Computational approaches and software tools for genetic linkage map estimation in plants

Jitender Cheema and Jo Dicks
Submitted: 13th July 2009; Received (in revised form): 4th September 2009

Abstract
Genetic maps are an important component within the plant biologist’s toolkit, underpinning crop plant improvement programs. The estimation of plant genetic maps is a conceptually simple yet computationally complex problem, growing ever more so with the development of inexpensive, high-throughput DNA markers. The challenge for bioinformaticians is to develop analytical methods and accompanying software tools that can cope with datasets of differing sizes, from tens to thousands of markers, that can incorporate the expert knowledge that plant biologists typically use when developing their maps, and that facilitate user-friendly approaches to achieving these goals. Here, we aim to give a flavour of computational approaches for genetic map estimation, discussing briefly many of the key concepts involved, and describing a selection of software tools that employ them. This review is intended both for plant geneticists as an introduction to software tools with which to estimate genetic maps, and for bioinformaticians as an introduction to the underlying computational approaches.

Keywords: Genetic mapping; marker grouping; marker ordering

BACKGROUND
A genetic map is a list of genetic elements ordered according to their co-segregation patterns. Historically, those elements were inferred genes underlying phenotypic characters such as seed shape in pea or eye colour in Drosophila, though not being restricted to morphological traits. More recently, DNA markers have become prominent in genetic mapping studies. Consequently, today we most often think of genetic maps as ordered sets of markers, together with inter-marker distances, in essence representing ‘milestones’ along a chromosome or part of a chromosome.

A genetic map serves many practical biological purposes and is a key tool in both classical and modern plant research. Clearly, for the vast majority of plants whose genomes are yet to be sequenced it provides an essential resource to understand the order and spacing of markers (and relative order when compared to those of other plants), and for leverage of additional genetic information through comparative mapping with genetic maps and genome sequences of other plant species. Once developed, the genetic map underpins studies of plants genes, including quantitative trait loci (QTL) implicated in key plant traits. However, the influence of the genetic map is not limited to plant species whose genomes have yet to be sequenced. It provides a scaffold for genome sequence assembly and validation. The bridge between the genetic map and assembled genome sequence then enables the suggestion of candidate genes corresponding to QTL. Finally, it forms a vital tool in marker-assisted plant breeding programs, enabling plant breeders to develop in a targeted fashion new plant varieties in response to demands such as increased yield and resistance to pests and pathogens.

The algorithmic problem of genetic linkage map estimation is approaching its centenary. Following the discovery of genetic linkage, first reported in 1905
in the sweet pea by Bateson and colleagues [1] (although referred to there as coupling), and the observation by Morgan that the amount of crossing over between genes might indicate the distance between them on a chromosome [2], Morgan’s student Sturtevant used these ideas to develop the first genetic map in 1913 [3]. Sturtevant established the genetic map unit, the centiMorgan (cM), defining it to correspond to ‘a portion of the chromosome of such length that, on the average, one crossover will occur in it out of every 100 gametes formed’. Edwards [4] gives a fascinating and detailed account of early approaches in human genetic mapping before the widespread use of computers. Notably, many of the key concepts used in modern genetic linkage analysis were developed within the first 50 years after the discovery of linkage, whereas more recent advances have focussed on algorithmic approaches to solving established functions and procedures. Moreover, the method of maximum likelihood [5] was developed in the early days of genetic linkage analysis and influenced its development, with likelihood-based methods coming to dominate within a fairly short time.

One key concept in genetic linkage analysis is that of the log odds, or LOD, score. It was first used in genetic mapping in 1947 by Haldane and Smith [6] and was further developed by Morton in 1955 [7]. The LOD score compares the likelihoods of a dataset exhibiting r crossovers out of a potential N between a pair of markers under the hypothesis of linkage (i.e. \( \theta < 0.5 \), where \( \theta \) represents the recombination fraction) versus the same observation under the hypothesis of independent segregation (i.e. \( \theta = 0.5 \)):

\[
LOD = Z(\theta) = \log_{10} \left( \frac{1 - \theta^{N-r} \times \theta^r}{0.5^N} \right)
\]

The LOD function is maximised at \( \hat{\theta} = r/N \), the maximum likelihood estimate of \( \theta \), and the convention that \( Z(\hat{\theta}) > 3 \) lends strong support for linkage between the two markers is used frequently in mapping analysis. This value corresponds to a likelihood of observing the dataset, given that the two markers are unlinked, of <1/1000. Furthermore, given a prior probability of linkage for two markers chosen at random of 0.02, this likelihood corresponds to a probability \( P < 0.05 \) of a false positive. Ott showed for simple autosomal human diseases that such a value was well justified [8] but suggested different LOD thresholds for different disease types.

An analogous function exists for a set (rather than a pair) of markers, here considering all recombination fractions between adjacent pairs in the set and maximising them simultaneously. In general, functions that consider only a pair of markers are known as pairwise or two-point functions, whereas those that consider many markers simultaneously are known as multipoint functions.

Further early concepts related the genetic map distance \( D \) between two marker loci and the recombination fraction \( \theta \). The Haldane map function [9] assumes that all crossovers occur independently of one another:

\[
D = \frac{-(\log(1 - 2\theta))}{2}
\]

whereas the analogous Kosambi map function [10] models the effect of positive interference, where one chiasma deters the occurrence of a second in close proximity to the first:

\[
D = \log \frac{(1 + 2\theta) / (1 - 2\theta)}{4}
\]

with both functions acting over the range \( 0 < \theta < 0.5 \). Interested readers are also directed to the genetic mapping reviews by Birren and Green [11], Nelson [12] and Semagn et al. [13] which discuss these, and other, concepts in a more descriptive and detailed manner than we have scope for here.

The personal computing revolution in the 1980s led to a wide range of software tools for genetic mapping, many of them implementing and extending the likelihood ideas developed decades earlier. However, the parallel revolution in DNA marker technology presented serious computational challenges for these methods. This in turn led to the development of new algorithmic solutions, some of which we will discuss in later sections, to check these growing run-times. More recently, we have seen a new wave of high-throughput marker technologies, with single nucleotide polymorphisms (SNPs) perhaps the most widely-used presently. Dense marker datasets are pushing current approaches to their limits, in essence reinvigorating this field with new computational challenges, thereby leading researchers in a new hunt for effective methodological and algorithmic approaches.

### THE GENETIC MAP PROBLEM

In plant studies, a genetic map is estimated from a dataset derived from a mapping population,
which possesses certain characteristics that must be taken into account within the estimation procedure. The mapping population consists of \( p \) plants that result from a crossing experiment with a given experimental design. In plant analysis, commonly used designs include backcrossing, F2, doubled haploid (DH) and recombinant inbred lines (RILs). Many other experimental designs are also in use, although not all software tools for genetic mapping support all experimental designs. Furthermore, marker data can consist of different types: co-dominant or dominant. Thus our starting dataset consists of an \( m \times p \) matrix, with \( p \) members of a mapping population each scored for \( m \) markers. Taken together, the experimental design and the marker type will define the way in which distances and other functions are calculated between distinct markers.

The computational problem to be solved can essentially be split into three parts: grouping, ordering and spacing. The first part, grouping, divides the DNA marker set into distinct linkage groups. Clearly, the ideal would be a one-to-one correspondence between linkage group and chromosome but this will depend on the density and proximity of the underlying markers (a consequence of the co-ancestry of the two parents in addition to marker development strategies), as well as regional recombination rates. Of course, in the event that a researcher knows their DNA markers all derive from a single chromosome, this analytical step is unnecessary. The second part, ordering, takes each of the linkage groups in turn and aims to find the relative orders of the markers within the group. For a linkage group of \( m \) markers, there are \( m! / 2 \) possible orders and so for most large datasets this is not a task that can be undertaken exhaustively due to the prohibitive computational time required to carry it out. The final step, spacing, for an ordered set of markers in a given linkage group is to find the map distances (hereafter referred to simply as ‘distances’), in cM, between each adjacent pair of marker loci and hence the length of the linkage group as the sum of those distances. Many solutions to this problem involve taking or refining the two-point distances calculated in the grouping step. Figure 1 illustrates the genetic mapping process, showing the different steps taken to reach a genetic map.

**COMPUTATIONAL APPROACHES**

Early genetic mapping software concentrated on estimating the most likely positions of markers

---

**Figure 1:** The genetic map estimation process. In (a) a crossing experiment is performed. The resulting dataset of 15 markers scored in 12 individuals is seen in (b). The grouping process leads to three linkage groups in (c) which are then ordered in (d).
on a map, given a fixed order. However, from the late 1980s onwards, the imperative shifted to the ordering problem, as the number of markers meant it was no longer possible to examine all potential orders within a reasonable timeframe. Here, we will give a general description of the two most active areas of research: grouping and ordering. In later sections, we will describe sources of error in genetic map estimation and we will describe several software tools, outlining particular approaches taken to solve the genetic map problem.

**Marker grouping**

Several types of solution have been proposed for the marker grouping problem. One type recognises the underlying similarity to the well-studied area of agglomerative hierarchical clustering. Here, in methods such as nearest neighbour locus, clusters of markers (i.e. linkage groups) are grown by sequentially adding that marker which shows the lowest recombination value to the current members of the cluster. For example, the strategy employed by MAPMAKER [14, 15] is of this type. It begins by calculating all two-point maximum likelihood distances and corresponding LOD scores, with linkage established between pairs of markers if the LOD score is >3 and the inter-marker distance is <80 Haldane cM (default values that can be changed by the user). MAPMAKER considers linkage to be transitive such that if marker A is linked to marker B, and if B is linked to C, then A, B and C are candidates for belonging to the same linkage group (but which may be excluded later if they show significant deviation from additivity of their map distances). Another type of grouping method adopts ideas from graph theory. For example, MadMapper [16] and MST_MAP [17] both use graph partitioning approaches, creating a complete graph of all markers connected to all other markers and with connecting graph edges weighted by some two-point function of the data. Then, all edges over a certain threshold value are chopped, leaving a number of distinct subgraphs, each of which corresponds to a linkage group. It is notable that many grouping methods require input parameters to be specified by the user, thereby influencing their output. Consequently, linkage group content can be changed to some extent by a user's expert knowledge and opinion.

**Marker ordering**

Given a linkage group, we wish to find the order of its markers that maximises or minimises some scoring function. This scoring function is commonly known as an objective function. Put simply, we want some way to (i) evaluate the quality of a given marker order and (ii) to describe how one marker order is better or more suitable than another. Furthermore, we require an objective function that is simple to calculate yet is also biologically and statistically meaningful. An example of a simple objective function, to be minimised, is the sum of adjacent recombination fractions (SARF) [18]. Since adjacent marker loci tend to have the smallest recombination fractions, the marker order that minimises SARF was referred to by its developer as the minimum distance map. Examples of other popular objective functions are the maximum sum of adjacent LOD scores (SALOD) [19], the minimum number of crossovers [20, 21], the product of adjacent recombination fractions (PARF) [22], the minimum entropy [16], the minimum weighted least squares marker order [23], the maximum likelihood (ML) [14, 24] and the maximum number of fully informative meioses (SALEQ) [25]. Olson and Boehnke [26] presented a comparison of eight objective functions, analysing their performance with regard to locus informativeness, inter-marker distances, map length and sample size. Using a simulation study, they found that SALEQ and SARF performed best overall, while also being simple to calculate. However, linkage group size was limited to six markers and not all of those objective functions noted above were then developed or included in the analysis. More recently, Hackett and Broadfoot [27] compared the performance of SARF, ML and the weighted least squares procedure of JoinMap [23] (see below), in the presence of missing data, genotyping errors and segregation distortion. They found that the SARF and ML methods gave highly similar results. Furthermore, the ML method gave a higher number of correct marker orders than the weighted least squares method with similar results of the two methods for the mean rank correlations between estimated and true marker orders.

As we noted earlier, optimising an objective function over all $m!/2$ possible marker orders is not feasible for most datasets. Finding an optimal marker order for a particular objective function is known in computer science terminology as a non-deterministic
Genetic linkage map estimation

599

polynomial (NP)-hard combinatorial problem and necessitates the use of a search strategy that significantly reduces the space of marker orders to explore. Initially, search strategies such as seriation [28] and branch-and-bound [14] were used. In a seriation approach, a marker order is grown in a greedy fashion from an initial pair of tightly linked markers, adding at each step the single most informative marker in the position that optimises the objective function. In the branch-and-bound strategy an initial good solution is found, perhaps based on a two-point method. Subsequently, the initial marker order is probed by incrementally constructing partial orders, with those less good than the current full order eliminated, along with all full orders based on, or descended from, it. Once a full order better than the current is discovered, it becomes the next current order to be investigated. In this way, the objective function never decreases from the initial solution to the time the method terminates.

Subsequently to these approaches, a convenient relationship was discovered between the marker ordering problem and the symmetric wandering salesman problem, a variant of the travelling salesman problem (TSP), perhaps one of the best researched and understood problems in computer science [29]. In this problem, a given set of \( m \) cities has to be traversed so that every city is visited exactly once in such a way that the total distance travelled is minimised and that the choice of the first and last cities is free. Thus algorithms for solution of the TSP can be used within genetic map estimation, with the \( m \) cities recoded as our \( m \) markers. The type of strategy that seems to cope best with the presence of missing data, and hence that lends itself well to genetic mapping where missing data are common, is that of the local search procedure (e.g. simulated annealing, genetic algorithms, stochastic gradient, tabu search). Finally, geometric approaches [30, 31] transform inter-marker distances in a way that negates the need for a marker order search strategy. Instead, the optimal marker order is ‘read’ directly from the transformed datapoints.

SOURCES OF ERROR

It is necessary to be aware that genetic map estimation, like any estimation procedure, is prone to error. Error may arise due to many factors, including missing data, chiasma interference, genotyping error and segregation distortion. Missing data can lead to an incorrect marker order, particularly in dense regions of a map [27]. Some scoring failures are likely to be the results of random processes. However, there is also an element of systematic bias and we often see, for example, a particular marker for which several plants are not scored. In such a case, we may wish to delete the marker from our analysis. For less systematic cases we may wish to infer missing values through some computational method.

In the presence of chiasma interference, the Haldane map function is not valid, since it assumes no interference has taken place. However, many map functions account for chiasma interference in varying degrees. For example, the Rao map function is a versatile function that accounts for interference along a sliding scale with \( p = 1 \) (no interference) and \( p = 0 \) (complete interference) at its extremes [32]. Although the Rao map function is not widely implemented in software tools, the Kosambi map function, which accounts for interference at \( p = 0.5 \), is supported by many such software.

Genotyping errors can have a large impact on the accuracy of a map, inflating map lengths (particularly when applying multipoint maximum likelihood methods), reducing estimates of chiasma interference and supporting incorrect marker orders. In practice, many researchers will deal with genotyping errors by searching for double recombinants on an estimated genetic map (and sometimes recombinants over short distances), followed by checking of potentially erroneous scores. However, such an approach will not always be practical and is unlikely to uncover all cases of genotyping error. Consequently, two types of computational approach to this problem have been developed. The first type concerns the identification of potentially erroneous scores. For example, the JoinMap software [23] implements a method that calculates a probability for each genotype, given the scores of the two flanking markers and the inter-marker distances. Genotypes with low probabilities can then be investigated further. The second type concerns modifying the map either during or following the estimation process. For example, Shields et al. [33] described both an error filter for pairwise methods that corrected map length while considering the level of interference \( p \) and error corrections for multipoint methods. Although they showed that both methods performed well for certain datasets, also highlighting the underestimation of interference in their absence, they noted that the multipoint correction was
potentially not as satisfactory as the error filter method as it was performed on a marker order obtained under the assumption of no error.

Where segregation distortion is found to have occurred (calculated via Chi-squares), the mapping population deviates from allele and genotype frequencies expected under the Hardy–Weinberg Law (which states that population frequencies remain in equilibrium across generations unless disturbed by some phenomenon). For plant mapping populations such deviations from the expected frequencies typically arise as the result of gametic or post-zygotic selection (see ref. [13] for a larger list of potential factors), resulting in a marker locus which, though appropriate for the marker scores, does not correspond to the physical location of the marker. Although Hackett and Broadfoot [27] showed in their simulation analysis that the presence of segregation distortion had little effect on the accuracy of marker order or map length, this contradicts the results of other studies [13] and may be dataset specific. Consequently, methods which allow such markers to be identified prior to analysis are useful, as they give the researcher the opportunity to analyse the dataset either with or excluding such markers (or potentially both).

The interplay between these sources of error is complex. For example, we have already mentioned briefly the interaction between genotyping errors and chiasma interference. Hackett and Broadfoot [27] also noted that missing values led to shorter map lengths for more widely spaced markers, particularly in the presence of segregation distortion, when using the weighted least squares method of JoinMap. The authors also noted that, in their study, missing values had a lesser effect on the accuracy of marker order than did genotyping errors. Interestingly, Hackett and Broadfoot also showed that marker order is robust to both missing data and genotyping errors for well-spaced data-points (10 cM intervals, although not always in combination), perhaps validating a two-step approach of developing a framework map followed by marker addition, as advocated by Vision et al. [34] and by Kozik et al. [16].

Furthermore, other sources of error exist. For example, we have recently observed in the model grass Brachypodium distachyon (data unpublished) that mixing marker types within a single scoring scheme can result in ‘attraction’ of similar types of marker independent of their chromosomal locations. Consequently, diagnostic tests and methods that allow researchers to interact with their mapping data are desirable features of genetic mapping software.

Additional data may help to resolve errors. For example, physical mapping data and in particular complete genome sequences, will also present a marker order. This is highly attractive, as estimating marker order is the most difficult part of genetic map estimation for large datasets. Indeed, at least one software tool, CarthaGène [35], has recently implemented a novel method to take into account marker order in the same or another organism. However, we should also be aware when comparing genetic and physical marker orders, that the genome sequence is itself an estimate gained from a sequence assembly process and may not be highly accurate for up to several years following initial sequencing. Furthermore, when comparing genetic and physical marker orders from different organisms, we must not underestimate the effect that micro-rearrangements could have on making inferences on the accuracy of the genetic map. Rather the comparison between the two datasets gives an opportunity to highlight areas of conflict that may then be investigated further.

SOFTWARE TOOLS
Here we give a brief description of 11 software tools for genetic map estimation, approximately in the chronological order of their development, describing the various computational strategies they have taken. We chose these tools for discussion because either we knew they were popular amongst our colleagues in the plant community or because we felt they played an important role in informing readers about the recent history of genetic map estimation software and the range of computational solutions developed to date. However, this is not an exhaustive list of such tools and it is important to note that some of them are either no longer under development or are no longer available. In Table 1, we give details on the availability and status of the software discussed, together with an indication of the experimental designs supported by each tool. Most, if not all, support dominant, co-dominant and a mixture of marker types. A broader list of software tools for genetic analysis, including those for genetic map estimation, can be found at http://www.nslig-genetics.org/soft/.
<table>
<thead>
<tr>
<th>Tool</th>
<th>Availability</th>
<th>Software status</th>
<th>Experimental designs</th>
</tr>
</thead>
<tbody>
<tr>
<td>CarthaGene</td>
<td><a href="http://www.inra.fr/mia/T/CarthaGene/">http://www.inra.fr/mia/T/CarthaGene/</a></td>
<td>Version 1.0 released in October 2004, with no apparent subsequent development.</td>
<td>F2 intercross, F2 backcross, RIL, phase known outbreds</td>
</tr>
<tr>
<td>DGMAP</td>
<td>No longer available</td>
<td>No longer under development.</td>
<td>Various including F2 backcross</td>
</tr>
<tr>
<td>JoinMap</td>
<td><a href="http://www.kyazma.nl/">http://www.kyazma.nl/</a></td>
<td>Version 4 released in July 2006 with most recent maintenance release in August 2008.</td>
<td>BCI, F2 intercross, RIL (self), DH, DHI, DH2, HAP, HAPI, CP, BCpxFy, IMxFy</td>
</tr>
<tr>
<td>MadMapper</td>
<td><a href="http://cgpdb.ucdavis.edu/XLinkage/MadMapper/">http://cgpdb.ucdavis.edu/XLinkage/MadMapper/</a></td>
<td>Most recent version released in February 2008.</td>
<td>Specialises in RILs but flexible scoring scheme can be employed for many other design types</td>
</tr>
<tr>
<td>Map Manager QTX</td>
<td><a href="http://www.mapmanager.org/">http://www.mapmanager.org/</a></td>
<td>Last release date unknown. No longer under development.</td>
<td>Advanced intercross, advanced backcross, RILx</td>
</tr>
<tr>
<td>MSTMAP</td>
<td><a href="http://www.BB.23.91.145/mstmap/">http://www.BB.23.91.145/mstmap/</a></td>
<td>Last release date unknown.</td>
<td>BCI, DH, HAP, RIL</td>
</tr>
<tr>
<td>Neighbour Mapping</td>
<td>Available from the author</td>
<td>No longer under development.</td>
<td>RIL</td>
</tr>
<tr>
<td>RECORD</td>
<td><a href="http://www.plantbreeding.wur.nl/UK/software.record.html">http://www.plantbreeding.wur.nl/UK/software.record.html</a></td>
<td>Last release date unknown. No longer appears to be under development.</td>
<td>BCI, F2, F3, RIL</td>
</tr>
<tr>
<td>THREaD Mapper</td>
<td><a href="http://cbr.jic.ac.uk/dick/software/threadmapper/index.html">http://cbr.jic.ac.uk/dick/software/threadmapper/index.html</a></td>
<td>Version 1.0beta released in July 2009.</td>
<td>F2 intercross, F2 backcross, RIL (self), DH</td>
</tr>
</tbody>
</table>

Software largely superseded by more recent tools, or no longer available for download or analysis by new users, are shaded in grey. See, for example, JoinMap 4 Manual (at http://www.kyazma.nl/index.php/mc/JoinMap/sc.Manual/) for details of each type of experimental design.
MAPMAKER (1987–92)
In the 1980s a new wave of molecular marker technologies, such as RFLP, had led to datasets consisting of hundred of markers. MAPMAKER [14] led the way in computationally efficient strategies for datasets of such a size. Building on previous research, its authors developed a new way to maximise the multipoint likelihood objective function for a given marker order [15]. This approach combines an EM algorithm for recombination fraction estimation with a Hidden Markov Model (HMM) method for calculating the expected number of inter-marker recombination events, significantly lowering computation time for large datasets. Marker order estimation is carried out using a range of search procedures including an exhaustive search for small datasets, a greedy algorithm, a branch-and-bound method and simulated annealing. The greedy process was further optimised through a pre-processing step that involves a three-point analysis of all linked triples. Once a marker order is found it can be locally perturbed through a ‘ripple’ procedure, whereby the order of neighbouring markers is permuted, to check for improvements in the objective function. Finally, MAPMAKER reports the log-likelihoods of alternative solutions to the user, not simply the best one found during the search procedure. MAPMAKER uses a simple two-point approach to infer linkage groups, using a transitive procedure on two-point maximum likelihood distances and LOD scores, as described in the previous section. Following map estimation, it enables the calculation of a LOD-error score regarding evidence for genotyping error, recommending that markers with scores >1 are investigated further. The MAPMAKER software was written for DOS and for UNIX operating systems on Sun and Apple Macintosh computers. It includes an interactive command language that enables data exploration and has been very widely used and cited in the construction of linkage maps for many species groups including human and plants.

JoinMap (1993–)
JoinMap [23] is one of the most widely used software tools for the estimation of genetic maps. It is commercially available software and benefits from a highly advanced MS-Windows user interface for data management and analysis, professional support and continued development. JoinMap’s linkage grouping strategy is essentially a clustering on recombination frequencies (known as linkages), the result of which can be examined using several grouping test procedures. JoinMap has two distinct marker order search strategies: the regression mapping and maximum likelihood mapping algorithms. Its original regression mapping approach uses a seriation/greedy strategy within its search procedure, with additional local reshuffling via the ripple operation, permuting markers in a sliding window of size 3. The objective function, to be minimised, is a weighted least squares function which compares observed pairwise distances (calculated by applying a mapping function such as Haldane or Kosambi to observed recombination frequencies) to their expected counterparts, weighted by the squares of the associated LODs. At each step of the algorithm, a goodness-of-fit statistic known as a \( G^2 \) likelihood ratio statistic is calculated. This statistic compares the observed recombination frequencies with their map-derived counterparts (found using the inverse mapping function) and its value is used to judge, at each step, whether the addition of a marker should be accepted. The newer maximum likelihood approach was added to deal with larger datasets and is much faster than the regression mapping approach. The new approach follows [24], maximising a multipoint likelihood objective function using Gibbs sampling to estimate multipoint recombination frequencies, simulated annealing as a marker order search strategy, and spatial sampling to escape from local optima. As noted above, the JMCHK module permits a posteriori calculation of genotype error. JoinMap also has a range of other functionality, such as various diagnostic procedures, data import facilities, plots, and its key capability to integrate data from multiple populations.

DGMAP (1995)
DGMAP [30] uses a distance geometry approach to estimate an optimal marker order from an initial \( m \times m \) matrix of weighted inter-marker distances (e.g. recombination fractions). The approach is inspired by the Distance Geometry (DG) algorithm [36], which uses inter-object distances to calculate in N-dimensional space co-ordinates for the objects that are consistent with the inter-object distances. DGMAP adapts this algorithm for mapping data and subsequently refines the marker loci via a weighted least-squares method, minimising the differences between the inter-marker distances in the major axis (i.e. the 1D linear map) and the
initial distances. In essence, the sum of differences between the original and transformed distances is the objective function, though the approach does not involve marker order searching in the conventional sense as the order is implicit from the transformation process. Instead, the majority of the computational effort is spent on this transformation process and is practical for the order of hundreds of markers. The authors suggest use of the output for further analysis in maximum likelihood-based software. However, DGMAP does not appear to tackle the marker grouping problem. In the initial \( n \times m \) distance matrix calculation, DGMAP deals with missing and imprecise pairwise distances by constructing/ replacing them with those calculated via intermediate, significantly linked marker pairs. This process is carried out using Floyd’s all pair shortest path algorithm [37]. The authors also note that markers containing high levels of genotyping error will be easily identified as those lying far from the major axis in N–D space. In addition, DGMAP supports a generic mapping function to account for differing levels of chiasma interference. The DGMAP software was developed in the C programming language for the UNIX platform but is no longer available.


CarthaGène [29, 35] focuses on the marker ordering problem. Recognising the close relationship of this problem to the TSP, it uses several algorithmic approaches such as nearest neighbour, simulated annealing, tabu search and the Lin–Kernighan heuristic. For search strategies in general, when the landscape of possible marker orders is studded with many local optima, most local search algorithms include a mechanism that enables them to escape from a local optimum. For example, one of the simplest neighbourhood structures is the 2-change neighbourhood of a map, the set of all maps obtained by an inversion of a subsection of the map. CarthaGène’s algorithms exploit this neighbourhood to achieve high quality results. Their high level of sophistication also make CarthaGène robust to missing data, a fact it exploits to enable joining of maps from distinct crosses. In addition, it optimises the EM algorithm introduced in MAPMAKER for backcross datasets. Furthermore, CarthaGène produces a set of best marker orders rather than a single optimum, enabling further user interaction. As noted previously, it has recently included a comparative mapping method that enables genome sequence data with known orthologous relationships to a subset of the markers to be exploited. The method, in which a reference order gained from the sequence is merged with the dataset undergoing analysis, uses a Bayesian model based on the multi-point likelihood and the number of breakpoints between a map and the reference order. The CarthaGène software was written in C++ and three interfaces are available: a graphical user interface enabling day-to-day use of the software and two programmable interfaces (shell and shared library) that permit advanced interaction and extension.

Neighbour mapping (1997)

The Neighbour mapping algorithm [38] is based on the Neighbor-Joining (NJ) method of phylogenetic analysis, developed by Saitou and Nei [39]. The NJ algorithm begins with an \( n \times m \) distance matrix and a “star” tree topology. In a stepwise greedy process, that pair of nearest tree nodes that minimises a function of the branch lengths is found at each step and is placed at an appropriate position in the tree. The resulting topology is an estimate of the minimum branch length tree and the properties of NJ, such as its rapid computation time and relatively good accuracy, are well studied. The Neighbour Mapping approach modifies this algorithm for genetic map estimation, using a matrix of two-point distances as the starting point. Test datasets show the method to be fast and accurate and capable of placing all markers in a given map, with ‘inappropriate’ markers often easy to spot as flanked by large, bare map intervals. Neighbour Mapping naturally gives most weight to the most closely linked marker pairs, minimising the effect of erroneous markers on the map. However, an inherent problem with the method is its tolerance and inclusion of inappropriate markers, which would be screened by other algorithms. Neighbour mapping has features in common with seriation approaches [28] though it was developed independently of them. It was originally developed as a method to explore the consequences of data errors on the resulting maps and potentially to provide a suitable starting marker order to MAPMAKER or an order of marker inclusion to JoinMap. The related software was written in the BASIC programming language. The speed of execution warrants its examination for high throughput markers, perhaps with the addition of a local perturbation approach.
The marker grouping problem was not touched upon but the author suggested the use of a two-point approach.

**MapManager QTX (2001)**

Map Manager QTX [40] is an enhanced and amalgamated version of the previous Map Manager QT and Map Manager programs for QTL and genetic mapping respectively. For genetic mapping, it carries out a combined marker grouping and ordering procedure that can be further refined using rippling. The algorithm, derived from the rapid chain delineation method [41], creates new linkage groups by establishing tightly linked pairs of markers and then redistributes markers between linkage groups in order to optimise the objective function of the sum of LOD scores for adjacent markers. The process of creating new linkage groups and redistributing markers continues until the remaining (unlinked) markers contain no linked pairs. The whole procedure is repeated up to three times, with the stringency of linkage decreasing at each cycle. The authors found through simulation that the time to identify linkage groups and marker orders varied approximately as $C^{1.6} L^{2.0}$, where $C$ is the number of linkage groups and $L$ the number of markers. Map Manager QTX reconstructs missing marker scores from the genotypes of adjacent markers, using the Markov Chain method of Jiang and Zeng [42]. It also allows markers under segregation distortion to be identified and is capable of analysis of many types of crosses, including complex crossing schemes. The software is no longer under development and until recently was distributed for Microsoft Windows and Mac OS operating systems.

**RECORD (2005)**

The RECORD (R_Ecombination Counting and ORDering) software [21] was developed for the construction of dense maps, such as those with over 500 markers per linkage group. Its objective function is the sum of recombination events over pairs of adjacent markers, known as COUNT, which is minimised. This function has the attractive property that it gives similar results to those obtained by the likelihood function and SARF (identical for datasets with no missing or unknown data), yet is much more efficient to calculate. For its search strategy, RECORD combines a greedy algorithm with local reshuffling. The method begins with a random selection of two markers, with remaining markers chosen again at random and added one at a time. The location of each added marker is that at which COUNT is minimised. Once all markers are added, a sliding window moves across the marker order, inverting the local order and accepting the change if COUNT is reduced. $m−2$ rounds of this sliding window procedure are carried out, starting with a window of size 2 and ending with size $m−1$. A consequence of this approach is that two different runs of the same dataset may differ. The RECORD developers recommend that ~10 runs of a given dataset are made. Similarly to other tools such as CarthaGène and MAPMAKER, a series of near-optimal marker orders are given in addition to the optimal order. The RECORD developers also developed a method called SMOOTH [43], which used an iterative process to remove genotyping errors during map construction, using neighbouring marker scores to identify potentially erroneous datapoints. They compared RECORD’s performance to that of JoinMap 2.0 and showed that it was considerably faster and, for most experiments performed, less sensitive to missing data and scoring errors. The original version of RECORD did not appear to perform linkage grouping or calculate genetic distances in cM. The RECORD software was written in the C++ programming language for the DOS operating system.

**AntMap (2005–06)**

AntMap [44] uses a TSP-based search algorithm to rapidly find the marker order either that minimizes SARF or the sum of adjacent absolute two-point log likelihoods. This goal is achieved through ant colony optimisation (ACO), a set of algorithms inspired by the co-operative behaviour of ants in finding the shortest path from their nest to a food source. ACO has been used successfully to solve the TSP problem and here, markers were considered to correspond to the TSP cities and the absolute values of log likelihoods (or recombination fractions) between marker loci to the distances between them. To enhance the ACO algorithm, procedures known as random and elite selections were implemented to avoid solutions becoming trapped in local optima. AntMap provides two options for linkage grouping, a nearest neighbouring locus strategy and a strategy similar to that used by MAPMAKER. It implements a segregation test (chi-squared goodness of fit), categorising markers with distorted frequencies according to $P$-value thresholds.
Furthermore, it employs a bootstrap test to gauge the reliability of an estimated marker order. The associated software tool is written in the Java programming language and has a visually attractive graphical user interface (GUI).

**MadMapper (2006—)**

MadMapper [16] was developed for the analysis of high-throughput molecular marker datasets, specifically for RILs of sixth or higher generation (i.e. an inbred line whose genome is a ‘mosaic’ of two inbred parental lines, developed by repeated selfing and single seed descent over a given number of generations from the segregating F2 generation [45]). It consists of three programs, each written in the Python scripting language, and is consequently cross-platform software. MadMapper_RECBIT calculates distances between pairs of markers using two novel scoring schemes known as BIT and REC and performs various quality control procedures. It also creates linkage groups on these pairwise distances using the transitive property, denoting non-trivial groups as ‘complete’ if all group members are linked to one another and as ‘linked’ otherwise. For each linkage group, MadMapper_XDELTA uses a pairwise matrix of recombination fractions to find the marker order with lowest entropy by minimising the total sum of differences between adjacent cells. Notably, pairwise scores for both linked and unlinked markers are considered in this analysis, unlike in many other methods. This novel function, similar perhaps only to the entropy-like measure of SALEQ, is optimised by first considering only the members of a framework marker dataset and adding markers to it one by one, or by calculating a framework map for up to 10 markers if one is not already available. Once a map has been generated it can be further refined through a ripple process. Finally, CheckMatrix provides highly useful and attractive data visualisations for genetic map validation. It generates both graphical genotype plots and 2D heat maps of two-point scores, allowing users to spot mis-ordered markers and potential genotyping errors without necessitating further computational analysis.

**MST\text{MAP} (2008—)**

MST\text{MAP} [17] is a recently developed tool which uses concepts from graph theory and machine learning to rapidly estimate high density genetic maps. For linkage grouping, a graph is constructed, where nodes consist of markers and edges all pairwise inter-marker Hamming distances (simply the number of individuals whose scores differ for the two markers). Using the popular Chernoff bound (Heoffding’s inequality) concept from machine learning, a user-influenced distance threshold is found and all edges with distances greater than this value are cut. The result is a number of connected components, each representing a linkage group. For each linkage group MST\text{MAP} then uses a TSP-based marker ordering strategy. Many TSP search methods begin to explore the solution space either from a purely random order or from a rapidly obtained greedy solution. MST\text{MAP} shows that when data quality is high the optimal solution can be identified rapidly, simply by calculating the minimum spanning tree (MST) of the marker graph, based on a chosen objective function. For noise-free datasets, the MST is proven to be the requisite TSP solution to the marker ordering problem. In the presence of noise, the MST may not be a ‘path’ (i.e. the tree is not of a completely linear form but looks like a branch with twigs sprouting from it). In such cases, three simple heuristics are used to find an optimal path solution from the longest path, or *backbone*, in the MST. MST\text{MAP} also uses an EM algorithm to deal with missing data and a further neighbourhood-based algorithm to detect and remove scoring errors, which may then be dealt with as missing data. Consistent with the findings of Hackett and Broadfoot [27], the authors concluded that missing data have a much less negative impact on the quality of the final map than genotyping errors. Additionally, they found that the choice of objective function does not have a significant impact on the quality of the map. In a comparison with RECORD, Cartha\text{G}ène and JoinMap, MST\text{MAP} performed very well, producing for most experiments the most accurate maps in a fast time. The MST\text{MAP} software was written in C++ and is available for Linux workstations.

**THREaD Mapper (2009—)**

Like DGMAP, THREaD Mapper [31] first calculates an $m \times m$ matrix of inter-marker distances, using a range of scoring schemes, and uses a geometric approach to calculate marker positions. However, here a Principle Co-ordinate analysis combined with a variant of the Local Principal Curve algorithm are used to map the data into three-dimensional (3D) space and to thread an optimal 3D trendline through the markers. Markers are then projected onto the
trendline, thereby producing an optimal marker ordering. Consequently, the approach is deterministic and requires no marker ordering search strategy. For marker grouping, a spectral embedding approach on the pairwise matrix is used to separate distinct groups. This approach has the effect of bringing together well connected pairs of markers and dragging apart remote pairs. Linkage groups can then be separated either by eye or by various algorithmic approaches. The 3D nature of THREaD Mapper lends itself well to visual interpretation of the data. For example, ‘suspicious’ markers (either those with missing data or subject to genotyping errors) can be easily seen as lying far from the trendline, much as seen by Newell et al. [30] in the DGMAP software. Furthermore, individual markers can be colour-coded by user-defined categories. Examples of colour codings include marker type, physical chromosome location (in the same or another species), and linkage group assignment in another tool. These functions make the visualisation process of use in validation of results. THREaD Mapper implements a Chi-squared segregation test, reporting \( P \)-values of each marker to the user and allowing them to include or exclude markers accordingly. THREaD Mapper has been developed in the first instance as a web-browser enabled tool.

**DISCUSSION**

We have described a wide range of approaches and algorithms for genetic map estimation and have introduced briefly eleven software tools that employ them. For the linkage grouping problem, the user is typically given control of various thresholds or rules that influence the separation of distinct groups. From our experience, linkage grouping is best performed through the setting of soft rules based on heuristics and on the user’s expert knowledge. Consequently, we believe that this strategy is well chosen. However, we also feel that it is unfortunate that most effort in genetic mapping algorithm development has been spent researching the marker ordering problem, for the large part leaving the marker grouping problem as a simple clustering exercise. We hope that further research into this area is carried out, particularly as datasets are becoming so large and the opportunity for mis-classification is growing. In addition, we believe that the marker grouping and ordering problems are inter-related, and perhaps we might see more approaches that consider these two problems in tandem, as embodied in the Map Manager QTX and THREaD Mapper algorithms.

With regard to the marker ordering problem, there is a clear trade-off between complexity of the ordering model and speed. For example, many consider the multi-point likelihood to be the gold standard objective function due to its desirable statistical properties. However, runtime for large datasets can be prohibitively large and users may become drawn towards faster methods. This choice can only become more difficult as datasets grow and it is a challenge to developers to discover new search methods that can facilitate the fast and accurate use of excellent objective functions or that develop new fast methods that also show good accuracy.

In general, the accuracy of any genetic map estimation method relies on the distribution of recombination frequencies, the proportion of missing data, the quantity of noise due to genotyping errors and genetic interference phenomena. While sources of error have only been touched upon lightly in this review, their importance to the area should not be underestimated. We have seen that current techniques for dealing with errors include putative error identification following map estimation and error reduction both during and following map estimation. A rigorous comparison of these types of method, together with individual techniques within the two categories, would be of great utility to both users and developers of such tools and would perhaps stimulate further research. Furthermore, a greater use of quality control mechanisms for marker order accuracy, such as those adopting re-sampling procedures such as bootstrapping and jack-knifing, would also be helpful for users.

It is notable that most of the tools described here share several common features in their algorithmic approaches in addition to their functionality. Furthermore, the influence of computer science has grown significantly in this field since the 1980s, with many tools resulting from collaborations between biologists, statisticians and computer scientists. Consequently, the goal of fast, biologically meaningful and statistically valid approaches are becoming achievable. We have not attempted here to identify the ‘best’ tool, but rather to talk about features of each that may be important to different researchers. For example, the most recently developed or updated software tools are, in general, the fastest. Of these MST_MAP is undoubtedly one of
the fastest currently available, if the not the fastest, but users may also appreciate the excellent visualisation tools of MadMapper or the ease of use of JoinMap. In practice, it is almost certainly best to use a mixture of approaches in developing and refining a map, not only because they each bring something unique to the analysis but also because we do not know which approach will succeed best for a new dataset and we do not know enough about the behaviour of each tool to judge this in advance. For example, we recently took part in a project that used JoinMap and MapManager QTX to develop maps, which we then validated within THREaD Mapper. We strongly believe that map estimation is an iterative process, where researchers should first grasp the global pattern of their dataset before revaluing and revising the grouping and ordering of markers rather than performing a rigid, linear three stage methodology of grouping, ordering and spacing. Consequently, tools that enable users to carry out such an iterative process will be of greatest value in future genetic map estimation.

Key Points

- The genetic linkage map estimation problem is currently approaching its centenary.
- High throughput genetic marker technologies are producing enormous datasets that are pushing genetic mapping software to their limits.
- Two of the most important sub-problems are the marker grouping problem and the marker ordering problem.
- Many different algorithmic techniques have been developed and adapted for genetic map estimation but there are shared features of all software tools.
- In practice, using several tools with different advantages in an iterative map development and refinement process may achieve the best results.

Acknowledgements

The authors would like to thank their colleague Noel Ellis for re-introducing them to this problem, for his insightful ideas that led to the THREaD Mapper project and for his help in improving this review. They would also like to thank David Garvin and John Snape for their valuable guidance on the practicalities of genetic map estimation. In addition, they would like to thank three anonymous reviewers for their thoughtful comments and helpful suggestions for improving this review.

FUNDING

Biotechnology and Biological Sciences Research Council.

References


