# AMSI VACATIONRESEARCH SCHOLARSHIPS 2021–22

## Get a taste for Research this Summer
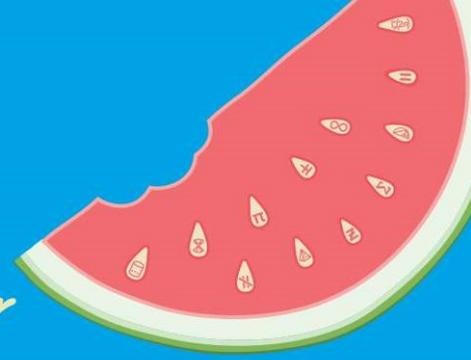
# Mean Exit Time for Diffusion on Irregular Domains

Joshua Roebuck

Supervised by Prof. Matthew Simpson & Prof. Scott McCue

Queensland University of Technology

AMSI AUSTRALIAN MATHEMATICAL SCIENCES INSTITUTE

# 1    Table of Contents

## 2    Abstract

First passage time is an incredibly useful metric in fields across biology, chemistry, physics, and economics. In particular, when looking at the first passage time of a particle crossing a boundary, the mean time it takes for this to occur is a scenario of interest. The mean exit time across several regular geometries can be found through standard solution of a partial differential equation. We were interested in finding computationally efficient numerical methods to solve this problem that could be applied to irregular shapes. The boundary integral method was researched, developed, and applied to example cases to see how well it extended to irregular shapes. The results were positive, and the boundary integral method provided an effective numerical solution to various shapes. This method was determined to be significant as it provided a robust approach to real life applications of this problem.

## 3    Introduction

First passage time is a broad mathematical concept that involves describing processes by their mean time to travel from an initial state to some final state [1]. Applications of this idea are important in various areas including biology [2], physics [3], and economics [4].
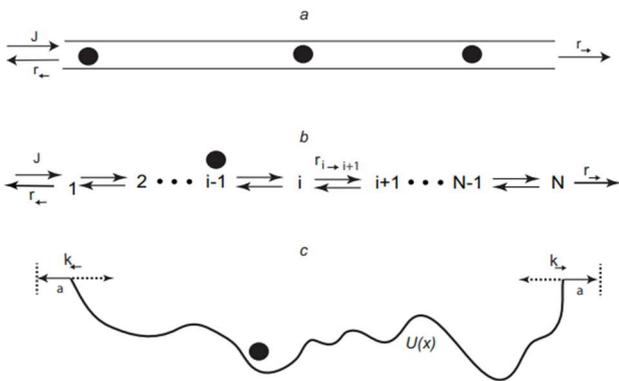


Figure 1: Channel transport representations in biology [2]
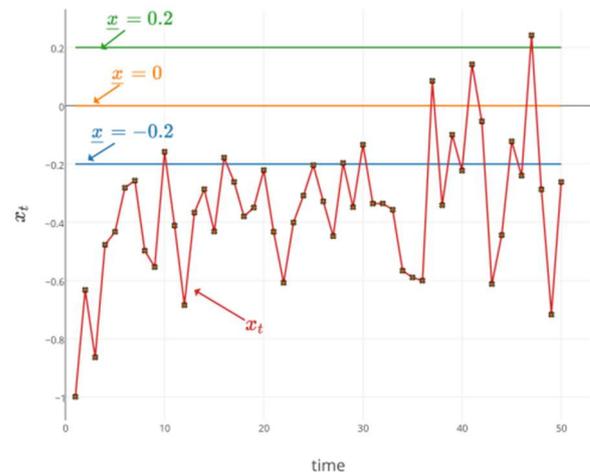


Figure 2: Sample of path processes with different hitting barriers in econometrics [4]

When investigating diffusion of particles, a certain application of first passage time is a useful property to be able to determine – the mean time a particle takes to reach an absorbing boundary. The average lifetime of the particle will depend on where the particle starts and the shape of the geometry.

A stochastic random walk model is constructed in MATLAB to simulate particle diffusion. Certain customisable parameters are implemented to fit the model to real life applications. The simulation was run for an increasing number of trials to get a feel for the patterns we are investigating, and to see visually how the random walk worked.

Analytical and numerical methods are then explored to solve the problem on different domains. Exact solutions for disc and elliptical domains are known as a solution to a partial differential equation [5], however numerical methods are then investigated to use for more irregular shapes, where an exact solution is harder to derive.

The boundary integral method is found to provide accurate numerical solutions with limited use of computational power. It is compared to exact solutions on domain where those are known, and tested on more irregular domains such as larger polygons and periodically determined boundaries.

## Statement of Authorship

Under the direction of my academic supervisors, I constructed the stochastic model in MATLAB, and then used already developed plotting tools to plot my solutions. I analysed the already developed exact solutions, and learned how to manipulate the developed boundary integral code to develop solutions for different domains. I drafted the report and got supervisor feedback and approval.

# 4 Background

## 4.1 Parameters

Diffusion of a particle within a 2D geometry is modelled as a random walk in MATLAB. Setting up this model introduces parameters that are decided on and can be altered depending on the application.

The distance the particle moves at each step is a preset value ($\Delta$), along with the time between each step ($\tau$). The probability to actually move or stay still at each step (P) is another parameter to introduce to further customize the diffusivity model. These properties are required for the calculating the diffusivity ($D$) which is a key value in the model.

$$D = \frac{P\Delta^2}{4\tau} \tag{1}$$

The random aspect of the model is implemented through the aforementioned probability, as well as the direction that the particle chooses to move in at each step. The range of angles the particle can move in ($[\theta_1, \theta_2]$) is another parameter that is introduced. This parameter is useful as it allows motion to be directed if needed. A uniform distribution is assumed for these parameters, however can be altered if needed.

For ease of definition, a standard model is defined with certain parameters that will be kept constant (unless specifically stated) for the duration of the project.

$$\Delta = 0.01, \qquad \tau = 1, \qquad P = 1, \qquad \theta_1 = 0, \qquad \theta_2 = 2\pi$$

## 4.2 Algorithm

The algorithm for the random walk code is determined and implemented. For any given starting point,

1. Generate a random number to determine if the particle will move this step.
2. Generate a random angle within the defined range for the particle to move.
3. Move in the chosen direction for the defined step length.
4. Increment the step counter by the step time.
5. Check if new position of the particle has crossed the boundary.

The random walk model is tested using the basic geometry of a disc. It can be parameterised as follows:

$$x = R\cos\theta, \qquad y = R\sin\theta, \qquad \theta \in [0, 2\pi]$$

Using the origin as a starting point, five tests are run to produce Figure 3.
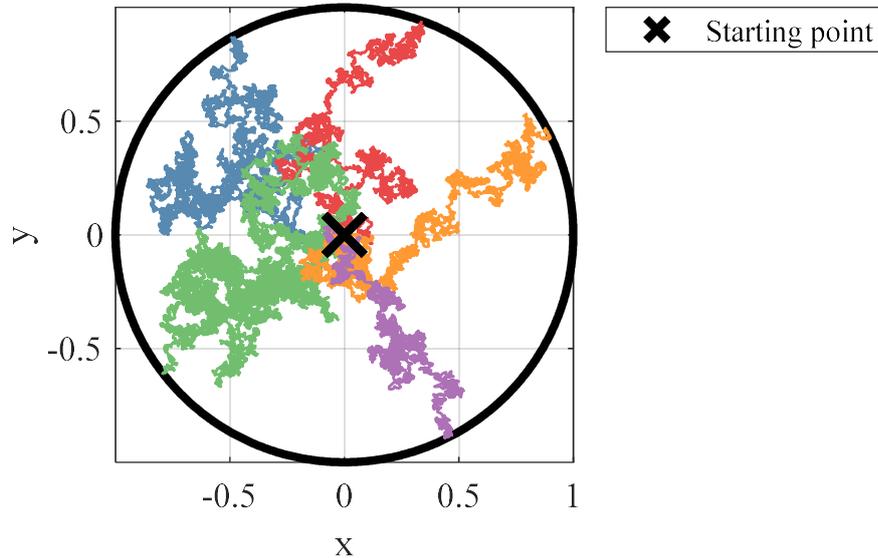


*Figure 1: Five random walk paths starting from the origin (geometry chosen to be circle of radius 1)*

## 4.3   Geometry Mesh and Starting Points

The random walk code runs as desired and so it can be implemented to run across the whole shape. To determine a set of points across the shape that encompasses enough information, a mesh can be generated across the geometry. This is done using GMSH software which outputs the full set of nodes, the triangles that connect these nodes, and the boundary nodes that form the outside of the shape. This is incredibly useful information to have both for the random walk and for numerical methods that will be explored later on. This mesh also introduces another parameter which is the refinement of the mesh. The more refined the mesh is, the more random walk starting points there will be. This leads to a more accurate solution, but as a result, more computational power is required.
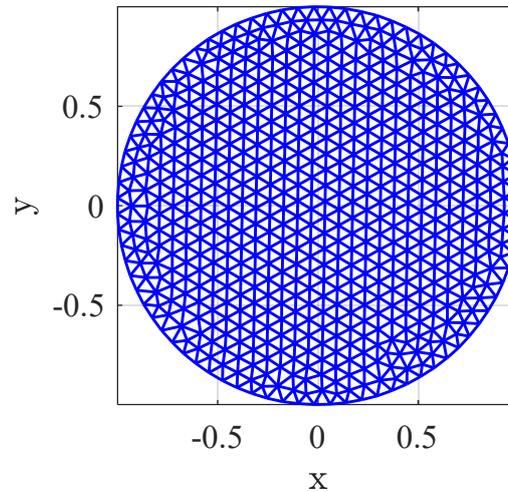
*Figure 4: Full geometry mesh of the disc domain with refinement parameter set to 0.08*

Once generating the mesh, the random walk is looped to run at each of these starting points. Another test is run, this time starting a five different points, as seen in Figure 5.
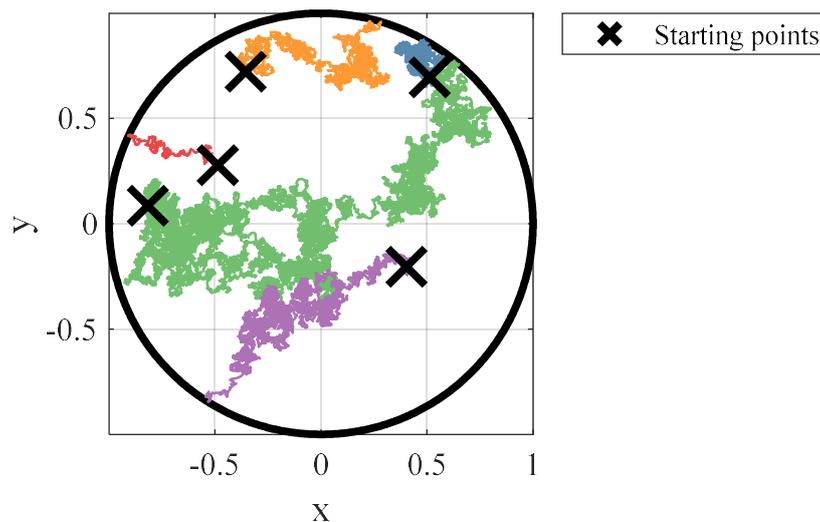


*Figure 5: Five random walk paths starting from five different starting nodes*

Figure 3 and figure 5 exemplify the main consequence of the random walk model, which is that each instance of starting at the same point can lead to wildly different paths. To characterise the exit time from each starting point, multiple trials will be run for each point and the times will be averaged, providing the mean exit time. This introduces one more parameter which is how many trials will be run at each starting point. More trials will lead to a more accurate result, but will proportionally increase the runtime of the simulation. The algorithm in Section 2.2 is modified to first loop for each trial at the current starting node, and then loop through each node and start the simulations there.

# 5 Random Walk Results

The random walk is run three times on the disc domain to see the effect of increasing the number of trials at each node.
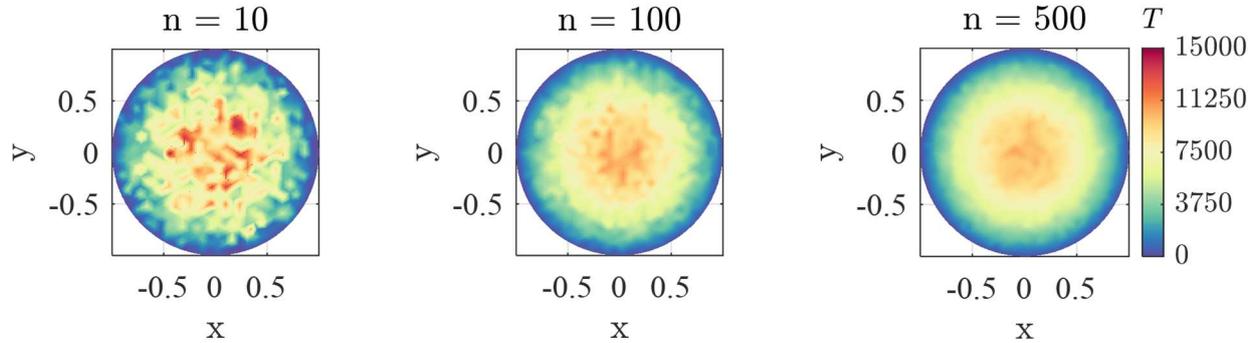


*Figure 6: Standard model run on the disc domain for 10, 100, and 500 trials per starting node*

It is observed that as the model uses an increasing number of trials, the solution approaches a smooth pattern where in general, particles starting toward the centre of the domain have a longer mean exit time to those starting near the edge. It is also seen that while the 10 trials plot shows times upwards of 15000 units, the maximum time after using more trials looks to be around 10000-11000. This reinforces the importance of taking as many trials as possible to obtain an accurate solution, as this allowed high and low outliers to be accounted for. It is noted that while the 500 trials plot shows a relatively accurate solution, it still is patchy and is not entirely reliable. Further means of solving need to be explored to obtain better solutions with more efficient use of computational power.

# 6 Exact Solutions

Exact solutions can be determined analytically by solving partial differential equation (2).

$$\nabla^2 T = -\frac{1}{D} \tag{2}$$

A boundary condition is placed on the solution by drawing some conclusions about the desired solution. If a particle starts on the boundary ($r = R$), its exit time will be 0, as it has already reached the boundary. In the centre of the disc ($r = 0$), The partial derivative in all directions of $r$ can also be assumed to be 0 to keep the solution symmetrical. Hence,

$$T(r) = 0, \qquad r \in \partial\Omega$$

$$\frac{\partial T}{\partial r} = 0, \qquad r = 0$$

7

Solving equation (2) and applying the boundary conditions yields the following exact solution:

$$T(r) = \frac{R^2 - r^2}{4D}, \qquad r \in [0, R] \tag{3}$$

It is important to note that the disc is a special case where the exact solution can be represented in radial coordinates. Other solutions will be expressed in regular Cartesian coordinates. This solution is observed visually using a surface plot to compare to the random walk plot.
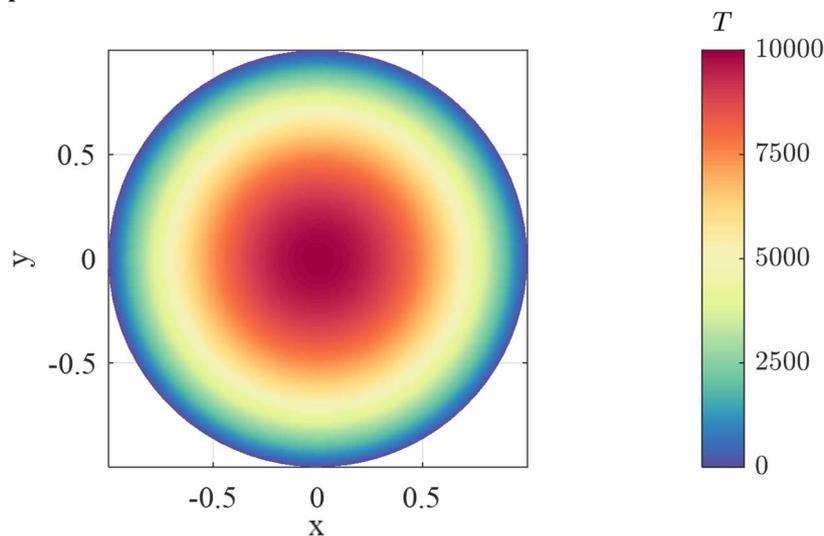


*Figure 7: Exact solution for mean exit time on a disc of radius 1*

Visually, the exact solution follows the same pattern that the random walk solutions were tending towards as the number of trials at each node increased. Therefore, it is clear the exact solution is the preferred method to solve mean exit time problems. However, this method is only convenient for simple shapes. As shapes get more complex and irregular, this method becomes incredibly difficult. Hence, numerical methods are explored to tackle these problems computationally.

# 7   Numerical Methods

## 7.1  Different Meshes

The main mesh encountered so far is the full geomtry mesh seen in Figure 4. While this was used for the random walk starting points, these nodes also lend themselves to use in numerical methods. One method that utilises this full mesh is the finite volume method.

However, when solving for an accurate approximation of the solution, more mesh nodes will be required, making this method extremely computationally demanding. The full mesh shown in Figure 4 has 1183 nodes, meaning MATLAB will need to solve a 1183x1883 matrix system. Rather than using a full mesh, another way to capture the shape of the geomtry is to just mesh the boundary.
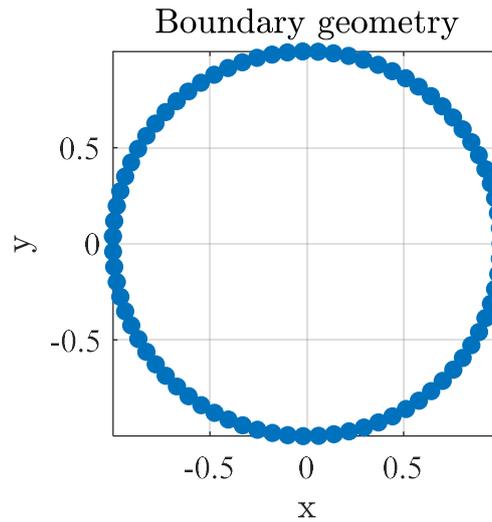


*Figure 8: Boundary nodes of the mesh isolated*

In comparison, only 79 nodes are needed to capture the boundary with the same accuracy as the full mesh. Developing a numerical method based on these nodes is desired to save computational power and develop an accurate solution.

## 7.2 Boundary Integral Method

The boundary integral method is a method based on complex variable algebra to develop an accurate approximation to the mean exit time solution. Firstly, the solution is split into a particular solution and a homogenous solution.

$$T(x, y) = T_P(x, y) + \phi(x, y) \tag{4}$$

The particular solution is chosen, and so it is known. The homogenous solution must be a harmonic function satisfying Laplace's equation:

$$\nabla^2 \phi = 0 \tag{5}$$

This is important as it leads to the next step. $\phi$ being harmonic implies that a complex function, $f$, can be defined such than $f$ is analytic.

$$f(z) = \phi(x, y) + i\psi(x, y) \tag{6}$$

Along the boundary, some information about $\phi$ can be determined. Applying the boundary condition, $T = 0$, to equation (4) allows the following relatinship to be identified:

$$\phi = -T_P \tag{7}$$

From here, this information can be used to determine $\psi$ along the boundary. To do this, Cauchy's integral formula will be applied. This formula is an elementary result from complex variable methods.

$$\oint_{\partial\Omega} \frac{f(z)}{z - z_0} dz = a i f(z_0), \qquad a = \begin{cases} \pi & z_0 \in \partial\Omega \\ 2\pi & z_0 \in \Omega \end{cases} \tag{8}$$

When applied on the boundary, an equation can be formed in which $\psi$ is the only unknown and can hence be solved for.

$$f(z_0) = \frac{1}{\pi i} \oint_{\partial\Omega} \frac{f(z)}{z - z_0} dz \tag{9}$$

This integral is essentially a contour integral around the boundary of the shape, and so this introduces a condition on this method – the boundary has to be paramterisable. Furthermore, the paramaterisation of the boundary must be anticlockwise around the domain for equation 9 to be valid. When subsituting in the known parameters into this equation, a large and difficult to solve integral is formed. Rather than trying to solve this manually, or use a package to solve it symbolically, numerical quadrature can be used.

In this case, the trapezoid rule provides enough accuracy with less computational power than higher order methods. Using the trapezoid rule will provide a linear system which can be solved simply in MATLAB. This system will be much smaller than if a full gemotry mesh were used, highlighting the benefit of this method. Once solved, all the information about $f$ is known on the boundary.

From here, Cauchy's integral formula can be reapplied to evaluate $\phi$ at points on the interior. Then, the solution can be recounstructed, as shown in equation 4.

### 7.3  Numerical Method Results

To test the boundary integral method, it is tested on the circle and compared to the exact solution. The circle parameterisation was shown earlier, and these can easily be differentiated. Hence, the circle is a valid shape to use the method on.
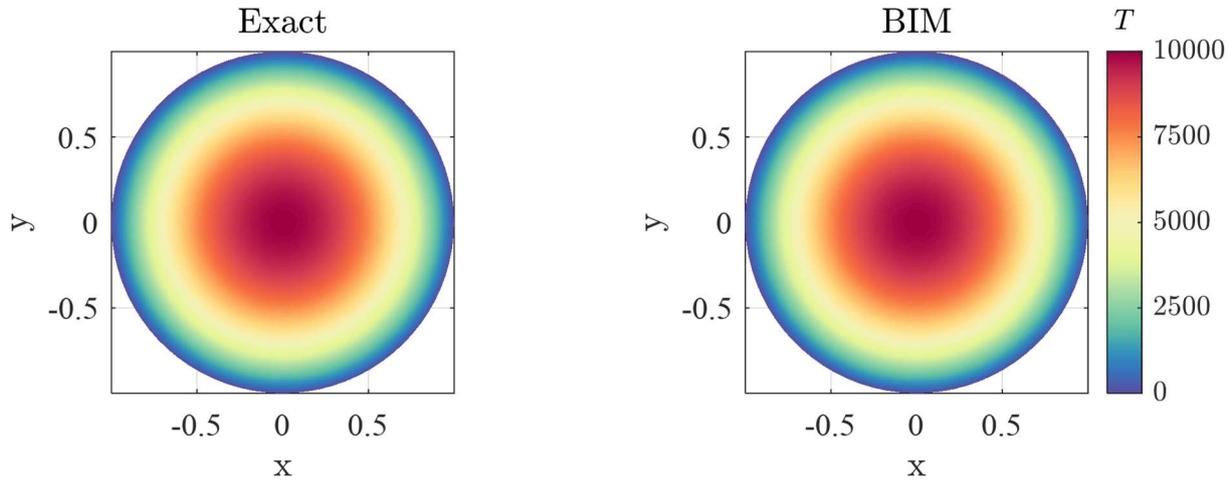


*Figure 9: Exact solution compared to the solution generated by the boundary integral method*

Visually, the plots look identical. There will be some small error due to the approxmation stage of the process – this can be quantified using a discrepancy measurement equation posed in [5].

$$e(x, y) = 100 \frac{|T_{Exact}(x, y) - T_{BIM}(x, y)|}{\max_{(x,y) \in \Omega} |T_{Exact}(x, y)|} \tag{10}$$

An error plot can be created by calculating the error at each node using this equation.
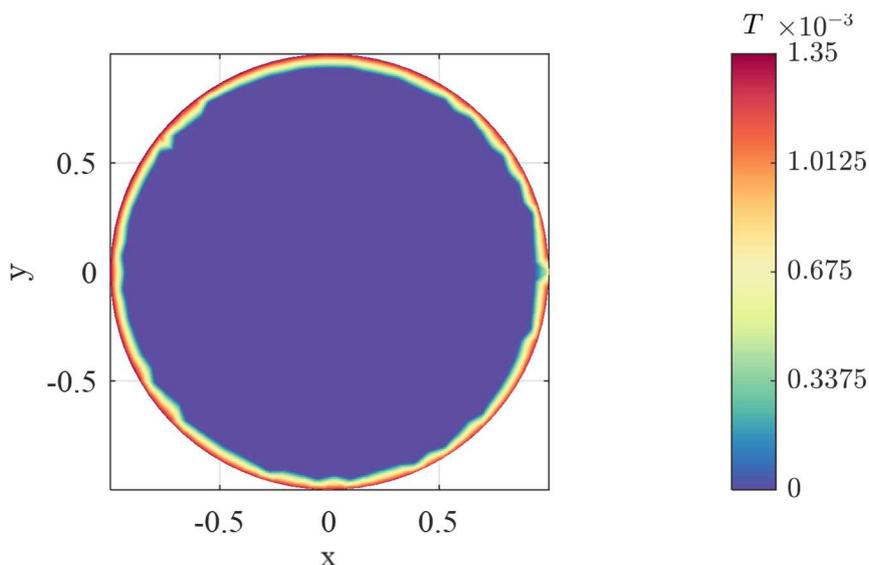


*Figure 10: Plot of the error between the exact solution and the boundary integral approximation*

For the majority of the shape, the error is near 0, however the area at the boundary shows a spike in the error to just under 0.00135. This is not a relatively small error, and is negligible for most purposes, however is important to note.

# 8 Other Applications

## 8.1 Ellipse

An ellipse with semi-major axis, a, and semi-minor axis, b, can be parameterised as follows:

$$x = a\cos\theta, \qquad y = b\sin\theta, \qquad \theta \in [0, 2\pi]$$

These are easily differentiable, and so are valid to use the boundary integral method on. A random walk with 500 trials at each node was also run for the elliptical domain for comparison. Furthermore, an exact solution is actually known for the elliptical domain.

$$T(x, y) = \frac{a^2 b^2}{2D(a^2 + b^2)}\left[1 - \frac{x^2}{a^2} - \frac{y^2}{b^2}\right] \tag{11}$$

These three plots are shown in Figure 11 for comparison.
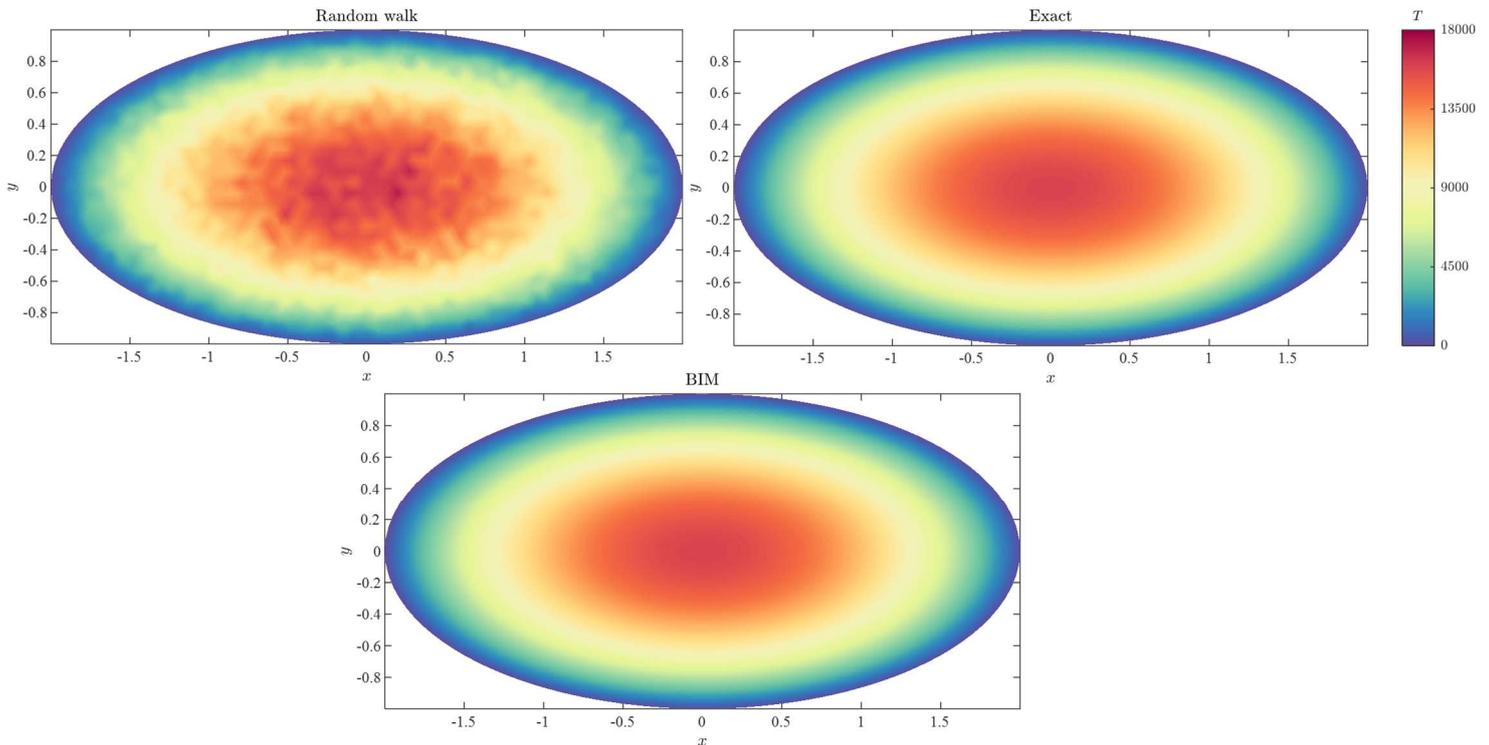


*Figure 11: Random walk plot (using 500 trials at each node), exact solution plot, and boundary integral solution plot, on ellipse of semi-major axis length 2 and semi-minor axis length 1*

12

It is observed that, similar to the disc domain, the stochastic model seems to be approaching a pattern as the number of trials at each node increases. This smooth pattern is shown fully formed by the exact solution, as expected. The solution formed by the boundary integral method solution again looks identical to the exact solution. The error at each node can again be measured using equation 10 and plotted.
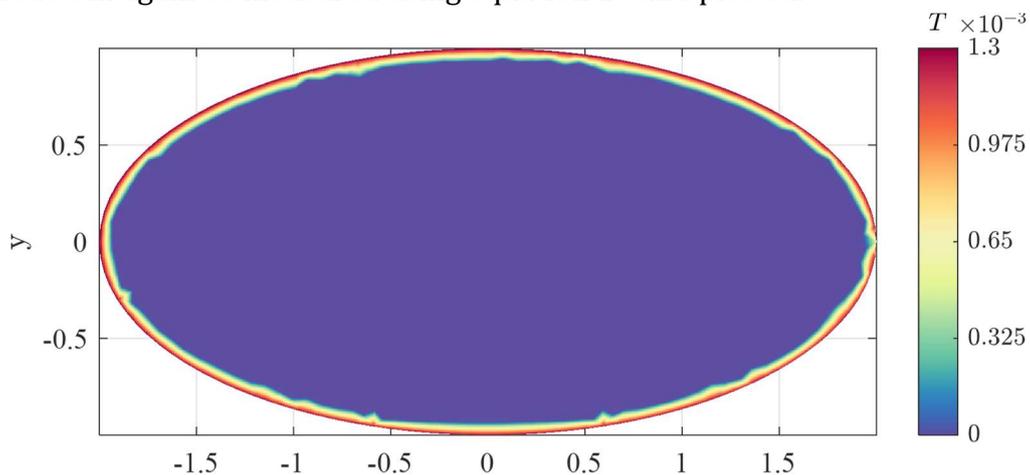


*Figure 12: Plot of the error between the exact solution and the boundary integral approximation on the elliptical domain*

Similarly to the error on the disc domain, the error appears to be near 0 across the majority of the domain. Around the boundary, the error spikes close to 0.0013, however this is small and neglible when considering the actual size of the values at these nodes.

## 8.1 Pentagon

Next, a pentagon domain is investigated. Pentagons, along with along any regular polygon, can be parameterised relatively easily as they are solely comprised of straight lines. If the coordinates of each vertex are known, then line segments can be paramterised between each can be determined. Linear functions are also easily differentiated, making them suitable for the boundary integral method. Exact solutions can be developed for polygons, however, they become increasingly complex, and so this was not found in this study. A random walk was run, and the boundary integral method was applied.
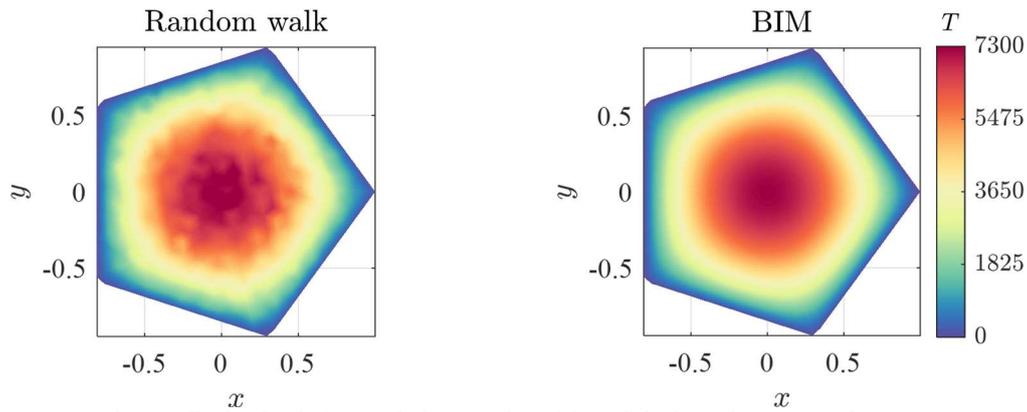
*Figure 13: Random walk plot (with 500 trials from each node), and the boundary integral approximation on the pentagon domain*

Once again, it is observed here that the boundary integral method provides an accurate solution that the random walk appeared to be approaching. While there is no numerical error to calculate here, a visual analysis shows the boundary integral method works on the pentagon domain.

## 8.3 Irregular Shape

Finally, a more perturbed, irregular shape was tested. Its parameterisation was made by a sum of different cosine and sine functions, both of which are straightforward to differentiate. Both the random walk and boundary integral method were run for this domain also.
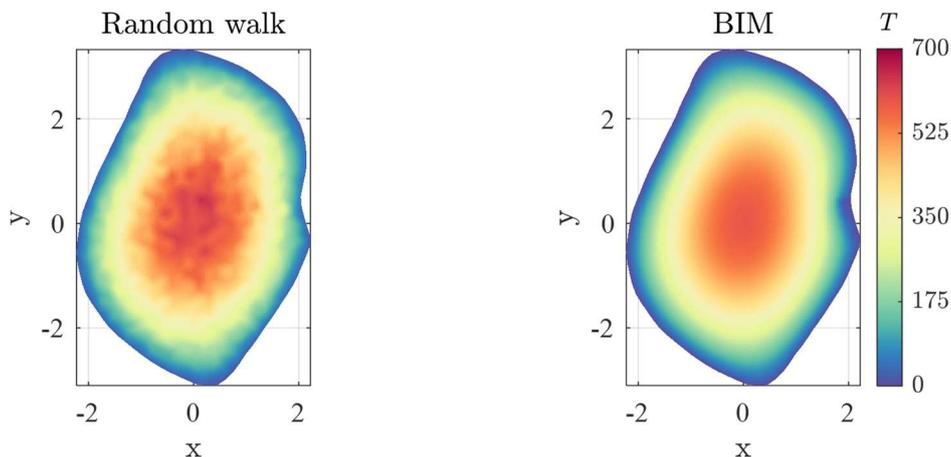


*Figure 14: Random walk plot (with 500 trials from each node), and the boundary integral approximation on the irregular domain*

The boundary integral method seems to work as intended here, again providing a smooth solution to the problem that the random walk seemed to be tending toward. Small errors can be noticed here around the 'sharper' points of the boundary, suggesting these are the points where the approximation differs the most. This could be investigated further by using a more refined mesh and running the simulation and approximation again.

# 9 Conclusion and Recommendations

This project investigated techniques to modelling and solving mean exit time problems for diffusion on different domains. Stochastic modelling with appropriate parameters allowed for realistic modelling of the diffusion process. Exact solutions could be formed on regualr domains through analytical techniques to solving partial differential equations. For irregular domains, the boundary integral method provided an efficient and accurate numerical method.

From here, the project can be furthered and extended down various avenues. Firstly, the application of the boundary integral method in this project showed some errors and places for improvement. This can be further investigated by using more refined meshes, and testing more different domains. Different boundary conditions could be applied to explore their effect. For example, half of the disc domain could be made to be a reflective boundary rather than an absorbing boundary. Furthermore, the model could begin to be applied to real life appplcations like those mentioned in the introduction. The tools have been developed, so parameter measurement could be completed and deployed onto the model.

# 10 References

[1] Polizzi, N. F., Therien, M. J., & Beratan, D. N., 2016, 'Mean First-Passage Times in Biology', Israel journal of chemistry, 56(9-10), 816–824

[2] Iyer-Biswas, S., Zilman, A., 2015, 'First Passage processes in cellular Biology'

[3] Ryazanov, V. V., 2021, 'First passage time and statistical thermodynamics'

[4] Jaskowski, M., van Dijk , D., 2015, 'First-Passage-Time in Discrete Time'

[5] Simpson, M.J., VandenHeuvel D.J., Wilson J.M., McCue S.W., Carr E.J., 2021, 'Mean exit time for diffusion on irregular domains', New Journal of Physics. 23, 043030.