On the Complexity of DNA Physical Mapping

Martin Charles Golumbic
IBM Israel Scientific Center
Technion City, Haifa, Israel
and Bar-Ilan University
Ramat Gan, Israel.
email: golumbic@israearn.bitnet

Haim Kaplan   Ron Shamir
Department of Computer Science
Sackler Faculty of Exact Sciences
Tel Aviv University
Tel-Aviv 69978, Israel.
email: shamir@math.tau.ac.il

January 1993, revised April 1993

Abstract

The Physical Mapping Problem is to reconstruct the relative position of fragments (clones) of DNA along the genome from information on their pairwise overlaps. We show that two simplified versions of the problem belong to the class of NP-complete problems, which are conjectured to be computationally intractable. In one version all clones have equal length, and in another, clone lengths may be arbitrary. The proof uses tools from graph theory and complexity.

1 Introduction

Physical mapping [7, 25] is a central problem in molecular biology. In order to replicate and study a certain contiguous stretch of the DNA (a chromosome or a part of one), it must be cut into relatively small fragments called *clones*. In the process, the information on the order of these clones is lost. The goal of physical mapping is to reconstruct the relative position of the clones along the original stretch of DNA. A physical map is an essential part of most sequencing, gene locating and cloning projects. In fact, one of the main goals set for the first five years of the Human Genome Project is to obtain a detailed physical map of all human chromosomes [32].

There are various biological techniques for determining if two clones intersect. All of these techniques involve obtaining some *fingerprint* for each clone, and deciding that two clones intersect if their fingerprints are sufficiently similar. (A fingerprint can be, for example (1) a list of subfragment lengths obtained by digestion of the clone with restriction enzymes [8, 21, 26], or (2) a 0-1 vector denoting if the clone does or does not hybridize to each of a set of short synthetic oligonucleotides [24] or (3) the hybridization pattern of the clone with each of a set of probes, where each probe appears in a unique region of the genome [27, 19]). Ideally one would like to compute the intersection probability for each pair of clones from their fingerprints, and then, based on the set of pairwise probabilities, determine the most likely placement of all the clones.

In this note we show that even a simplified version of the physical mapping problem is very probably intractable. In this simpler problem we restrict the pairwise information so that for every two clones either

(1) we know that they must intersect, or

(2) we know that they must not intersect, or

(3) we do not know if they intersect or not.

We ask if there exists any solution (i.e., placement of the clones along a line) which satisfies these constraints. This is equivalent to attaching intersection probabilities 1,0, and 1/2 to cases (1), (2) and (3) respectively, and asking if there is a placement whose overall probability is positive.

We discuss this problem in two versions: (A) when fragment lengths may vary, and (B) when all fragments are assumed to have the same length. Biologically, the first case corresponds to 'libraries' of YACs (Yeast artificial chromosomes) [6, 9], in which the fragments may vary in length between 100K and over 1M bp (base pairs), or bacteriophage P1 libraries with 20Kbp- to 100Kbp-long clones [30]. Version (B) corresponds to experiments with phage [8, 26, 21] or cosmid clones [10, 31], in which all cloned fragments are approximately of equal length (40Kbp for cosmids, 15Kbp for phage Lambda). We show that both computational problems belong to the class of NP-complete problems, which are conjectured to be intractable. (For background on NP-completeness, see [13].)

We use graph theoretical concepts to formulate the problem. (For general background and terminology on graph theory, see, e.g., [16].) A graph $G(V, E)$ is called an *interval graph* if one can assign to each vertex $v \in V$ an interval on the real line $I(v)$, so that edge $(u, v) \in E$ if and only if the intervals $I(u)$ and $I(v)$ have a non-empty intersection. In that case, the set of intervals $\{I(v)|v \in V\}$ is called *an interval realization* for the graph G. The use of interval graph models in molecular biology dates back to Benzer's studies on the linearity of genes [4]. Waterman and Griggs [33] have shown that restriction mapping can be modeled by interval graphs.

An *interval sandwich problem* is defined by a vertex set V and a partition of the set of all edges of the complete graph on V into three sets $E^1 + E^0 + E^3$. Denote $E^2 = E^1 \cup E^0$. One asks whether there exists set of edges E such that $E^1 \subseteq E \subseteq E^2$ so that $G(V, E)$ is an interval graph. (G is called *a sandwich graph* since it is "sandwiched" between the graph $G^1 = (V, E^1)$ and its supergraph $G^2 = (V, E^2)$.) The problem data can be specified by (V, E^1, E^2) or (V, E^1, E^3).

To see the equivalence of the interval sandwich problem with the simplified physical mapping problem described above, assign a vertex to each clone, and assign edge (i, j) to E^1, E^3 or E^0 if the pair of clones satisfies case (1), (2) or (3), respectively. There exists an interval sandwich graph if and only if there is a placement of the clones satisfying all the input data.

The interval graph sandwich problem has been defined in [18] in the context of temporal reasoning, where it was shown to be NP-complete. A study of sandwich problems for other graph families was subsequently done in [17]. In section 2 we give a simpler proof of the NP-completeness of the interval sandwich problem, based on ideas from [17]. If we assume that all clones have equal length, then in the graph model this is equivalent to the requirement that all the intervals in the realization must have equal length. In Section 3 we give an NP-completeness proof for this special case too. Here we cannot use the simpler proof of section 2. (The NP-completeness of the former case does *not* imply the NP-completeness of the latter or vice versa:

A unit interval sandwich is in particular an interval sandwich, but the converse is not always true, so for a given input there may be an interval sandwich but no unit interval sandwich. Hence, there are instances for which the answers of the two problems will differ.) The proof we present is a careful modification of the reduction in [18], for the case of unit intervals. Section 4 discusses some open related problems.

There have been several statistical and algorithmic studies of the physical mapping problem. Lander and Waterman [22], Barillot et al [3] and Arratia et al [2] study probabilistic models for the distribution of islands (connected components in the map) as a function of experimental parameters and its implications for mapping project design. Alizadeh et al [1] have recently investigated the model of [22] for the special case of hybridization fingerprints with equal-length clones. Using heuristics for a combinatorial version of the problem, they obtained encouraging results with simulated data. They also prove NP-hardness of two variants of the problem. Goldstein and Waterman [15] have shown that restriction mapping is NP-complete. In [14] two other models of physical mapping are shown to be NP-complete: One model allows only false positive errors in determining intersections, and wishes to minimize their number. The other allows no errors but assumes that clones can actually be chimers (the union of two disjoint intervals), a common problem in YAC clones. All these hardness results are related but not equivalent to those presented here.

2 Interval Graphs

Given a set of intervals on the real line, one can define a partial order on the intervals by $a \prec b$ if and only if the interval a is completely to the left of interval b. We use this *interval order* to give a reduction from the following problem:

BETWEENNESS:

INPUT: A set of elements $S = \{a_1, \ldots, a_n\}$, and a set $T = \{T_1, \ldots, T_m\}$ of ordered triplets of elements from S, where $T_i = (a_{i_1}, a_{i_2}, a_{i_3})$ $i = 1, \ldots, m$.

QUESTION: Does there exist a one-to-one function $f : S \rightarrow \{1, 2, \ldots, n\}$ such that either $f(a_{i_1}) < f(a_{i_2}) < f(a_{i_3})$ or $f(a_{i_1}) > f(a_{i_2}) > f(a_{i_3})$ for $i = 1, \ldots, m$?

The problem was shown to be NP-complete by Opatrny [28].

Theorem 2.1 *The interval sandwich problem is NP-complete.*

Proof. Membership in NP follows since a given interval realization can be checked to fit the data in polynomial time. Given an instance of BETWEENNESS, we transform it to an interval sandwich instance as follows: Define a vertex v_i for each element a_i, and two vertices x_j^1 and x_j^2 for triplet T_j. The vertex set is $V = \{v_1, \ldots, v_n\} \cup \{x_j^1, x_j^2 | j = 1, \ldots, m\}$. The edges sets are

$$E^1 = \{(v_{i_1}, x_i^1), (x_i^1, v_{i_2}), (v_{i_2}, x_i^2), (x_i^2, v_{i_3}) \mid i = 1, \ldots, m\}$$

$$E^3 = \{(v_i, v_j) \mid i \neq j\} \cup \{(v_{i_1}, x_i^2), (v_{i_3}, x_i^1), (x_i^1, x_i^2) \mid i = 1, \ldots, m\}$$

In other words, to each triplet corresponds a 5-chain (see figure 1(a)) which should also appear as a 5-chain in the sandwich graph. In addition, all the v_i-s should form an independent set in that graph. The reduction is clearly polynomial.

The key to the reduction is the following simple observation: In any interval realization of a 5-chain (a, x, b, y, c), either $a \prec b \prec c$ or $c \prec b \prec a$ (see figure 1(b)).

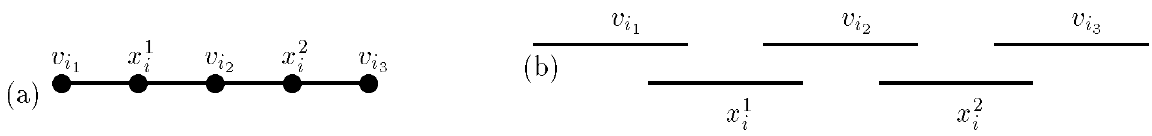

Figure 1: (a) The 5-chain $(v_{i_1}, x_i^1, v_{i_2}, x_i^2, v_{i_3})$. (b) An interval realization of the 5-chain.

Suppose there is an ordering f of the set S which satisfies the betweenness conditions. Create an interval realization for an interval sandwich as follows: Draw n disjoint intervals corresponding to the v_i-s in the same order as the a_i-s appear in f. That is, $I(v_i) \prec I(v_j)$ if and only if $f(a_i) < f(a_j)$. Since the order conditions on triplets are satisfied, additional intervals corresponding to x_i^1 and x_i^2 can be added for each triplet T_i such that the subgraph induced by the five vertices $\{v_{i_1}, v_{i_2}, v_{i_3}, x_i^1, x_i^2\}$ is a 5-chain. The resulting interval graph on V is a sandwich graph.

For the converse, suppose there exists an interval sandwich solution with realization I. Since in that graph the set $\{v_1, \ldots, v_n\}$ induces an independent set, a complete ordering on S is generated: Number the elements in the order of their intervals from left to right, namely, $f(a_i) <$

$f(a_j)$ if and only if $I(v_i) \prec I(v_j)$. Since for each triplet T_i the five vertices $\{v_{i_1}, v_{i_2}, v_{i_3}, x_i^1, x_i^2\}$ must induce a 5-chain in every sandwich graph, by the above observation, every triplet satisfies the betweenness condition. ■

Interestingly, we can also show that a restricted version of the above problem is NP-complete: A function $c : V \rightarrow Z$ is called a *coloring* of the graph $G = (V, E)$ if for every $(u, v) \in E$, $c(u) \neq c(v)$. The problem is defined as follows:

COLORED INTERVAL SANDWICH:
INPUT: A graph $G = (V, E)$ and a coloring c of G.
QUESTION: Does there exist a set E' so that $E \subseteq E'$, $G' = (V, E')$ is an interval graph and c is also a coloring for G'?

Notice that this is a special case of the sandwich problem where $E^1 = E$ and $E^3 = \{(u, v) \mid c(u) = c(v)\}$ is given implicitly by the coloring. The problem is also motivated by physical mapping: Suppose the data consists of sets of clones, where each set is obtained by *complete digestion* of the genome using one or more restriction enzymes (cf. [11]). Then all the clones in a set must be disjoint. In the graph presentation, their vertices will all be assigned the same color. If we use data only for positive intersections, and the non-intersections are implied from the partition into such sets, then the resulting decision problem corresponds to that defined above. Interestingly, the analogous problem where the property 'interval' is replaced by 'chordal' also arises in biology: It is a graph-theoretical reformulation of the perfect phylogeny problem in evolutionary trees, see [5] and [20].

Theorem 2.2 *The colored interval sandwich problem is NP-complete.*

Proof. Reduction from BETWEENNESS, where the vertex set and $E = E^1$ is defined as in Theorem 2.1 and the coloring is $c(x_i^1) = c(x_i^2) = i$ $(i = 1, \ldots, m)$ and $c(v_1) = c(v_2) = \cdots c(v_n) = m + 1$. A slight modification of the arguments in Theorem 2.1 proves the result. ■

Another proof of the result above has been recently given independently in [12].

3 Unit Interval Graphs

The reduction of the previous section does not carry over to the case of unit intervals: It is easy to make all the intervals $I(v_j)$ have the same length, but an interval $I(x_i)$ may properly contain one or many $I(v_j)$ intervals, as well as other $I(x_k)$ intervals. We need to work a little harder to prove the complexity of this problem:

Theorem 3.1 *The unit interval sandwich problem is NP-complete*

Proof. The problem is clearly in NP since a given unit interval realization can be verified to fit the input in polynomial time. We give a reduction from the following problem, which was shown to be NP-complete by Schaefer [29]:

> **Not-All-Equal 3-Satisfiability (NAE-3SAT):**
> INPUT: A boolean formula in 3CNF form.
> QUESTION: Is there a truth assignment such that each clause contains one or two true literals?

Let Φ be a CNF-formula with variables $X_1, \ldots, X_n$ and clauses $C_1, \ldots, C_m$, i. e., $\Phi = \Pi_{i=1,\ldots m} C_i$ where $C_i = [Z_{i1} \bigvee Z_{i2} \bigvee Z_{i3}]$, and each Z_{ij} is a literal from the set $\{X_1, \ldots, X_n$ $\overline{X}_1, \ldots, \overline{X}_n\}$. Without loss of generality, no clause contains a variable and its negation. We now show how to construct for Φ an instance (V, E^1, E^2) of the sandwich problem. The construction uses two "gadgets"; each gadget is a small sandwich instance which is contained as an induced subproblem in (V, E^1, E^2), and imposes certain constraints on the solution of the problem.

1. Assign two vertices to each literal: Vertex x_i (resp., $\overline{x}_i$) is called *the vertex corresponding to literal* X_i (resp., $\overline{X}_i$), and vertex x'_i (resp. $\overline{x}'_i$) is called *the private vertex of literal* X_i (resp. $\overline{X}_i$). For each i, $i = 1, \ldots, n$, connect the vertices $x_i, x'_i, p, \overline{x}'_i, \overline{x}_i$ to form a *variable gadget* as shown in figure 2(I). The same vertex p is involved in all the 5-chains of the variable gadgets.

2. For clause C_i define three vertices v_i^1, v_i^2, v_i^3, which will be called the *private vertices* of clause C_i. Denote the vertices corresponding to the three literals of that clause by x_{i1}, x_{i2} and x_{i3}. Connect these six vertices to form the *clause gadget* as shown in figure 2(II). (Here there are both E^1 and E^0 edges, unlike the variable gadget, in which there were only E^1 edges.)

In summary, the vertex set is:

$$V = \{p\} \cup \{x_i, \overline{x}_i, x'_i, \overline{x}'_i | i = 1, \ldots, n\} \cup \{v_i^1, v_i^2, v_i^3 | i = 1, \ldots m\}.$$

The mandatory edge set is:

$$E^1 = \{(x_i, x'_i), (x'_i, p), (p, \overline{x}'_i), (\overline{x}'_i, \overline{x}_i) | i = 1, \ldots, n\} \cup \{(x_{ik}, v_i^k) | i = 1, \ldots, m, k = 1, 2, 3\},$$

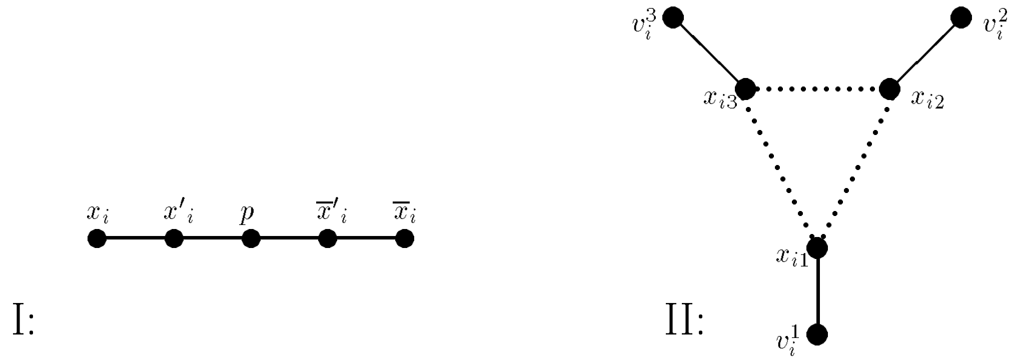

Figure 2: I: variable gadget. II: clause gadget (Solid edges are E^1-edges, dotted edges are E^0-edges. Non-edges are in E^3.)

and the forbidden edge set is:

$$E^3 = \{(x_i, \overline{x}_i), (x_i, p), (\overline{x}_i, p), (x'_i, \overline{x}'_i), (x_i, \overline{x}'_i), (\overline{x}_i, x'_i) | i = 1, \ldots, n\}$$
$$\cup \{(v_i^k, v_i^j) | i = 1, \ldots, m, k \neq j\} \cup \{(v_i^j, x_{ik}) | i = 1, \ldots, m, k \neq j\}$$

where x_{ik} is the vertex corresponding to literal Z_{ik}. Clearly the construction is polynomial.

Suppose first that there exists a unit interval sandwich G for the input. Take a unit interval realization of G, and denote the interval corresponding to vertex v in the realization by $I(v)$. By changing scale and moving the origin, if necessary, we can assume that all intervals are of length 1 and $I(p) = [0, 1]$. Denote by p_1 the point -1 and by p_2 the point +2 on the line. For each variable X_i, the intervals $I(x_i)$ and $I(\overline{x}_i)$ must be disjoint, since $(x_i, \overline{x}_i) \in E^3$. Suppose $I(p) \prec I(x_1)$. Since $(p, x'_1) \in E^1$, if we denote $I(x'_1) = [l, r]$ then $0 < l < 1$, and since $(x'_1, x_1) \in E^1$, $(p, x_1) \in E^3$ and all intervals have unit length, the interval $I(x_1)$ must contain the point p_2. In that case by the construction of the gadget for variable 1, $I(\overline{x}_1)$ must contain the point p_1. More generally, for every $i = 1, \ldots, n$, either $p_1 \in I(x_i)$ and $p_2 \in I(\overline{x}_i)$, or $p_2 \in I(x_i)$ and $p_1 \in I(\overline{x}_i)$. Assign truth values to the variables as follows: $t(X_i) = TRUE$ if and only if $p_1 \in I(x_i)$. By the above argument, exactly one literal for each variable is true.

Consider now the clause C_i. The intervals $I(x_{i1}), I(x_{i2}), I(x_{i3})$ cannot be pairwise disjoint, since each of them contains either p_1 or p_2. Hence two of the corresponding vertices are adjacent in G. If exactly two of the edges $(x_{ij}, x_{ik}), k \neq j$ were included in G then the graph would have contained the induced subgraph $K_{1,3}$ (also called a *claw*), which is forbidden in a unit interval

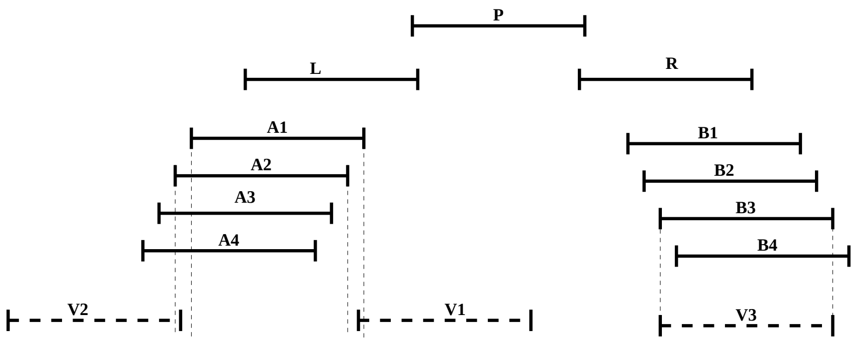

Figure 3: A unit interval realization for a satisfiable instance of NAE-3SAT.

graph. If all three are included, the sandwich graph induced by the gadget is not an interval graph. Consequently, one or two but not all three intervals contain p_1. Hence, one or two of the literals in each clause are true, as required in NAE-3SAT.

Conversely, suppose there is a NAE truth assignment which satisfies the formula Φ. We shall prove that there exists a unit interval sandwich for (V, E^1, E^2) by creating a realization for it: Assign unit intervals $P, L, R, A1, \ldots, An, B1, , \ldots, Bn$ as depicted in figure 3. All the intervals $A1, \ldots An$ intersect with L, and do not intersect with P, R or Bi. Moreover, they form a 'staircase' so that for every pair of intervals Ai and Aj, each one has a 'tail' which is not covered by the other. The situation for the B intervals is analogous. Note that intervals $A1$ and $B1$ are more than one unit apart.

Set the interval $I(p) = P$. Given the NAE truth assignment t, for $i = 1, \ldots, n$, if $t(x_i) = TRUE$, set $I(x_i) = Ai$, $I(x'_i) = L$, $I(\overline{x}_i) = Bi$ and $I(\overline{x}'_i) = R$. If $t(x_i) = FALSE$, set $I(x_i) = Bi$, $I(x'_i) = R$, $I(\overline{x}_i) = Ai$ and $I(\overline{x}'_i) = L$. In other words, we put all the true (resp., false) literals on the left (resp., right) staircase and identify all their private literal intervals with L (R). This takes care of all the constraints induced by the variable gadget.

Since t is NAE, for every clause at most two literals are true and at most two are false. Hence, only one or two of their intervals can appear in the same staircase. The intervals of the private vertices of each clause can therefore be placed as follows: (Compare the dotted intervals in figure 3). Suppose in clause C_i two variables are true, say $t(Z_{i1}) = t(Z_{i2}) = TRUE$ and $t(Z_{i3}) = FALSE$. Then $I(x_{i1}) \cap I(x_{i2}) \neq \emptyset$, and both intervals have non-overlapping tails. Add the private clause (unit) intervals $I(v_i^1)$ or $I(v_i^2)$ extending from the respective tails, and set $I(v_i^3) = I(x_{i3})$. Since Ai and Bj are more than one unit apart, both $I(v_i^1)$ and $I(v_i^2)$ can be placed without overlap with $I(v_i^3)$ and $I(x_{i3})$. The case where the two variables are false is symmetric. The result is an interval realization of a sandwich unit interval graph. ■

4 Discussion

We have shown that two simplified versions of the physical mapping problem are NP-complete. One version refers to the situation where all clones are of equal length, which is approximately the case for cosmid or phage clones. The other assumes that clones can have arbitrary lengths, as is the case with YAC clones. One possible way to cope with the hardness of the problems is to perform probabilistic analysis, and show that under certain probabilistic assumptions the expected behavior of certain mapping algorithms is polynomial. The probabilistic model proposed by Lander and Waterman [22] is a natural candidate here. Another possible approach is to incorporate additional constraints of the biological problem into the mathematical model. The resulting restricted problem may still be polynomial.

In the other direction, one may wonder whether devising fingerprinting methods which give different kind of data may help. This interesting question is motivated by the fact that the computational complexity is dramatically reduced if additional information of particular type is known. Specifically, for the case that the *relative order* of two fragments is known whenever they do not intersect, it was shown in [18] that the decision problem can be solved in linear time. Moreover, the set of all possible realizations (in case there are several orders of the endpoints which fit the input data) has a compact and convenient representation. Hence, if such order information can be provided by biological experiments for every two definitely-non-overlapping clones, the computational problem becomes quite easy. One experimental technique which provides such order data is CISS hybridization [23].

References

[1] F. Alizadeh, R. M. Karp, L. W. Newberg, and D. K. Weisser. Physical mapping of chromosomes: A combinatorial problem in molecular biology. Technical Report TR-92-066, International Computer Science Institute, Berkeley, CA, 1992.

[2] R. Arratia, E. S. Lander, S. Tavaré, and M. S. Waterman. Genomic mapping by anchoring random clones: A mathematical analysis. *Genomics*, 11:806–827, 1991.

[3] E. Barillot, J. Dausset, and D. Cohen. Theoretical analysis of a physical mapping strategy using random single-copy landmarks. *PNAS*, 88:3917–3921, 1991.

[4] S. Benzer. On the topology of the genetic fine structure. *PNAS*, 45:1607–1620, 1959.

[5] H. L. Bodlaender, M. R. Fellows, and T. J. Warnow. Two strikes against perfect phylogeny. In *Proc. 19th ICALP*. Springer, 1992.

[6] D. T. Burke, G. F. Carle, and M. V. Olson. Cloning of large segments of exogenous DNA into yeast by means of artificial chromosome vectors. *Science*, 236:806–812, 1987.

[7] A. V. Carrano. Establishing the order of human chromosome-specific DNA fragments. In A. D. Woodhead and B. J. Barnhart, editors, *Biotechnology and the Human Genome*, pages 37–50. Plenum Press, New York, 1988.

[8] A. Coulson, J. Sulston, S. Brenner, and J. Karn. Toward a physical map of the genome of the nematode *caenorhabditis elegans*. *PNAS*, 83:7821–7825, 1986.

[9] A. R. Coulson, J. Waterson, J. Kiff, J. Sulston, and Y. Kohara. Genome linking with yeast artificial chromosomes. *Nature*, 335:184–186, 1988.

[10] A. G. Craig et al. Ordering of cosmid clones covering the herpes simplex virus type I (HSV-I) genome: A test case for fingerprinting by hybridization. *NAR*, 18:2653–2660, 1990.

[11] K. J. Danna, G. H. Sack, and D. Nathans. Studies of Simian Virus 40 DNA VII. A cleavage map of the SV40 genome. *J. of Molec. Biol.*, 78:363–376, 1973.

[12] M. R. Fellows, M. T. Hallett, and H. T. Wareham. DNA physical mapping: three ways difficult (extended abstract). Technical report, Dept. of Computer Science, University of Victoria, Victoria, BC, Canada, 1993.

[13] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Co., San Francisco, 1979.

[14] P. W. Goldberg, M. C. Golumbic, H. Kaplan, and R. Shamir. Three strikes against physical mapping of DNA (extended abstract). Technical report, Institute of Computer Science, Tel Aviv Univeristy, 1993.

[15] L. Goldstein and M. S. Waterman. Mapping DNA by stochastic relaxation. *Advances in Applied Mathematics*, 8:194–207, 1987.

[16] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[17] M. C. Golumbic, H. Kaplan, and R. Shamir. Graph sandwich problems. Technical Report 270/92, Computer Science Dept., Tel Aviv University, 1992. submitted.

[18] M. C. Golumbic and R. Shamir. Complexity and algorithms for reasoning about time: A graph-theoretic approach. Technical Report 91-54, DIMACS Center, Rutgers University, NJ, 1991. To appear in *JACM.*

[19] E. D. Green and M. V. Olson. Chromosomal region of the Cystic Fibrosis gene in yeast artificial chromosomes: a model for human genome mapping. *Science*, 250:94–98, 1990.

[20] S. K. Kannan and T. J. Warnow. Triangulating 3-colored graphs. *SIAM J. of Discrete Math.*, 5(2):249–258, 1992.

[21] Y. Kohara, K. Akiyama, and K. Isono. The physical map of the whole *E. coli* chromosome: application of a new strategy for rapid analysis and sorting of large genomic libraries. *Cell*, 50:495–508, 1987.

[22] E. S. Lander and M. S. Waterman. Genomic mapping by fingerprinting random clones: A mathematical analysis. *Genomics*, 2:231–239, 1988.

[23] P. Lichter et al. High-resolution mapping of human chromosome 11 by in situ hybridization with cosmid clones. *Science*, 247:64–69, 1990.

[24] F. Michiels, A. G. Craig, G. Zehetner, G. P. Smith, and H. Lehrach. Molecular approaches to genome analysis: a strategy for the construction of ordered overlapping clone libraries. *CABIOS*, 3(3):203–210, 1987.

[25] R. Nagaraja. Current approaches to long-range physical mapping of the human genome. In R. Anand, editor, *Techniques for the Analysis of Complex Genomes*, pages 1–18. Academic Press, London, 1992.

[26] M. V. Olson et al. Random-clone strategy for genomic restriction mapping in yeast. *PNAS*, 83:7826–7830, 1986.

[27] M. V. Olson, L. Hood, C. Cantor, and D. Botstein. A common language for physical mapping of the human genome. *Science*, 234:1434–1435, 1985.

[28] J. Opatrny. Total ordering problems. *SIAM J. Computing*, 8(1):111–114, 1979.

[29] T. J. Schaefer. The complexity of satisfiability problems. In *Proc. 10th Annual ACM Symp. on Theory of Computing*, pages 216–226, 1978.

[30] N. Sternberg. Bacteriophage P1 cloning system for the isolation, amplification and recovery of DNA fragments as large as 100 kilobase pairs. *PNAS*, 87:103–107, 1990.

[31] K. Tynan et al. Assembly and analysis of cosmid contigs in the CEA-gene family region of human chromosome 19. *NAR*, 20(7):1629–1636, 1992.

[32] U. S. Congress. Mapping our genes–the genome projects, how big, how fast? Technical Report OTA-BA-373, Office of Technology Assessment, Washington, D.C., 1988.

[33] M. S. Waterman and J. R. Griggs. Interval graphs and maps of DNA. *Bull. Math. Biol.*, 48:189–195, 1986.