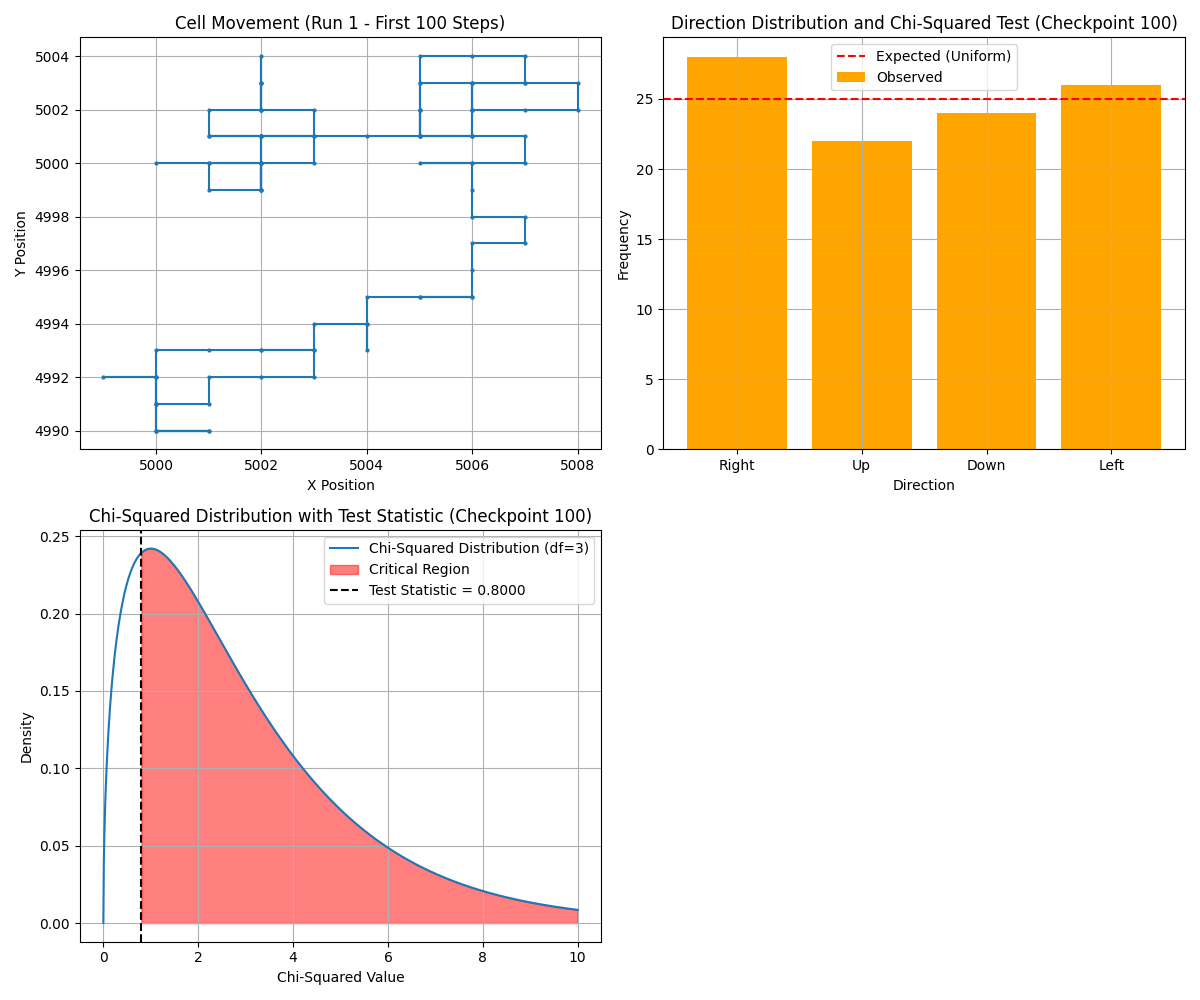
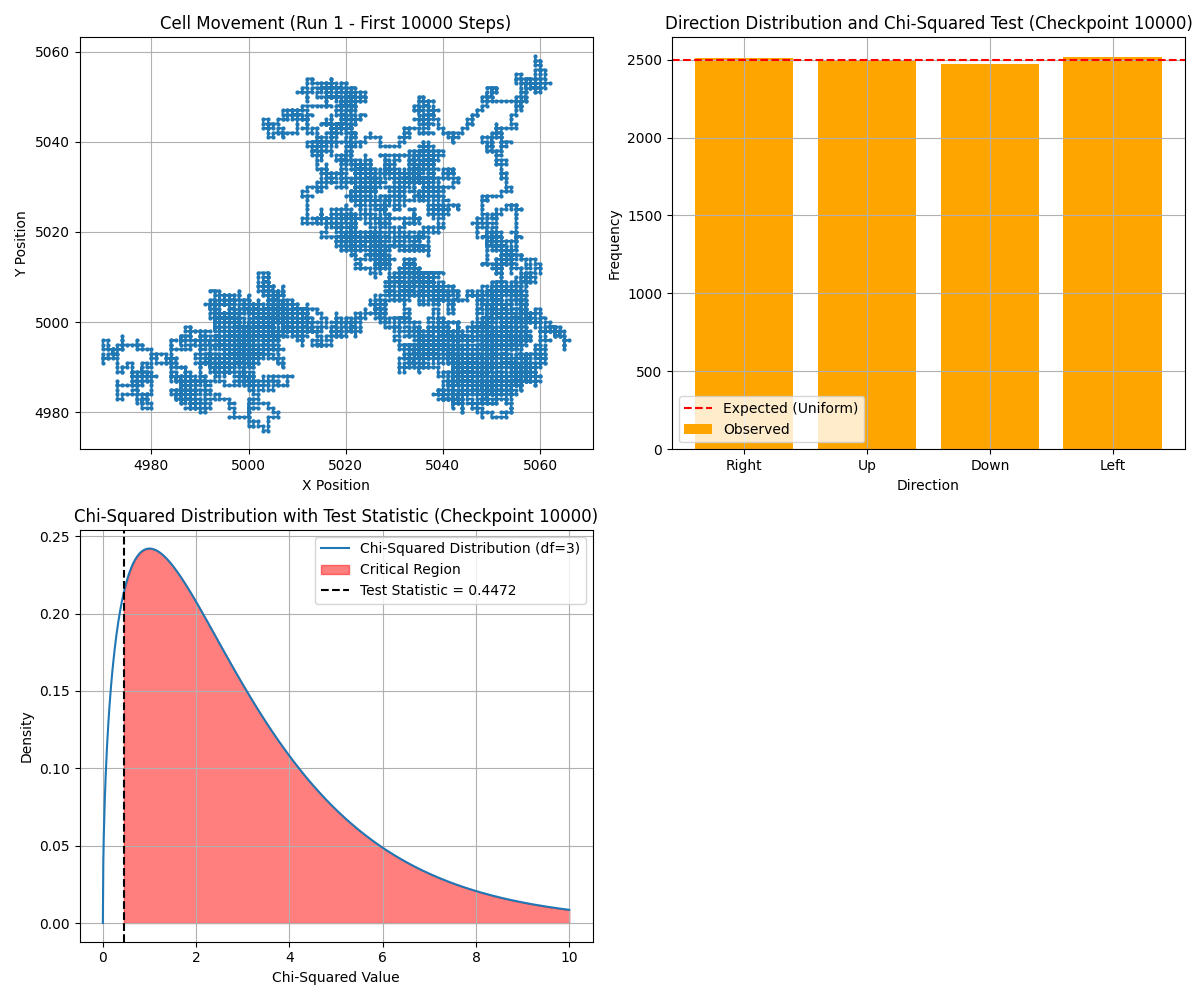
## Introduction

This report explores computational simulations to explore randomness and distributions through the means of modelling biological processes such as random cellular movements and tumour growth dynamics. Using cellular automata and the Gompertz growth model, the simulations aim to replicate and analyse the interaction between randomness, growth, and complexity within computing. By understanding how distributions evolve and applying mathematical principles, this report provides insights into randomness, the law of large numbers, and the computational trade-offs inherent in modelling complex systems.

## Task 1.1: Cellular Automata Movement in Four Directions

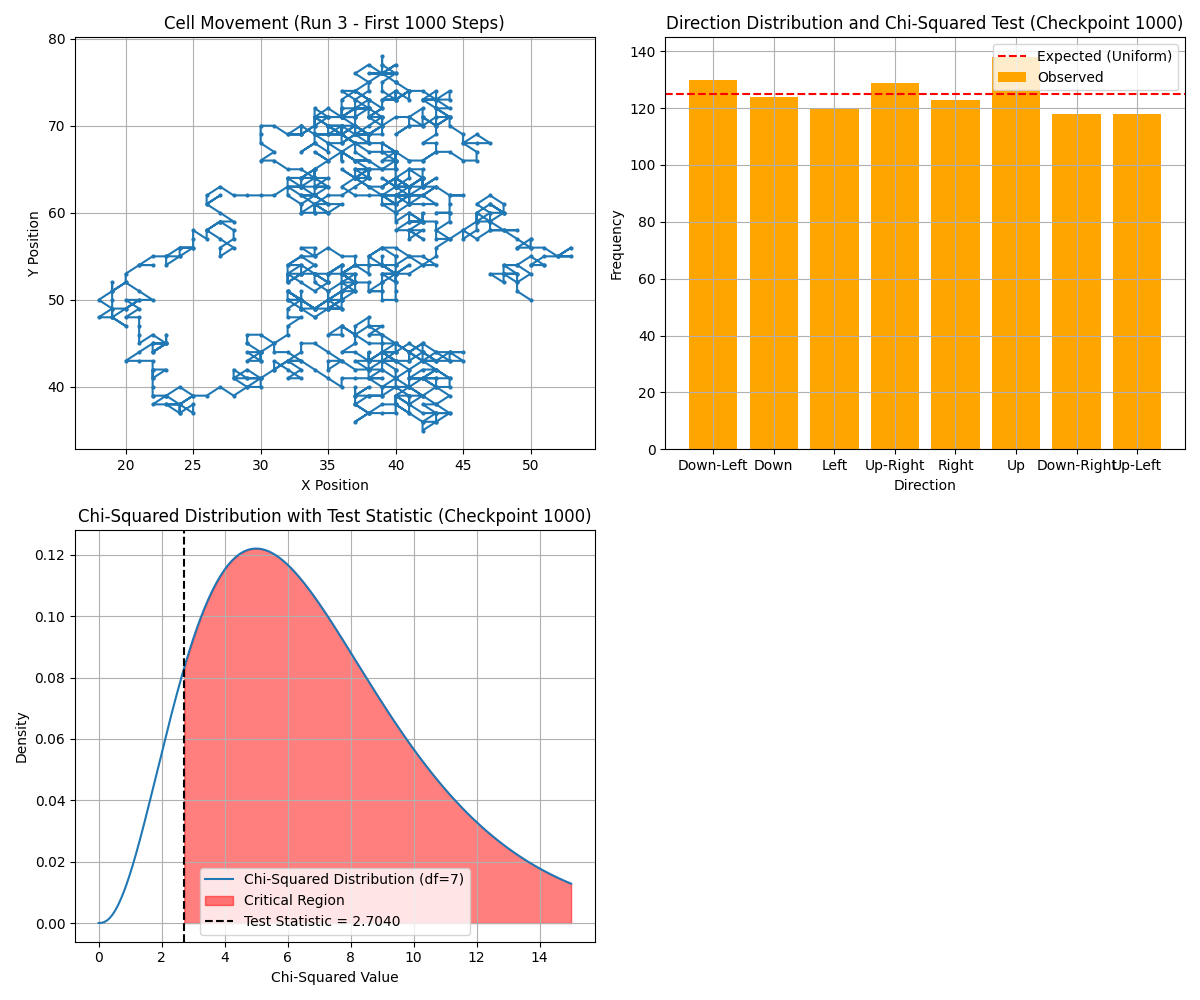
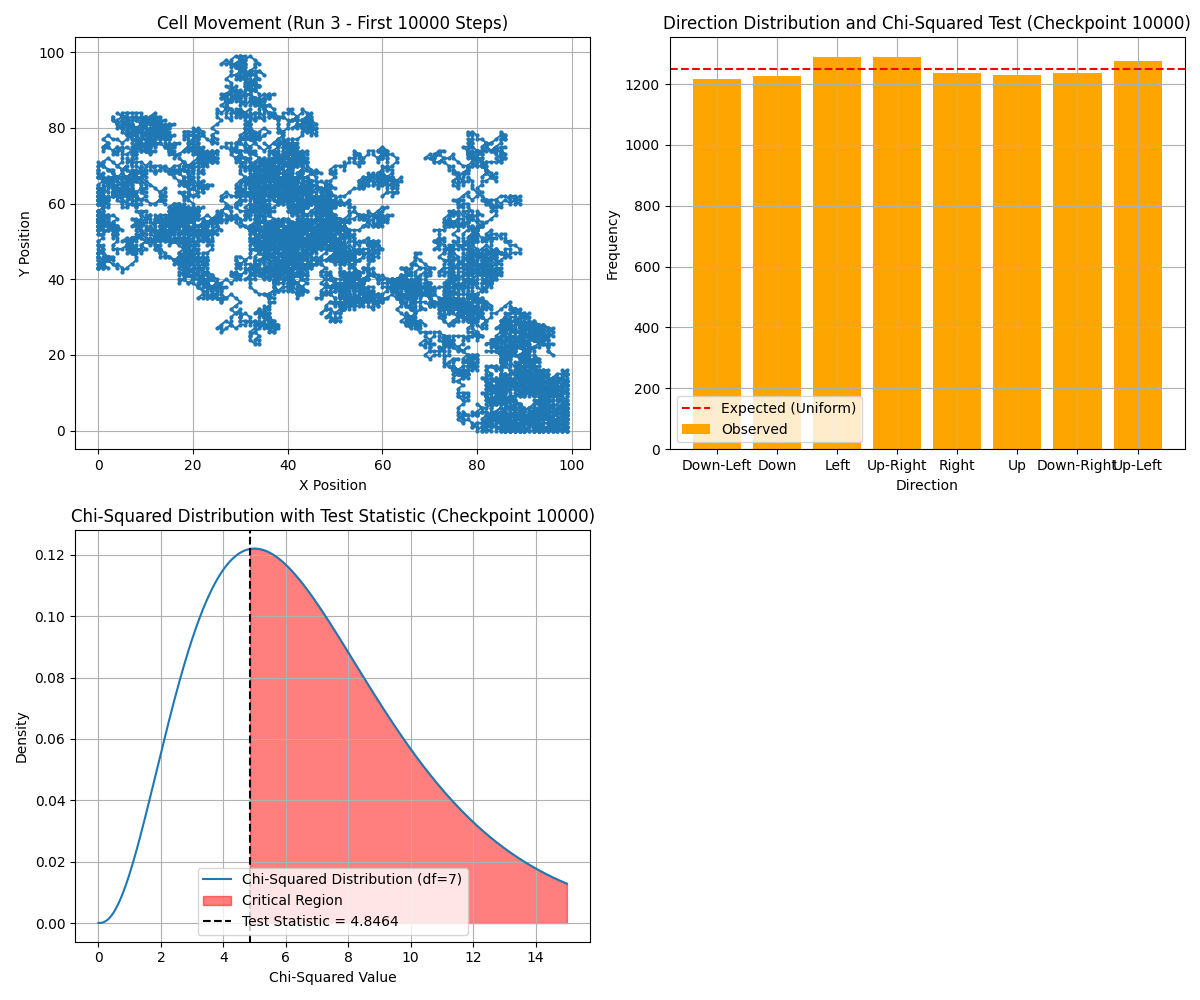
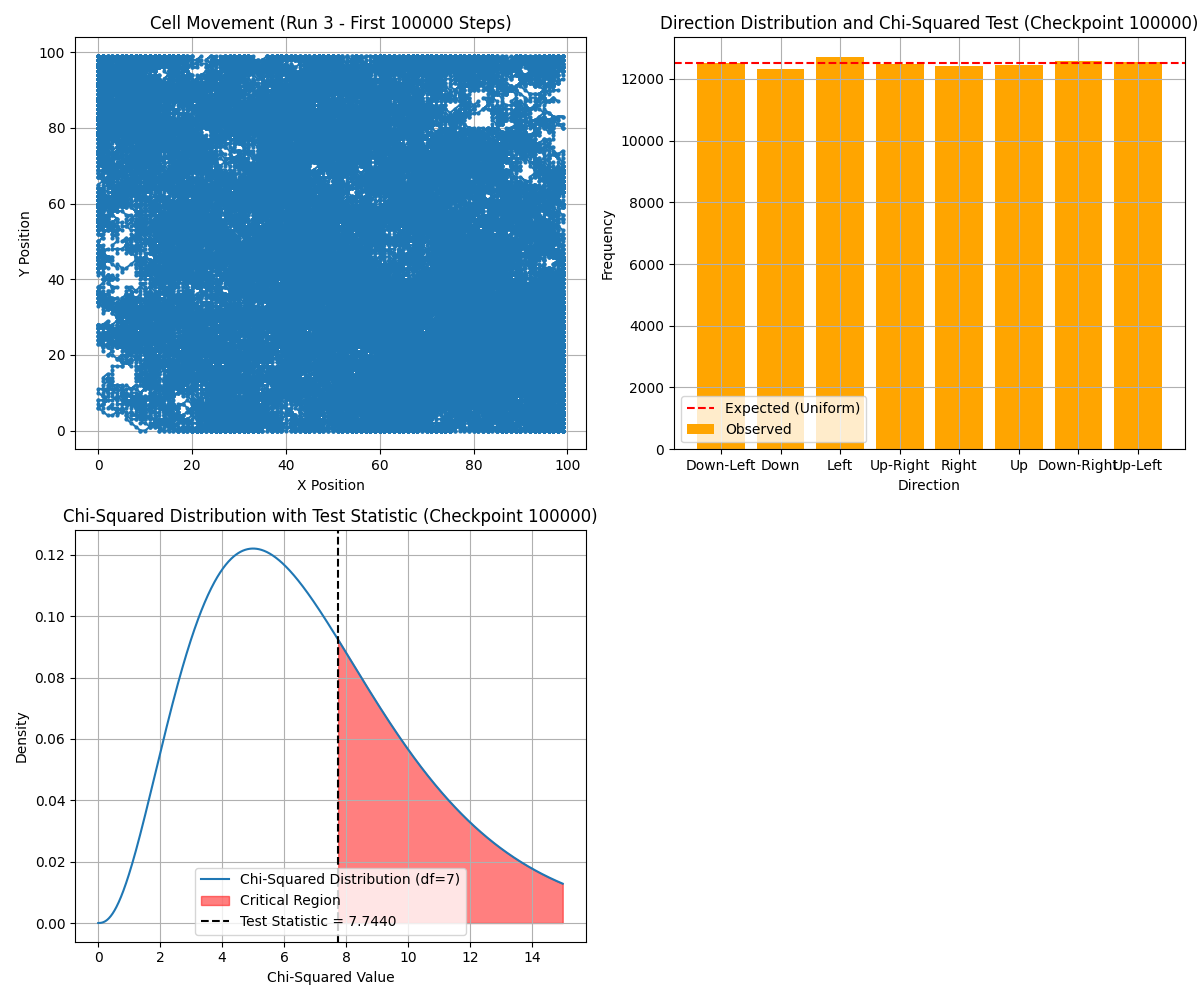
The first task involves simulating the random movement of a cell on a 100x100 grid in four cardinal directions. Randomness is generated using a uniform distribution, ensuring each direction has an equal probability of being chosen. A Chi-Squared test was employed at multiple checkpoints (e.g., 100, 1000, and 10000 steps) to evaluate the uniformity of the distribution there are many different kinds of distribution such as the normal distribution, which describes data using a probability density function with mean μ and standard deviation σ and is often used in random number generation (Shiflet & Shiflet, 2014). The uniform distribution ensures that all outcomes have an equal probability in discrete cases, or that equal-length intervals have an equal chance of occurring in continuous cases (Shiflet & Shiflet, 2014). The Bernoulli distribution arises when modelling binary outcomes, and its conditional version, the conditional Bernoulli distribution, is useful when several Bernoulli random variables are conditioned on their sum equalling a specific value, with applications in sampling and hypothesis testing (Chen and Liu, 1997). The beta-binomial distribution applies when the probability of success in a binomial process itself follows a beta distribution, offering a model for over-dispersed data, common in biological and reliability studies (Skellam, 1948; Altham, 1978). Lastly, the Poisson distribution generalizes the binomial distribution by allowing different probabilities of success in each trial, which is useful in survey sampling and logistic regression (Chen and Liu, 1997).

Initial results reveal noticeable biases in direction frequency due to the limited number of steps. For example, at 100 steps, the observed frequencies often deviate significantly from the expected values, resulting in higher Chi-Squared statistics and lower p-values. These deviations occur because smaller sample sizes make it abundantly clear random fluctuations, a phenomenon explained by the law of large numbers. As step counts increase to 1000 or 10000, the observed distributions converge closer to uniformity. This convergence reflects how larger datasets mitigate the effects of random variability, resulting in more balanced direction frequencies this can be seen in the figures below.  
  
Figure: Movement Trajectory in 4 Directions at 100 Steps  
  
  
Figure: Movement Trajectory in 4 Directions at 10,000 Steps

Graphical plots of cell trajectories provide a visual representation of the movement pattern. For instance, at 100 steps, the movement path is jagged and exhibits clustering in certain directions, indicating the influence of random noise. By contrast, the trajectory at 10000 steps shows a more evenly distributed path across the grid. These observations highlight the importance of sample size in achieving reliable results.

Computationally, the task is relatively straightforward due to the simplicity of the random number generation and movement logic. However, optimizing the initialization of the random number generator, such as using consistent seeding, could further reduce early-stage biases. Additionally, computational efficiency was maintained by leveraging NumPy’s optimized functions for random number generation, ensuring the model scales effectively for larger grids or step counts. Future extensions of this simulation could include non-uniform probabilities for movement directions to reflect potential real-world scenarios.

Task 1.2: Cellular Automata Movement in Eight Directions

Expanding the movement to eight directions introduces additional complexity by enabling diagonal directions as well as the cardinal directions. To achieve this, random numbers are mapped to one of eight directional vectors, with each vector corresponding to a specific movement pattern (e.g., up-right or down-left). This modification required updating the algorithm to ensure consistent boundary handling and uniformity analysis.  
  
Figure: Movement Trajectory in 8 Directions at 1,000 Steps  
  
  
Figure: Movement Trajectory in 8 Directions at 10,000 Steps  
  
  
Figure: Movement Trajectory in 8 Directions at 100,000 Steps

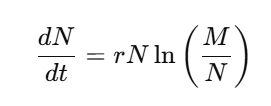
Uniformity was evaluated at multiple checkpoints using both Chi-Squared test. At early checkpoints, such as 1000 steps, the results indicate moderate deviations from uniformity. For example, observed frequencies of diagonal movements may slightly exceed those of cardinal directions, reflecting the added complexity of balancing eight choices. As step counts increase, these discrepancies diminish, and the observed distributions align more closely with expected values.

Alternative approaches for direction selection were explored, such as generating random angles and mapping them to movement vectors. While this method offers smoother transitions and avoids discrete boundaries, it introduces higher computational costs due to the need for trigonometric calculations and angle normalization. For example, using random angle generation resulted in smoother trajectories but required additional computational resources to maintain grid boundaries. Comparing these methods revealed that the original approach is computationally efficient and sufficiently accurate for large step counts, where uniformity naturally improves.

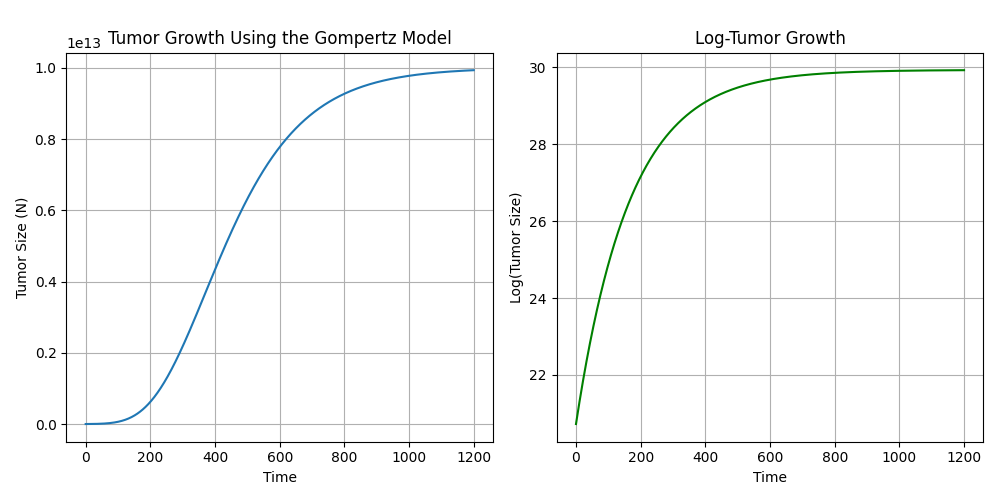
Graphical plots of movement trajectories further illustrate the differences, with the diagonal paths adding more variability to the overall pattern. Despite the increased computational demands, the task remains manageable within the given constraints. Future enhancements could involve introducing weighted probabilities for diagonal versus cardinal directions to simulate environmental biases or testing with larger grid sizes to assess scalability.

## Task 2.1: Tumour Growth Using the Gompertz Model

The Gompertz model was used to simulate tumour growth, governed by the equation:



where N represents tumor size, M is the carrying capacity, and rrr is the growth rate. Simulations were conducted for M=1013, with r=0.006and an initial size N0=109, numerical integration was performed using Euler’s method with a small step size of 0.001 to minimize discretization errors.

The growth curve exhibited a sigmoidal pattern, characterized by an initial exponential phase followed by a plateau as the tumour approached steady-state size. For instance, when, the tumour size grew rapidly during the early phase but stabilized near after approximately 1000 units of time. This behaviour reflects the biological reality of limited resources constraining growth. For a visual representation of these dynamics, refer to the plot below:  
  
Figure: Tumor Growth vs. Log-Tumor Growth (Gompertz Model)

By comparing results across different values of, it became evident that higher carrying capacities allowed the tumour to grow larger before reaching steady state. This variation underscores the sensitivity of the growth dynamics to system parameters.

Error analysis focused on the impact of step size and machine precision. Smaller step sizes reduced discretization errors, allowing the numerical solution to closely match theoretical predictions. However, they also increased computational effort and amplified cumulative rounding errors. For example, with a step size of 0.001, the simulation required significantly more iterations than with a step size of 0.01, leading to longer runtimes. Including error bars in the growth plots provided a visual representation of these uncertainties, enhancing the interpretability of the results.

Additionally, calculating the time required to reach 66% of the carrying capacity offered insights into growth rates, which varied proportionally with and. These simulations highlight how biological parameters interact with numerical methods, emphasizing the trade-offs between accuracy and computational cost. Future simulations could incorporate stochastic variations in or to reflect environmental or genetic variability in tumour growth dynamics.

## Task 2.2: Tumour Growth with Movement Across a Grid

Integrating tumour growth and movement, this task simulated spatial propagation on a 10x10 grid. Starting from the grid centre, the tumour grew to its steady-state size before moving to a randomly selected neighbouring cell. Each movement reset the initial conditions, reflecting localized growth processes. Constraints ensured that the tumour did not revisit previously occupied cells, preserving biological plausibility.

The simulation demonstrated how spatial constraints influenced tumour dynamics. For example, as the tumour moved across the grid, the cumulative time to fill multiple cells highlighted the interplay between growth and movement rates. The time required to achieve steady state in each cell remained consistent due to uniform initial conditions, but the overall time to fill the grid varied based on movement patterns. Random movement introduced variability, with some paths taking longer to explore unvisited cells.

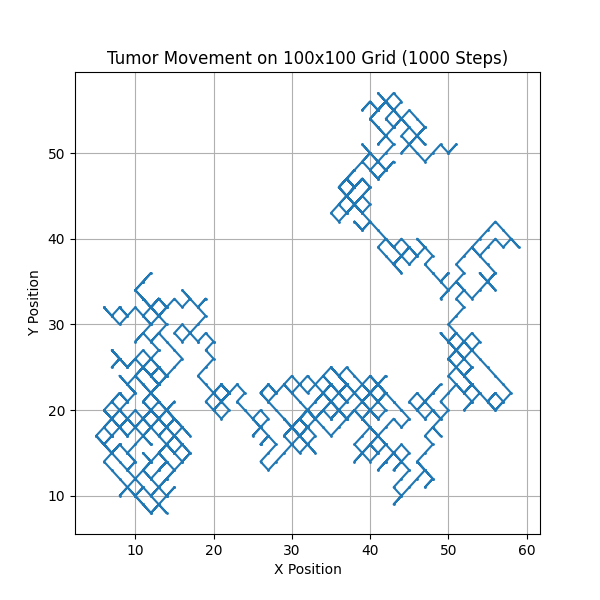
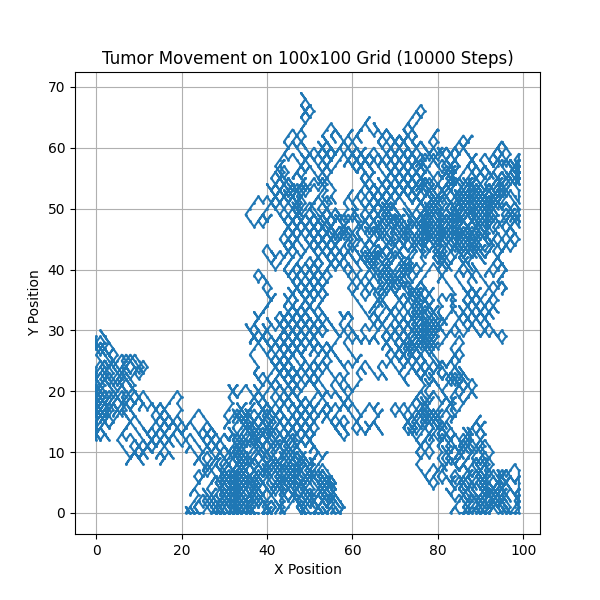
Refer to the visualization below to see an example of the tumour movement across the grid:  


Figure: Tumour Movement Across the Grid   
Figure: Tumour Movement Across the Grid

Computationally, the integration of growth and movement increased complexity. The algorithm required additional checks to ensure valid movement and avoid overlaps. Graphical visualizations showed tumour trajectories, emphasizing how random paths covered the grid over time.

Additional changes could explore adaptive step sizes to optimize computational efficiency. Introducing additional rules, such as resource competition or interaction between neighbouring tumours, could provide deeper insights into the dynamics of cancer spread. Furthermore, larger grid sizes or three-dimensional grids could be tested to simulate more realistic environments.

## Conclusion

The simulations demonstrate the interplay between randomness, growth, and computational complexity. Tasks 1.1 and 1.2 highlight how the law of large numbers ensures convergence towards uniform distributions with increased trials. The Gompertz model effectively captures biological growth dynamics, with results influenced by step size, carrying capacity, and numerical precision.

Task 2.2 integrates growth and movement, showcasing how localized processes interact with spatial dynamics. While the simplicity of the movement algorithm ensures computational feasibility, future improvements could explore adaptive step sizes or stochastic elements to capture finer details of tumour propagation.

The computational complexity across tasks varies, reflecting the trade-offs between accuracy and runtime. For example, increasing step counts improves uniformity in cellular automata but requires more computational resources. Similarly, smaller step sizes in tumour growth enhance precision but introduce higher cumulative errors due to machine precision limitations.

## References

• Chen, S.X. and Liu, J.S., 1997. Statistical applications of the Poisson-binomial and conditional Bernoulli distributions. Statistica Sinica, 7(4), pp.875–892.

• Altham, P.M.E., 1978. Two generalizations of the binomial distribution. Journal of the Royal Statistical Society: Series C (Applied Statistics), 27(2), pp.162–167.

• Skellam, J.G., 1948. A probability distribution derived from the binomial distribution by regarding the probability of success as variable. Journal of the Royal Statistical Society: Series B (Methodological), 10(2), pp.257–261.

• Shiflet, A.B. & Shiflet, G.W., 2014. Introduction to computational science: Modeling and simulation for the sciences. Princeton University Press.

## Appendices

Code is located at: https://github.com/JoeWCassidy/Computational\_ScienceACW