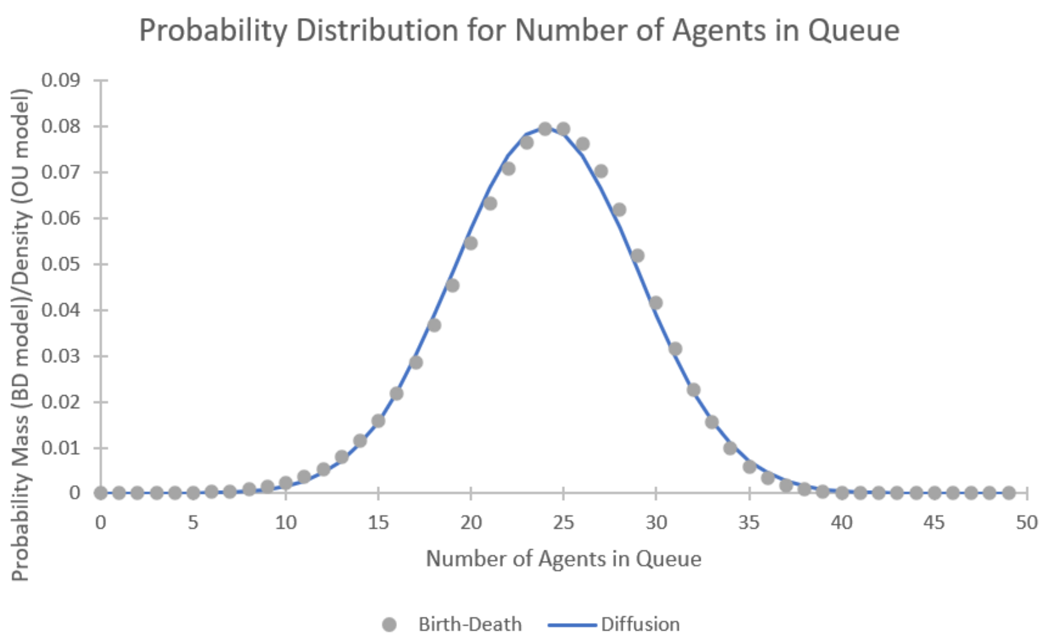
and thus the variance of the equilibrium Ornstein Uhenbeck model as

$$\begin{aligned}Var(X) &= \frac{2}{2 \times 0.04} \\ &= 25.\end{aligned}$$

So the OU model approximates the agent queue lengths as being normally distributed with mean 24 and variance 25. Pretty good approximation, no?

(c) On a single graph, plot the probability distributions for the number of agents in queue resulting from the birth-death model of (a) and the Ornstein Ulenbeck model of (b). Which model provides the most succinct description of the resulting steady state queue length distribution?

Here's my graph:



While it is easy enough just to report the mean and variance for either model, having an approximate normal distribution is much easier for discussion (and likely subsequent calculations) than trying to explain the intricacies of the birth-death process.

3. Big Data at the NSA

According to *Washington Post* investigative reporters, "Every day, collection systems at the National Security Agency intercept and store 1.7 billion e-mails, phone calls and other types of communications." (D. Priest and W. M. Arkin, 2010, "National Security Inc.," *Washington Post*, (July 20) A1.) Suppose that the NSA intercepts communication messages (SIGINT) in accord with a Poisson process where the mean message arrival rate is as stated above. Assume that the NSA initially processes such intercepts for further study (or discard) using automated algorithmic data analyzers or ADEs (i.e. dedicated computers), each of which is capable of analyzing and classifying at most 5 million intercepts per day.

(a) What is the absolute minimum number of ADEs the NSA requires to meet this workload?

From the problem description, we have an arrival rate $\lambda = 1.7 \times 10^9$ and a per-server processing rate of $\mu = 5 \times 10^6$. First principles state the the expected number of busy servers is then given by

$$\begin{aligned} E(B) &= \frac{\lambda}{\mu} \\ &= \frac{1.7 \times 10^9}{5 \times 10^6} \\ &= 340 \end{aligned}$$

which means that the NSA needs 340 ADEs to meet the workload.

(b) Suppose the number of ADEs deployed by the NSA is set equal to 1 + your answer from (a). Using the Halfin-Whitt approximation, what is the probability that all ADEs are busy when a new intercept arrives? Again using Halfin-Whitt, what is the expected length of the queue of communications intercepts waiting for processing by an ADE?

We are told to set the number of servers $n = 341$. To use the Halfin-Whitt approximation, we first compute

$$\begin{aligned}\beta &= \frac{n - \lambda/\mu}{\sqrt{n}} \\ &= \frac{1}{\sqrt{341}} \\ &= 0.0542.\end{aligned}$$

With this value for β, the probability that all 341 servers are busy is given by

$$\begin{aligned}\alpha &= \frac{\phi(\beta)}{\phi(\beta) + \beta\Phi(\beta)} \\ &= 0.9337.\end{aligned}$$

The expected Halfin-Whitt queue length is then given by

$$\begin{aligned}L_q &= \frac{\alpha}{\beta}\sqrt{n} \\ &= \frac{0.9337}{0.0542}\sqrt{341} \\ &\approx 318.\end{aligned}$$

Just for comparison, the Queueing ToolPak results for the probability of delay and expected queue length for the M/M/n queue with the same arrival rate, service rate, and number of servers are Pr{Delay} $= 0.9346$ and $L_q \approx 318$.

(c) Given the potentially serious consequences of a communication intercept, the NSA decides it would like to deploy a sufficient number of ADEs such that the probability that all of them are busy only equals 1%. Using square root staffing and the Halfin-Whitt regime, how many ADEs should the NSA deploy? What is the expected number of intercepts waiting for processing in this situation?

First we need to find that value of β that solves the equation

$$\alpha(\beta) = \frac{\phi(\beta)}{\phi(\beta) + \beta\Phi(\beta)} = 0.01$$

To three decimals I find the root of this equation as $\beta^* = 2.375$. Now we employ square-root staffing to obtain

$$\begin{aligned} n &= \frac{\lambda}{\mu} + \beta^* \sqrt{\frac{\lambda}{\mu}} \\ &= 340 + 2.375\sqrt{340} \\ &= 383.79 \approx 384. \end{aligned}$$

Note that we need to round up as we can't have 0.79 of a server. With 384 servers, the expected number waiting in queue will essentially vanish as

$$\begin{aligned} L_q &= \frac{\alpha}{\beta}\sqrt{n} \\ &= \frac{0.01}{2.375}\sqrt{384} \\ &= 0.08. \end{aligned}$$

4. Terror Queues with All Agents Busy

Recall the Terror Queues model from Policy Modeling, and also described in the paper by the same name posted on the course website. Read over Section 3.1 (Boundary Approximations) on p. 778 of the paper, and focus on the case where the number of busy agents $Y \approx f$, the total number of agents. Taking $Y = f$ reduces the model to one dimension, namely the number of undetected terror plots X. Why? Because in steady state, the number of undetected plots that are detected per unit time equals the the number of detected plots that are interdicted per unit time, and if $Y = f$ this latter rate just equals ρf. Noting that X will increase with the arrival of a new plot, decrease with the detection of a previously undetected plot, and also decrease with the execution of a terror plot:

(a) What is the differential equation for the fluid model for the number of undetected terror plots over time? What is the fluid limit for the number of undetected terror plots?

The fluid model is simply given by

$$\begin{aligned} \frac{dx(t)}{dt} &= \alpha - \mu x - \rho f \\ &= -\mu\left(x - \frac{\alpha - \rho f}{\mu}\right). \end{aligned}$$

Yippee – we have this in exactly the form of an Ornstein Uhlenbeck model, and we see that the fluid limit for the number of undetected plots is given by

$$x^* = \frac{\alpha - \rho f}{\mu}$$

(b) Formulate a steady-state birth-death model for the number of undetected terror plots assuming all agents are busy, and plot the resulting probability distribution for the number of undetected plots assuming $\alpha = 100$, $\mu = 1$, $\rho = 4$, and $f = 15$. Report the mean and standard deviation of the number of undetected plots.

Let's get the model first, and plug in the numbers second. Letting x denote the number of undetected terror plots, the birth rate $\lambda(x) = \alpha$ while the death rate $\mu(x) = \mu x + \rho f$ (the first term reflects successfully executing undetected plots, while the second term represents the detection (and interdiction) of undetected plots). Solving the usual balance equations

$$\begin{aligned} \lambda(x)p(x) &= \mu(x+1)p(x+1),\ x = 0, 1, 2, ... \\ \sum_{x=0}^{\infty} p(x) &= 1 \end{aligned}$$

we obtain the solution

$$p(x) = \frac{\prod_{j=0}^{x-1} \frac{\alpha}{j\mu + \rho f}}{\sum_{k=0}^{\infty} \prod_{j=0}^{k-1} \frac{\alpha}{j\mu + \rho f}}$$

Plugging in the parameter values $\alpha = 100$, $\mu = 1$, $\rho = 4$, and $f = 15$ I obtain (from Excel) $E(X) \approx 40$ and $Var(X) \approx 100$, which means $StDev(X) \approx 10$. The plot of $p(x)$ appears below in part (c).

(c) Formulate an Ornstein Uhlenbeck diffusion model for this situation using the methods described in class. Derive the steady state distribution of the number of undetected terror plots and plot the resulting density for the same parameters in part (b) on the same graph. Report the mean and standard deviation of the number of undetected plots. Which approach was easier to implement?

From part (a) we already know that the fluid limit, which in the Ornstein Uhlenbeck approximation is taken as the expected value, is given by

$$\begin{aligned} x^* &= \frac{\alpha - \rho f}{\mu} \\ &= \frac{100 - 4 \times 15}{1} \\ &= 40. \end{aligned}$$

That agrees perfectly with the birth-death model. To get $Var(X)$, we first estimate the infinitesimal variance of the diffusion approximation at the fluid limit which yields

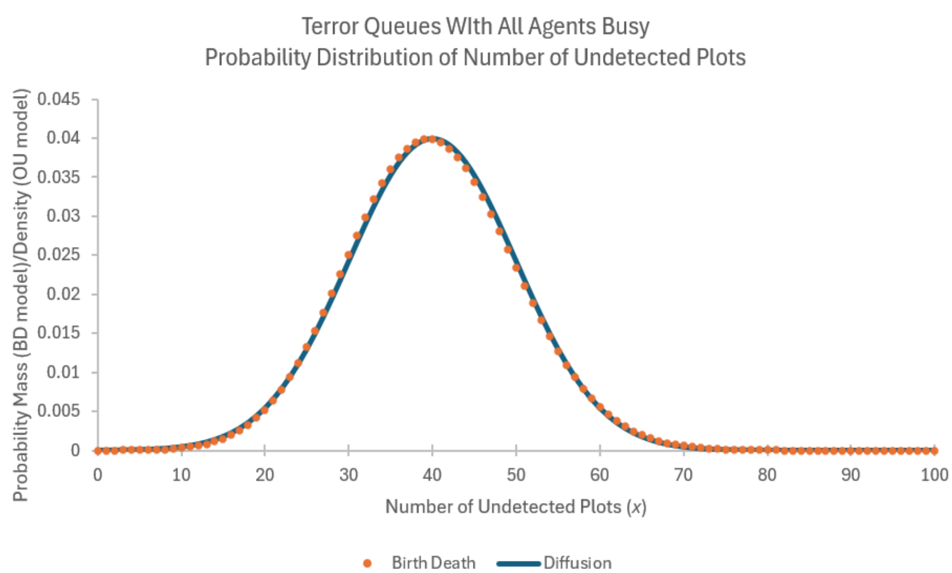
$$\begin{aligned} \sigma^2 &= \lambda(x^*) + \mu(x^*) \\ &= 100 + 1 \times 40 + 4 \times 15 \\ &= 200. \end{aligned}$$

We get $Var(X)$ by dividing the infinitesimal variance by twice the drift coefficient ($\mu = 1$) which yields

$$Var(X) = \frac{200}{2 \times 1} = 100$$

and thus $StDev(X) = 10$, in perfect agreement with the birth-death model.

Here is a graph showing both the birth-death probability mass function (dots) and the Ornstein Uhlenbeck normal density (smooth curve).



It is pretty much impossible to tell these two probability models apart. I think implementing the OU model was much simpler – the required equations for the mean and variance are simple, whereas solving for the birth-death probabilities, though not hard, is more work. Again we have a very succinct summary of the situation – the number of undetected terror plots follows a normal distribution with a mean of 40 and a standard deviation of 10. And this is when all the agents are busy. Sounds like we need more counterterrorism agents!