Know Thy Neighbor: LCBB

Understanding Evolution Through Computation

Bernard M.E. Moret
Nothing makes sense in biology except in the light of evolution.

Overview

• LCBB’s main focus: evolution
• Modelling choices and challenges
• Some nice results
• One tough problem
• Bringing it all back together
Overview

- LCBB’s main focus: evolution
- Modelling choices and challenges
- Some nice results
- One tough problem
- Bringing it all back together
What Kind of Evolution?

Evolution occurs across time and space and affects population, morphology, metabolism, genome, and molecules.

We focus on long-term evolution at the level of molecular characteristics:

- Thousands of generations
- DNA sequences, genomic architecture, etc.
What Are the Goals?

Sure, but...

Our interest (and expertise) is model and algorithm development.

So our goals are:

- to develop models that are useful in biological research;
- to develop models amenable to computational approaches; and
- to design and characterize algorithms for these models.
Why Should We Care?

Evolution provides *a framework* around which to organize biological data and within which to interpret them.

An evolutionary framework enables comparative approaches; in turn, these allow us to *transfer knowledge* accumulated about selected organisms to many others.

- **Necessary for a high-throughput approach.**
- **Crucial to medical research: we can learn from short-lived organisms that are also easy to keep in a lab (mice, yeast, etc.).**
Example: Co-Evolution

Parasites: lice

- Pediculus schaeffi
- Pediculus humanus
- Pthirus pubis
- Pthirus gorillae
- Pediculus
- Pediculus capitis

Symbionts: gut bacteria

- Buchnera species
- aphids

- Si. chinensis
- Me. mois
- P. betae
- M. persicae
- C. viminalis
- M. kinseyi
- U. sonchi
- U. rurale
- S. graminum
- M. persicae
- E. coli

Association (~200 MY)
Example: Comparing Two Bacteria

R. conorii (Med. spotted fever) vs. R. prowazeckii (typhus)

Example: Pathogenicity in Fungi

Soanes et al.,
The Plant Cell
Example: Dispersion of Organisms

Avian flu

Katydid (Hawaii)
Why Do We Need Computation?

Scale:
The data are accumulating very fast, thanks to high-throughput pipelines for sequencing whole genomes, genotyping individuals, gathering gene expression profiles, etc.

Complexity:
Evolution is a very complex stochastic process. Even the simplest models give rise to hard computational problems.

Integration:
Most biological data bear traces of evolution. The totality of evidence should help us improve our understanding.
Overview

- LCBB’s main focus: evolution
- Modelling choices and challenges
- Some nice results
- One tough problem
- Bringing it all back together
Computational Characteristics

Very unusual characteristics for Computer Science:

- No training set—we do not know the true history.
- Only one instance of the problem—until we discover new lifesystems elsewhere in the universe. . .
- Any optimization function is a surrogate—the truth cannot be quantified.

and some complicating attributes:

- Time scales range from minutes to billions of years.
- Nearly all (molecular) data are about current organisms.
- Parts of the data are irrelevant and some important parts are missing—but we do not know which.
Approaches to Modelling

Consider modelling rearrangements in a genome—through which the allotment of genes among chromosomes and orderings of these genes along each chromosome are altered.

We can model this type of evolution as follows:

- **no assumed model (prediction only):** use only observed results (e.g., one adjacency list of genes for each chromosome);
- **biological model (prediction and explanation):** use biologically documented operations (inversions, transpositions, translocations, fusions, fissions, etc.); or
- **mathematical model (prediction and characterization):** devise a single mathematical operator that can produce all observed results and leads to good theory.
Which Approach to Modelling?

These three choices are typical of any modelling activity.

Which one should a computer scientist choose?
Which Approach to Modelling?

These three choices are typical of any modelling activity.

Which one should a computer scientist choose? All of them, in the order given!

- The first approach is simplest and can yield early results. (Nothing is ever model-free, however: in this case, for instance, are all adjacencies of equal importance?)
- The second approach has explanatory power and will be preferred by life scientists. It is usually too complex, but parts of it may be solvable.
- The third approach comes with maturity—a good, productive mathematical model is the hardest to develop.
Overview

- LCBB’s main focus: evolution
- Modelling choices and challenges
- Some nice results
- One tough problem
- Bringing it all back together
Some Nice Results

- Sorting signed permutations by inversions in $O(n \log n)$ time.
- Reconstructing accurate phylogenies from whole-genome data.
- Refining inferred transcriptional networks using evolutionary models.
Given a signed permutation of the set $S = \{1, 2, \ldots, n\}$, find a shortest sequence of inversions (take a subsequence of the permutation and flip its order and all its signs) to transform it into the identity permutation.
Given a signed permutation of the set $S = \{1, 2, \ldots, n\}$, find a shortest sequence of inversions (take a subsequence of the permutation and flip its order and all its signs) to transform it into the identity permutation.
Given a signed permutation of the set $S = \{1, 2, \ldots, n\}$, find a shortest sequence of inversions (take a subsequence of the permutation and flip its order and all its signs) to transform it into the identity permutation.
Given a signed permutation of the set $S = \{1, 2, \ldots, n\}$, find a shortest sequence of inversions (take a subsequence of the permutation and flip its order and all its signs) to transform it into the identity permutation.
A simplified version: the (one-spatula) pancake-flipping problem. Given a stack of pancakes of different sizes, produce a stack sorted by size with the largest pancake on the bottom, using a sequence of “stack flips” (prefix reversals).

This was the only research paper published by Bill Gates (with Christos Papadimitriou, Discr. Math., 1979).

Sorting by inversions is then sorting “burnt” pancakes using two spatulas, so that all burnt sides are down.
Sorting by Inversions

Polytime alg. in 1995 (Hannenhalli & Pevzner, STOC)
runs in $O(n^5)$ time.
Subquadratic solution in 2004 (Tannier & Sagot, CPM)
runs in $O(n\sqrt{n \log n})$ time.

We gave an algorithm to sort almost all signed permutations
by inversions in $O(n \log n)$ time (RECOMB 2009).

Main ideas: (i) lazy data structure; (ii) faster to recover from errors than
to avoid (or even recognize) them; (iii) work done after the last error,
but before recovery starts, can be reused; (iv) errors are very rare.
Given a collection of “genomes,” reconstruct their evolutionary relationships in the form of an evolutionary tree. (A genome is a partition of the “genes” into chromosomes; a chromosome is a signed permutation.)

Phylogenetic reconstruction was formalized by Willi Hennig in 1950. DNA data created a huge demand for such reconstructions, so there are many software packages.

In contrast, reconstruction from rearrangement data has been limited to small collections of small genomes (10–20 genomes of 100–200 genes).
We designed two different methods to reconstruct phylogenies from complex genomes (20’000 genes) with very high accuracy (better than 95%).

One is based on a fast approximation algorithm to solve the NP-hard median problem for three genomes: given three genomes, find a fourth (the median) that minimizes the sum of the distances to the given three. The decomposition is optimality-preserving—the approximation comes about when no decomposition can be found.

The other is based on a very accurate estimation of the true evolutionary distance between two genomes and then uses a very simple algorithm to construct the tree.
We designed two different methods to reconstruct phylogenies from complex genomes (20’000 genes) with very high accuracy (better than 95%).

One is based on a fast approximation algorithm to solve the NP-hard median problem for three genomes: given three genomes, find a fourth (the median) that minimizes the sum of the distances to the given three. The decomposition is optimality-preserving—the approximation comes about when no decomposition can be found.

The other is based on a very accurate estimation of the true evolutionary distance between two genomes and then uses a very simple algorithm to construct the tree.
True Rearrangement Distances

The edit (minimum) distance between two genomes underestimates the true distance. Can we estimate the true distance from the edit distance?

We have done this for several operations, most recently for the “good” mathematical model (ISMB 2008) and for an extension that allows for gene duplication and deletion (APBC 2010).

Main idea: set up a state descriptor for a signed permutation undergoing rearrangements, using only a few variables, then write and solve the steady-state state-transition equations.
What confidence can we have in any given edge of the tree?

Nonparametric bootstrapping is used in sequence-based reconstruction. New datasets are created by resampling the input, a tree is built from each dataset, then a frequency count is made. However, this approach cannot be used with permutations.

We devised a parametric bootstrapping test. For some $\varepsilon$, apply $\varepsilon \cdot n$ random rearrangements to each input permutation, re-estimate true distances, subtract from them $2\varepsilon \cdot n$, and build a tree from the resulting distance matrix. Repeat to obtain enough samples and do a frequency count.
Overview

• LCBB’s main focus: evolution
• Modelling choices and challenges
• Some nice results
• One tough problem
• Bringing it all back together
Modelling genomic evolution under rearrangements as well as duplication and losses of “genes” (or genomic blocks).

The “blocks” must be homologous: they must be descended from the same block in some last common ancestor. But duplication creates a number of initially identical blocks, which then diverge slowly from each other within the same organism. Thus duplications, losses, and rearrangements are not independent.

Say genome A has 6 copies of block X and genome B has 4 copies—now which copy corresponds to which? It is not even automatic that each of the 4 copies of block X in genome B has a corresponding block among the 6 copies in genome A: there could have been gains as well as losses of blocks in both A and B.
Things are in fact worse.

The DNA sequences of these blocks is usually available, so sequence-based evolutionary analyses can be run. Evolution at the level of nucleotides and at the level of duplications, losses, and rearrangements must be combined into a single model.

Nothing of the kind exists at present.
Overview

• LCBB’s main focus: evolution
• Modelling choices and challenges
• Some nice results
• One tough problem
• Bringing it all back together
So why do we work on these problems?

- It is a fun mix of design, optimization, and testing.
- The models must connect at some level to biological reality.
- We constantly have to learn new things outside CS.
- Working between communities is a challenge.
- Biology moves so fast that we are never going to run out of work.
- Somebody needs to bring algorithms into this or else the data will just get warehoused.
- There is nonzero probability of getting some insights into biology.