LA-SUB--95-111

FINAL REPORT
ON
CONTRACT 9-L72-Z9321

ABSTRACT

This report documents work done under contract 9-L72-Z9321. An algorithm was developed for genome contig assembly which extended the range of data types that could be included in assembly and which ran on the order of a hundred times faster than the algorithm it replaced.

OBJECTIVES

The objective of Contract 9-L72-Z9321 was to "develop and test new optimization algorithms for searching the space of all possible integrated genome maps, to find those that fit available experimental data as well as possible."

PERFORMANCE

In my original technical proposal, I proposed to extend work done in 1991 on applying genetic algorithms to the problem of genome maps. I began by exploring ways of improving the efficiency of the existing genetic algorithm (GCAA) for contig assembly. One of the approaches I explored was to initialize the GA with a population of so-so maps created by a greedy algorithm, rather than with a population created at random, but after developing an appropriate greedy algorithm I found that, with a slight enhancement, it outperformed our GA (and using it to initialize the GA gave no better results than using it alone). After discussions with Jim Fickett, the University's technical representative, I shifted my attention to this approach and extended the algorithm to allow for subcloning. I also began work on a more general version of this algorithm which would allow for a much wider range of relations between elements.

In the latter part of 1992 the human genome project at LANL began a big push to assemble maps from their existing data, and I spent a good deal of time tuning the algorithm presented below to deal with their data sets, and in generating actual maps for them. Work on more general data types (genetic linkage data etc.) was suspended to concentrate on their requests.

In January 1993, the folks in Life Sciences and the Genome Project were expressing increasing interest in localized approaches to map assembly rather than the sort of global probabilistic approach envisioned in this project, and at the same time personal distractions were making it more difficult for me to pursue the work, so I asked Jim Fickett to discontinue the contract and we agreed to discontinue it following a talk which I gave in late January, summarizing my results.

RESULTS

1) An algorithm for contig assembly, hereafter called "Iterative Contig Assembly" or ICA, is summarized below. An implementation of the algorithm was left with Jim Fickett.

2) Maps of all existing cosmid clone and YAC data at the Human Genome Information Resource were assembled using ICA. The resulting maps are summarized below.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

3) A talk was given in late January for members of the Human Genome Information Resource and Life Sciences.

ICA ALGORITHM:

The data consisted of estimated clone lengths, pairwise probabilities of overlap between clones (which in some cases was constrained to 'containment' of one clone by another clone), estimates of length of overlap for clone pairs with > .5 probability of overlap, and for certain YAC pairs a probability of the pair being "either identical or disjoint". In this later case, assuming the probability of the clones lapping was p and the probability of them being either identical or disjoint was q, we divided the possible relations between the two clones into three cases:

identical: q x p
disjoint: q x (1-p)
otherwise: 1-q

The fitness of a given map was defined to be:

$\prod P_{ij}$

taken over all (ij) pairs, where P_{ij} is the probability that clones i and j were related to one another (identical, overlapped or disjoint) as in the map. This fitness assumes that the relation between each pair is independent of the relations between other pairs, which is not the case, but may be a reasonable first-order approximation. In the talk I gave in January I suggested one way of avoiding this assumption, and in the discussion that followed David Torney said that he believed he could include the interdependence between pairs directly in the calculation. If a global, probabilistiic approach is pursued, David's ideas ought to be explored.

In this context, a 'map' is an unordered list of contigs such that every clone in the map is in one and only one contig; a 'contig' is a spacing of the endpoints of a set of clones along the real line to span a closed interval (one end of which can be arbitrarily chosen to be 0). ICA builds maps by deterministically adding clones to smaller maps, but the order in which clones are added is chosen at random. In simple cases adding clones in any order yields the same (optimal) map; in more complicated cases adding clones in different orders yields different maps, but a small number of different orderings (a few hundred) typically turns up maps which are as good or better than GCAA maps. ICA also runs so much faster than GCAA that maps involving hundreds of clones can be assembled; such large data sets are impractical for GCAA.

To build a map, begin with an empty map and add clones to it in a random order. To add a clone to a map, evaluate adding it to each separate contig of the map; consider three possibilities: 1) Overall fitness would be most increased by connecting two contigs with the new clone (forming a single larger contig) 2) Overall fitness would be most increased by adding the clone to an existing contig 3) Overall fitness would be decreased if the new clone overlaps any contig. In the first case, we made need to flip a contig over to connect two contigs. In the third case, we start a new contig.

To evaluate adding a clone to a contig: consider the set of at most $2n+1$ contiguous intervals defined by the endpoints of the n clones in the contig (the first and last intervals extend infinitely far and overlap no clone). Moving an endpoint of the new clone within a given interval will not alter $\prod P_{ij}$, so we need only consider the effect on $\prod P_{ij}$ of putting the new endpoints in any of $2n+1$ places. The number of places to consider is further reduced by the minimum and maximum allowed lengths of the new clone. At the same time, note if

overhanging the contig improves $\prod P_{ij}$; when each contig has been evaluated separately, consider connecting any pair of contigs for which overhanging improves $\prod P_{ij}$.

To place a new clone once an optimal set of intervals is found for its endpoints: pick the clone length nearest to the estimated length, that fits within the intervals, and then position the clone to minimize the sum of the squares of errors in estimated overlap lengths.

ICA, as outlined above, works well with clones of similar length. In order to map clones of widely disparate lengths such as the cosmid and YAC clones which were included in the data set at LANL, one often needs to rearrange short clones that have already been mapped to within some longer clone. Define any set of clones that are contained within a longer clone to be a sub-map, and consider moving and connecting contigs within the sub-map. In this case the overall length of the sub-map is constrained to remain less than the length of the containing clone.

SUMMARY OF ICA MAPS:

A total of 3955 cosmid clones and 485 YACs were used in mapping. The resulting maps were left with the Human Genome Information Resource. A number of specific questions about coverage, raised by people at Life Sciences, were addressed.

There was some question about how useful YACs were in bridging gaps between cosmid clones, and at the request of Life Sciences maps were assembled both with and without YACs. Below is a histogram of the resulting contigs:

	with YAC		without YAC	
clones per contig	# of contigs	bp	# of contigs	bp
1	1047	72892131	827	30421656
2- 9	689	52823324	569	23236526
10-19	46	6704838	56	2531835
20-29	3	544000	10	486439
30-39	2	637313	2	97519
40-49	1	255000		
50-59	3	1032544	1	50000
60-69	1	280000		
		135169150		56823975

218 YAC contigs involved no cosmids.

61 YAC contigs involved cosmids from more than 1 YAC-less contig, and if we assume that whenever cosmids from *n* YAC-less contigs appear in one YAC contig, *n-1* gaps were closed, then a total of 110 gaps were closed by YACs. (This may be a little misleading because, given the effects of YACs on the mapping process and the non-deterministic algorithm used in mapping, the assignment of cosmids to contigs is not necessarily the same in both cases.) A list of the 61 YAC contigs bridging YAC-less contigs and the contigs they bridged was given to Life Sciences.

152 YAC-less contigs involve clones from more than one of the YAC contigs. These are either cases of incorrect YAC-less contigs being legitimately broken up by constraints imposed on the YACs, or noise.

Within the YAC runs all subclone relations were satisfied.

Life Sciences provided a list of sets of clones that were already thought to belong in single contigs. A total of 31 clones were in the list, 20 of these were grouped by ICA as expected by Life Sciences.

It was suspected that some of the YACs were in fact identical; these ICA runs were, as far as I know, the first systematic attempt to figure out the identities, and we found 22 sets of YACs which are strong candidates for being identical. The list of likely identities was given to Life Sciences.

799 YAC pairs were initially assigned a 90% probability of being either disjoint or identical, based on the experimental techniques used to generate them. 48 of these pairs wound up overlapping without being identical; this is in keeping with the assigned probability. The overlaps may be due to the expected experimental error in the technique used to generate them, or to chimera and 'repeat' problems, or to faulty mapping by ICA. The list of overlapping but not identical pairs was given to Life Sciences.

82 clones had been previously assigned 'Sigma coordinates' using localization techniques. This data was not used in generating ICA maps, but ICA maps were compared with Sigma coordinates and did well at grouping clones judged to be nearby by localization, within contigs. There were some clear failures: cases in which clones localized to points far apart on the chromosome were placed within a single, short ICA contig. A list of ICA groupings of localized clones was given to Life Sciences.

CONCLUSION:

ICA 'maps' reveal (or confirm) serious problems in our data. Finding or highlighting problems is itself worthwhile. When I began working on map assembly the only map assembly being done at LANL was done by individuals, assembling contigs of 15 or 20 clones by hand. At that time the effort went to finding a 'best' arrangement. With GCAA it was relatively easy to find best or nearly best arrangements of contigs involving 30 or 40 clones, and some of us began to wonder whether a single 'best' arrangement was really what was needed. ICA, developed during this project, makes it easy to find excellent maps involving hundreds of clones — if by excellent one means only that they fit the available probabilities as well as any other map.

Unfortunately for most sets of hundreds of clones, there are many excellent maps ('many' here is deliberately vague) — maps whose likelihood ratio with the best map we can find is essentially 1. If used right, ICA and other probabilistic mapping strategies might help direct efforts to refine the probabilities, but it seems clear that the probabilities will have to be refined before meaningful maps can be made. I think LANL's Human Genome effort now should focus on other techniques.

I'd like to thank Jim Fickett of LANL and the Human Genome Information Resource for his help, insights and calm on this project.