Scientific Computing: Set II

Kasper Nicholas (10678859) & Steven Raaijmakers (10804242)

I. Introduction

Diffusion limited aggregation (DLA) and the Gray-Scott diffusion model are systems that can incorporate the growth of particle clusters and the interaction between multiple chemicals[1]. Physical experiments incorporating the two respective mechanisms can be costly, time consuming and even dangerous. Hence, numerical simulations are a commonly used alternative for assessing the (time) evolution and subsequent characteristics of systems pertaining these models

In this report we focus on two DLA models and one Gray-Scott diffusion model. Firstly, the theoretical framework of all three models is elaborated upon. In the following section we discuss the numerical implementation of the derived discretized schemes. Thereafter, the two variants of the DLA are implemented and compared in order to assess the respective efficiency and features of each method. Subsequently, the Gray-Scott model is assessed for varying initial conditions and parameter settings. Finally, we present and discuss the results of all simulations and subsequent visualisations in order to assess the potential qualitative and quantitative benefits of the three models.

II. Theory

Before the numerical implementation of our experiments is discussed in detail, we briefly touch upon the theory underlying the three variations of simulation methods for diffusion limited aggregation.

A. Diffusion Limited Aggregation

The diffusion limited aggregation is a growth model based on diffusing particles. For our simulations we take the diffusion to adhere the solution of the time-independent diffusion equation. The partial differential equation (PDE) governing the system is

$$\nabla^2 c = 0. \tag{1}$$

We discretize this PDE by means of successive over relaxation (SOR) method due to its advantageous iterative efficiency compared to other diffusion schemes. The SOR method is obtained from the Gauss-Seidel method by an over-correction of the new iterate:

$$\begin{aligned} c_{i,j}^{k+1} = \frac{\omega}{4}(c_{i+1,j}^{k} + c_{i-1,j}^{k+1} + c_{i,j+1}^{k} + c_{i,j-1}^{k+1}) \\ +(1-\omega)c_{i,j}^{k}. \end{aligned} \tag{2}$$

For a more extensive derivation of the discretization process we wish to refer to *Set I* where this is discussed in greater detail. The steady state of the diffusive system is subsequently reached when the following convergence for all values of (i, j) occurs for a predetermined small enough ϵ:

$$\delta \equiv \max_{i,j} |c_{i,j}^{k+1} - c_{i,j}^{k}| < \epsilon. \tag{3}$$

We have now determined the diffusive framework of our model that is required for the DLA. The initial diffusion, in which the simulation is started, is however determined by the solution of the time-dependent diffusion equation:

$$\begin{aligned} c(x,t) = \sum_{i=0}^{\infty} \text{erfc}\left(\frac{1-x+2i}{2\sqrt{Dt}}\right) \\ -\text{erfc}\left(\frac{1+x+2i}{2\sqrt{Dt}}\right). \end{aligned} \tag{4}$$

where erfc(...) is the Gauss error function. The growth is usually started with an initial single seed. Growth candidates around this seed get assigned a growth probability p_g as a function of the concentration of diffusing nutrients at the coinciding growth location. The growth probability p_g for each growth candidate is calculated by

$$p_g(i,j) = \frac{c_{i,j}^{\eta}}{\sum_{\text{growth candidates}} c_{i,j}^{\eta}}. \tag{5}$$

The parameter η determines the shape of the subsequent cluster. A larger numerator and/or a smaller denominator lead to larger probabilities despite potentially low concentration levels of nutrients. Hence our cluster also grows in the direction of lower concentrations. In other words $\eta < 1$ results in a compact cluster, whereas $\eta > 1$ results in a more open cluster. These features are discussed and illustrated in more detail in Section II. After the growth step the diffusive system state is again calculated according to (2). The growth and diffusion are

calculated for a large number of steps, allowing intricate tree-like structures to form.

B. Monte Carlo simulation of DLA

A variation to the growth method of a cluster in a diffusive system incorporates the use of Monte Carlo methods. As opposed to growth neighbours adding to the cluster with a probability p_g, the Monte Carlo simulation of DLA grows a cluster by releasing *random walkers* in the system and letting them move with each step until they hit the cluster. The memory-less property of a Monte Carlo simulations allows for the random walkers to move in a random direction at each step. Furthermore, the parameter η is no longer easily varied. Subsequent clustering is hence more consistent throughout simulations.

However, another parameter can be introduced in order to vary the cluster shape and size, namely the sticking probability p_s. If a particle lands in one of the neighbouring positions of the cluster, it can then become a part of the cluster according to this probability p_s. If the walker does not stick it continues its movement as before, but is not allowed to move into a location in which the cluster is already located.

C. The Gray-Scott model

The Gray-Scott model describes a system of chemical reactions between two chemicals U and V. A diffusive system with both chemicals incorporates interactions between the two substances, and a subsequent decay of V into a product P. The equations are:

$$U + 2 \cdot V \rightarrow 3 \cdot V, \tag{6}$$

$$V \rightarrow P. \tag{7}$$

The substance U is continuously fed into the system. Thereafter it reacts with V according to (6), which is referred to as an auto-catalytic reaction due to the product V enhancing its own production. Simultaneously V decays into P according to (7), independent from U.

In order to numerically assess the Gray-Scott model, we can formulate two partial differential equations that describe the interaction between U, V and P as

$$\frac{\partial u}{\partial t} = D_u \nabla^2 u - uv^2 + f(1-u), \tag{8}$$

$$\frac{\partial v}{\partial t} = D_v \nabla^2 v + uv^2 + (f+k)v. \tag{9}$$

The concentrations are denoted by u and v, with each a diffusion rate D_u and D_v respectively. The parameter f determines the rate at which U is added to the system, and $f + k$ controls the rate at which V decays into P. Varying these parameters leads to a large variety of diffusive systems, some of which end up in a stable pattern, while others remain time-dependent.

III. Methods

In this section the numerical implementation of the three DLA methods is explained. Pseudo-code is added to clarify the complete process and for replication purposes. In the attachments our exact code can be found.

All our methods pertain a two-dimensional cell grid. Most parameter values are further elaborated upon in Section IV for specific simulations due to the varying necessities of figures respectively. In order to decrease computational times the Python package Numba, an open source JIT compiler, is used to speed up our simulations.

A. Diffusion Limited Aggregation

We start our DLA implementation with a single cell above the lower border in a 100x100 grid. The diffusion is then calculated according to (4). As in *Set I* the left and right boundaries are periodic for the diffusion scheme, whereas the top boundary is set to $c(x, y) = 1$ and the bottom boundary is set to zero. Due to the periodicity of the left and right boundary, we can simply extrapolate the one-dimensional diffusion from 4 in to two dimensions by setting each cell in the y-direction with the same concentration. Thereafter, Algorithm 0 is run until the predetermined $max\_cluster\_size$ is reached. Every complete iteration of the algorithm firstly runs the SOR diffusion scheme and subsequently grows the cell cluster according to $p_g(i, j)$ defined in (5). The SOR scheme is calculated for parameter values $D = 1$ and $\epsilon = 10^{-5}$. Due to the locality of the SOR method's functionality only a fraction of the grid has to be updated during each iteration; the majority of cells remains unchanged. It is therefore advantageous to incorporate the solution of the previous SOR iteration.

Algorithm 1 *Simple DLA*

1. Place initial seed (with $c(x, y) = 0$) on 2-dimensional grid
2. Add neighbors of cells with $c(x, y) = 0$ to growth_candidates
3. **for each** growth_candidate:
4. Determine growth probability p_g
5. **if** random number $< p_g$:
6. Add growth_candidate to cluster
7. Set cell concentration $c(x, y) = 0$
8. Repeat steps 2-7 with the new cluster until the cluster reaches a size $max\_cluster\_size$.

It should be noted that each cell that is a part of the cluster acts as a sink in the diffusion scheme; the concentration $c(x, y)$ is set to zero. Growth candidates are simply lattice sites whose north, east, south, and/or west neighbor is part of the cluster.

In order to assess the optimization of the SOR iteration in the DLA method, we calculate the optimal ω with respect to the number of iterations k for regular SOR which incorporates DLA clusters as sink-like objects. For each cluster size we use multiple instances, since the stochasticity in the DLA results in a different cluster shape each run.

B. Monte Carlo simulation of DLA

A second simulation method incorporates the release of random walkers on our grid, and letting them walk until they hit the cluster or wander off the grid. Each step of the random walker is determined by Algorithm 0, in which the neighbors of a cell are determined by the Von Neumann neighborhood for $r = 1$. To simulate with the same boundary conditions as the previous DLA method, traversing the top and bottom boundaries leads to the removal of a random walker. The left and right boundary, however, are again periodic and hence allow the random walker to appear on the other side once it crosses one of the two boundaries.

Algorithm 2 *Random Walker*

```
1. for each neighbor of current position:
2.     if neighbor.x = (−1 ∨ N − 1):
3.         remove_neighbor()
4.     if neighbor.y = (−1 ∨ N − 1):
5.         remove_neighbor()
6.     if neighbor overlaps with cluster:
7.         remove_neighbor()
8. Pick random neighbor from remaining options.
```

Once a random walker has a neighbor that is a part of the cluster, it also becomes part of the cluster according to the sticking probability p_s. Contrary to the growth probability p_g in the previous section, p_s is simply a pre-determined probability that does not depend on the concentration $c(x, y)$. Is is therefore not required to calculate the SOR diffusion scheme for this method. It is important to note that only one random walker is in the system at any time. A new one is generated at a random location on the top border. The complete Monte Carlo simulation is described by Algorithm 0.

As for the initial DLA method we let the Monte Carlo simulations run until the number of cells in the cluster is equal to a predefined maximum cluster size. This allows for a subsequent comparison of the two methods in terms of computational efficiency and cluster features.

Algorithm 3 *Monte Carlo DLA*

```
1. Run SOR diffusion scheme
2. A random walker is randomly placed at the top boundary.
3. while cluster_size < max_cluster_size:
4.     new_position = random_walker(current)
5.     if new_position is off the top or bottom:
6.         create new random walker
7.     else if new_position neighbors the cluster:
8.         if random number < p_s
9.             Add random walker to cluster
```

C. The Gray-Scott model

The third simulation scheme that is implemented is the Gray-Scott model. In Section II we touch upon the underlying theory and reactions between two chemicals U and V. In order to implement the reaction-diffusion equations in (8 and (9) we are required to apply a discretization. If we let k denote the current iteration and $k+1$ denote the next iteration, we arrive to the following equation for the concentrations u and v in coordinate (i, j).

$$u_{i,j}^{k+1} = u_{i,j}^{k} + \frac{\Delta t D_u}{(\Delta x)^2} \cdot (u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,j-1}^{k} - 4u_{i,j}^{k}) - u_{i,j}^{k} \cdot (v_{i,j}^{k})^2 + f(1 - u_{i,j}^{k}) \tag{10}$$

and

$$v_{i,j}^{k+1} = v_{i,j}^{k} + \frac{\delta t D_v}{(\delta x)^2} \cdot (v_{i+1,j}^{k} + v_{i-1,j}^{k} + v_{i,j+1}^{k} + v_{i,j-1}^{k} - 4v_{i,j}^{k}) + u_{i,j}^{k} \cdot (v_{i,j}^{k})^2 + (f + k)v_{i,j}^{k}. \tag{11}$$

The derivation of these equations is similar to the one in *Set I* in which we use Taylor's theorem and the difference quotient. The time step δt and cell size δx are both set to 1. The variable parameters in this model consist of the diffusion coefficients D_u and D_v, and f and k. In our experiments we vary these parameters to explore their effect on U and V.

Contrary to the two previous methods we now consider all boundaries to be periodic rather than just the left and right boundary.

Now that we have discrete representations of U and V we can implement the Gray-Scott model in our two-dimensional grid. We choose to model this on a 200x200 grid, where the initial concentration $u(x, y)$ is set to 0.5

in all cells, and $v(x, y)$ is set to 0 except for square objects which contain a concentration of 0.25.

For the first experiment we put a square of 40x40 containing 0.25 in the centre of V, while we vary the parameter set. After this we explore the effect of noise in the form of arbitrary squares placed on different positions on $v(x, y)$.

IV. Results

The three simulation methods discussed in the previous section are run for varying initial conditions and parameter values. Results are presented in figures in the text, and in files pertaining animations of a growth model.

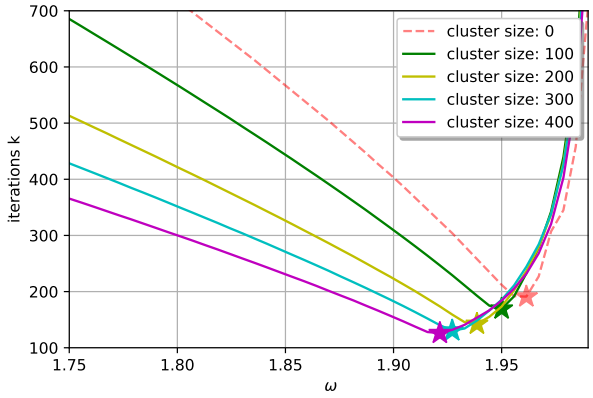

Fig. 1: The optimal ω is calculated for varying cluster sizes. The average number of iterations k is determined by simulating each cluster ten times. The minima of k are indicated with a star.

A. Diffusion Limited Aggregation

For our experiment concerning the optimal ω we used five clusters with sizes ranging from 0 to 400. Each cluster size has ten different instances to account for the variance in shape within a cluster size. In Fig. 1 the omega for the minima of k are indicated with a star. We observe a downward slope of k as ω decreases and the cluster size increases, like we have seen before in *Set I*.

Another aspect of interest in our DLA simulations is the influence of the parameter η on the cluster formation. In Fig. 2 we plot the 2-D grids of DLA simulations for five values of η ranging from 0 to 2. We notice a increase in branching and cluster length as η increases.

B. Monte Carlo simulation of DLA

In the Monte Carlo simulation of DLA the parameter η is fixed to 1. Thus we compare the Monte Carlo simulation against the diffusion equation with $\eta = 1$. The result of both implementations for a cluster of size 300 can be found in Fig. 3. Since both models contain a form of stochasticity we make a comparison between three different instances. This shows us that the DLA model has a more dense and shorter cluster compared to the Monte Carlo simulation. The Monte Carlo clusters in Fig. 3 have longer branches and are more spread out in general.

The influence of p_s on the clustering behaviour of Monte Carlo DLA is reflected in the results presented in Fig. 4. Five different simulations are depicted for p_s between 0.1 and 1. Similar to the influence of η on clusters in simple DLA we notice denser cluster for small p_s and more branching for large p_s.

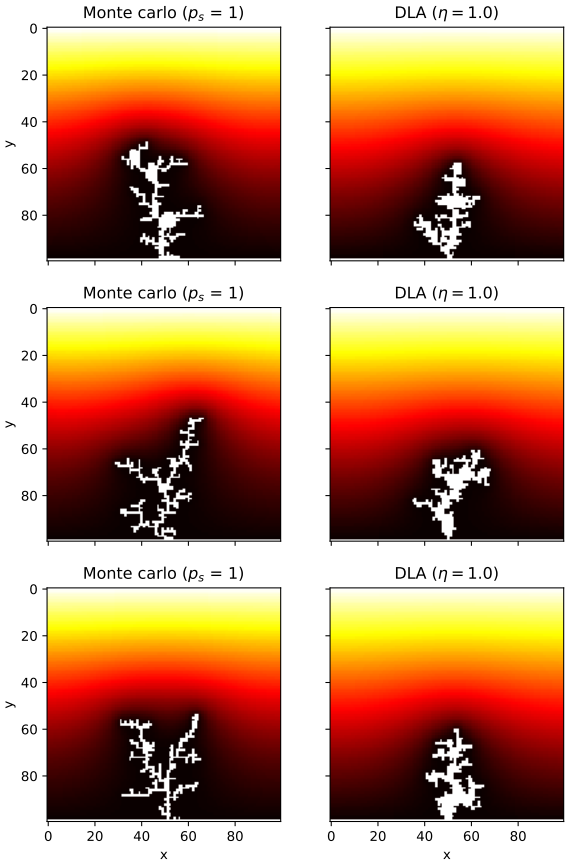

Fig. 3: Simple DLA and Monte Carlo DLA simulations are simulated, with cluster size = 300.

In Fig. 4 we see the influence of various p_s on the cluster shape. We see that for a lower probability the cluster is more dense. This is explained by Algorithm 0

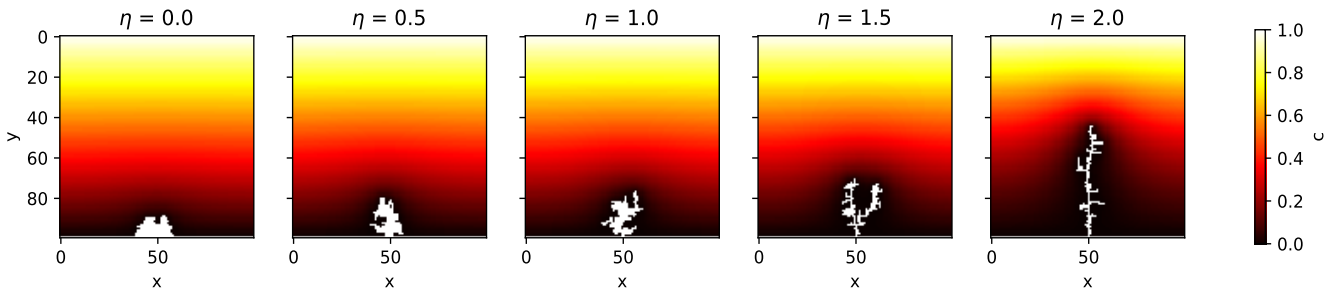

Fig. 2: DLA is simulated for varying η, with cluster size $= 150$.

since the random walker would have a higher probability of stopping at the cluster if it is neighboring multiple cells at once.

C. The Gray-Scott model

The result of different set of parameters for the Gray-Scott model can be found in Fig. 5, with U_t and V_t denoting respectively the concentration of U and V at time t. The figure shows us that a small perturbation in the parameters leads to significant differences in the outcome of the diffusion. For a more clear overview of this behaviour, we have added animations of these plots for 10,000 iterations into the appendix. These animations illustrate the differences in diffusive speed and behaviour throughout the simulations.

In Fig. 6 we illustrate the Gray-Scott model implementation for a different initial V, in which squares of various sizes are drawn on different positions within the grid. The parameters are the same as the top row plots in Fig. 5. We see a similar pattern emerging, but with a slight skewering of the concentration formations, more irregularities and less symmetry.

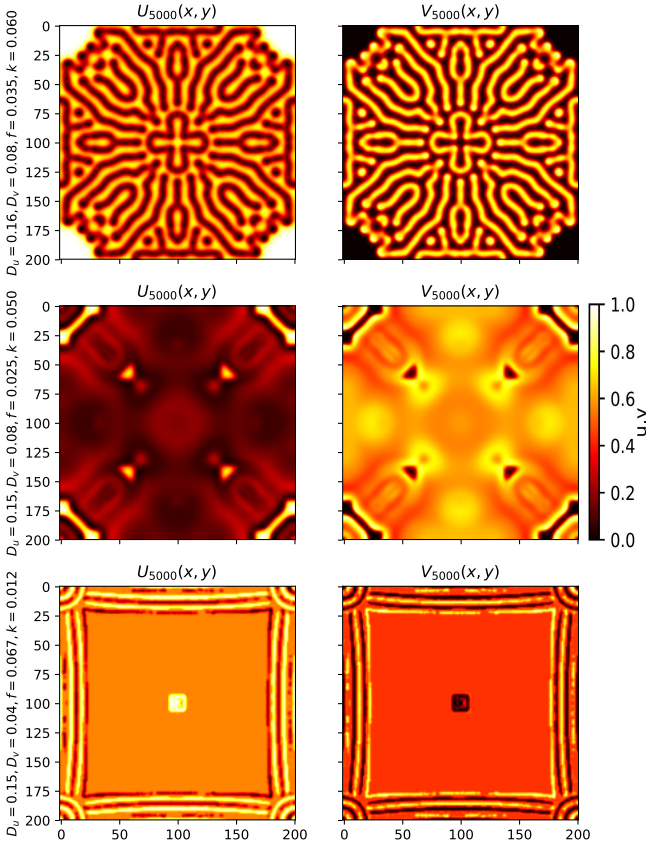

Fig. 5: The concentrations U and V are plotted for the initial parameters listed on the left-hand side of the graphs. All concentrations are plotted after 5,000 steps.

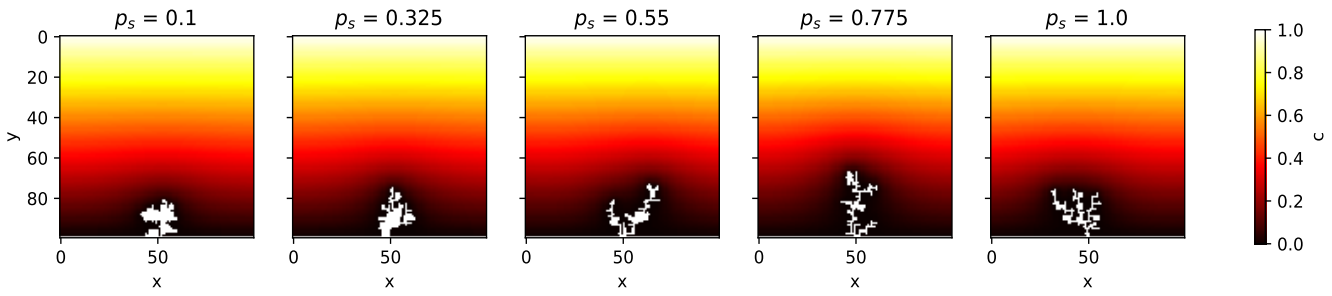

Fig. 4: Monte Carlo DLA is simulated for varying p_s, with cluster size $= 150$.

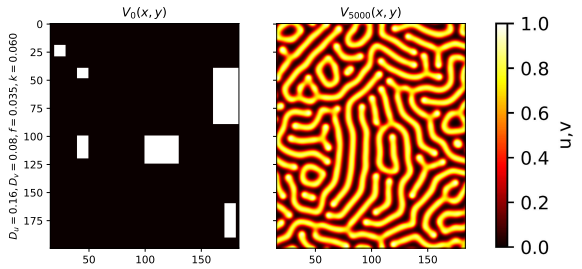

Fig. 6: The parameters of the top graphs in Fig. 5 are used with random initial rectangles of V on the grid. The left plot shows the system at the start of the simulation, the right plot shows the concentration of V after 5,000 steps.

V. Discussion

The results of the simulations of the simple DLA method clearly indicate an optimal ω for different cluster sizes. Moreover, these results are in line with our results and conclusion in Set I. Larger values of η in Fig. 2 lead to longer clusters with more branching, whereas small values lead to dense and short clusters. This is in line with our theoretical definition of p_g in (5). In this equation we observe that smaller values of η increase p_g, allowing for cluster growth in places with small concentrations and vice versa.

The Monte Carlo simulation results presented in Fig. 3 indicate the increased branching characteristics of this method compared to the simple DLA simulations for $p_s = 1$ and $\eta = 1$. This is intuitive due to a $p_s = 1$ directly causing random walkers to stick to the first cell belonging to the cluster. Therefore few random walkers manage to traverse beyond the outer branches/cells and end up prolonging branches rather than thickening them. If we consider the influence of p_s further in Fig. 4 this intuition is indeed correct. For smaller values of p_s we do indeed observe denser clusters. The inverse effect of p_s on Monte Carlo DLA is similar to that of η on simple DLA. These parameters are hence an adequate means to vary cluster density and length in both simulation methods. We do note a significant computational efficiency difference between the two methods and therefore prefer the simple DLA method due to its short computational times. Moreover, the simple DLA method can incorporate the dependency of cluster growth on the diffusive concentration $c(x, y)$, whereas our Monte Carlo DLA method fails to do so.

The Gray-Scott model results are more intricate and not as straightforward to interpret with respect to the parameter values and initial conditions. From Fig. **??** we can infer that a symmetric placement of the concentration V combined with a uniform concentration U throughout the grid lead to symmetric yet complex diffusive and reactionary patterns. This is further illustrated in Fig. 5 but for varying parameter values. The different parameter values also lead to varying stability of the patterns. When the initial concentration V is added as noise in a random manner as in Fig. 6 the patterns lose their symmetry and regularity. The concentration distribution, however, with and without noise for the same parameter values is relatively similar. We wish to emphasize that our evaluation of the Gray-Scott is limited. More extensive research should be done upon the influence of parameters and initial conditions to determine the exact features of subsequent diffusion schemes. Our results do, however, indicate the complexity of Gray-Scott systems despite the simplicity of its theoretical framework. This concludes the discussion and comparison of the two DLA methods and the Gray-Scott diffusion scheme.

References

[1] M Heath. *Computing: An introductory survey*. McGraw-Hill, 1998.