

A cluster miscellany

This appendix is aimed at a general audience. Topics include the cognitive aspects of cluster structure, the role of the computer in cluster analysis, general aims and ideas, a short history of clustering and classification, and some remarks on the usage and etymology of words in mathematics, in particular those words often occurring in this thesis. The appendix concludes with a short account of the main contribution of this thesis, the Markov Cluster algorithm.

1 Of clusters and cognition

A cluster is a) *a close group or bunch of similar things growing together* or b) *a close group or swarm of people, animals, faint stars, gems, et cetera*, according to the Concise Oxford Dictionary¹. Examples of usage are ‘a cluster of berries’, ‘galaxy cluster’, ‘cluster of molecules’, and ‘cluster of computers’. The usage of the word *cluster* in the mathematical discipline *Cluster Analysis* is really the same, except that only the generic parts of the definition remain. Thus, plants, animals, people, computers, molecules, and galaxies are submerged in the sea of things, and as consolation ‘things’ is changed to the classier ‘entities’. The meaning of cluster in a mathematical setting then becomes *a close group of entities*. Clearly, a ‘close group’ indicates a remarkable degree of fellowship that is in contrast with the surrounding parts. Thus, if a garden has seven yew trees (*Taxus Baccata*) which are dispersed homogeneously across the garden, they cannot be regarded as forming a cluster together. A point of interest is that in mathematics it is perfectly acceptable for a cluster to consist of a single element, as this makes reasoning about clusters a lot easier. So, in this case, the yews are better viewed as seven separate clusters, just considering the garden they are in. However, if all neighbouring gardens have exactly one yew, then on a larger scale the seven yews may be seen as forming a single cluster. Furthermore, if a great yew baron has a plantation farming thousands of yew trees, then a group of seven yews picked out in the middle is hardly a cluster. On a larger scale again, all of the thousands of yews do form a giant cluster in the surrounding landscape of meadows and pastures.

More complex arrangements can be pictured, as in a garden with twelve groups (Figure 30), each consisting of three yews, where at each different point of the compass three groups of yews are planted together. This results in different clusters on different scales, one where twelve groups of three yews are distinguished, and, on a larger scale, one where four groups of nine yews are distinguished. The picture becomes blurred if a few extra yews are scattered around the (neglected) garden, some of them standing close to the neatly arranged groups (Figure 31). At a certain degree of closeness the mind and

¹Ninth Edition.

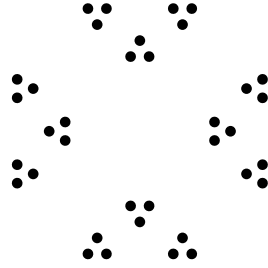


Figure 30. Garden with yews.

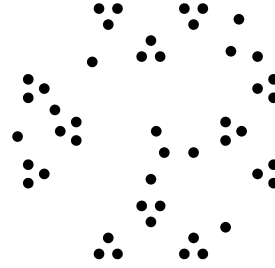


Figure 31. Neglected garden.

eye will tend to see a group of yews swallowing up yews standing closely near, but there is no rule predicting exactly when this will happen. The process is influenced by the relative size and location of other groups. In general, the way in which local cluster structure is perceived is affected by the overall perceived structure; a number of yews which the eye tends to see as a cluster in one configuration may be seen as less related in another configuration. On the other hand, the perception of cluster structure is also influenced by the relative sizes of the different yews, i.e. by highly local parameters. Furthermore, the eye has a tendency to group things together in such a way as to produce balanced groups, and the degrees of regularity and symmetry of an arrangement also plays a role. In the picture on the left of Figure 32 the force of regularity tends to prevail, i.e. enforcing the perception of two clusters of three elements and four clusters of a single element. On the right the scales tend to favour a balanced grouping, with two groups of three elements and one of four. Even in these simple examples it is seen that perceiving cluster structure is a cognitive event influenced by many parameters. The concept of cluster is inherently vague, and much to the chagrin of mathematicians, the situation is more or the less the same in mathematics.



Figure 32. Regularity and balancedness competing.

2 Aims and ends I

Cluster analysis can be described as the study of characterization and recognition of close groups in a class of entities. The study of recognition means the study of methods that label entities as belonging to the same group or to different groups. A *clustering* is such a labelling or division of the entities into groups, and a *method* is a recipe which, if

followed, produces a clustering if one is given a class of entities and a notion of similarity (closeness) or distance between those entities. Usually, recipes are called *algorithms*. With this terminology settled, let us try and see if it is useful, and why the problems in cluster analysis are interesting and difficult.

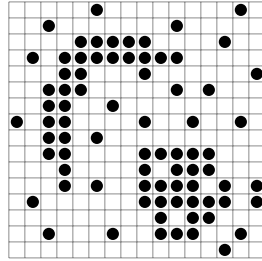


Figure 33. Image A.

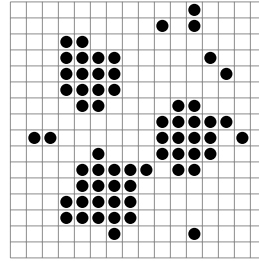


Figure 34. Image B.

What is the use of finding clusterings? One use is as a means of creating a *classification* of the entities by partitioning them into groups. The classification then corresponds with 'the big picture'; it means that structure is attached to some source of information or observations on entities. This is perhaps best explained by looking at what it takes to equip a machine with a simple form of vision. Images *A* and *B* in Figure 33 represent camera generated pictures which are sent to a computer, that has to take some action (follow some recipe) depending on how many objects are present in a picture. For interpreting a picture, it needs another recipe, because in the picture itself there is no information about objects, there is only a grid of boxes (called pixels) which may be black or white. Thus, the entities in this case are black pixels, and a close group of black pixels represents some object. Clustering amounts to recognizing higher level entities (shapes or objects) from lower level entities (black pixels).

Animals, c.q. hominides, are very good at extracting patterns from this kind of image. In fact, *they see nothing but structure*, so it is hard to recognize the difficulty of the task. First, it should be stressed that the task is not to find a way of analysing a particular picture, but to find a method that can be used for analysing an enormous range of possible pictures. Second, the images look deceptively simple to us because of our cognitive skills. What if one is asked to recognize 'objects' in the following array of pairs of numbers?

(0 7)	(2 14)	(3 11)	(5 11)	(8 4)	(9 12)	(10 12)	(12 5)	(14 0)
(1 3)	(3 3)	(4 2)	(6 2)	(8 7)	(9 13)	(10 14)	(12 9)	(14 7)
(1 12)	(3 4)	(4 3)	(6 3)	(8 9)	(9 14)	(11 7)	(12 10)	(14 14)
(2 1)	(3 5)	(4 4)	(6 6)	(8 10)	(10 1)	(11 9)	(12 12)	(15 4)
(2 5)	(3 6)	(4 5)	(6 14)	(8 11)	(10 3)	(11 10)	(12 13)	(15 11)
(2 6)	(3 7)	(5 0)	(7 2)	(8 12)	(10 5)	(11 11)	(13 2)	(15 12)
(2 7)	(3 8)	(5 2)	(7 3)	(9 3)	(10 9)	(11 12)	(13 11)	
(2 8)	(3 9)	(5 3)	(8 2)	(9 9)	(10 10)	(11 13)	(13 12)	
(2 9)	(3 10)	(5 8)	(8 3)	(9 11)	(10 11)	(11 14)	(13 15)	

This mess of numbers contains exactly the same information² as image *A*, and is in fact essentially the only way that an image can be stored in a computer. In designing a recipe one is forced to create a list of instructions that can be applied to arbitrary lists of numbers.

Suppose it is known in advance that the image may contain any number of objects between 0 and 5, and that an object is never hiding part of another object, but that objects might be located close to each other. That is at least something; this knowledge can be used and incorporated into the recipe that the computer is going to follow in deciding in how many objects there are. Now, it is not hard at all to design a recipe which works for the picture in Figure 33. But that is not what is required: a recipe is needed which works well for all different kinds of pictures that can be sent to the computer. There may be *noise* present in the picture, like the many loose black pixels in image *A*, the objects may have different sizes, shapes can be long and stretched or compact, bent or straight, and one shape can possess all of these characteristics simultaneously. Two shapes close together can be hard to distinguish from one long bent shape and noise may further cloud the issue.

The previous examples illustrate some of the typical challenges in cluster analysis. Devising a good clustering method for this kind of problem is at least as difficult as answering how the eye and mind perceive cluster structure. In cases where the complexity of a problem can be visualized, people, scientists included, expect the results of cluster methods to match their own interpretation. Now on the one hand the perception of cluster structure is influenced by changes in context, but on the other hand the perception of cluster structure may equally well lead to a perceived change in context. Perception of cluster structure has to do with interaction of low-level and high-level cognitive functions, and such interaction is very difficult to catch in a recipe.

The general rule of method design applies that the more restricted the problem area is (i.e. the less uncertainty there is about possible contexts), the easier it is to design methods excelling in this area. Different methods that are respectively optimized for recognizing different kinds of images such as in Figure 31 and Figure 33 will perform not as well on other kinds of images, and methods that are widely applicable are unlikely to excel everywhere. The best thing possible is to have a method that can be easily tuned to different contexts.

3 Aims and ends II

A second use of finding clusters lies in cutting apart a structure such that it stays intact as much as possible. That sounds funny, so consider Figure 35. Twelve cities are pictured as little circles, labelled 1, ... 12, and a number of roads connects the cities. Suppose that the cities should be grouped into different regions, where each region gets its own road maintenance department. One region would be inefficient (for reasons unknown to us), and if there are more regions then the number of roads between regions should

²e.g. the (7, 2) pair says that starting from the upper left pixel, a black pixel is found when going 7 pixels to the right and 2 to below.

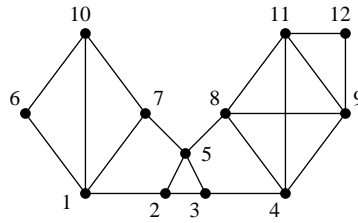


Figure 35. Cities and roads.

be minimal (in order to minimize the chance of colliding maintenance teams). A good candidate for such a grouping or clustering is $\{1, 6, 7, 10\}$, $\{2, 3, 5\}$, $\{4, 8, 9, 11, 12\}$. The main contribution of this thesis is a powerful new method, *the Markov Cluster Algorithm*, that is specifically suited to attacking this kind of problem — where the number of dots may be orders of magnitudes larger.

There are many real-life domains which can be modelled as a structure like the one in Figure 35, that is, as a combination of dots and lines. These structures are called *graphs* and they play a role whenever there is a notion of objects (or entities) being connected or not. Entities are then called *nodes*, and lines are called *edges* or *links*. The most appealing and largest in scale examples are the Internet and the World Wide Web. The Internet consists of millions of computers which are linked via intricate networks. The World Wide Web consists of an ever growing amount of web pages³, which refer to each other in myriads of ways via the so called hyperlinks. Diving into the computer itself, circuits in chips consist of transistors (nodes), wired together (links). The list of examples is virtually without limit. Take people as entities, and say there is a connection between two people if they ever shook hands. This is a graph where the dots are people and a line corresponds with (at least) one handshake. It is a theory or legend of some kind that there are on average only six handshakes between two arbitrary different people on earth⁴. Note that in this example the link does not correspond with a connection that also resembles a physical distance, contrary to the cities and roads example from Figure 35. However, this figure is an abstract picture; it could equally well represent twelve people, the lines telling which pairs of people ever shook hands.

Many different kinds of connections between people can be considered; e.g. being relatives of the first (or second, third, ...) degree; both having a parent born in the same town; having worked for the same company. In the scientific community, one may define being connected as having co-authored an article. Many mathematicians know their Erdős number, which is the distance from the famous and prolific mathematician Paul Erdős⁵. If mathematician P has co-authored an article with Q , where Q has co-authored an article with Erdős, then Q has Erdős number 1, and P has Erdős number 2.

³Currently probably within the billions.

⁴This may seem improbable, but giving it some thought should at least refute the spontaneous disbelief. The origin of this conjecture is unknown to me.

⁵Paul Erdős wrote more than 1500 articles in his career, many of which were jointly written with other mathematicians.

Another graph constructed from science takes as nodes, say, all mathematical articles ever written; two articles are connected in this graph if the subject lists describing their content share one or more common elements. Such a graph is in a sense a map of the whole of mathematics, and it is interesting to investigate properties of this map. One idea is that articles may be related to each other without sharing any element of their subject lists, perhaps because the same subject is known under different wordings. It is then natural to assume that the articles must still be close to each other on the map previously introduced.

Finally it should be noted that the examples in this chapter were chosen because they have some visual appeal and a low degree of complexity. In the vector model, for slightly more complex input data, the problems can no longer be visualized such that an observer or practitioner can test methods against her own intuition. This applies even more to the graph model; graphs with dozens of nodes and hundreds of links already yield a picture of an inextricably entangled web.

4 Perspectives

Cluster analysis, aiming at dissecting structures such that they stay intact as much as possible, yields a relatively new perspective, inspired by the increasing number of phenomena that can be modelled as graphs (structures with nodes and links), such as computer networks and computer chips, databases, document collections, c.q. web pages, et cetera. In this thesis I argue that there is a subtle but significant difference with the classic applications of cluster analysis, which are better described by the notion of *vector models* rather than graphs. This notion will be introduced in the course of a short account of cluster analysis history.

The early origins of cluster analysis are found in the classification of populations in species⁶ in biology, a discipline which is commonly known as *taxonomy*. Aristotle was already herein engaged (writing the book *Scala Naturae*, i.e. scale of nature), and Carl Linnaeus is its most famous contributor. In taxonomy, the entities are different populations of animals, and observations on how different populations differ in their characteristics establish a notion of similarity (or conversely, distance) between them. The characteristics chosen by current taxonomists vary among others from morphological attributes (e.g. skeleton or bone related measures like type, curvature, weight, measures on fur, feather, teeth, digestive system and so on) to ecological and geographical data describing the habitat of populations. Essentially, populations are described by their number scores on the chosen characteristics, and two populations are 'close' if their respective scores are close. This picture is far from complete, as taxonomists take other factors into account, such as the ability between populations to produce fertile offspring, the extent of DNA hybridization, and the number and degree of pairing between chromosomes (quoted from [95], page 133). Sticking to the simple model, a list of numbers (here characterizing a population) is in mathematics called a *vector*, hence the phrase vector model.

⁶And, following species, genera, families, classes, orders, phyla.

In the 20th century, taxonomists began to seek objective, unbiased methods using such numerical characteristics of individuals and populations (see [155], page 13). This research is collectively labelled *numerical taxonomy*. The division between method and application, i.e. formulation of methods in such abstract terms that they can be applied to behavioural, biological, or any other kind of data, eventually lead to the recognition of *cluster analysis* as a research area of its own, where generic clustering and classification methods are studied not tied to any particular context. Taxonomists did not get rid of the issue of objectivity however, because it turns out that different methods and different ways of preparing the data yield different results, and that it is impossible to establish that one method is better than another, except in very specific cases.

Following taxonomy, subsequent applications of cluster analysis are still best described by the vector model. These include the grouping of medical records in order to find patterns in symptoms and diseases related to characteristics of patients, and the analysis of behavioural and sociological data for similar purposes. In the first case the data for each entity (a medical record or patient) is a set of scores on symptoms, body characteristics, or a combination of both, in the second case the entities are either people or populations (e.g. cities), and the scores can pertain to economic status, education, crime, health, or any other sociological phenomenon of interest. Everitt lists several of such applications ([54], page 8).

The difference between vector and graph settings is one of genuinely different types of topology. In both settings there is a notion of distance or similarity between entities, but they are conceived in different ways. In the vector model, the distance between two entities is derived by comparing a set of scores, and calculating a number which represents the distance between the two entities. The vector model can be applied to the entities in images *A* and *B*, which are the black pixels. Their 'scores' are just their coordinates, and the distance between the black pixels (7, 3) and (2, 1) in image *A* is then for example calculated as the horizontal distance plus the vertical distance, amounting to $5 + 2 = 7$.

In the graph model, there are the two notions of a) being connected or not and b) longer distance paths going from one entity to another, notions lacking in the vector model. The Markov Cluster algorithm was inspired by the implications of the *path* notion for properties of clusters in the settings of graphs. It hardly could have been conceived in the classic setting of vector models, and experiments indicate that it is very powerful particularly in the setting of graphs. Still, the vector model and the graph model have so many resemblances that it is easy to try and apply the Markov Cluster algorithm to vector models. This honours a good engineering and scientific principle that theories, methods, designs, and even machines should always be tried to the limit of what is possible. By doing so, the practitioner learns about the strengths and weaknesses of that what is tested, and it may well lead to new insights. This is also the case for the Markov Cluster algorithm. It is applied to (highly abstract variants of) images such as in Figures 33 and 34. For some cases it works well, and for others it does not, and it can be explained and predicted for which classes this is the case. This leads to a shift of perspective as it is shown that the *MCL* process can be put to a different use, namely that of border detection in images. These issues are discussed in Chapter 10.

5 Words of history and history of words

Since this thesis is all about a recipe called the *Markov Cluster Algorithm*, some explanatory words may be of interest.

1. *Algorithm* has the meaning of ‘recipe’, a set of instructions for the purpose of achieving some goal. The man who fathered this word did not live to know that he did. Ja’far Mohammed Ben Musa lived at the court of the caliphs of Bagdad, and was also known as al-Khowarazmi (also often transliterated as al-Chwarizmi), meaning ‘the man from Khwarazm’. Around the year 825 he wrote an arithmetic book explaining how to use (i.e. giving methods or recipes) the Hindu-Arabic numerals⁷. This book was later translated with the Latin title *Liber Algorismi*, meaning ‘Book of al-Khowarazmi’. Schwartzman writes in [148]: “The current form *algorithm* exhibits what the *Oxford English Dictionary* calls a ‘pseudo-etymological perversion’: it got confused with the word *arithmetic* (which was one of its meanings, and which has several letters in common with it); the result was the current *algorithm*.”

An algorithm is something which can be programmed on a computer, that is, the computer can carry out the instructions on the recipe. The most important part of cluster analysis is cooking up good recipes, and understanding why certain ingredients and procedures work well together, and why others fail to do so. It should be noted that all the hard work is done outside of the computer; humans have to supply the recipes, the ingredients, and the cooking equipment. The computer is just a wired together piece of junk⁸ that does exactly what the recipe tells it to do, using the ingredients and equipment that come with the recipe.

2. By pulling himself up by his bootstraps *Baron von Münchhausen* fathered the word *bootstrapping*. In science its abstract meaning is to derive high-level structural descriptions from low-level data without using (a lot of) a priori knowledge. The origin suggests that bootstrapping problems require some miraculous feat. However, the dull truth is that no method solves bootstrapping problems entirely satisfactory, which is exactly what makes them so interesting. The phrase can be used in sentences like *The basic problem in cluster analysis is that of bootstrapping*, or *Building a sophisticated software environment requires a lot of bootstrapping*, and perhaps *Life is the mother of all bootstraps*.

3. *Botryology* is an obscure term which was meant as a dignified label for the discipline of cluster analysis, meaning ‘the theory of clusters’. It is formed from the Greek *βοτρυον*, meaning ‘a cluster of grapes’. In the article ‘The Botryology of Botryology’ the author I.J. Good puts in an heroic effort to lift the term into mainstream usage, though his argumentation seems somewhat humorous ([68], page 73):

⁷Source: [148], page 21.

⁸Of course, computer manufacturers put a lot of hard work in wiring the junk such that it can carry out very large recipes very swiftly.

It seems to me that the subject of clustering is now wide enough and respectable enough to deserve a name like those of other disciplines, and the existence of such a name enables one to form adjectives and so on. For example, one can use expressions such as “a botryological analysis” or “a well-known botryologist said so and so”. There is another word that serves much the same purpose, namely “taxonomy”, but this usually refers to biological applications whereas “botryology” is intended to refer to the entire field, provided that mathematical methods are used. The subject is so large that it might not be long before there are professors and departments of botryology. Another possible name would be *aciniformics*, but it sounds inelegant. On the other hand, “agminatics” is a good contender, forming “agminaticist”, etc.

It is an example of scientific word usage that never quite made it. Occasionally, the term still surfaces though, as in the title of [163]. In my mind I.J. Good will always be well-known as a botryologist, and as one who profoundly appreciates the aesthetic aspects of word usage. He gives several references to earlier uses of the word. The earliest reference is an article written by himself, which may indicate that he fathered the word. It is left as an exercise for the reader to find the etymology of the constructions *aciniformics* and *agminatics*.

Perhaps it is a witness to the fragmented origin of cluster analysis, i.e. its simultaneous growth in different disciplines, that a wealth of labels has been associated with it. Hartigan gives the following list in [79], page 1: *numerical taxonomy, taximetrics, taxonotics, morphometrics, botryology, nosology, nosography, and systematics*.

4. *Cluster* is related to the Low German *kluster*, and the Old English *clott*, meaning lump or mass, courtesy of Webster’s dictionary. Webster’s concludes its etymological summary of cluster with ‘more at CLOT’. Looking up ‘clot’ yields the related Middle High German *kloz*, meaning lumpy mass or ball, and the indirection ‘more at CLOUT’. Then ‘clout’ yields among others Russian *gluda* or lump, Latin *galla* (gall-nut), and the reference ‘more at GALL’. The word *gall* is ‘perhaps akin to Greek *ganglion* cystic tumor, mass of nerve tissue, [and] Sanskrit *glau* round lump; basic meaning: ball, rounded object’. The intricate ways of language and dictionaries! This gives yet another example of graph structure in daily life; the nodes or entities are dictionary entries and the links are the cross-references between them. Linguists study properties of this type of structure.

5. *Andrei Andreyevich Markov* was a famous Russian mathematician, born 14 June 1856 in Ryazan, died 20 July 1922 in St Petersburg. He is best known for his contributions to probability theory. His name is connected to many mathematical notions, among them *Markov chain, Markov matrix, Markov moment, and Markov process*. It should be noted that such attributions are made by other mathematicians, usually after the principal contributor’s work has gained widespread acceptance. A Markov matrix is a special kind of matrix (see below) which has the property that it is nonnegative, i.e. all its elements are greater than or equal to zero, and that all its columns (or rows, depending on the chosen orientation) sum to 1. This thesis rests mainly on two pillars; Markov theory, and the theory of nonnegative matrices, for which Markov theory formed a thriving source of inspiration.

In the mathematical landscape, this thesis is situated somewhere near the disciplines of *the study of nonnegative matrices* and *matrix analysis*. Many concepts are named after people who made profound contributions, concepts such as Geršgorin discs, Hermitian matrices, the Hadamard-Schur product, the Jordan canonical form, the Kronecker product, Perron-Frobenius theory, and the Schur product theorem.

6. A *matrix* is the mathematical object which lies at the heart of the Markov Cluster algorithm, in particular the matrix subspecies *Markov matrix*. A matrix is a rectangular array of entities that are usually just numbers or indeterminates. Examples of matrices can be found on pages 50, 53, and 66. The word matrix has a rather impressive heritage and several meanings. Most of them refer to something in which the evolution of new life or substance is embedded; the word is etymologically related to *mater* or mother. Examples of this are the meanings *uterus* or *womb*, *cradle*, and *mould* (note: the Dutch word for mould is *matrjjs*, which is etymologically very close to matrix). Webster's dictionary⁹ gives the example 'an atmosphere of understanding and friendliness that is the matrix of peace'. This generic meaning must be certainly what inspired the makers of the 1999 Warner Bros science-fiction action movie *The Matrix* in titling their creation, in which robots running amok have subjected mankind and cast nearly all humans into artificial wombs. That is not even the worst part, as the humans are wired into a computer, which causes them to believe that they are leading a normal life. Thus, bodies are grown, supported, and constrained by a matrix in the form of artificial wombs, and the mind is similarly treated by a computer-created virtual matrix. In mathematics however, matrices are very likable beasts, which are used in many different disciplines to great avail.

Schwartzman ([148], page 132) names two possible entry points for the word *matrix* in mathematics. However, Jeff Miller gives a much clearer explanation¹⁰ and even cites the person who actually introduced the word, James Joseph Sylvester (1814-1897). Schwartzman's first remark is that mathematically speaking a matrix *generates* geometric or algebraic transformations. The second is that matrices are arrays of numbers *surrounded* by large brackets or parentheses. In both cases the meaning of the verb refers to womb or matrix-like qualities, and Schwartzman suggests that these congruences lead to the use of the word matrix. Sylvester himself has the following to say ([15], page 150):

This homaloidal law has not been stated in the above commentary in its form of greatest generality. For this purpose we must commence, not with a square, but with an oblong arrangement of terms consisting, suppose, of m lines and n columns. This will not in itself represent a determinant, but is, as it were, a Matrix out of which we may form various systems of determinants by fixing upon a number p , and selecting at will p lines and p columns, the squares corresponding to which may be termed determinants of order p .

Miller gives the following citation of Kline, found in [107], page 804:

⁹Webster's Third New International Dictionary, 1971.

¹⁰Source: <http://members.aol.com/~jeff570/mathword.html>.

The word matrix was first used by Sylvester when in fact he wished to refer to a rectangular array of numbers and could not use the word determinant, though he was at that time concerned only with the determinants that could be formed from the elements of the rectangular array.

It is clearly the determinant-generating ability that inspired Sylvester.

7. *Walk, random.* A random walk is a walk that is governed by the flipping of a coin, the rolling of a dice, or more generally by any event for which the outcome is a priori uncertain. Suppose you are walking in a city, and each time you arrive at a crossing you flip two coins, one after another. There are four possible outcomes, writing H for heads and T for tails. and you decide to: turn left if the outcome is TH (the first coin is T , the second is H), go straight on if the outcome is TT , turn right if it is HT , and turn around if the outcome is HH . This is a good example of a random walk. One important aspect is that if you start two or more times from the same departure point, then the resulting walks will in general be different; a random walk cannot be predicted. Still, a lot of things can be said about the *probability* that certain things happen. For example, one may ask what the chances are that you return to the point from which you departed after visiting a hundred crossings, or what the chances are that you have returned at least one time. What are the odds that you visit only different crossings, or that you visit no more than fifty different crossings? If ten thousand people start a random walk from the same point, how far will they be away from the departure point after a hundred steps, on average? One random walk is not predictable, but if you combine very many of them, then it is often possible to make surprisingly strong statements about them. The concept of a random walk is a very rich source for scientific research, because many questions about them (such as posed here) can be answered using mathematical tools such as Markov matrices, and because many real-world phenomena are well described by the concept of a random walk.

The Dutch physicist and publicist Ad Lagendijk dedicated an entire column to the random walk in *De Volksrant* d.d. Saturday 11 December 1999. The column is titled¹¹ *Walk of the century*. Lagendijk lists several phenomena which can be described using random walks: the transport of fluids through porous media (e.g. oil through rock), the diffusion of molecules in gas mixtures or chemical solutions, the way in which people navigate supermarkets and large stores, and the transport of molecules through the cells of organisms. He argues that the concept of a random walk deserves to be called the biggest scientific discovery of the 20th century, because of the power and the fundamental nature of the concept, its wide applicability, its common use in many different scientific disciplines, and because it has become part of scientific mainstream to the extent that scientists use it even unconsciously.

The *MCL* process utilizes random walks for the retrieval of cluster structure in graphs. It is described in some more detail in the next section. Perhaps confusingly, the word *flow* is also used throughout this thesis to describe the working of the *MCL* process. Obviously one associates flow with a good swim rather than a random walk, at least for ordinary mortals. The right perspective is found by pairing the concept of flow with a

¹¹The original title is *Wandeling van de eeuw*.

vast collection of random walks. Picture ten thousand people each starting their own random walk from the same departure point. An observer floating high above them will see the crowd slowly swirling and dispersing, much as if a drop of ink is spilled into a water-filled tray.

8. The Markov Cluster Method is *Yet Another* Cluster Method, in the sense 1 as listed below. The Jargon File 4.0 has this to say¹² about the qualifier *yet another*, which is an example of popular language from the world of computers and computer science.

Yet Another: /adj./ [From Unix's 'yacc(1)'. 'Yet Another Compiler-Compiler', a LALR parser generator] 1. Of your own work: A humorous allusion often used in titles to acknowledge that the topic is not original, though the content is. As in 'Yet Another AI Group' or 'Yet Another Simulated Annealing Algorithm'. 2. Of others' work: Describes something of which there are already far too many.

6 The Markov Cluster algorithm

The main contributions in this thesis are centred around a new method in cluster analysis, which I named the *Markov Cluster algorithm*, abbreviated as *MCL* algorithm. As stated before, the algorithm is specifically suited to graph structures such as in Figure 35. The *MCL* algorithm is inspired by a simple yet powerful idea. The aim of a cluster method is to dissect a graph into regions with many edges inside, and with only few edges between regions. A different way of putting this is that if such regions exist, then if inside one of them, it is relatively difficult to get out, because there are many links within the region itself, and only a few going out. The idea is now to simulate random walks or flow within the whole structure, and to further strengthen flow where the current is already strong, and to weaken flow where the current is weak. In different wordings, random walks are promoted, e.g. by broadening the pavement, where the number of pedestrians (i.e. current) is already high, and random walks are demoted where this number is low, e.g. by narrowing the pavement. The hypothesis supporting this idea is that cluster structure corresponds with regions of strong current (many random walks pass through), separated by borders where the current is weak (relatively few random walks pass through). If this idea is put to the test with the right tools, it turns out to work. Flow can be manipulated in this way such that it eventually stabilizes, where most of the flow weakens to such a large extent that it actually dries up. Different regions of constant flow remain which are separated by dry borders; these regions can be sensibly interpreted as clusterings. A symbolic picture representing such a situation for the graph in Figure 35 is seen in Figure 36, and a sequence of pictures representing different stages of flow is found on page 7. In these pictures the grey level of a node indicates how many random walks pass through at a given stage: the darker the node, the more walks.

¹²An interesting web resource edited by famous Open Source advocate Eric S. Raymond, found at <http://www.tuxedo.org/~esr/jargon/>. Its sibling the *Free On-line Dictionary Of Computing*, <http://foldoc.doc.ic.ac.uk/>, edited by Denis Howe, is also noteworthy.

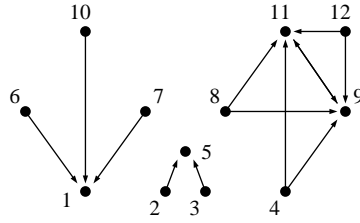


Figure 36. Regions of flow.

Markov theory supplies almost everything needed to manipulate flow as described above. Flow simulation can be done by taking a suitable Markov matrix, and computing powers of this matrix. This is the well-known concept of a discrete Markov chain. The only thing lacking is the strengthening and weakening of flow. That part is supplied by inserting a new operator in the Markov chain, which can be described in terms of the so called *Hadamard-Schur* product. In effect the *MCL* algorithm draws upon two well-developed disciplines of mathematics, and this crossbreeding yields fertile offspring.

The beauty of the *MCL* algorithm is that the method is not actively engaged in finding clusters. Instead, it simulates a process which is inherently affected by any cluster structure present. The cluster structure leaves its marks on the process, and carrying the process through eventually shows the full cluster structure. The process parameters can be varied, i.e. the flow can be strengthened and weakened to a greater or lesser extent. This parametrization affects the scale on which the cluster structure leaves its marks. It is shown in Chapter 7 that the process (in particular, the matrices created in it) has mathematical properties which have a straightforward interpretation in terms of cluster structure. These results are of particular interest, as it is for the first time that cluster structure is found via and within a simple algebraic process.

The *MCL* algorithm generates many new research questions. In this thesis a few of them are answered, and these answers help in gaining insight in the algebraic process employed by the algorithm. Mathematics is, contrary to common belief, an ever changing field of research where results lead to new questions and questions lead to new results.

