COMPARING GENOMES

Can it then be that there is…something of use for unraveling the universe to be learned from the philosophy of computer design?—J.A. Wheeler[1]

The theory behind biocomputing is to look to biological structures and processes for new ways of solving difficult computational problems. But this need not be a one-way street: advances in computing can feed back into the study of biology, leading to better biotechnological tools.

As new ways of using biological material to solve difficult computational problems continue to emerge, several fundamental questions arise: Are these techniques practical? If so, what are the key applications? Do the techniques scale to larger problems? Do they give us anything more than a few elegant theoretical insights into the nature of computation? Ultimately, is this a productive endeavor?

Before exploring these questions, it might be fruitful to examine a quotation by Richard Feynmann, as it reflects on similar questions in the context of quantum-mechanical computers:

> The discovery of computers and the thinking about computers has turned out to be extremely useful in many branches of human reasoning. For instance, we never really understood how lousy our understanding of language was, the theory of grammar and all that stuff, until we tried to make a computer which would be able to understand language. We tried to learn a great deal about psychology by trying to understand how computers work. There are interesting philosophical questions about reasoning and relationship, observation, and measurement and so on, which computers have stimulated us to think about anew, with new types of thinking. And all I was doing was hoping that the computer type of thinking would give us some new ideas, if any are really needed.[2]

In a similar vein, I argue that although biocomputing approaches use classical biotechnological tools, ultimately the reasoning and design style emerging in the biocomputing field will lead to more sophisticated, robust, and high-throughput biotechnology—a technology primarily centered around manipulating biological material in living cells (in vivo), in test tubes (in vitro), or in computational models (in silico) with the hope of creating a detailed picture of how living organisms function.

To develop these ideas, this article focuses on just one example: using a randomized technique from the world of computer algorithms to compare two related genomes. This method of genome comparison, which has many applications in cancer research, was originally developed in collaboration with my colleague Mike Wigler and his laboratory at Cold Spring Harbor. The description here, focusing simply on the problem's computational aspect, uses some ideas from a recent article.[3]

1521-9615/02/$17.00 © 2002 IEEE

BUD MISHRA
Courant Institute, New York University, and Cold Spring Harbor Laboratory

The problem: Comparing genomes

The motivation for comparing two related genomes comes from our efforts to understand the genetic basis of cancer. Roughly, to deduce what makes a cell go into uncontrolled growth, we must focus on the genes involved in a cell's important decisions about growth, growth arrest, and cell death (apoptosis). The genes involved in these processes fall into two categories: *oncogenes*, of which there are about 100, and *tumor suppressor genes*, of which there are about 1,000.

A healthy cell deviates from its normal function to initiate tumor formation because of various changes to the genome: *amplifications*, *deletions*, *translocations*, and *point mutations*. Both amplification and deletion are fluctuations in a gene's *copy number*—the number of occurrences of that gene in the genome: an amplification increases the copy number, and a deletion decreases it. Thus, detecting regions of amplification can lead us to the locations of oncogenes; detecting regions of deletion can lead us to tumor suppressor genes. A translocation occurs when a gene moves from its original location to another without changing its copy number, and a point mutation occurs when a single base pair is replaced by another. The differences between the genomes from healthy tissue and those from cancer tissue tell us a lot about where the oncogenes and tumor suppressor genes might be located.

Comparing two genomes rapidly appears to be an elusive goal. Recently, Douglas Hanahan and Robert Weinberg wrote pessimistically, "At present, description of a recently diagnosed tumor in terms of its underlying genetic lesions remains a distant prospect. Nonetheless, we look ahead 10 or 20 years to the time when the diagnosis of all somatically acquired lesions present in a tumor cell genome will become a routine procedure."[4]

Clearly, we cannot simply sequence the genomes completely and compare them; such an approach is not cost-effective and won't be for the foreseeable future. So instead, my colleagues and I are focusing on a randomized approach that is quite common in the computer algorithm field.

Tools of the trade

Before we get to the algorithm, however, let's start with some biological background, leading to the three key biotechnological operations that are the tools of our trade.

The usual configuration of DNA is a *double helix* consisting of two chains or strands coiling around each other, with two alternating grooves of slightly different spacing. The backbone in each strand is made of alternating big sugar molecules (Deoxyribose residues) and small phosphate molecules.

Connected to each sugar molecule is one of four bases—*adenine* (*A*), *thymine* (*T*), *cytosine* (*C*), or *guanine* (*G*). Reading the sequence of bases defines the information encoded by the DNA. Complementary base pairs (*A*-*T* and *C*-*G*) in the two strands are connected by hydrogen bonds, and each of these base pairs forms an essentially coplanar "rung" connecting the two strands. This characteristic, known as *Watson-Crick complementarity*, makes DNA chemically inert and mechanically stable; hence, it is an ideal molecule for mechanical and computational devices. However, we can manipulate DNA molecules with various biochemical tools: scissors, glues, and copiers.

Scissors: Restriction activity

Type II sequence-specific restriction endonucleases are enzymes that can "cut" a double-stranded DNA by breaking the phosphodiester bonds on the two DNA strands at specific target sites. These target sites, known as *restriction sites*, are determined completely by their base pair composition—usually, a short sequence of four, six, or eight base pairs. For instance, the restriction enzyme Hpa II cuts DNA at any occurrence of the tetranucleotide *CCGG*. Such enzymes have been extremely useful in biotechnology as biochemical *scissors* and biochemical *markers*, because they always cut DNA at the same short, specific patterns.

In our application, we use restriction enzymes to cut a genome into small pieces and then select only a subset of these fragments for further use as probes. (I'll explain exactly how we use the probes when I get into the details of our genome comparison method.) Because we know exactly how the restriction enzymes will cut the DNA, the probes we generate are reproducible, reliable, and consistent. Furthermore, parallel representations—probe sets selected from two genomes—preserve gene ratios and hence provide a crucial tool for our application.

Glues: Ligation and hybridization

In contrast to scissors enzymes, a DNA ligase is a cellular enzyme that can join two strands of DNA molecules by repairing a phosphodiester bond. We do not make explicit use of DNA ligases in our application, but they are widely used as a key biotechnological tool.

We do focus, however, on the process of *hybridization*, by which hydrogen bonding between two complementary, single-stranded DNA fragments (or an RNA fragment and a complementary single-stranded DNA fragment) creates a double-stranded DNA (or a DNA–RNA complex). In our application, we use hybridization primarily to determine whether a short string (a probe, in our case) appears as a substring in a longer string (a clone or subgenomic DNA). To achieve this, we create a DNA fragment encoding the sequence that is complementary to that of the probe; then we experiment to see whether the complementary fragment hybridizes to a DNA fragment encoding the longer sequence. If it does, the longer sequence includes our probe.

We can parallelize the method by spotting on a surface several probe sequences as a matrix of a very large number of spots (several thousand) and hybridizing all the probes with one or more clone sequences in parallel. If more than one clone sequence is involved, this approach lets us determine whether a particular probe sequence belongs to any one of the clone sequences. This technology, embodied as microarrays, has widespread application in measuring gene expressions, classifying genes, mapping markers on the genome, and detecting polymorphisms.

Copiers: Cloning and PCR

For our purposes, a *clone* is a rather large DNA fragment that has been preselected and kept in a library, which we can use to make many faithful copies. We use four different kinds of clones in the laboratory: yeast artificial chromosomes (YACs), which range in size from 1 to 2 million base pairs (Mb); bacterial artificial chromosomes (BACs), which range from 100 to 200 thousand base pairs (Kb); cosmids, which range from 20 to 45 Kb; and lambdas, which range from 2 to 20 Kb. Molecular cloning is an in vivo approach involving a living host organism (usually the *E. coli* bacteria or yeast) that replicates a suitably modified foreign DNA as if the foreign DNA were one of its own. The modification involves combining a cloning *vector* with the foreign DNA to be amplified (the *insert*) to create a circular recombinant DNA molecule called the *replicon*. The cell will not replicate any foreign DNA without a suitable vector.

In our application, we use BACs more or less as measuring devices. If two probes cohybridize to the same BAC, we know that those two probes are within a distance shorter than the length of the BAC. However, hybridizing with just one BAC at a time would be inefficient. Hybridizing with several thousand randomly selected BACs can simultaneously give us distance information for many pairs of probes. The fact that we can make vast numbers of copies of the same BAC reliably and rapidly is the key to our approach's overall robustness.

PCR, or *polymerase chain reaction*, is an in vitro technique for replicating a fragment of DNA to produce many copies of a short, specific DNA sequence. The biochemical process involved in PCR operates iteratively: In the first step, we denature (separate) two strands of the DNA by heating. In the subsequent step, we add short sequences of a single DNA strand (primers), together with a supply of free nucleotides and DNA polymerase, to create two double-stranded copies, each originating from the two complementary single strands obtained in the earlier step. The original DNA sequence doubles in each repetition of the heating and cooling cycle, resulting in rapid amplification.

PCR is commonly used as an alternative to in vivo cloning to amplify DNA material. This technique finds use in many medical and biological applications (measuring gene expressions, DNA sequencing, and so on), but its most prominent applications are in forensic science, where it is used to amplify minuscule traces of genetic material for DNA fingerprinting.

Sampling rather than sequencing

Now let's turn to how we use these biotechnological tools with a randomized approach to actually compare genomes. We can sample the genome uniformly to create a large number of probes—150,000—located every 20 Kb. These probes, which are short subsequences of 200 to 1,200 base pairs, come from regions of the genome that do not share homologous sequences elsewhere in the genome, so each probe is almost surely unique. Our approach then boils down to determining the relative locations of these probes in the two genomes: their relative ordering, their presence (possibly multiple times) or absence, or simply the changes in their relative distances from each other within a small chromosomal region.

Thus, if we can create an inexpensive biotechnological method of measuring the distance between any two probes, we can then shift the focus of our research to the algorithmic problem of finding the probes' locations along the two genomes, or even to the simpler problem of determining

when the relative locations of a small group of closely clustered probes are perturbed from one genome to another. Of course, any biochemical method we develop will be subject to the corrupting effects of many independent error sources; modeling these errors will be a key challenge.

The fundamental idea of our algorithm, which localizes the probes along the genome, comes from the simple observation that if we can determine the pairwise distances among all the probes, then we can place these probes along the genome correctly. If we know the distances with accuracy, then any three probes satisfy a triangle equality; with the known locations of any two of the three probes, we can uniquely determine the third probe's location.

When the pairwise distance data are inaccurate, the triangle equality (and other similarly higher-order constraints) are violated, and the distance data is inconsistent. Thus, the algorithmic question becomes, "How can the distance data be minimally perturbed so that they become consistent?" We can formulate this question as an optimization problem for a weighted sum-of-square cost function. Although in the most pathological context such problems can be computationally infeasible, we have developed a simple, almost-linear-time probabilistic algorithm that works well for a carefully designed experiment—for example, choosing the expected number of probes per clone, the number of hybridization experiments, and so on.[3]

Thus, the focus of our research moves to the following key questions: How do we model the errors in the distance function? How do we design the experiments' parameters?

Roughly, a single biochemical hybridization experiment (conducted with a microarray) assigns a discrete value—a "color"—to each probe: B = blank, R = red, G = green, and Y = yellow. A sequence of such experiments assigns a *color vector* to each probe, and the number of places in which these color vectors differ for any two probes gives us a clue about the distance between these two probes. Thus, we derive the distance metric between two probes from a Hamming distance between every pair of color vectors assigned to the probes. As we conduct a succession of these hybridization experiments, the Hamming distance between two probes is incremented by one every time the probes disagree on the outcomes of any hybridization experiment. Thus, the probabilistic modeling of the errors in distance simply involves deriving a conditional probability that the two probes will disagree in an experiment, given that they are some particular distance apart.

Probes and their distances

To measure the pairwise distances among a large number of probes, we've devised a method that relies on the available microarray technology. The basic technology uses unordered probes that are microarrayed and hybridized to an organized sampling of arrayed but unordered members of libraries of large insert genomic clones. In this article, we'll focus on BACs, but the basic ideas can be applied to other types of clones, chromosomal fragments, or random PCR products derived from genomic DNA. (A detailed discussion of this technology's challenges as well as its full potential would include our knowledge of genome organization, DNA hybridization, repetitive DNA, gene duplication, and the varieties of microarrays. For the sake of simplicity, however, this article omits these details.)

Imagine a set of P points on a line segment of length G (representing probes on a chromosome or a genome, which denotes the collection of all the chromosomes). Further imagine a set of random intervals of length L from the line segment (representing a BAC or YAC library or the chromosomal fragments contained in a panel of radiation hybrid cell lines). For our purposes, the line segments will be BACs, and length L will be 160 Kb.

Now, we perform the following *array hybridization*. We pick two random subsets of K intervals each and denote one set as the red set and the other as the green set. We assign each point a color based on whether the point belongs to neither the union of intervals in the red set nor the union of intervals in the green set (blank); or does belong to either the former (red), the latter (green), or both (yellow). That is,

- B = blank ($\neg$ red $\wedge$ $\neg$ green),
- R = red (red $\wedge$ $\neg$ green),
- G = green ($\neg$ red $\wedge$ green), or
- Y = yellow (red $\wedge$ green).

We can easily achieve these logical steps by an *array hybridization* step with microarrays. The P probes are Watson-Crick complements of short, "unique" subsequences of the genomes; we can produce them reliably and in large quantity by using restriction enzymes, or we can synthesize them as oligoes. Each probe is spotted at a fixed physical location on a microarray.

Now, if we hybridize a collection of several BACs to this microarray, the BACs that contain a subsequence complementary to the probe sequence hybridize to the probe. Because these BACs each possess a color, which we achieve physically by attaching a colored fluorescent dye, the probe acquires the colors of the BACs that it hybridizes to. For instance, if the complement of the probe sequence is contained in a BAC sequence dyed red, but not in any BAC sequence dyed green, we will see that probe as red. Analogously, we can see the relation between points and intervals in our earlier discussion to be biochemically determined for the probes and BACs through hybridization. Thus, array hybridization lets us observe a color outcome for each of the 150,000 probes in a short, constant amount of time.

The probability that two probes have different color outcomes in a single array hybridization depends on how far apart they are and monotonically increases with the distance. Thus, if we can estimate this probability by several array hybridization experiments, we can estimate the distance between two probes. We can easily estimate the probability by counting the number of experiments in which the probes have different color outcomes and expressing it as a fraction of the total number of experiments. In other words, we can present the outcomes of M different experiments as color vectors of length M, one associated with each probe, and estimate the distance between two probes from the Hamming distance between their associated color vectors. The Hamming distance between two discrete-valued vectors is defined as the number of positions where the entries of the two vectors differ.

To explore the relation between the "true" distance between probes and the Hamming distance between their color vectors, we proceed as follows: Represent the probes as points $\{p_1, ..., p_P\}$. Assume that the probes are independent and identically distributed (i.i.d.) with uniform random distribution over the interval $[0,G]$. Let S be a collection of intervals of the genome, each of length L. Suppose the left-hand points of the intervals of S are i.i.d. uniform random variables over the interval $[0,G]$. Take a small subset—$2K$—of intervals $S' \subset S$, chosen randomly from S. Divide S' randomly into two equal-sized, disjoint subsets $S' = S'_R \cup S'_G$, where R indicates a red color set and G indicates a green color set. Now specify any point p_i in $[0,G]$ and consider the possible associations between p_i and the intervals in S':

- Point p_i is not covered by any interval in S'. Probe p_i hybridizes to no BACs. We say the outcome is blank, B.
- Point p_i is covered by at least one interval of S'_R but no intervals of S'_G. Probe p_i hybridizes to at least one red BAC and no green BACs. We say the outcome is red, R.
- Point p_i is covered by at least one interval of S'_G but no intervals of S'_R. Probe p_i hybridizes to at least one green BAC and no red BACs. We say the outcome is green, G.
- Point p_i is covered by at least one interval of S'_R and at least one interval of S'_G. Probe p_i hybridizes to at least one green BAC and at least one red BAC. We say the outcome is yellow, Y.

We call these events i_B, i_R, i_G, and i_Y respectively. If we perform a sequence of M such experiments, for each p_i we get a sequence of M outcomes represented as a color vector of length M. The parameter domain for the full experiment is $\langle P, L, K, M\rangle$, where P is the number of probes, L is the average length of the genomic material used (for BACs, L = 160 Kb), K is the sampling size, and M is the number of samples. The output is a color sequence for each probe. The sequence corresponding to probe p_j is $\mathbf{s}_j = \langle s_{j,k}\rangle^M_{k=1}$, with $s_{j,k} \in \{B, R, G, Y\}$.

With the resulting color sequences $\mathbf{s}_j$ we can compute the pairwise Hamming distance. Let

- $H_{i,j}$ = the number of places where $\mathbf{s}_i$ and $\mathbf{s}_j$ differ,
- $C_{i,j}$ = the number of places where $\mathbf{s}_i$ and $\mathbf{s}_j$ are the same but $\mathbf{s}_i \neq B$, and
- $T_{i,j}$ = the number of places where $\mathbf{s}_i$ and $\mathbf{s}_j$ are B.

The Hamming distance $H_{i,j}$ defines a distance metric on the set of probes. The roles of the functions $C_{i,j}$ and $T_{i,j}$ will become clear as we go on.

Because the M array hybridization experiments are independent, we must look at any single experiment—that is, $M = 1$ case. Let's define events $T = (i_B \wedge j_B)$, $C = ((i_R \wedge j_R) \vee (i_G \wedge j_G) \vee (i_Y \wedge j_Y))$, and $H = (\neg T \wedge \neg C)$. We will compute the conditional probabilities of these events when we know the distance between the corresponding probes—that is, $x = | p_i - p_j |$.

Given a set of $2K$ BACs on a genome $[0,G]$, the probability that none starts in an interval of length l is $(1 - \alpha)^l \approx e^{-\alpha l}$, where $\alpha = 2K/G$. Similarly, the probability that no red BACs start in an interval of length l is $(1 - \alpha_R)^l \approx exp[-\alpha_R l]$ (and

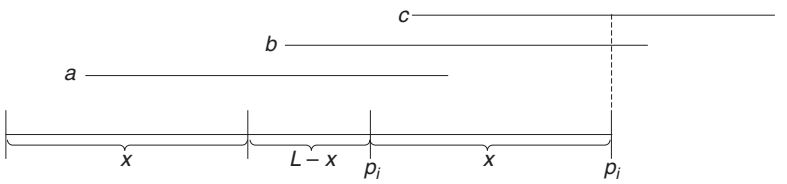

Figure 1. Computing probabilities for events *C, H,* and *T:* (a) set of BACs that covers probe p_i but not p_j; (b) set of BACs that covers probe p_i and p_j; and (c) set of BACs that covers p_j but not p_i. Note that as the distance between p_i and p_j increases, the probability that a single BAC covers both probes will progressively diminish to zero. As a result, in the extreme cases, the only way two probes will receive the same color would be by hybridizing to two different BACs of the same color. Thus by carefully choosing the number of BACs of each color in an experiment, you can reliably predict the conditional probability of each event (*C, H,* or *T*) as a direct function of the distance between two probes. Using Bayes' theorem, you can then estimate the distance between two probes as a function of the outcomes of a series of hybridization experiments. All these pairwise interprobe distances give us the basic means to determine how the organization of the probes may have changed from one genome to another.

the probability that no green BACs start in an interval of length l is $exp[-\alpha_G l]$), where $\alpha_R = \alpha_G = K/G = \alpha/2$. Let c denote $\alpha L = 2KL/G$, the coverage by the BAC sublibrary $S' \subset S$.

The diagram in Figure 1 is helpful in computing the probabilities for events C, H, and T when $x < L$. Hence, we can compute various conditional probabilities:

$$P(T \mid x \le L) = exp[-(\alpha_R + \alpha_G)(L+x)],$$

$$P(i_R \wedge j_R \mid x < L) = exp[-\alpha_G(L+x)]\{1 - 2\, exp[-\alpha_R L] + exp[-\alpha_R(L+x)]\},$$

$$P(i_G \wedge j_G \mid x \le L) = exp[-\alpha_R(L+x)]\{1 - 2\, exp[-\alpha_G L] + exp[-\alpha_G(L+x)]\},$$

$$P(i_Y \wedge j_Y \mid x \le L) = (1 - 2\, exp[-\alpha_R L] + exp[-\alpha_R(L+x)]) \times (1 - 2\, exp[-\alpha_G L] + exp[-\alpha_G(L+x)]),$$

$$P(C \mid x \le L) = P(i_R \wedge j_R \mid x \le L) + P(i_G \wedge j_G \mid x \le L) + P(i_Y \wedge j_Y \mid x \le L), \text{ and}$$

$$P(H \mid x \le L) = 1 - [P(T \mid x \le L) + P(C \mid x \le L)].$$

Similarly, when $x \ge L$ the probabilities are

$$P(T \mid x \ge L) = exp[-(\alpha_R + \alpha_G)(2L)],$$

$$P(i_R \wedge j_R \mid x \ge L) = exp[-\alpha_G(2L)]\{(1 - exp[-\alpha_R L])^2\},$$

$$P(i_G \wedge j_G \mid x \ge L) = exp[-\alpha_R(2L)]\{(1 - exp[-\alpha_G L])^2\},$$

$$P(i_Y \wedge j_Y \mid x \ge L) = (1 - exp[-\alpha_R L])^2\,(1 - exp[-\alpha_G L])^2,$$

$$P(C \mid x \ge L) = P(i_R \wedge j_R \mid x \ge L) + P(i_G \wedge j_G \mid x \ge L) + P(i_Y \wedge j_Y \mid x \ge L), \text{ and}$$

$$P(H \mid x \ge L) = 1 - [P(T \mid x \ge L) + P(C \mid x \ge L)].$$

Recall that $\alpha_R L = \alpha_G L = c/2 = KL/G$. Let $q = q(x) = P(H)$ and $p = p(x) = P(C)$. In general, $q(x)$ and $p(x)$ are complicated functions of x:

$$q(x) = P(H) = \frac{2c \exp\left(\frac{-c}{2}\right)x}{L} + O(x^2) \qquad (1)$$

$$p(x) = P(C) = 1 - e^{-c} + c/2(e^{-c} - 2e^{-c/2})x + O(x^2) \qquad (2)$$

With independent sampling, we now have the following Binomial probability distribution functions:

$$P(H_{i,j}) \sim \text{Binomial}\,(M, q(x))$$
$$P(C_{i,j}) \sim \text{Binomial}\,(M, p(x))$$

By solving equations 1 and 2, and neglecting higher-order terms, we get

$$\tilde{x} \approx \left(\frac{q}{q + 2p}\, e^{c/2}\right) L\,.$$

We can use the following estimator of x_{ij} to measure the distance between two probes:

$$\tilde{x}_{i,j} = \frac{H_{i,j}}{H_{i,j} + 2C_{i,j}} e^{\frac{H_{i,j}+2C_{i,j}}{4M}} L .$$

This estimator takes into account the variation of sample coverage over the genome. Using a simplifying normal approximation, we have, for $x < L$, the measured distance

$$\tilde{x} \sim x + \left(\frac{e^{c/4}}{\sqrt{2c}} \right) \sqrt{\frac{L}{M}} \sqrt{x} \mathrm{N}(0,1) .$$

When $x \geq L$, similarly we have

$$\tilde{x} \sim L + \left(\frac{e^{c/4}}{\sqrt{2c}} \right) L \sqrt{\frac{1}{M}} \mathrm{N}(0,1) .$$

Here, N(0,1) represents a standard normal distribution of mean 0 and variance 1.

In summary, our biochemical process lets us measure the distance between any two probes. Furthermore, we have a good model of the errors in the measurements, and we can accurately control the amount of error by appropriately choosing various experimental parameters such as *K*, the number of BACs (affecting parameter *c*); *L*, the clone length; and *M*, the number of array hybridization experiments. We should note that if two probes are further apart than the BAC length (L = 160 Kb), the distance measured does not provide any useful information.

Applications

Now let's take a look at how we can use the probe distance technology to compare two genomes. In the simplest applications, we can use the probe distance data to find the relative locations of the probes along the genome. The information created this way provides us a low-resolution reference map of the probes. We can compare this map to a specific genome (for example, from tumor tissues) to see which probes are present multiple times in the genome and which probes are omitted. The simplest analysis could involve hybridization with whole genomic DNA to microarrays of probes. If a region surrounding a probe is missing from the selected genome, the genomic DNA lacks material that could hybridize to the probe. Conversely, if a certain region surrounding a probe has been amplified in the selected genome, the genomic DNA has material that could hybridize to the probe in abundance. Applying such an analysis to cancer genomes could tell us the regions of amplification and deletion, but not translocations. Nonetheless, this analysis would be sufficient to find the oncogenes and tumor suppressor genes.

Although the ideas I've just described are sound in principle, they are impractical, because a genome's complexity is high, and the signal-to-noise ratio is inadequate to detect all but the grossest amplifications. My colleagues and I, and other researchers, have modified the basic technology in several ways to improve the signal-to-noise ratio and detect copy number changes accurately—amplifications and deletions are specific examples.[3,5–8]

We can further improve the basic technology by measuring the probe distances with genomic chromosomal fragments rather than clones. When we use clones from a library, the distances measured are distances with respect to a reference genome; these depend on how the clone library was created. If we avoid clones and use instead genomic materials from a selected genome to measure the distances between probe pairs, the measured distances reflect the locations of the probes along the selected genome; these measurements are much more informative. As before, the signal-to-noise ratio in the hybridization creates problems that we can solve through various modifications to the basic technology.

In general, comparative genomics has many applications of the utmost biological significance. The technology described here can be adapted to many different applications in those contexts. Most important, the ideas developed here indicate how the design principles developed for computer algorithms, information theory, systems sciences, and so on are likely to find applications in biotechnology. The greatest impact of biocomputing will be on biotechnology.

Acknowledgments

The following research grants supported the work reported in this article: "High-Density Gene Copy

Number Microarrays," National Institutes of Health; "Genomics via MicroArrays," NYU University Research Challenge Fund; "Bioinformatics Prototyping Language for Mapping, Sequence Assembly, and Data Analysis," US Department of Energy; "Faculty Development Program for Bioinformatics and Genomics," New York State Office of Science, Technology, & Academic Research (NYSTAR); "Algorithmic Tools and Computational Frameworks for Cell Informatics," DARPA; and "Algorithmic and Mathematical Approaches in Cell Informatics," HHMI Biomedical Support Research Grant.

References

1. J.A. Wheeler, "The Computer and the Universe," *Int'l J. Theoretical Physics,* vol. 21, no. 6/7, 1982, p. 557.
2. R. Feynmann, "Simulating Physics with Computers," *Int'l J. Theoretical Physics*, vol. 21, no. 6/7, 1982, p. 486.
3. W. Casey, B. Mishra, and M. Wigler, "Placing Probes along the Genome Using Pairwise Distance Data," *Algorithms in Bioinformatics, Proc. First Int'l Workshop, WABI 2001*, Lecture Notes in Computer Science vol. 2149, Springer-Verlag, Berlin, 2001, pp. 52–68.
4. D. Hanahan and R. Weinberg, "The Hallmarks of Cancer," *Cell,* vol. 100, no. 1, Jan. 2000, pp. 57–70.
5. J. Håstad, L. Ivansson, and J. Lagergren, "Fitting Points on the Real Line and Its Application to RH Mapping," *Algorithms—ESA '98*, Lecture Notes in Computer Science, vol. 1461, Springer-Verlag, Berlin, 1998, pp. 465–467.
6. N. Lisitsyn and M. Wigler, "Cloning the Differences between Two Complex Genomes," *Science,* vol. 259, no. 5097, Feb. 1993, pp. 946–951.
7. R. Lucito et al., "Detecting Gene Copy Number Fluctuations in Tumor Cells by Microarray Analysis of Genomic Representations," *Genome Research,* vol. 10, no. 11, Nov. 2001, pp. 1726–1736.
8. R. Lucito et al., "Genetic Analysis using Genomic Representations," *Proc. Nat'l Academy of Sciences USA,* vol. 95, no. 8, 14 Apr. 1998, pp. 4487–4492.

Bud Mishra is a professor of computer science and mathematics at New York University's Courant Institute and a professor at Cold Spring Harbor Laboratory. He is also a cofounder of OpGen Inc., a biotechnology company specializing in the construction of genomewide restriction maps using single molecules. His current research focuses on modeling cellular and genomic evolutionary processes; Valis, a bioinformatics language and environment; single-molecule-based technology for mapping, sequencing, karyotyping, and haplotyping; microarray-based technology for correspondence mapping; and gene-expression analysis with applications to cancer study. Mishra holds an ISc in physics from Utkal University, a BTech in electronics and electrical communication engineering from IIT, and an MS and a PhD in computer science from Carnegie Mellon University. Contact him at Courant Institute of Mathematical Sciences, NYU, 251 Mercer St., New York, NY 10012; mishra@nyu.edu.

Further Reading

For the basic ideas of the algorithm described in this article and their extension to create genomewide maps of probes, see Will Casey, Bud Mishra, and Mike Wigler.[1] Other researchers have published the algorithms for and the algorithmic complexity of constructing probe maps, RH maps, and similar physical maps.[2–6] Robert Lucito and his colleagues have published the experimental work as well as the underlying foundations for detecting gene copy number fluctuations.[7] Several publications cover related ideas, such as the low-complexity representation of genomes, cloning genomic differences, application to genetic analysis, and so on.[7–9] Finally, the recent book by Charles Cantor and Cassandra Smith provides a good reference for the biotechnology revolution spurred by the human genome project.[10]

References

1. W. Casey, B. Mishra, and M. Wigler, "Placing Probes along the Genome Using Pairwise Distance Data," *Algorithms in Bioinformatics, Proc. First Int'l Workshop, WABI 200*, Lecture Notes in Computer Science vol. 2149, Springer-Verlag, Berlin, 2001, pp. 52–68.
2. F. Alizadeh et al., "Physical Mapping of Chromosomes Using Unique Probes," *J. Computational Biology,* vol. 2, no. 2, Summer 1995, pp. 159–185.
3. A. Ben-Dor and B. Chor, "On Constructing Radiation Hybrid Maps," *Proc. First Int'l Conf. on Research Computational Molecular Biology,* ACM Press, New York, 1997, pp. 17–26.
4. J. Håstad, L. Ivansson, and J. Lagergren, "Fitting Points on the Real Line and Its Application to RH Mapping," *Algorithms—ESA '98*, Lecture Notes in Computer Science, vol. 1461, Springer-Verlag, Berlin, 1998, pp. 465–467.
5. M. Jain and E.W. Myers, "Algorithms for Computing and Integrating Physical Maps Using Unique Probes," *J. Computational Biology,* vol. 4, no. 4, Winter 1997, pp. 449–466.
6. D. Slonim et al., "Building Human Genome Maps with Radiation Hybrids," *J. Computational Biology,* vol. 4, no. 4, Winter 1997, pp. 487–504.
7. R. Lucito et al., "Detecting Gene Copy Number Fluctuations in Tumor Cells by Microarray Analysis of Genomic Representations," *Genome Research,* vol. 10, no. 11, Nov. 2001, pp. 1726–1736.
8. N. Lisitsyn and M. Wigler, "Cloning the Differences between Two Complex Genomes," *Science,* vol. 259, no. 5097, Feb. 1993, pp. 946–951.
9. R. Lucito et al., "Genetic Analysis using Genomic Representations," *Proc. Nat'l Academy of Science USA,* vol. 95, no. 8, 14 Apr. 1998, pp. 4487–4492.
10. C. Cantor and C. Smith, *Genomics: The Science and Technology Behind the Human Genome Project,* John Wiley & Sons, New York, 1999.