Constructing Large-Scale Genetic Maps Using an Evolutionary Strategy Algorithm

D. Mester, Y. Ronin, D. Minkov, E. Nevo and A. Korol[1]

Institute of Evolution, University of Haifa, Haifa 31905, Israel

Manuscript received July 23, 2003
Accepted for publication August 28, 2003

ABSTRACT

This article is devoted to the problem of ordering in linkage groups with many dozens or even hundreds of markers. The ordering problem belongs to the field of discrete optimization on a set of all possible orders, amounting to $n!/2$ for n loci; hence it is considered an NP-hard problem. Several authors attempted to employ the methods developed in the well-known traveling salesman problem (TSP) for multilocus ordering, using the assumption that for a set of linked loci the true order will be the one that minimizes the total length of the linkage group. A novel, fast, and reliable algorithm developed for the TSP and based on evolution-strategy discrete optimization was applied in this study for multilocus ordering on the basis of pairwise recombination frequencies. The quality of derived maps under various complications (dominant *vs.* codominant markers, marker misclassification, negative and positive interference, and missing data) was analyzed using simulated data with ~50–400 markers. High performance of the employed algorithm allows systematic treatment of the problem of verification of the obtained multilocus orders on the basis of computing-intensive bootstrap and/or jackknife approaches for detecting and removing questionable marker scores, thereby stabilizing the resulting maps. Parallel calculation technology can easily be adopted for further acceleration of the proposed algorithm. Real data analysis (on maize chromosome 1 with 230 markers) is provided to illustrate the proposed methodology.

An important step in generating multilocus genetic maps using the results of linkage analysis is the determination of the true marker order. One of the possibilities in addressing this problem is to recover the linear marker order from the known pairwise marker distance matrix d_{ij}. A primary difficulty in ordering genetic loci using linkage analysis is the large number of possible orders: for n loci on a chromosome, $n!/2$ distinct orders should be evaluated. In real problems, n might vary from dozens to 200–500 markers and more (*e.g.*, www.maizemap.org/ibm_frameworkmaps.htm; see also Ott 1991). Clearly, even for $n \sim 30$, it would not be feasible to evaluate all $n!/2$ possible orders using two-point linkage data. This is why multilocus ordering is considered as a nonpolynomial (NP)-hard combinatorial problem (Wilson 1988; Olson and Boehnke 1990; Falk 1992; Ellis 1997). A solution to this problem can be obtained on a Pentium-IV (1500 Mhz) computer even for a modest case such as $n = 10$ after 1 hr.

Several methods have been proposed for determination of marker order (Lathrop *et al.* 1985; Lander and Green 1987; Knapp *et al.* 1995; Newell *et al.* 1995; Liu 1998) and implemented in software packages like LINKAGE (Lathrop *et al.* 1984), MapMaker (Lander *et al.* 1987), FastMap (Curtis and Gurling 1993), and JoinMap (Stam 1993). Historically, the main approach of ordering markers within linkage groups was based on multipoint maximum-likelihood analysis. Several effective algorithms have been proposed using various optimization tools, including the branch and bound method (Lathrop *et al.* 1985), simulated annealing (Thompson 1984; Weeks and Lange 1987; Stam 1993; Jansen *et al.* 2001), and seriation (Buetow and Chakravarti 1987).

Olson and Boehnke (1990) compared eight different methods for marker ordering. In addition to multilocus likelihood, they also considered more simple criteria for preliminary multipoint marker ordering in large-scale problems based on two-point linkage data (by minimizing the sum of adjacent recombination rates or adjacent genetic distances). The simple criteria are founded on the biologically reasonable assumption that the true order of a set of linked loci will be the one that minimizes the total map length of the chromosome segment. Simple methods work quickly but their accuracy may depend on the number of markers, distribution of recombination frequencies (presence of large gaps), percentage of missing data, type of the employed optimization criterion, noise caused by misclassification, and genetic interference. That is why there is a tendency to combine two-point analysis with more general multipoint methods. However, even for simple methods, based on pairwise analysis, there is a pressing need for efficient algorithms enabling high-quality "preliminary" multipoint ordering. Keeping in mind the large number of markers employed in mapping projects of different organisms (humans, experimental model organisms, and agricul-

[1]*Corresponding author:* Institute of Evolution, University of Haifa, Haifa 31905, Israel. E-mail: korol@esti.haifa.ac.il

tural plants and animals), such algorithms should cope with many dozens and even hundreds of markers (*e.g.*, 100 ÷ 1000) per chromosome and in a reasonable executing time.

We present in this article a new, highly efficient algorithm of multilocus ordering based on two-locus linkage data that employs the evolutionary optimization strategy (ES). ES is a heuristic algorithm mimicking natural population processes. The numerical procedures in such optimization are based on simulation of mutation and reproduction, followed by selection of the fittest "genotypes," representing the obtained values of the optimization criterion. Together with genetic algorithm (Holland 1975) and evolutionary programming (Fogel 1992), evolution strategies form the class of evolutionary algorithms (Nissen 1994). The evolutionary strategies were proposed in the 1970s (Rechenberg 1973; Schwefel 1977, 1987) to solve optimization problems with real-value variables. A recent survey of search strategies for combinatorial problems was provided by Muhlenbein *et al.* (1998). ES for optimization problems is presented as a random search by asexual reproduction, which uses *mutation*-derived variation and *selection*. The mutation change of the current vector of parameters can be introduced by adding a vector of normally distributed variables with zero means. The level of changes can be adapted by variances of these disturbances.

In contrast to ES, genetic algorithms, introduced by Holland (1975), simulate sexual reproduction that is characterized by recombination of two parental strings to build the offspring generation. Clearly, the contributions of *mutation* and *recombination* as sources of variation in the search strategy are different: mutation is based on chance only, and the success of a single mutation is largely unpredictable. Crossover can be viewed as a history-preserving operation, which at the same time introduces a new structure to be tested in competition. Homberger and Gehring (1999), Mester (1999, 2000), and D. Mester (unpublished results) adopted the ES algorithm to solve the vehicle routing problem with time-window restrictions, which is similar, to some extent, to multipoint analysis of markers belonging to several chromosomes (linkage groups). In this article, we applied the ES algorithm for multipoint marker ordering using the similarity between this problem and the well-known traveling salesman problem (TSP; Press *et al.* 1986; Weeks and Lange 1987; Falk 1992; Schiex and Gaspin 1997).

EVOLUTION STRATEGIES AND THE HEURISTICS IN THE DEVELOPED ALGORITHM

The employed procedure as a simulated analog of evolutionary processes: Usually, the optimization process of an objective function $f(\boldsymbol{x})$ with n real-value variables $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ can be represented as an evolution of the solution vector $\boldsymbol{x} \in R^n$. The main elements of ES algorithms and their correspondence with the elements and processes of an "evolving population" are presented in Table 1.

The common ES algorithm steps: Evolution strategies define the size of a population and the rules for the selection process. Various approaches were proposed for choosing the population size in the ES, including the (1 + 1) strategy (Rechenberg 1973) and (μ, λ) strategy (Schwefel 1977). With the (1 + 1) strategy, population size is equal to one individual used to obtain offspring individuals via mutation operation. If a new individual is better than the "parent," it replaces the parent. The (μ, λ) strategy works with a population of size λ. The selection operator chooses μ best individuals to establish the new generation. Both versions, (1 + 1) and (μ, λ), employ the following steps:

1. Create λ individuals ($\boldsymbol{x}^k$) of initial population $\boldsymbol{P}^0$.
2. Compute the fitness $f(\boldsymbol{x}^k)$, $k = 1, \ldots, \lambda$.
3. If the optimization process is terminated, then stop.
4. Select the $\mu \leq \lambda$ best individuals.
5. Create λ/μ offspring $\boldsymbol{x}^{k+1}$ of each of the μ individuals by small variation.
6. Return to step 2.

Peculiarities of the combinatorial version of ES: Clearly the multilocus ordering problem cannot be directly represented in terms of ES with real-value formulation. Combinatorial versions of ES differ from the real-value formulation by specific representation of the solution vector $\boldsymbol{x}$ and mutation mechanisms (Homberger and Gehring 1999). In combinatorial formulation, the solution (an "individual") can be represented as a vector $\boldsymbol{x} = (x_1, x_2, \ldots x_n)$ that consists of n ranked discrete coordinates (chromosomes) or as a directed graph $G(A, B)$ with a set of nodes $A = \{a_1, a_2, \ldots a_n\}$ and set of arcs $B = A \times A$, where node a_j, $j > 0$, represents the chromosome. The fitness function assigns to each of the $n(n-1)/2$ arcs (a_i, a_j) [or pair of coordinates (x_i, x_j)] a nonnegative d_{ij} cost of moving from element i to element j. The problem is symmetric if and only if $d_{ij} = d_{ji}$ for all arcs. For optimization of a combinatorial problem, one needs to define such an order of the vector coordinates (or nodes) that will provide minimum total cost.

The mutation operator (referred to hereafter as *mutator*) changes the vector $\boldsymbol{x}^k$, thereby producing a new solution vector $\boldsymbol{x}^{k+1}$. For this goal, one can use the *move-generation* and the *solution-generation* mechanisms (Osman 1995; Homberger and Gehring 1999) or the *remove-insert* mechanism (Mester 1999). Our version of the combinatorial ES algorithm employs multiparametric mutator (MPM), which changes the solution vector via removing and inserting β coordinates of $\boldsymbol{x}^k$ (Mester 1999; D. Mester, unpublished results). The common heuristic *remove* defines a random proportion $\beta = (0.1 + 0.5r)\,n$ of rejected coordinates in the solution vector, where n is the number of coordinates in the solution and r is a random value (*e.g.*, evenly) distributed between 0 and

TABLE 1

Main components of ES algorithm as a simulation analogue of evolutionary models

Natural elements	Simulation elements
Chromosome	Variable value x_i
Individual, a set of chromosomes	A solution vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$
Mutation, a change of the chromosome for a small value	Operator **M:** $\boldsymbol{x}^k \rightarrow \boldsymbol{x}^{k+1}$
Population, a set of (parental) individuals	A set $\boldsymbol{P}$ of solution vectors $\{\boldsymbol{x}^k\}$
Fitness, a quantitative characteristic of organism's performance	Optimization criterion value $f(\boldsymbol{x}^k)$
Selection, choosing the fittest individual(s) for the next generation	Operator **S**: $f(\boldsymbol{x}^k) \rightarrow$ min (max)

1. The heuristic also defines a set of removing rules R (to take out specific parts of $\boldsymbol{x}^k$ or the full vector). At this mutation stage, the solution vector $\boldsymbol{x}^k$ is divided into two subvectors: $\boldsymbol{x}^k_{\text{remainder}}$ and $\boldsymbol{x}^k_{\text{reject}}$. Another common heuristic, *insert*, defines a set of rules I to insert, consequently one by one, all $x_i \in \boldsymbol{x}^k_{\text{reject}}$ into $\boldsymbol{x}^k_{\text{remainder}}$. This is the *construction phase* of the mutator, which builds some new solution vectors $\boldsymbol{x}^{k+1}$ using the variation of the problem-specified criterion (Mole and Jameson 1976; Or 1976; Osman 1993; Mester 1999).

At the *mutation stage*, mutator $\boldsymbol{M}(R, I, \beta, \boldsymbol{x}^k)$ produces an offspring $\boldsymbol{x}^{k+1}$ from the parent $\boldsymbol{x}^k$. If the first offspring appears to surpass the parent, the mutator with the same parameters is applied again to the new parent, and so on. If the offspring does not surpass the parent, then to generate the new offspring, the algorithm uses the mutator with other parameters. After mutation, the vector $\boldsymbol{x}^{k+1}$ "is improved" by standard combinatorial procedures of order $O(n^2)$: (1) 2-Opt (Lin and Kernighan 1973), (2) Or-Opt (Or 1976), and (3) 1-interchange (Osman 1993).

This two-phase approach (mutation-improving) reflects the principles of solution diversification and upgrading (Rochat and Taillard 1995). We combine the last three improving procedures into one composite procedure (*Composite*). At the initial solution phase, Composite is applied five times. We refer to such an algorithm (multiple application of the Composite procedure starting from random initial points) as the *Multi-Start* procedure. In Table 2 we compare the solutions of standard TSP obtained by four different powerful heuristics: guided local search (GLS), simulated annealing (SA), tabu search (TS; for the comparison of these three algorithms, see Voudoris and Tsang 1999), and the ES-MPM algorithm proposed by Mester (1999, 2000, and unpublished results). In addition, we present for comparison also three simple heuristics: 3-OPT of Lin and Kernighan (1973), the Composite, and the Multi-Start (Table 2). ES-MPM is a two-phase algorithm that first produces an initial solution using the simple Multi-Start procedure and then moves to a more powerful, albeit less fast, ES-search (*ES-phase*). The presented benchmark clearly demonstrates that the ES-MPM algorithm provides quality solutions and is faster than other adaptive algorithms (GLS, SA, and TS).

Multipoint marker ordering as a TSP problem: The proposed algorithm of multipoint ordering employs two-point linkage data (see also Press *et al.* 1986; Weeks and Lange 1987; Falk 1992; Schiex and Gaspin 1997). Although this approach is usually considered as "preliminary ordering," the good quality of the maps produced by our version of the ES algorithm (see below) allows us to consider it not only as a complement to the more sophisticated multilocus maximum-likelihood (ML) ordering, but also, to some extent, as a competitor to ML algorithms (especially for a large number of marker loci and various complications like missing data, misclassification, etc.). We consider n markers enumerated arbitrarily by n coordinates $x_i \in \boldsymbol{x}$ and, for each $n - 1$ marker pairs (x_i, x_j), a "distance" ρ_{ij}. As ρ_{ij}, either pairwise recombination fractions r_{ij} or map distances d_{ij} (*e.g.*, in Haldane or Kosambi metrics) are employed.

Different criteria can be used to discriminate between competitive orders, for example, total distance measured as a sum of distances between consecutive adjacent markers or the total number of recombination events. These criteria are founded on a biologically reasonable assumption that the true order of a set of linked loci will be the one that minimizes the total length of the chromosomal map (Press *et al.* 1986; Weeks and Lange 1987; Falk 1992; Schiex and Gaspin 1997). In our model, the minimum of sum of distances between adjacent markers was applied as optimization criterion (OC),

$$\text{OC} = \sum_{ij}^{n} \rho_{ij}\delta_{ij}, \tag{1}$$

where $\delta_{ij} = 0$ or $\delta_{ij} = 1$ represents in the criterion only $u \leq n - 1$ distances out of all $n(n - 1)/2$ pairwise distances; $\rho_{ij}\delta_{ij} > 0$, $i = \overline{1, n-1}$; $j = \overline{2, n}$.

The program for simulations was written in Visual Basic 6.0. Monte Carlo testing experiments were conducted on a double-processor Pentium 3 (800 Mhz). To compare different situations, the following coefficient of *restoration quality* [proximity between the "true" (simulated) and estimated orders] was employed,

TABLE 2

Comparison of different heuristics and the ES-MPM algorithm on standard (51–318 points) TSP

N	Problem name	Best published solutions		GLS	Inaccuracy[a] (*I*, %) and executing time (*T*, sec) of the TSP solutions ES-MPM	SA	TS	3-Opt	Multi-Start	Composite
1	Ei1-51	426	*I*	0	0	0.73	0	5.9	2.0	3.4
			T	1.3	0.1	6.3	1.1	0.2	0.04	0.01
2	Eil-101	629	*I*	0	0	1.76	0	4.8	5.0	5.0
			T	5.0	1.3	33.3	61.4	0.2	0.2	0.04
3	Eil-76	538	*I*	0	0	1.21	0	3.5	4.3	4.7
			T	2.3	1.1	18	5.2	0.1	0.08	0.01
4	KroA-100	21,282	*I*	0	0	0.42	0	0	0.2	6.5
			T	0.7	0.6	37.4	21.4	0.12	0.3	0.06
5	KroA-150	26,524	*I*	0	0	1.86	0.03	8.4	5.2	4.8
			T	24		103.3	413	0.8	0.35	0.27
6	KroA-200	29,368	*I*	0	0	1.04	0.72	4.6	4.8	6.6
			T	187	34	229.4	776	4.3	0.3	0.9
7	KroC-100	20,749	*I*	0	0	0.8	0.25	4.5	4.3	7.7
			T	1.8	1.5	36.6	4.8	0.3	0.2	0.07
8	Lin-318	42,029	*I*	0	0	1.34	1.31	4.0	4.2	5.6
			T	335	245	829	2672	13.8	7.6	0.8

[a] Inaccuracy is employed as a score of the quality of the solution; it is presented as a deviation (%) of the obtained result (by the inspected method) from the best-known solution.

$$K_r = (n-1)/\sum_{i=1}^{n-1} |x_i - x_{i+1}|, \qquad (2)$$

where x_i is the digit code of the ith marker in the currently ordered marker sequence. Figure 1 illustrates a typical dependence of K_r on executing time using different heuristics.

SIMULATED DATA SETS

The data for analysis were produced using a pseudo-random generator. The simulation algorithm repeatedly generated a single-chromosome mapping population, F_2, for a chosen number of markers with:

1. Variation of recombination rates between adjacent markers along the chromosome;
2. defined proportion of dominant *vs.* codominant markers;
3. chosen proportion of missing data;
4. chosen proportion of markers with erroneous classification and level of errors; and
5. chosen mode of recombination interference for adjacent markers, Haldane, Kosambi, or arbitrary interference. In the last case, we define a few ranges of coincidence values and the probabilities to sample the coincidence values from these ranges (with even distribution of the coincidence values from the chosen range).

The following are the numerical values (ranges) of the main parameters in the majority of experiments:

1. The number of markers per chromosome: $m = 80$.

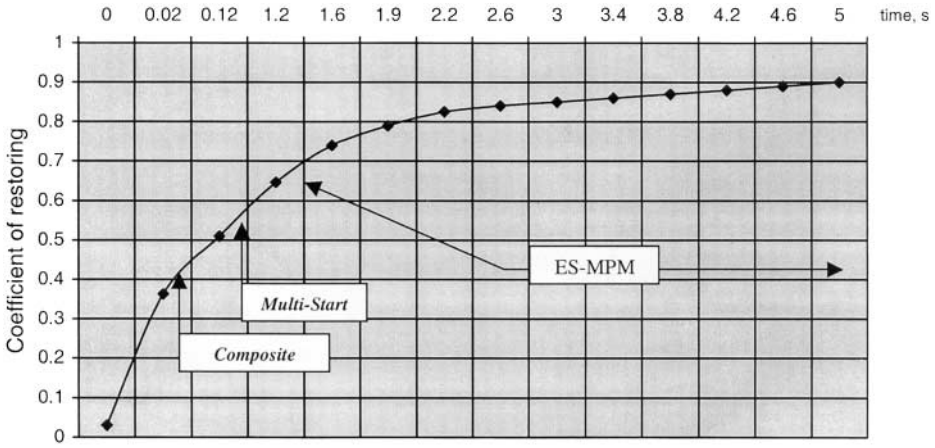


FIGURE 1.—Typical dependence of order quality (K_r) on executing time using Composite, Multi-Start, and ES algorithms (100-markers problem).

2. Probability distributions for distances between adjacent markers:
   $P(3 < d\ (\text{cM}) < 5) = 0.8$, $P(5 < d\ (\text{cM}) < 10) = 0.15$, $P(10 < d\ (\text{cM}) < 20) = 0.05$, with even distribution within each of the three ranges.
3. Proportions of codominant and dominant markers (in coupling phase, unless noted otherwise): 0.5 and 0.5, respectively.
4. Three levels for missing data: 0, 10, and 20%.
5. Two levels for the proportion of loci with classification errors, 0 and 40%, and in the last case, two levels of misclassification, 10 and 20%.
6. In the case of arbitrary interference, the distributions of coincidence coefficients: $P(0 < c < 1) = 0.6$ (positive interference), $P(1 < c < 2) = 0.2$ (slight-to-moderate negative interference), and $P(2 < c < I_c) = 0.2$ (moderate-to-strong negative interference), where $I_c = 5$, 10, 20, and 40.

Therefore, the efficiency of the preliminary multilocus ordering was considered upon complications caused by negative interference, erroneous marker scoring, and incomplete mapping information due to dominant markers and missing data, known to affect the quality of multipoint ordering. Motivation to consider such complications derives from the simple fact that in real mapping work no one can guarantee that the data are free of such complications. Moreover, in numerous previous attempts at building efficient multilocus ordering tools, some of these problems were usually ignored.

RESULTS

The considered types of disturbances (see the end of SIMULATED DATA SETS) proved to affect the quality of restoration of the true order of markers. These disturbances mainly caused local distortion of the order, *e.g.*, interchanging of two to three neighboring markers (referred to as "local disturbances"). There could be several inverted islands per linkage group. As expected, the number of these islands increases with the percentage of missing data, classification errors, and the level of negative interference. Less frequent were violations caused by excision of a large segment and its transposition to another place with or without inversion within the segment ("global disturbances"). Clearly, such violations result in an appreciable reduction of the coefficient of restoration quality (Equation 2).

Dominance: When all dominant markers were in coupling phase, the proportion of dominant and codominant markers had no effect on the quality of marker ordering. For three proportions of dominant markers (50, 66, and 100%) with Kosambi, Haldane, and a slight negative interference, nearly full recovery of marker order was reached ($K_r \approx 0.997 \div 0.999$). A different result was obtained with dominant markers in repulsion phase. It appears that the higher the proportion of repulsion phase, the lower the quality of multilocus ordering (MESTER *et al.* 2003). The employing of the ES-phase of the ES-MPM algorithm (see above, *Peculiarities of the combinatorial version of ES*) after getting some initial solution through Multi-Start positively affected the quality of the final solution. It is noteworthy that the application of ES-phase also stabilizes the ordering results (as displayed by the reduction of σ_{Kr}, the standard deviation of K_r between the Monte Carlo experiments). High precision of ordering in the coupling-phase data and low precision in the repulsion-phase data justify splitting the data into two sets, each with coupling-phase markers only and generating two complementary maps for each linkage group (KNAPP *et al.* 1995; PENG *et al.* 2000; MESTER *et al.* 2003). Clearly, the next step should be integration of the two maps. The last step may encounter difficulties caused by local and global map disturbances affecting codominant markers common for both maps, if the density of such codominant markers is relatively low (*e.g.*, in cases when codominant markers serve as anchors). In fact, the availability of shared codominant markers enables mutual control during multilocus ordering, which, together with computing-intensive jackknife and bootstrap techniques (EFRON 1979), significantly improves the quality of the resulting map (MESTER *et al.* 2003).

TABLE 3

Effect of negative interference on the quality of multilocus ordering

	Initial solution by Multi-Start		Improved solution by ES-phase		
I_c	K_r	σ_{Kr}	K_r	σ_{Kr}	N_{ES}(%)
5	0.944	0.150	0.993	0.013	9
10	0.926	0.160	0.985	0.018	10
20	0.900	0.129	0.937	0.043	11
40	0.866	0.125	0.901	0.056	14

I_c is the maximum value of the coincidence coefficient for cases of negative interference; as noted in the description of the simulation procedure, $P(2 < c < I_c) = 0.2$ (moderate-to-strong negative interference; in more detail, the analyzed situations are described in SIMULATED DATA SETS). Note the increased stability of ordering owing to application of the ES-phase of the ES-MPM algorithm (displayed in a substantial reduction in the standard deviation, σ_{Kr}, of the coefficient of restoration quality, K_r. Here and in the following tables, N_{ES} is the proportion of cases (Monte Carlo runs) where application of the ES-phase after the Multi-Start procedure improved the solution.

Negative interference: As expected, negative interference complicates the ordering problem that is manifested in reduction of K_r (Table 3). However, the decline in K_r with an increase in the maximum value I_c of coefficient of coincidence c is unexpectedly slow, pointing to robustness of the employed ordering procedure. A detailed anatomy of misordered situations shows that

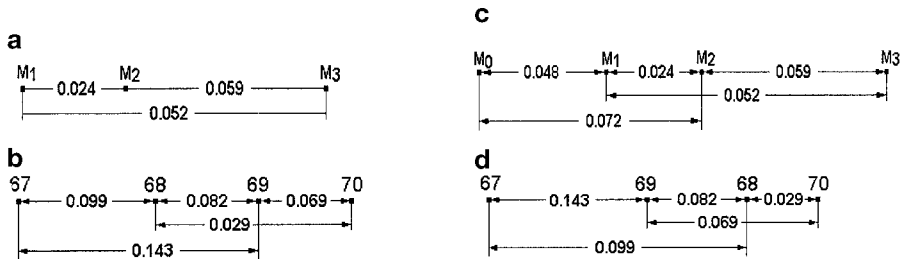


FIGURE 2.—Local disturbances of the order due to negative interference.

deviations from the true order in such cases are due mainly to interchanges of adjacent markers. The relatively low effect of negative interference can be explained by a stabilizing role of the neighboring intervals. This can be illustrated by the following example (Figure 2), in which recombination rate between the flanking markers M_1 and M_3 is smaller than that for the subinterval M_2–M_3 (see Figure 2a).

Without taking into account the information from neighboring intervals, the criterion "minimum of total distance between markers" will give the local order M_2–M_1–M_3 (Figure 2b) that differs from the true one. The stabilizing effect of the neighbor M_0 allows us to obtain the true order. Indeed, the optimization criterion value for the true order (Figure 2c) is OC = 0.048 + 0.024 + 0.059 = 0.131, whereas the order corresponding to the foregoing inversion between M_1 and M_2 (Figure 2d) caused by negative interference results in OC = 0.072 + 0.024 + 0.052 = 0.148. Therefore, despite high negative interference on interval M_1–M_2–M_3 (c = 15.8), which violates the rule that "the entire entity is supposed to be larger than its parts," the algorithm recovers the true order.

Misclassification: Errors in marker scoring inflate recombination distances and can also violate the principle, the entire entity is supposed to be larger than its parts by imitating "negative interference." This is why some mapping packages allow for error filtration by selecting out double recombinants. Our simulations showed that in the majority of such local violations the true order could be recovered due to the stabilizing effect of the neighboring markers (Table 4).

In a typical example (Table 5) with a maximum level of noise (20% of marker scoring errors were simulated for 40% of marker loci), there were 14 pairs of adjacent intervals (out of 49 possible pairs) in which either $r_{i,i+1}$ or $r_{i+1,i+2}$ was larger than $r_{i,i+2}$, but in only 4 of these pairs the true order could not be recovered. We conclude from the obtained results that despite the biases in pairwise estimates of recombination rates and inflation of the map length, the employed criteria of ordering are fairly robust to errors in marker scoring, unless the errors occur on a catastrophic level (say, at half of the loci and with a rate $\geqslant$20%).

Missing data: The results presented in Table 6 for several levels of missing marker scores (m = 0, 10, and 20%) show the same tendencies as those found for complicating factors considered above. Thus, participation of the ES-phase in the optimization procedure increased the precision of ordering (reducing the deviation of K_r from unity) and stabilized the ordering among Monte Carlo runs (as displayed in reduction of σ_{Kr}).

Comparison with multilocus algorithms: The foregoing results illustrate the advantages of the ordering procedure on the basis of minimization of the total length of the map (sum of recombination rates or distances between consecutive pairs of markers). Combined with our novel, highly efficient method of discrete optimization, a unique performance and rather high robustness with respect to various disturbances (like classification errors, negative interference, and missing data) are provided. It is noteworthy that ordering 100, 200, 400, and 800 markers takes ~1.3 sec, 14 sec, 2 min, and 9 min on a Pentium-4 2.0-GHz computer in the most complicated of the aforementioned situations. Note that even better performance was found in the first trials of our new optimizer based on guided evolution strategies (GES): on the same computer, map ordering for the foregoing variants proved fivefold (!) faster (MESTER and BRAYSY 2003). It would be of interest to compare

TABLE 4

Effect of marker misclassification (f_m) on the quality of multilocus ordering

	Initial solution by Multi-Start		Improved solution by ES-phase		
f_m (%)	K_r	σ_{Kr}	K_r	σ_K	N_{ES} (%)
Haldane mapping function					
0	0.938	0.162	0.997	0.008	8
10	0.908	0.153	0.966	0.027	23
20	0.764	0.144	0.843	0.058	66
Kosambi mapping function					
0	0.915	0.187	0.999	0.004	9
10	0.901	0.167	0.970	0.026	15
20	0.772	0.157	0.860	0.069	57

Note that the proportion of cases in which application of the ES-phase after the Multi-Start procedure improved the solutions (N_{ES}) increased severalfold for the nonzero level of misclassification.

TABLE 5

Effect of violations of the principle "entire is larger than its parts" caused by typing errors (20% at 40% of loci) and "self-correction" of the order owing to the stabilizing role of adjacent markers

True order	c or d	r_{12}	r_{23}	r_{13}	Resulting order	Sign
3–4–5	c-c-c	0.125	0.197	0.123	3–4–5	+
5–6–7	c-c-c	0.163	0.114	0.086	5–7–6	−
12–13–15	c-c-d	0.127	0.291	0.247	12–13–15	+
13–15–16	c-d-c	0.291	0.227	0.254	13–15–16	+
17–18–19	c-c-c	0.174	0.148	0.092	17–18–19	+
23–24–25	c-d-c	0.196	0.190	0.131	23–24–25	+
25–26–28	c-c-c	0.145	0.157	0.078	25–26–28	+
32–33–34	c-d-c	0.249	0.216	0.216	32–34–33	−
35–36–37	c-c-d	0.270	0.196	0.166	35–37–36	−
39–40–41	d-c-d	0.184	0.274	0.258	39–40–41	+
40–41–42	c-d-c	0.274	0.240	0.265	40–41–42	+
42–43–45	c-d-c	0.178	0.300	0.210	42–43–45	+
43–45–46	d-c-d	0.300	0.119	0.231	43–45–46	+
50–51–53	c-c-c	0.152	0.259	0.188	51–50–53	−

c and d denote codominant and dominant markers, respectively; r_{12}, r_{23}, and r_{13} are recombination rates between markers within a triad 1, 2, and 3; recovering of the true order despite violation is denoted by "+," whereas "−" denotes distorted order.

our algorithm with other procedures, like those of Ott (1991) and Lander and Green (1987). Ott (1991) proposed a criterion on the basis of sliding summation of three-locus LODs along the chromosome. This criterion was compared with the foregoing OC criterion (see Equation 1), using our optimization tools, on the basis of 10 Monte Carlo samples. The simulated F_2 data were for a 100-marker map (total length 500–600 cM), population size $n = 200$ with a very high noise caused by misclassification (40% of markers were simulated with 20% of typing errors!). The pairwise comparison shows (Table 7) that OC does invariably better than S_{LOD} (higher values of the coefficient of restoration K_r were obtained for OC).

TABLE 6

Effect of the missing data proportion (m) on the efficiency of multilocus ordering

	Initial solution by Multi-Start		Improved solution by ES-phase		
m (%)	K_r	σ_{K_r}	K_r	σ_{K_r}	N_{ES} (%)
Haldane mapping function					
0	0.938	0.162	0.997	0.008	8
10	0.953	0.143	0.992	0.013	16
20	0.917	0.158	0.974	0.023	17
Kosambi mapping function					
0	0.915	0.187	0.999	0.004	9
10	0.927	0.172	0.996	0.009	14
20	0.926	0.154	0.981	0.020	14

m (%), percentage of missing data.

To compare the efficiency of the OC criterion (Equation 1) with the multilocus-likelihood method, MapMaker 3.0 software was employed in a simulated data set of 200 markers with high negative interference in several regions. First, we revealed on the simulated map all islands where for three consecutive markers i, $i + 1$, and $i + 2$, either $r_{i,i+1}$ or $r_{i+1,i+2}$ was larger than $r_{i,i+2}$. For each such island, three "windows" involving 5, 7, and 9 markers, respectively, were analyzed using MapMaker. Simultaneously, the entire set of 200 markers was ordered with our program. Despite the fact that a local order that one could derive by comparing multilocus likelihoods for all possible candidate orders of such local neighborhoods (of 5, 7, or 9 markers) cannot be considered as a final solution, it makes sense to compare the local properties of the MapMaker solutions and those of the OC-based procedure (with OC defined by Equation 1). This is especially important for situations in which the natural condition $r_{i,i+1}$ and $r_{i+1,i+2} < r_{i,i+2}$ is

TABLE 7

Pairwise comparison of the ordering criterion OC and S_{LOD} for 10 Monte Carlo samples

N_{run}	1	2	3	4	5	6	7	8	9	10
K_r (OC)	0.81	0.89	0.82	0.92	0.82	0.68	0.83	0.85	0.96	0.95
K_r (S_{LOD})	0.55	0.31	0.42	0.78	0.71	0.64	0.76	0.62	0.71	0.75

K_r is the coefficient of restoration, whereas OC and S_{LOD} denote our criterion (Equation 1) and the criterion based on sliding summation of three-locus LODs.

violated, causing the highest instability of the result under sampling variation, *e.g.*, by using jackknife or bootstrap procedures. It should be noted, however, that application of these last techniques seems impossible with MapMaker for ~100 and more markers because of CPU limitations. The following are the details of this comparison. The simulated data of 200 markers included (a) for 95% of intervals $L = 0.75$ cM, for 2.5% $L = 30$ cM, and for the remainder 2.5% $L = 60$ cM; (b) 80% of the markers dominant in coupling phase, and 20% codominant; (c) population size $N = 400$; and (d) interference, with probability $P = 0.6$, $c \in (0, 1)$, with $P = 0.2$, $c \in (1, 2)$, with $P = 0.2$, $c \in (2, 20)$.

This example included 10 3-marker islands with violation of the condition $\{r_{i,i+1}$ and $r_{i+1,i+2} < r_{i,i+2}\}$. In addition to negative interference or classification errors, such a violation may derive from sampling fluctuations, especially when two adjacent intervals are of very different lengths. At 8 out of 10 such islands, our algorithm recovered the true order (the entire solution for 200 markers took <1 sec). MapMaker recovered the true order in 5 out of 10 islands on the basis of the 5-marker window; the remaining 5 islands were treated using the 7-marker window and recovered the true order in an additional 3 islands, and the last 2 were treated using the 9-marker window with a 50% success. The last two tasks took 6 hr.

Possibilities to validate the solution: Clearly, the foregoing comparisons using simulated data are only to illustrate the quality of the solution provided by the simple OC-based procedure. In dealing with real data, one needs some tools to validate the obtained order, and it is hard to choose the solution from several (sometimes dozens) candidate solutions (like those provided by MapMaker). To cope with this problem, some authors proposed computing-intensive procedures based on various combinations of jackknifing and bootstrapping (Efron 1979; Mott *et al.* 1993; Wang *et al.* 1994; Liu 1998). With a sufficiently large number of markers, the feasibility of such analysis strongly depends on the performance of the ordering algorithm employed and the quality of solution. We believe that our algorithm perfectly fits both of these demands: its high performance allows us to conduct the ordering procedure many times under different jackknife or bootstrap iterations of the initial sample (Figure 3).

The first step is ordering of markers using the whole set of data. To validate (or correct) the obtained map, the following analysis is conducted on the basis of a large series of jackknife runs (*e.g.*, 1000–10,000). In each run based on a subsample of individuals (*e.g.*, 90%), we first order the markers and for each marker determine its two (left and right) neighbors. Then, for each marker, the frequency distribution of its closest left and right neighbors is calculated and the *unstable neighborhoods* are detected using the entire set of generated jackknife runs. Such cases are classified according to

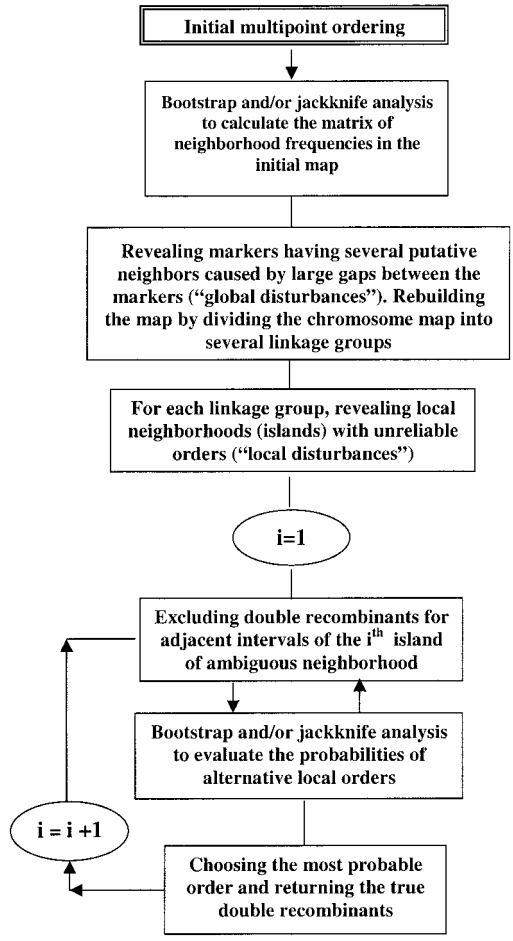


Figure 3.—Scheme of the algorithm for map verification.

the putative causal factor of instability: (a) double recombinants in adjacent intervals (resulting from negative interference or misclassification) and (b) sampling variation of recombination in large intervals. The first problem can be treated by taking out marker scores qualified as "double recombinants" (without affecting the scores of other markers of the same individual). The revised data set is reanalyzed by a repeated ordering procedure using the same jackknife approach. If the instability was caused by the second problem, one could split the map into two linkage groups until additional markers are available for the revealed gap. The following simulated example illustrates the application of the algorithm. The simulated data of 100 codominant markers included: (i) for 80% of intervals, $L \in (5, 10)$ cM, and for 20% $L \in (10, 20)$ cM; (ii) population size $N = 300$; and (iii) interference, with $P = 0.6$, $c \in (0, 1)$; with $P = 0.2$, $c \in (1, 2)$; and with $P = 0.2$, $c \in (2, 20)$.

Each jackknife run employed 275 (92%) individuals at both steps: initial ordering and validation were based on a revised data set. One thousand runs were analyzed. A typical fragment of the matrix characterizing the stability of neighborhoods is shown in Table 8a. It can be

TABLE 8

A fragment of the matrix of neighborhood frequencies based on the jackknife procedure

Marker	65	66	67	68	69	70	71	72	73	74
a. Initial data set										
64	1									
65		1								
66	1		1							
67		1		0.737	0.263					
68			0.737		0.993	0.263	0.007			
69			0.263	0.993		0.744				
70				0.263	0.774		0.993			
71				0.007		0.993		1		
72							1		1	
73								1		1
74									1	
b. After removing marker 69										
64	1									
65		1								
66	1		1							
67		1		1						
68			1			1				
70							1			
71						1		1		
72							1		1	
73								1		1
74									1	
c. After removing marker 69; scores qualified as double recombinants										
64	1									
65		1								
66	1		1							
67		1		0.974	0.026					
68			0.974		0.999	0.026	0.001			
69			0.026	0.999		0.975				
70				0.026	0.975		0.999			
71				0.001		0.999		1		
72							1		1	
73								1		1
74									1	

Multilocus ordering was conducted using the sum of recombination rates along consecutive pairs of adjacent markers.

easily seen from this fragment that two local orders are possible for this part of the map, $\mathbf{s}_1$ = (67, 68, 69, 70) and $\mathbf{s}_2$ = (67, 68, 69, 70), with probabilities $P(\mathrm{OC}(\mathbf{s}_1) > \mathrm{OC}(\mathbf{s}_2)) = 0.737$ and $P(\mathrm{OC}(\mathbf{s}_1) < \mathrm{OC}(\mathbf{s}_2)) = 0.263$. The recombination rates for the two orders calculated on the initial data set are shown in Figure 4. Thus, the OC values are $\mathrm{OC}(\mathbf{s}_1) = 0.099 + 0.082 + 0.069 = 0.250$ and $\mathrm{OC}(\mathbf{s}_2) = 0.143 + 0.082 + 0.029 = 0.254$. Therefore, on the basis of OC values, one will choose the true order $\mathrm{OC}(\mathbf{s}_1)$, but it is clear that for another sample $\mathrm{OC}(\mathbf{s}_2)$ may be selected as well, due to sampling variation of recombination rates. This is why it is important not only to detect such questionable neighborhoods, but also to evaluate the probabilities of the local competitive orders. The same is true for any other ordering criterion, *e.g.*, maximum likelihood, because $P(L(\mathbf{s}_1) > L(\mathbf{s}_2))$ is also *a priori* unknown. For our numerical example, the probabilities of the compared alternative orders do not

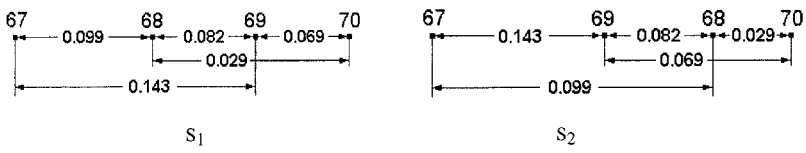


FIGURE 4.—Two most probable local orders for markers 67, 68, 69, and 70 of the simulated example with negative interference (see Table 8).

differ significantly, so that further steps are needed to obtain a solution with higher confidence.

The simplest way to improve the quality of the solution is to remove the questionable marker (Mott *et al.* 1993; Liu 1998). In our example, the source of the difficulties in the island 67–70 was marker 69 that inflated fivefold the size of the spanning interval 68–70. After removing marker 69, the jackknife procedure was applied 1000 times again with results shown in Table 8b. Thus, by deleting the problematic marker, one can obtain an unequivocal local ordering.

A more complex approach is based on temporal exclusion of marker scores considered as double recombinants (without affecting other markers of the same individuals; see Figure 3). This increases the probability of recovering the true order by excluding (albeit artificially) the local violations of the condition $r_{i,i+1}$ and $r_{i+1,i+2} < r_{i,i+2}$. After such editing of the data, we again applied the jackknife procedure. In the above example, after 1000 runs we obtained the result shown in Table 8c. Thus, as expected, removing double recombinants resulted in an increased stability of the derived ordering: the weakest connection between the neighbors in the locality 67–68–69–70 increased from $P = 0.737$ up to $P = 0.974$. Note that after removing double recombinants, $r_{68–69}$ and $r_{69–70}$ decreased from 0.082 and 0.069 to 0.025 and 0.011, respectively, and $r_{68–70} = 0.029$. Therefore, the condition $r_{i,i+1}$ and $r_{i+1,i+2} < r_{i,i+2}$ is not violated anymore. The same procedure could be applied to test the second local order, namely 67–69–68–70.

An example of application to real data: We employed the proposed approach to recently published mapping data on the maize *Intermated B73 × Mo17 (IBM) population* (Lee *et al.* 2002). For demonstration, the first chromosome (with 231 markers) was chosen from the *Map* database (www.maizemap.org/ibm_frameworkmaps.htm, framework_302.xls file). In our treatment of this data set, several questions that could be addressed during the map construction and its validation based on jackknife were of interest: (a) to reveal the map segments with stable neighborhoods ($P = 1$ for each pair of adjacent markers) that fully coincide with the published map (Lee *et al.* 2002); (b) to reveal the map segments with neighborhood probability higher than some threshold (*e.g.*, $P = 0.90$ or 0.95) that coincide with the published map; (c) to reveal the map segments with neighborhood probability higher than some threshold ($P = 0.90$ or 0.95) that do not coincide with the published map; (d) to demonstrate alternative (competitive) orders of the same region with unreliable neighborhoods (*i.e.*, with neighborhood probability lower than the threshold) that could be resolved by excluding 1–2 markers to fit the conditions b or c; and (e) revealing the segments of the map for which an exclusion of a larger group of markers (*e.g.*, ≥2) is needed to fit the conditions b, c, or d for the remaining subgroups.

For simplification of presentation of the results, the "natural" marker numbers as they are represented in the Excel data file (see also Lee *et al.* 2002) were used as a reference ordering. To address the foregoing questions (a–e), we employed our ordering algorithms for map construction and jackknife resampling procedure to test the reliability of the resulting orders (using 100 jackknife runs with sampled proportion of 90% of genotypes at each run). Following are the obtained results. First, marker 24 showed no close linkage with any of the remaining 230 markers; hence it was excluded from the map. The remaining marker groups were classified with respect to the jackknife test of neighborhood stability:

1. Regions with stable ($P = 1$) neighborhoods that fully coincide with the published IBM map were: 1–14, 37–48 (without marker 39 that was linked closer to other markers of chromosome 1; see below), 52–60, 81–84, 89–91, 112–115, 120–125, 128–133, 140–153, 205–208, and 218–221.
2. Segments with neighborhood probability $> P = 0.90$ that coincide with the published map were: 18–25 (but without marker 24), 28–33, 63–75, 160–165, 168–174, and 221–231.
3. Segments with neighborhood probability $> P = 0.9$ that did not coincide with the published map; the revised orders were: (1) 174–176–175–177; (2) 179–181–180, 184–185–186–188; (3) 204–202–201–205; and (4) 214–216–215–217.
4. Islands with simple unresolved alternatives; for resolution (*i.e.*, to reach the foregoing conditions b or c) it is necessary to exclude 1–2 markers:
   i. 15–16–17–18 with $P(15–16) \approx 0.62$ *vs.* 15–17–16–18 with $P(15–17) \approx 0.38$: after marker 17 is excluded, we obtain $P(15–16) = P(16–18) = 1$.
   ii. 25–27–26–39–28 with $P(25–27) \approx 0.72$ *vs.* 25–26–27–39–28 with $P(25–26) \approx 0.28$: after marker 27 is excluded, we obtain $P(25–26) = p(26–39) = p(39–28) = 1$. Resolving this situation has also improved the stability of the foregoing group 28 ÷ 33 that can be moved now from the set of groups with $P = 0.9$ to the set of fully stable ones with $P = 1$.
   iii. Stabilization of group 28 ÷ 33, in its turn, caused an improvement for group 33 ÷ 36. Instead of the initial dichotomy, 33–34–35–36 with $P(33–34) \approx 0.78$ *vs.* 33–35–34–36 with $p(33–35) \approx 0.22$, we obtained $P(33–34) = 0.96$, $P(34–35) = 1$, and $P(35–36) = 0.96$.
   iv. 48–49–51–50–52 with $P(48–49) \approx 0.84$ *vs.* 48–50–51–49–52 with $P(48–50) \approx 0.16$. By excluding marker 51, we can get $P(48–49) = 0.93$, $P(49–50) = 1$, and $P(50–52) = 0.93$.
   v. 60–62–61–63 with $P(60–62) \approx 0.62$ *vs.* 60–61–62–63 with $P(60–61) \approx 0.38$. By excluding marker 61, we obtain $P(60–62) = P(62–63) = 1$.
   vi. 75–78–77–76–79–80–81 with $P(75–78) \approx 0.6$ *vs.*

75–79–80–76–77–78–81 with $P(75–79) \approx 0.4$. After excluding markers 76 and 77, we obtain $P(75–78) = 0.95$, $P(78–79) = 0.99$, $P(79–80) = 1$, and $P(80–81) = 0.95$.

vii. 84–85–86–88–87–89 with $P(84–85) \approx 0.5$ *vs.* 84–86–85–88–87–89 with $P(84–86) \approx 0.5$. Excluding markers 86 and 87 results in $P(84–85) = P(85–88) = P(88–89) = 1$.

viii. 115–116–117–118–119 with $P(115–116) \approx 0.55$ *vs.* 115–117–118–116–119 with $P(115–117) \approx 0.45$. After 116 is excluded, $P(115–117) = P(117–118) = P(118–119) = 1$.

ix. 125–126–127–128 with $P(125–126) \approx 0.56$ *vs.* 125–127–126–128 with $P(125–127) \approx 0.44$; exclusion of marker 126 gives $P(125–127) = P(127–128) = 1$.

x. 133–134–135–136 with $P(133–134) \approx 0.69$ *vs.* 133–135–134–136 with $P(133–135) \approx 0.31$; exclusion of marker 134 gives $P(133–135) = P(135–136) = 1$.

xi. 136–138–137–139 with $P(136–138) \approx 0.66$ *vs.* 136–137–138–139 with $P(136–137) \approx 0.34$; exclusion of marker 138 gives $P(136–137) = P(137–139) = 1$.

xii. 153–155–154–157–156–158–159–160 with $P(153–155) \approx 0.55$ *vs.* 153–154–155–157–156–158–159–160 with $P(153–154) \approx 0.45$; exclusion of markers 154 and 155 gives $P(153–157) = 0.94$, $P(157–156) = 0.97$, $P(156–158) = 0.97$, $P(158–159) = 0.97$, and $P(159–160) = 1$.

xiii. 165–167–166–168 with $P(165–167) \approx 0.55$ *vs.* 165–166–167–168 with $P(165–166) \approx 0.45$; exclusion of marker 166 gives $P(165–167) = 1$, $P(167–168) = 0.95$.

xiv. 180–182–183–184 with $P(180–182) \approx 0.56$ *vs.* 180–183–182–184 with $P(180–183) \approx 0.44$; exclusion of marker 183 gives $P(180–182) = P(182–184) = 1$.

xv. 208–210–211–209–212 with $P(208–210) \approx 0.65$ *vs.* 208–209–210–211–212 with $P(208–209) \approx 0.35$; exclusion of marker 209 gives $P(208–210) = 1$, $P(210–211) = 0.99$, and $P(211–212) = 1$.

5. Segments of the map for which an exclusion of a larger group of markers (≥2 markers) is needed fit conditions b, c, or d for the remaining subgroups: (i) 91–112 and (ii) 187–200.

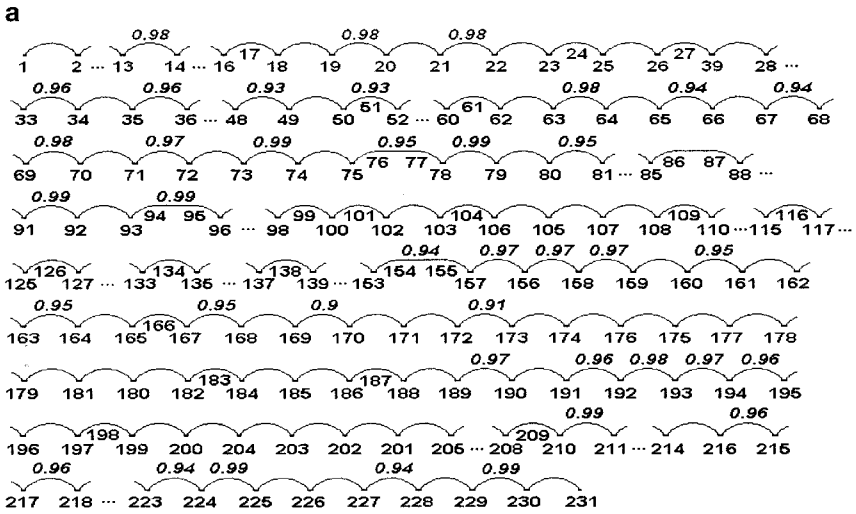
i. In the first group, $P(92–93) = 0.7$, $P(94–95) = 0.54$, $P(95–96) = 0.3$, $P(96–97) = 0.72$, $P(97–98) = 0.45$, $P(98–99) = 0.39$, $P(99–100) = 0.83$, $P(100–101) = 0.14$, $P(101–102) = 0.38$, $P(102–103) = 0.99$, $P(103–104) = 0.17$, $P(104–105) = 0.91$, $P(105–106) = 0.5$, $P(106–107) = 0.00$, $P(107–108) = 1$, $P(108–109) = 0.43$, $P(109–110) = 0.95$, $P(110–111) = 0.55$, and $P(111–112) = 0.88$. By excluding markers 94, 95, 99, 101, 104, and 109, we obtained $P = 1$ for pairs 92–93, 96–97, 97–98, 98–100, 100–102, 102–103, 103–106, 106–105, 105–107, 107–108, 108–110, and 110–111, and $P = 0.99$ for 93–96.

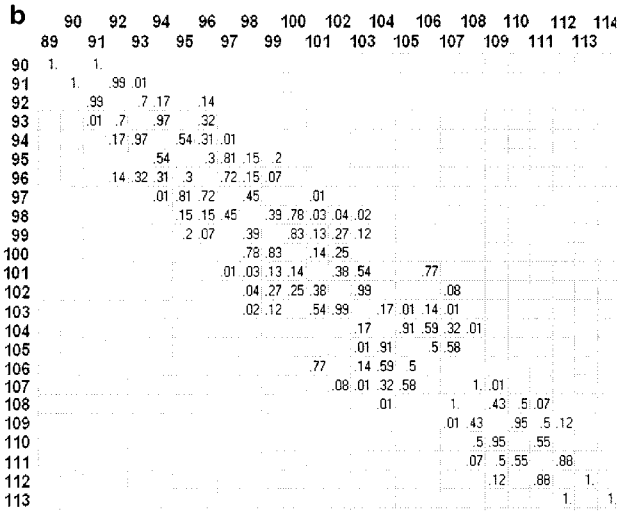
ii. In the second group, $P(187–188) = 0.47$, $P(188–189) = 0.61$, $P(189–190) = 0.9$, $P(190–191) = 0.99$, $P(191–192) = 0.41$, $P(192–193) = 1$, $P(193–194) = 0.66$, $P(194–195) = 0.5$, $P(195–196) = 1$, $P(196–197) = 0.75$, $P(197–198) = 0.5$, $P(198–199) = 0.61$, $P(199–200) = 0.83$. After excluding markers 187 and 198 we obtained $P(188–189) = 1$, $P(189–190) = 0.97$, $P(190–191) = 1$, $P(191–192) = 0.96$, $P(192–193) = 0.98$, $P(193–194) = 0.97$, $P(194–195) = 0.96$, and $P = 1$ for pairs 195–196, 196–197, 197–199, and 199–200.

The results of these manipulations are presented in Figure 5. It is noteworthy that the obtained map differs from the published one (see Lee *et al.* 2002). In the new version of the map, which was recently presented in www.maizemap.org/ibm_frameworkmaps.htm, the authors have deleted 40 markers, whereas in our version only 28 markers are deleted. The foregoing analysis allows us to suppose that our version is of a better quality compared to the revised map presented on the website.

DISCUSSION

This study is devoted to the problem of marker ordering in linkage groups with many dozens or hundreds of markers. We considered situations complicated by missing data, typing errors, high proportion of dominant markers, and high negative interference. The ordering problem belongs to the field of discrete optimization on a set of all possible orders (amounting to $n!/2$ for n loci). This formulation is quite similar to the well-known challenging TSP, and several authors attempted to employ the methods developed in the TSP for genetic mapping (Press *et al.* 1986; Weeks and Lange 1987; Falk 1992; Schiex and Gaspin 1997). New ES-optimization algorithms developed by Mester (1999, 2000, and unpublished results) significantly improved the quality of solution in the TSP field (see Table 2). Our simulation experiments showed that a need in optimization power provided by these ES-algorithms usually begins from ordering problems with >20 markers; with smaller-size problems the Composite algorithm seems to be sufficient. Composite is built from simple optimization procedures working faster than ES, but producing worse solutions. On all tested sizes of the ordering problem (50 and more), the ES algorithm provided the best solution after one to six evolutionary cycles. These results allowed us to define the threshold for the solution time (not more than six cycles) for the ES algorithm at different sizes of the problem. The advantage of ES over other selected algorithms of optimization, in particular simulated annealing (SA), as applied to combinatorial





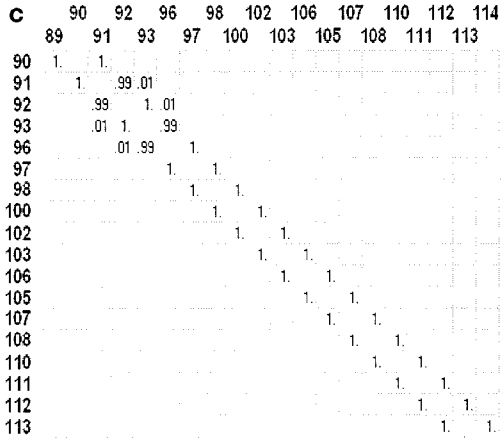


FIGURE 5.—Improving the reliability of multilocus marker ordering on the basis of results of jackknifing (example of maize chromosome 1): (a) The new order of markers after detecting and removing 26 markers that displayed unstable neighborhoods. The arcs represent stable ordered groups with $P = 1$ (not marked) or $P > 0.9$ (the estimated P is indicated above the arc), with the beginning and the end of the group marked by the marker number (the broken arc with marker numbers separated by ". . ." is to show a continuous series of markers with $P = 1$ for each pair of adjacent markers within the series; the numbers under the arcs are for deleted markers). (b and c) Fragments of the map (before and after removing problematic markers) for the group of markers 91–112 (for additional detail see description of this example in the text).

problems, can be clearly seen from Table 2. Therefore, it was quite natural to apply this fast and efficient approach for multilocus ordering. Applying the TSP-oriented methodology to mapping problems would be especially simple if, instead of multilocus likelihood, a faster criterion based on minimization of the total map length is employed. Combined with high performance of the optimization algorithm, this simplification allows

us to treat the problem systematically with verification of the obtained multilocus order on the basis of computing-intensive bootstrap and jackknife approaches.

To analyze the properties of derived maps under various complications (dominant *vs.* codominant markers, marker misclassification, alternating negative and positive interference, and missing data), simulated data with ~50–400 markers were employed, and the quality of the map was evaluated using a "coefficient of restoration" (based on comparison between the simulated and recovered orders). It appeared that the employed optimization criterion enabled us to achieve a very close proximity of the calculated orders to the simulated ones, despite missing data or misclassification. Two types of deviations from the true order were revealed: (i) local "inversion" usually involving adjacent markers and (ii) "excision" of a map fragment and its "insertion" (with or without inversion) to another map region. The first type of error is caused by violation of the condition $r_{i,i+1}$ and $r_{i+1,i+2} < r_{i,i+2}$ due to high negative interference or marker misclassification. The second type of error occurs mainly due to large gaps along the map (caused by low density of markers in some chromosomal regions).

To detect unreliable segments of the map, bootstrap and jackknife techniques could be employed. Unless the optimization procedure is highly efficient, the application of these approaches should be constrained to a relatively small number of markers due to CPU limitations. This is not the case with our ES-optimization algorithm: ordering of 100 markers takes ~0.2–1.5 sec on a Pentium 2-GHz computer. A further severalfold improvement in performance is expected by using our new optimizer based on guided evolution strategies (Mester and Braysy 2003). The diagnostic approach for detecting unreliable map regions, proposed in this article, differs in some aspects from other procedures [*e.g.*, from the bootstrap procedure described by Liu (1998)]. We employ an invariant description of marker orders on the basis of the notion of marker neighborhoods rather than marker map positions. Actually, such a consideration is closely related to our method of evaluation of restoration quality in simulated experiments, *i.e.*, proximity between the "true" (simulated) and recovered orders, which is independent of the specific coordinate system (*e.g.*, recombination rates or map positions). We believe that marker order is a much more objective indicator for comparison multipoint maps than map positions. Indeed, even with strict constancy of gene order within species, recombination rates (hence map positions) may widely fluctuate from experiment to experiment due to sampling variation, dependence on ecological conditions, sex, genotype, and age (Korol *et al.* 1994). Consequently, genetic mapping of any target trait, either qualitative or quantitative, through determining the marker brackets, will be less dependent on these fluctuations and more comparable across independent studies than a related effort based on map position in centimorgans. This is especially clear when the results of fine mapping serve as a starting point for map-based cloning. In such a case, what is really important is the information about close markers rather than precise map position.

There is also a technical advantage of using the marker orders rather than marker map positions (centimorgans) as final mapping results. Our verification procedure based on jackknife and bootstrap techniques reveals neighborhoods of questionable local ordering and enables us to detect the "weak connections" in the marker chain. If such a local "weakness" was caused by low marker density, one can split the map into two linkage groups and/or attempt to add new markers to fill the gap. In the case of an excess of double recombinants, the detected questionable marker scores can be removed from the data (without having to delete the marker entirely) with a subsequent reanalysis of the map, as in similar options available in other mapping tools (*e.g.*, MapMaker). This purifying operation may be sufficient to stabilize the resulting map, and it is reasonable if the questionable score derives from typing error (that can be tested by a repeated typing). However, there is some evidence that an excess of double recombinants may result from negative interference (Peng *et al.* 2000; Boyko *et al.* 2002; Esch and Weber 2002). Even then, such a treatment is useful as a diagnostic step, and after getting an idea of what factors caused the local problem, one may continue the analysis. For instance, it makes sense to deal with two versions of the defined region: one (purified) for mapping needs only and the other one for further in-depth study of the putative negative interference. Unlike many other procedures that remove double recombinants and conclude the analysis by recalculating the orders, in our case this step is complemented by reanalysis of the probabilities $P(\mathrm{OC}(\mathbf{s_1}) > \mathrm{OC}(\mathbf{s_2}))$ and $P(\mathrm{OC}(\mathbf{s_1}) < \mathrm{OC}(\mathbf{s_2}))$, thereby providing a direct tool for statistically justified decisions.

We should recall another reason to deal with two map versions simultaneously, which is related to the linkage phase of dominant markers, *i.e.*, *coupling vs. repulsion*. As shown above, higher precision of ordering coupling-phase dominant markers compared to repulsion-phase data justifies splitting the dominant marker data into two sets, each with the coupling phase only, and generating two maps for each linkage group (Knapp *et al.* 1995; Peng *et al.* 2000). Clearly, such a procedure should be followed by integration of the two maps. The availability of shared codominant markers enables mutual control during multilocus ordering (Mester *et al.* 2003), facilitating the integration that can be conducted by a proper algorithm (*e.g.*, Lalouel 1977; Stam 1993; Mester *et al.* 2003). Parallel calculation technology can easily be adopted for further expedition of the proposed algorithm.

The useful suggestions of G. Churchill and an anonymous reviewer are acknowledged with thanks. This study was supported by the Israeli Ministry of Absorption, the U.S. Agency for International Development Cooperative Development Research Program (grant TA-MOU-97-CA17-001), and the German-Israeli Cooperation Project [Deutsch-Israelische Projektkooperation project funded by the Bundesministerium für Bildung und Forschung (BMBF) and supported by BMBF's International Bureau at the Deutsch Zentrum Luft-und Raumfahrt].

LITERATURE CITED

Boyko, E., R. Kalendar, V. Korzun, J. Fellers, A. Korol *et al.*, 2002 A high-density cytogenetic map of the *Aegilops tauschii* genome incorporating retrotransposons and defense-related genes: insights into cereal chromosome structure and function. Plant Mol. Biol. **48:** 767–790.

Buetow, K. N., and A. Chakravarti, 1987 Multipoint gene mapping using seriation. Am. J. Hum. Genet. **41:** 189–201.

Curtis, D., and H. Gurling, 1993 A procedure for combining two-point lod scores into a summary multipoint map. Hum. Hered. **43:** 173–185.

Efron, B., 1979 Bootstrap method: another look at the jackknife. Ann. Stat. **7:** 1–26.

Ellis, T., 1997 Neighbour mapping as a method for ordering genetic markers. Genet. Res. **69:** 35–43.

Esch, E., and W. E. Weber, 2002 Investigation of crossover interference in barley (*Hordeum vulgare* L.) using the coefficient of coincidence. Theor. Appl. Genet. **104:** 786–796.

Falk, C. T., 1992 Preliminary ordering of multiple linked loci using pairwise linkage data. Genet. Epidemiol. **9:** 367–375.

Fogel, D., 1992 Evolving artificial intelligence. Ph.D. Thesis, University of California, San Diego.

Holland, J., 1975 *Adaptation in Natural and Artificial Systems.* MIT Press, Cambridge, MA.

Homberger, J., and H. Gehring, 1999 Two evolutionary metaheuristics for vehicle routing problem with time windows. INFOR **37:** 297–318.

Jansen, J., A. C. de Jong and J. W. van Ooijen, 2001 Constructing dense genetic linkage maps. Theor. Appl. Genet. **102:** 1113–1122.

Knapp, S. J., J. L. Holloway, W. C. Bridges and B. H. Liu, 1995 Mapping dominant markers using F2 mating. Theor. Appl. Genet. **91:** 74–81.

Korol, A. B., I. A. Preygel and S. I. Preygel, 1994 *Recombination Variability and Evolution.* Chapman & Hall, London.

Lalouel, J. M., 1977 Linkage mapping from pair-wise recombination data. Heredity **38:** 61–77.

Lander, E. S., and P. Green, 1987 Construction of multilocus linkage maps in human. Proc. Natl. Acad. Sci. USA **84:** 2363–2367.

Lander, E. S., P. Green, J. Abrahamson, A. Barlow, M. J. Daly *et al.*, 1987 MapMaker: an interactive computer package for constructing genetic linkage maps of experimental and natural populations. Genomics **1:** 174–181.

Lathrop, G. M., J. M. Lalouel, C. Julier and J. Ott, 1984 Strategies for multilocus linkage analysis in human. Proc. Natl. Acad. Sci. USA **81:** 3443–3446.

Lathrop, G. M., J. M. Lalouel, C. Julier and J. Ott, 1985 Multilocus linkage analysis in humans: detection of linkage and estimation of recombination. Am. J. Hum. Genet. **37:** 482–498.

Lee, M., N. Sharopova, W. D. Beavis, D. Grant, M. Katt *et al.*, 2002 Expanding the genetic map of maize with intermated B73×Mo17 (*IBM*) population. Plant Mol. Biol. **48:** 453–461.

Lin, S., and B. Kernighan, 1973 An effective heuristic algorithm for the TSP. Oper. Res. **21:** 498–516.

Liu, B. H., 1998 *Statistical Genomics: Linkage, Mapping, and QTL Analysis.* CRC Press, New York.

Mester, D., 1999 *The Parallel Algorithm for Vehicle Routing Problem With Time Windows Restrictions.* Scientific Report, Minerva Optimization Center, Technion, Israel.

Mester, D., 2000 *A Fast Evolutionary Algorithm for Vehicle Routing Problem.* Technical Report SYS-1/2000. Information and System Analysis Institute, University of Dortmund, Dortmund, Germany.

Mester, D., and O. Braysy, 2003 Guided evolution strategies for large scale vehicle routing problem with time windows. EURO/INFORMS, Join International Meeting, July 2003, Istanbul, Turkey.

Mester, D. I., Y. I. Ronin, Y. Hu, E. Nevo and A. B. Korol, 2003 Efficient multipoint mapping: making use of dominant repulsion-phase markers. Theor. Appl. Genet. **107:** 1102–1112.

Mole, R., and S. Jameson, 1976 A sequential route-building algorithm employing a generalized saving criterion. Oper. Res. **27:** 503–511.

Mott, R. F., A. V. Grigoriev, E. Maier, J. D. Hoheisel and H. Lehrach, 1993 Algorithm and software tools for ordering clone libraries: application to the mapping of the genome of *Schizosaccromyces pombe.* Nucleic Acids Res. **21:** 1965–1974.

Muhlenbein, H., M. O. Gorges-Scheuter and O. Kramer, 1998 Evolution algorithm in combinatorial optimization. Parallel Comput. **7:** 65–85.

Newell, R. W., R. Mott, S. Beck and H. Lehrach, 1995 Construction of genetic maps using distance geometry. Genomics **30:** 59–70.

Nissen, V., 1994 *Evolutionare Algorithmen.* Deutscher Universitats-Verlag, Wiesbaden, Germany.

Olson, J. M., and M. Boehnke, 1990 Monte Carlo comparison of preliminary methods of ordering multiple genetic loci. Am. J. Hum. Genet. **47:** 470–482.

Or, I., 1976 Traveling salesman-type combinatorial problems and their relations to the logistics of regional blood banking. Ph.D Thesis, Department of Industrial Engineering and Management Science, Northwestern University, Evanston, IL.

Osman, I., 1993 Metastrategy simulated annealing and tabu search algorithm for VRP. Ann. Oper. Res. **41:** 421–451.

Osman, I., 1995 An introduction of meta-heuristics, pp. 92–122 in *Operation Research Tutorial Papers,* edited by M. Lawerence and C. Wilson. Operational Research Society Press, Birmingham, UK.

Ott, G., 1991 *Analysis of Human Genetic Linkage.* John Hopkins University Press, Baltimore/London.

Peng, J., A. Korol, T. Fahima, S. Roder, Y. Ronin *et al.*, 2000 Molecular genetic maps in wild emmer wheat, *Triticum dicoccoides*: genome-wide coverage, massive negative interference, and putative quasi-linkage. Genome Res. **10:** 1509–1531.

Press, W. H., B. P. Flannery, S. A. Teucolsky and W. T. Vetterling, 1986 *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, London.

Rechenberg, I., 1973 *Evolutionstrategie.* Fromman-Holzboog, Stuttgart, Germany.

Rochat, Y., and E. Taillard, 1995 *Probabilistic Diversification and Intensification in Local Search for Vehicle Routing Problem.* Technical report CRT-95–13. Lausanne Federal Polytechnic School, Lausanne, Switzerland.

Schiex, T., and C. Gaspin, 1997 Carthagene: constructing and joining maximum likelihood genetic maps. ISMB **5:** 258–267.

Schwefel, H-P., 1977 *Numeriche Optimierung von Computer-Modelen Mittels der Evolutions-Strategie.* Birkauser, Basel, Switzerland.

Schwefel, H-P., 1987 *Collective Phenomena in Evolutionary System.* Interne Berichte und + +Skripten, Fachbereich Informatic, University of Dortmund, Dortmund, Germany.

Stam, P., 1993 Construction of integrated genetic linkage maps by means of a new computer package: JoinMap. Plant J. **3:** 739–744.

Thompson, E. A., 1984 Information gain in joint linkage analysis. IMA J. Math. Appl. Med. Biol. **1:** 31–49.

Voudoris, C., and E. Tsang, 1999 Theory and methodology: guided local search and its application to the traveling salesman problem. Eur. J. Oper. Res. **113:** 469–499.

Wang, Y., R. Prade, J. Griffith, W. Timberlake and J. Arnold, 1994 ODS_BOOTSTRAP: assessing the statistical reliability of physical maps by bootstrap resampling. Comput. Appl. Biosci. **10:** 625–634.

Weeks, D., and K. Lange, 1987 Preliminary ranking procedures for multilocus ordering. Genomics **1:** 236–242.

Wilson, S., 1988 A major simplification in the preliminary ordering of linked loci. Genet. Epidemiol. **5:** 75–80.

Communicating editor: G. Churchill