A New Voronoi-Based Surface Reconstruction Algorithm

Nina Amenta* UT - Austin
Marshall Bern Xerox PARC
Manolis Kamvysselis† M.I.T.

Abstract

We describe our experience with a new algorithm for the reconstruction of surfaces from unorganized sample points in $\mathbb{R}^3$. The algorithm is the first for this problem with provable guarantees. Given a "good sample" from a smooth surface, the output is guaranteed to be topologically correct and convergent to the original surface as the sampling density increases. The definition of a good sample is itself interesting: the required sampling density varies locally, rigorously capturing the intuitive notion that featureless areas can be reconstructed from fewer samples. The output mesh interpolates, rather than approximates, the input points.

Our algorithm is based on the three-dimensional Voronoi diagram. Given a good program for this fundamental subroutine, the algorithm is quite easy to implement.

Keywords: Medial axis, Sampling, Delaunay triangulation, Computational Geometry

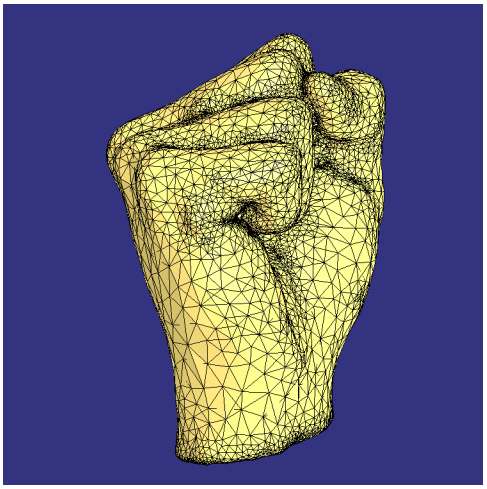

Figure 1. The fist mesh was reconstructed from the vertices alone. Notice that the sampling density varies. Our algorithm requires dense sampling only near small features; given such an input, the output mesh is provably correct.

1 Introduction

The process of turning a set of sample points in $\mathbb{R}^3$ into a computer graphics model generally involves several steps: the reconstruction of an initial piecewise-linear model, cleanup, simplification, and perhaps fitting with curved surface patches.

We focus on the first step, and in particular on an abstract problem defined by Hoppe, DeRose, Duchamp, McDonald, and Stuetzle [14]. In this formulation, the input is a set of points in $\mathbb{R}^3$, without any additional structure or organization, and the desired output is a polygonal mesh, possibly with boundary. In practice, sample sets for surface reconstruction come from a variety of sources: medical imagery, laser range scanners, contact probe digitizers, radar and seismic surveys, and mathematical models such as implicit surfaces. While the most effective reconstruction scheme for any one of these applications should take advantage of the special properties of the data, an understanding of the abstract problem should contribute to all of them.

The problem formulation above is incomplete, since presumably we should require some relationship between the input and the output. In this and a companion paper [2], we describe a simple, combinatorial algorithm for which we can prove such a relationship. A nontrivial part of this work is the fitting of precise definitions to the intuitive notions of a "good sample" and a "correct reconstruction". Although the actual definition of a good sample is rather technical, involving the medial axis of the original surface, Figure 1 gives the general idea: dense in detailed areas and (possibly) sparse in featureless ones.

The algorithm is based on the three-dimensional Voronoi diagram and Delaunay triangulation; it produces a set of triangles that we call the *crust* of the sample points. All vertices of crust triangles are sample points; in fact, all crust triangles appear in the Delaunay triangulation of the sample points.

The companion paper [2] presents our theoretical results. In that paper, we prove that given a good sample from a smooth surface, the output of our reconstruction algorithm is topologically equivalent to the surface, and that as the sampling density increases, the output converges to the surface, both pointwise and in surface normal.

Theoretical guarantees, however, do not imply that an algorithm is useful in practice. Surfaces are not everywhere smooth, samples do not everywhere meet the sampling density conditions, and sample points contain noise. Even on good inputs, an algorithm may fail to be robust, and the constants on the running time might be prohibitively large. In this paper, we report on our implementation of the algorithm, its efficiency and the quality of the output.

Overall, we were pleased. The program gave intuitively reasonable outputs on inputs for which the theoretical results do not apply. The implementation, using a freely available exact-arithmetic Voronoi diagram code, was quite easy, and reasonably efficient: it can handle 10,000 points in a matter of minutes. The main difficulty, both in theory and in practice, is the reconstruction of sharp edges.

*Much of this work was done while the author was employed by Xerox PARC, partially supported by NSF grant CCR-9404113.

†Much of this work was done while the author was an intern at Xerox PARC.

2 Related work

The idea of using Voronoi diagrams and Delaunay triangulations in surface reconstruction is not new. The well–known α-*shape* of Edelsbrunner et al. [9, 10] is a parameterized construction that associates a polyhedral shape with an unorganized set of points. A simplex (edge, triangle, or tetrahedron) is included in the α-shape if it has some circumsphere with interior empty of sample points, of radius at most α (a circumsphere of a simplex has the vertices of the simplex on its boundary). The *spectrum* of α-shapes, that is, the α-shapes for all possible values of α, gives an idea of the overall shape and natural dimensionality of the point set. Edelsbrunner and Mücke experimented with using α-shapes for surface reconstruction [10], and Bajaj, Bernardini, and Xu [4] have recently used α-shapes as a first step in the entire reconstruction pipeline.

An early Delaunay-based algorithm, similar in spirit to our own, is the "Delaunay sculpting" heuristic of Boissonnat [6], which progressively eliminates tetrahedra from the Delaunay triangulation based on their circumspheres. In two dimensions, there are a number of recent theoretical results on various Delaunay-based approaches to reconstructing smooth curves. Attali [3], Bernardini and Bajaj [5], Figueiredo and Miranda Gomes [11] and ourselves [1] have all given guarantees for different algorithms.

A fundamentally different approach to reconstruction is to use the input points to define a signed distance function on $\mathbb{R}^3$, and then polygonalize its zero-set to create the output mesh. Such *zero-set* algorithms produce approximating, rather than interpolating, meshes. This approach was taken by Hoppe et al. [14, 13] and more recently by Curless and Levoy [8]. Hoppe et al. determine an approximate tangent plane at each sample point using least squares on k nearest neighbors, and then take the signed distance to the nearest point's tangent plane as the distance function on $\mathbb{R}^3$. The distance function is then interpolated and polygonalized by the marching cubes algorithm. The algorithm of Curless and Levoy is tuned for laser range data, from which they derive error and tangent plane information. They combine the samples into a continuous volumetric function, computed and stored on a voxel grid. A subsequent hole-filling step also uses problem-specific information. Their implementation is especially fast and robust, capable of handling very large data sets.

Functionally our crust algorithm differs from both the α-shape and the zero-set algorithms. It overcomes the main drawback of α-shapes as applied to surface reconstruction, which is that the parameter α must be chosen experimentally, and in many cases there is no ideal value of α due to variations in the sampling density. The crust algorithm requires no such parameter; it in effect automatically computes the parameter locally. Allowing the sampling density to vary locally enables detailed reconstructions from much smaller input sets.

Like the α-shape, the crust can be considered an intrinsic construction on the point set. But unlike the α-shape, the crust is naturally two-dimensional. This property makes the crust more suitable for surface reconstruction, although less suitable for determining the natural dimensionality of a point set.

The crust algorithm is simpler and more direct than the zero-set approach. Zero-set algorithms, which produce approximating rather than interpolating surfaces, inherently do some low-pass filtering of the data. This is desirable in the presence of noise, but causes some loss of information. We believe that some of our ideas, particularly the sampling criterion and the normal estimation method, can be applied to zero-set algorithms as well, and might be useful in proving some zero-set algorithm correct.

With its explicit sampling criterion, our algorithm should be most useful in applications in which the sampling density is easy to control. Two examples are digitizing an object with a hand-held contact probe, where the operator can "eyeball" the required density, and polygonalizing an implicit surface using sample points [12], where the distribution can be controlled analytically.

3 Sampling Criterion

Our theoretical results assume a *smooth surface*, by which we mean a twice-differentiable manifold embedded in $\mathbb{R}^d$. Notice that this allows all orientable manifolds, including those with multiple connected components.

3.1 Geometry

We start by reviewing some standard geometric constructions. Given a discrete set S of sample points in $\mathbb{R}^d$, the *Voronoi cell* of a sample point is that part of $\mathbb{R}^d$ closer to it than to any other sample. The *Voronoi diagram* is the decomposition of $\mathbb{R}^d$ induced by the Voronoi cells. Each Voronoi cell is a convex polytope, and its vertices are the *Voronoi vertices*; when S is nondegenerate, each Voronoi vertex is equidistant from exactly $d+1$ points of S. These $d+1$ points are the vertices of the *Delaunay simplex*, dual to the Voronoi vertex. A Delaunay simplex, and hence each of its faces, has a circumsphere empty of other points of S. The set of Delaunay simplices form the *Delaunay triangulation* of S. Computing the Delaunay triangulation essentially computes the Voronoi diagram as well. See Figure 5 for two-dimensional examples.

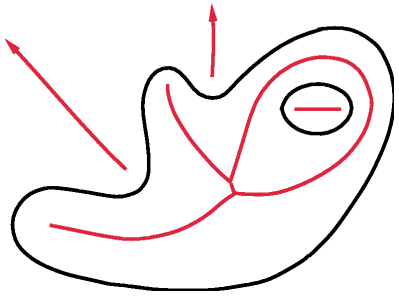

Figure 2. The red curves are the medial axis of the black curves. Notice that components of the medial axis lie on either side of the black curves.

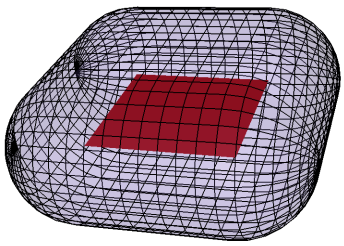

Figure 3. In three dimensions, the medial axis of a surface is generally a two-dimensional surface. Here, the square is the medial axis of the rounded transparent surface. A nonconvex surface would have components of the medial axis on the outside as well, as in the 2D example of Figure 2.

The *medial axis* of a $(d-1)$-dimensional surface in $\mathbb{R}^d$ is (the closure of) the set of points with more than one closest point on the surface. An example in $\mathbb{R}^2$ is shown in Figure 2, and in $\mathbb{R}^3$ in Figure 3. This definition of the medial axis includes components on the exterior of a closed surface. The medial axis is the extension to continuous surfaces of the Voronoi diagram, in the sense that the

Voronoi diagram of S can be defined as the set of points with more than one closest point in S.

In two dimensions, the Voronoi vertices of a dense set of sample points on a curve approximate the medial axis of the curve. Somewhat surprisingly—a number of authors have been misled—this nice property does not extend to three dimensions.

3.2 Definition

We can now describe our sampling criterion. A good sample is one in which the sampling density is (at least) inversely proportional to the distance to the medial axis. Specifically, a sample S is an *r-sample* from a surface F when the Euclidean distance from any point $p \in F$ to the nearest sample point is at most r times the distance from p to the nearest point of the medial axis of F.

The constant of proportionality r is generally less than one. In the companion paper [2], we prove our theorems for small values of r such as $r \leq .06$, but the bounds are not tight. Hence the theoretical results apply only when the sampling is very dense.

We observe that in practice $r = .5$ generally suffices. Figure 4 shows a reconstruction from a dense sample, and from a sample thinned to roughly $r = .5$. We did not compute the medial axis, which can be quite a chore. Instead, we used the distance to the nearest "pole" (see Section 4.2) as a reasonable, and easily computed, estimate of the distance to the medial axis.

Figure 4. The sampling spacing required to correctly reconstruct a surface is proportional to the distance to the medial axis. On the left is a surface reconstructed from a dense sample. The color represents estimated distance to medial axis—red means close. On the right, we use the estimated distance to thin the data to a .5-sample (meaning that the distance to the nearest sample for any point on the surface is at most half the distance to the medial axis), and then reconstruct. There were about 12K samples on the left and about 3K on the right.

Notice that our sampling criterion places no constraints on the distribution of points, so long as they are sufficiently dense. It inherently takes into account both the curvature of the surface—the medial axis is close to the surface where the curvature is high—and also the proximity of other parts of the surface. For instance, although the middle of a thin plate has low curvature, it must be sampled densely to resolve the two sides as separate surfaces. In this situation an r-sample differs from the distribution of vertices typically produced by mesh simplification algorithms, which only need to consider curvature.

At sharp edges and corners, the medial axis actually touches the surface. Accordingly, our criterion requires infinitely dense sampling to guarantee reconstruction. Sharp edges are indeed a problem in practice as well, although the reconstruction errors are not noticeable when the sampling is very dense. We discuss a heuristic approach to resolving sharp edges in Section 6, and propose a stronger theoretical approach in Section 7.

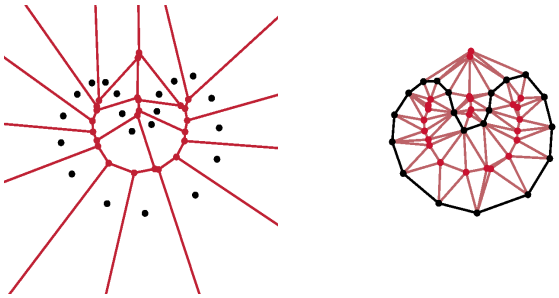

Figure 5. The two-dimensional algorithm. On the left, the Voronoi diagram of a point set S sampled from a curve. Just as S approximates the curve, the Voronoi vertices V approximate the medial axis of the curve. On the right, the the Delaunay triangulation of $S \cup V$, with the crust edges in black. Theorem 1 states that when S is an r-sample, for sufficiently small r, the crust edges connect only adjacent vertices.

4 The crust algorithm

4.1 Two Dimensions

We begin with a two-dimensional version of the algorithm [1]. In this case, the crust will be a graph on the set of sample points S. We define the crust as follows: an edge e belongs to the crust if e has a circumcircle empty not only of all other sample points but also of all Voronoi vertices of S. The crust obeys the following theorem [1].

Theorem 1. *The crust of an r-sample from a smooth curve F, for $r \leq .25$, connects only adjacent sample points on F.*

The medial axis provides the intuition behind this theorem. An important lemma is that for any sample S, an edge between two nonadjacent sample points cannot be circumscribed by a circle that misses both the medial axis and all other samples. When S is an r-sample for sufficiently small r, the Voronoi vertices approximate the medial axis, and any circumcircle of an edge between nonadjacent samples contains either another sample or a Voronoi vertex. An edge between two adjacent samples, on the other hand, is circumscribed by a small circle, far away from the medial axis and hence from all Voronoi vertices.

The definition of the two-dimensional crust leads to the following simple algorithm, illustrated in Figure 5. First compute the Voronoi diagram of S, and let V be the set of Voronoi vertices. Then compute the Delaunay triangulation of $S \cup V$. The crust consists of the Delaunay edges between points of S, since those are the edges with circumcircles empty of points in $S \cup V$. Notice that the crust is also a subset of the Delaunay triangulation of the input points; adding the Voronoi vertices filters out the unwanted edges from the Delaunay triangulation. We call this technique *Voronoi filtering*.

4.2 Three Dimensions

This simple Voronoi filtering algorithm runs into a snag in three dimensions. The nice property that all the Voronoi vertices of a sufficiently dense sample lie near the medial axis is no longer true. Figure 6 shows an example. No matter how densely we sample, Voronoi vertices can appear arbitrarily close to the surface.

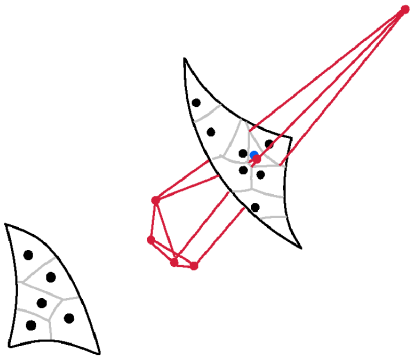

Figure 6. In three dimensions, we can use only a subset of the Voronoi vertices, since not all Voronoi vertices contribute to the approximation of the medial axis. Here, one sample on a curved surface is colored blue, and the edges of its three-dimensional Voronoi cell are drawn in red. One red Voronoi vertex lies near the surface, equidistant from the four samples near the center. The others lie near the medial axis, near the center of curvature on one side and halfway to an opposite patch of the surface on the other.

On the other hand, many of the three-dimensional Voronoi vertices *do* lie near the medial axis. Consider the Voronoi cell V_s of a sample s, as in Figure 6. The sample s is surrounded on F by other samples, and V_s is bounded by bisecting planes separating s from its neighbors, each plane nearly perpendicular to F. So the Voronoi cell V_s is long, thin and roughly perpendicular to F at s. V_s extends perpendicularly out to the medial axis. Near the medial axis, other samples on F become closer than s, and V_s is cut off. This guarantees that some vertices of V_s lie near the medial axis. We give a precise and quantitative version of this rough argument in [2].

This leads to the following algorithm. Instead of using all of the Voronoi vertices in the Voronoi filtering step, for each sample s we use only the two vertices of V_s farthest from s, one on either side of the surface F. We call these the *poles* of s, and denote them p^+ and p^-. It is easy to find one pole, say p^+: the farthest vertex of V_s from s. The observation that V_s is long and thin implies that the other pole p^- must lie roughly in the opposite direction. Thus in the basic algorithm below, we simply choose p^- to be farthest vertex from s such that sp^- and sp^+ have negative dot-product. Here is the basic algorithm:

1. *Compute the Voronoi diagram of the sample points S*
2. *For each sample point s do:*
   (a) *If s does not lie on the convex hull, let p^+ be the farthest Voronoi vertex of V_s from s. Let n^+ be the vector $sp+$.*
   (b) *If s lies on the convex hull, let n^+ be the average of the outer normals of the adjacent triangles.*
   (c) *Let p^- be the Voronoi vertex of V_s with negative projection on n^+ that is farthest from s.*
3. *Let P be the set of all poles p^+ and p^-. Compute the Delaunay triangulation of $S \cup P$.*
4. *Keep only those triangles for which all three vertices are sample points in S.*

Notice that one does not need an estimate of r to use the crust algorithm; the basic algorithm requires no tunable parameters at all. The output of this algorithm, the *three-dimensional crust*, is a set of triangles that resembles the input surface geometrically. More precisely, we prove the following theorem [2].

Theorem 2. *Let S be an r-sample from a smooth surface F, for $r \leq .06$. Then 1) the crust of S contains a set of triangles forming a mesh topologically equivalent to F, and 2) every point on the crust lies within distance $5r \cdot d(p)$ of some point p on F, where $d(p)$ is the distance from p to the medial axis.*

The crust, however, is not necessarily a manifold; for example, it often contains all four triangles of a very flat "sliver" tetrahedron. It is, however, a visually acceptable model.

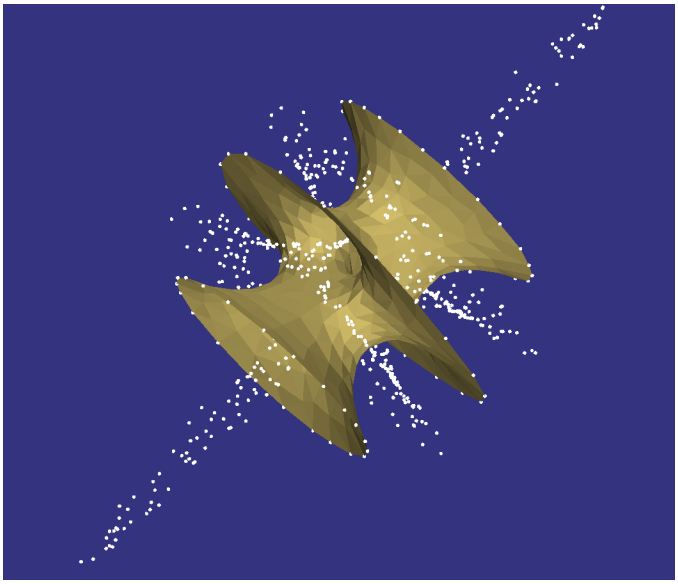

Figure 7. The crust of a set of sample points and the poles (white points) used in its reconstruction. Each sample selects the two vertices of its Voronoi cell that are farthest away, one on either side of the surface, as poles. The poles lie near the medial axis of the surface, sketching planes separating opposite sheets of surface that degenerate to one-dimensional curves where the cross-section of the surface is circular.

4.3 Normal Estimation and Filtering

Additional filtering is required to produce a guaranteed piecewise-linear manifold homeomorphic to F, and to ensure that the output converges in surface normal as the sampling density increases.

In fact, whatever the sampling density, the algorithm above may output some very thin crust triangles nearly perpendicular to the surface. We have an important lemma [2], however, which states that the vectors $n^+ = sp^+$ and $n^- = sp^-$ from a sample point to its poles are guaranteed to be nearly orthogonal to the surface at s. The angular error is linear in r. The intuition (put nicely by Ken Clarkson) is that the surface normal is easy to estimate from a point far away, such as a pole p, since the surface must be nearly normal to the largest empty ball centered at p.

We can use these vectors in an additional *normal filtering* step, throwing out any triangles whose normals differ too much from n^+ or n^-. When normal filtering is used, the normals of the output triangles approach the surface normals as the sampling density increases. We prove in [2] that the remaining set of triangles still contains a subset forming a piecewise-linear surface homeomorphic to F.

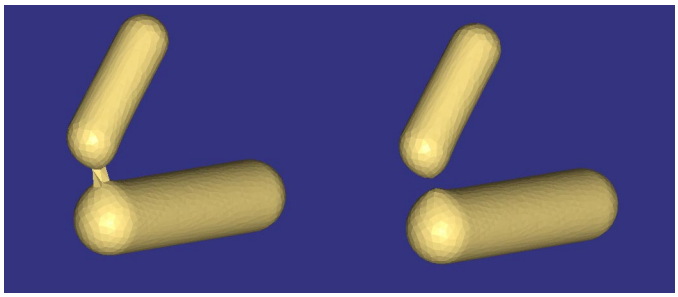

Figure 8. The crust of points distributed on an implicit surface (left). The additional normal filtering step is needed to separate the two connected components (right), which are undersampled at their closest point. Triangles are deleted if their normals differ too much from the direction vectors from the triangle vertices to their poles. These vectors are provably close to the surface normals.

Normal filtering can be useful in practice as well, as shown in Figure 8. In the usual case in which r is unknown the allowable difference in angle must be selected experimentally. Normal filtering can be dangerous, however, at boundaries and sharp edges. The directions of n^+ and n^- are not nearly normal to all nearby tangent planes, and desirable triangles might be deleted.

We note that n^+ and n^-, our Voronoi-based estimates of normal direction, could be useful in the zero-set reconstruction methods, which depend on accurate estimation of the tangent planes. For the algorithm of Hoppe et al. [14], a Voronoi-based estimate could replace the estimate based on the k-nearest neighbors. The Voronoi-based estimate has the advantage that it is not sensitive to the distribution; whereas, for instance, on medical image data, all k nearest neighbors might lie in the same slice, and so would the estimated tangent plane. In the algorithm of Curless and Levoy [8], the Voronoi-based estimate could be checked against the bounds on normal direction derived from the laser-range scanner.

4.4 Manifold Extraction

After the normal filtering step, all the remaining triangles are roughly parallel to the surface. We can define a sharp edge as one which is adjacent to triangles only on one side of a plane through the edge and roughly perpendicular to the surface. Notice that an edge of degree one counts as a sharp edge. If the surface F is indeed a smooth manifold without boundary, we are guaranteed that the normal-filtered crust contains a piecewise-linear manifold homeomorphic to F. Any triangle adjacent to a sharp edge cannot belong to this piecewise-linear manifold, and can be safely deleted. We continue recursively until no such triangle remains. A piecewise-linear manifold can then be obtained by a *manifold extraction* step which takes the outside surface of the remaining triangles on each connected component. This simple approach, however, cannot be applied when F is not a smooth manifold without boundary. In that case we do not know how to prove that we can extract a manifold homeomorphic to F.

4.5 Complexity

The asymptotic complexity of the crust algorithm is $O(n^2)$ where $n = |S|$, since that is the worst-case time required to compute a three-dimensional Delaunay triangulation. Notice that the number of sample points plus poles is at most $3n$. As has been frequently observed, the worst-case complexity for the three-dimensional Delaunay triangulation almost never arises in practice. All other steps are linear time.

5 Implementation

5.1 Numerical Issues

Robustness has traditionally been a concern when implementing combinatorial algorithms like this one. Our straightforward implementation, however, is very robust. This success is due in large part to the rapidly improving state of the art in Delaunay triangulation programs. We used Clarkson's *Hull* program. *Hull* uses exact integer arithmetic, and hence is thoroughly robust, produces exact output, and requires no arithmetic tolerancing parameters. The performance cost for the exact arithmetic is fairly modest, due to a clever adaptive precision scheme. We chose *Hull* so that we could be sure that numerical problems that arose were our own and did not originate in the triangulation. Finding the exact Delaunay triangulation is not essential to our algorithm.

Hull outputs a list of Delaunay tetrahedra, but not the coordinates of their circumcenters (the dual Voronoi vertices) which always contain some roundoff error. Fortunately, the exact positions of the poles are not important, as the numerical error is tiny relative to the distance between the poles and the surface. We computed the location of each Voronoi vertex by solving a 4×4 linear system with a solver from *LAPACK*. The solver also returns the condition number of the coefficient matrix, which we used to reject unreliable Voronoi vertices. Rejected Voronoi vertices were almost always circumcenters of "slivers" (nearly planar tetrahedra) lying flat on the surface; for a good sample such vertices cannot be poles. It is possible that this method also rejects some valid poles induced by very flat tetrahedra spanning two patches of surface. We have not, however, observed any problems in practice. Presumably there is always another Voronoi vertex nearby that makes an equally good pole.

5.2 Efficiency

Running times for the reconstruction of some large data sets are given in the table below; the reconstructions are shown in Figure 9. We used an SGI Onyx with 512M of memory.

Model	Time (min)	Num. Pts.
Femur	2	939
Golf club	12	16864
Foot	15	20021
Bunny	23	35947

The running time is dominated by the time required to compute the Delaunay triangulations. *Hull* uses an incremental algorithm [7], so the running time is sensitive to the input order of the vertices. The triangulation algorithm builds a search structure concurrently with the triangulation itself; the process is analogous to sorting by incrementally building a binary search tree. When points are added in random order, the search structure is balanced (with extremely high probability) and the expected running time is optimal. In practice, random insertions are slow on large inputs, since both the search structure and the Delaunay triangulation begin paging. We obtained better performance by first inserting a random subset of a few thousand points to provide a balanced initial search structure, and then inserting the remaining points based on a crude spatial subdivision to improve locality.

Most likely much greater improvements in efficiency can be achieved by switching to a three-dimensional Delaunay triangulation program that, first, does not use exact arithmetic, and second, uses an algorithm with more locality of reference.

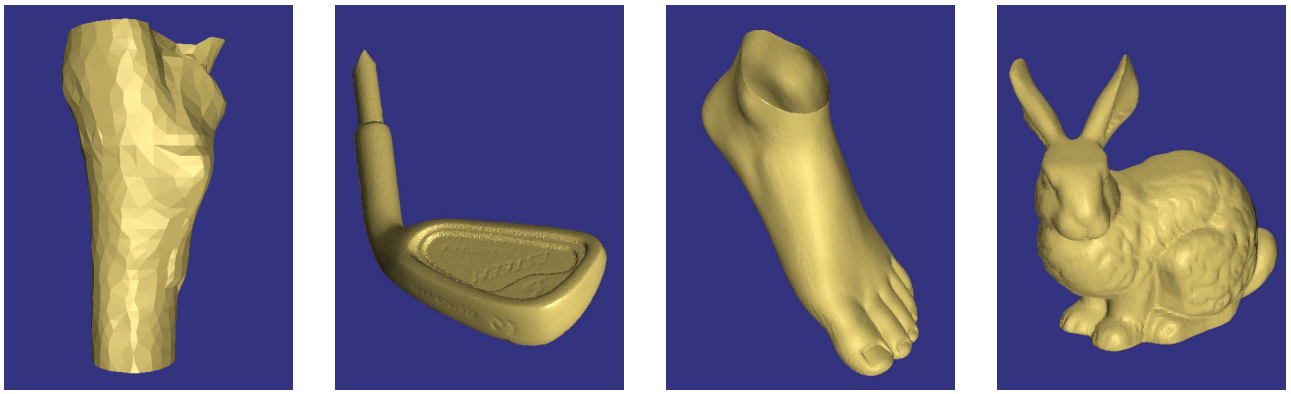

Figure 9. Femur, golf club, foot and bunny reconstructions. Notice the subtle "3" on the bottom of the club (apparently a 3-iron), showing the sensitivity of the algorithm. The foot, like all our reconstructions, is hollow. The bunny was reconstructed from the roughly 36K vertices of the densest of the Stanford bunny models in 23 minutes.

6 Heuristic Modifications

As we have noted, our algorithm does not do well at sharp edges, either in theory or in practice. The reason is that the Voronoi cell of a sample s on a sharp edge is not long and thin, so that the assumptions under which we choose the poles is not correct. For example, the Voronoi cell of a sample s on a right-angled edge is roughly fan-shaped. The vector n^+ directed towards the first pole of s might be perpendicular to one tangent plane at s, but parallel to the other. The second pole would then be chosen very near the surface, punching a hole in the output mesh.

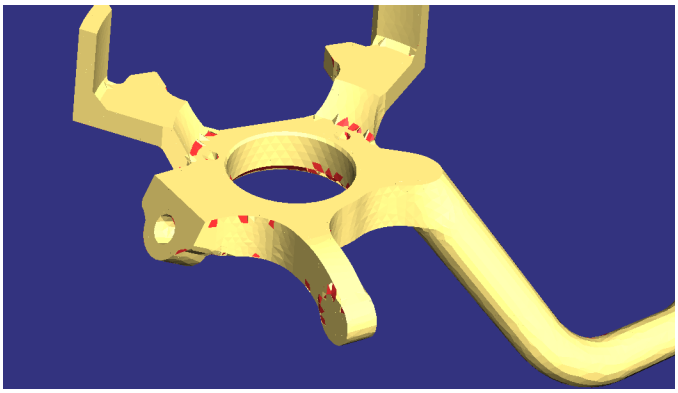

Figure 10. We resolve the sharp edges on this model of a mechanical part by using the two farthest Voronoi vertices as poles, regardless of direction. The basic algorithm forces the poles to lie in opposite directions, but is only guaranteed to work properly on a smooth surface. The red triangles do not appear in the reconstruction when using the basic algorithm.

We experimented with other methods for choosing the second pole. We found that choosing as p^- the Voronoi vertex with the greatest negative projection in the direction n^+ gave somewhat better results. This modification should retain the theoretical guarantees of the original algorithm. The best reconstructions, however, were produced by a different heuristic: choosing the farthest and the second farthest Voronoi vertices, regardless of direction, as the two poles (see Figure 10). This heuristic is strongly biased against choosing poles near the surface, avoiding gaps near sharp edges but sometimes allowing excess triangles filling in sharp corners. We believe that pathological cases could be constructed in which this fill causes a topologically incorrect reconstruction irrespective of the sampling density.

Boundaries pose similar problems in theory, but the reconstructions produced by the crust algorithm on surfaces with boundaries are usually acceptable. Figure 7 and the foot in Figure 9 are examples of perfectly reconstructed boundaries. When the boundary forms a hole in an otherwise flat surface, with no other parts of the surface nearby, the crust algorithm fills in the hole.

Undersampling also causes holes in the output mesh. For example, consider a sample in the middle of a a flat plate. Although its second pole lies in the correct direction, if there are two few sample points on the opposite side of the plate, the pole may fall near the surface on the opposite side and cause a hole. We experimented with heuristics to compensate for this undersampling effect, and for similar reconstruction errors in undersampled cylindrical regions. We found that moving all poles closer to their samples by some constant fraction allowed thin plates and cylinders to be reconstructed from fewer samples, while sometimes introducing new holes on other parts of the model. We were sometimes able to get a perfect reconstruction by taking the union of a crust made with this modification and one without.

7 Research Directions

We have identified a number of future research directions.

7.1 Noise

Small perturbations of the input points do not cause problems for the crust algorithm, nor do a few outliers. But when the noise level is roughly the same as the sampling density, the algorithm fails, both in theory and in practice. We believe, however, that there is a Voronoi-based algorithm, perhaps combining aspects of crusts and α-shapes, that reconstructs noisy data into a "thickened surface" containing all the input points, some of them possibly in the interior. See Melkemi [15] for some suggestive experimental work in $\mathbb{R}^2$.

7.2 Sharp Edges and Boundaries

We would like to modify the crust algorithm to handle surfaces with sharp edges and to provide theoretical guarantees for the reconstruction of both sharp edges and boundaries. Interpolating reconstruction algorithms like ours have an advantage here, since approximating reconstruction algorithms smooth out sharp edges. One important goal is to develop reliable techniques for identifying samples that lie on sharp edges or boundaries. As noted, the Voronoi cells of such samples are not long and thin. This intuition

could be made precise, and perhaps combined with more traditional filtering techniques.

7.3 Using Surface Normals

A variation on the problem is the reconstruction of surfaces from unorganized points that are equipped with normal directions. This problem arises in two-dimensional image processing when connecting edge pixels into edges. In three dimensions, laser range data comes with some normal information, and we have exact normals for points distributed on implicit surfaces. It should be possible to show that with this additional information, reconstruction is possible from much sparser samples. In particular, when normals are available, dense sampling should not be needed to resolve the two sides of a thin plate, suggesting that a different sampling criterion than distance to medial axis is required.

7.4 Compression

One intriguing potential application (pointed out by Frank Bossen) of interpolating, rather than approximating reconstruction, is that it can be used as a lossless mesh compression technique. A model created by interpolating reconstruction can be represented entirely by its vertices, and no connectivity information at all must be stored. A model which differs only slightly from the reconstruction of its vertices can be represented by the vertices and a short list of differences. These differences might be encoded efficiently using some geometrically defined measure of "likelihood" on Delaunay triangles. The vertices themselves could then be ordered so as to optimize properties such as compressibility or progressive reconstruction by an incremental algorithm. With the current best geometry compression method [16], most of the bits are already used to encode the vertex positions, rather than connectivity, but the connectivity is encoded in the ordering of the vertices. Allowing arbitrary vertex orderings could improve compression; we are experimenting with an octree encoding.

Our current crust algorithm is not incremental, and our implementation is too slow for real-time decompression, so this application motivates work in both directions.

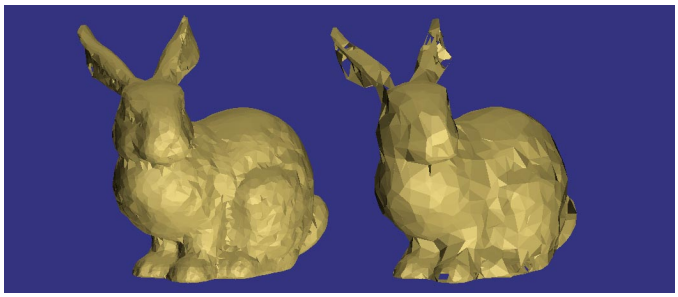

Figure 11. Reconstructions from subsets of the samples resemble the final reconstructions. The crust of the first 5 % of the points in an octree encoding of the bunny samples is still quite recognizable (right); the crust of 20 % of the points is on the left. Rough reconstructions like these could be shown during progressive transmission.

Acknowledgments

We thank David Eppstein (UC–Irvine) for his collaboration in the early stages of this research, and Frank Bossen (EPF–Lausanne) and Ken Clarkson (Lucent) for interesting suggestions. We thank Ping Fu (Raindrop Geomagic) for the fist and the mechanical part, Hughes Hoppe (Microsoft) for the head, the golf club and the foot, Chandrajit Bajaj (UT–Austin) for the femur, Paul Heckbert (CMU) for the hot dogs, and the Stanford Data Repository for the bunny. We thank Ken Clarkson and Lucent Bell Labs for *Hull*, and The Geometry Center at the University of Minnesota for *Geomview*, which we used for viewing and rendering the models.

References

[1] Nina Amenta, Marshall Bern and David Eppstein. The Crust and the β-Skeleton: Combinatorial Curve Reconstruction. To appear in *Graphical Models and Image Processing*.

[2] Nina Amenta and Marshall Bern. Surface reconstruction by Voronoi filtering. To appear in *14th ACM Symposium on Computation Geometry*, June 1998.

[3] D. Attali. r-Regular Shape Reconstruction from Unorganized Points. In *13th ACM Symposium on Computational Geometry*, pages 248–253, June 1997.

[4] C. Bajaj, F. Bernardini, and G. Xu. Automatic Reconstruction of Surfaces and Scalar Fields from 3D Scans. *SIGGRAPH '95 Proceedings*, pages 109–118, July 1995.

[5] F. Bernardini and C. Bajaj. Sampling and reconstructing manifolds using α-shapes, In *9th Canadian Conference on Computational Geometry*, pages 193–198, August 1997.

[6] J-D. Boissonnat. Geometric structures for three-dimensional shape reconstruction, *ACM Transactions on Graphics* 3: 266–286, 1984.

[7] K. Clarkson, K. Mehlhorn and R. Seidel. Four results on randomized incremental constructions. *Computational Geometry: Theory and Applications*, pages 185–121, 1993.

[8] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH '96 Proceedings*, pages 303–312, July 1996.

[9] H. Edelsbrunner, D.G. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane, *IEEE Transactions on Information Theory* 29:551-559, (1983).

[10] H. Edelsbrunner and E. P. Mücke. Three-dimensional Alpha Shapes. *ACM Transactions on Graphics* 13:43–72, 1994.

[11] L. H. de Figueiredo and J. de Miranda Gomes. Computational morphology of curves. *Visual Computer* 11:105–112, 1995.

[12] A. Witkin and P. Heckbert. Using particles to sample and control implicit surfaces, In *SIGGRAPH '94 Proceedings*, pages 269–277, July 1994.

[13] H. Hoppe. Surface Reconstruction from Unorganized Points. Ph.D. Thesis, Computer Science and Engineering, University of Washington, 1994.

[14] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface Reconstruction from Unorganized Points. In *SIGGRAPH '92 Proceedings*, pages 71–78, July 1992.

[15] M. Melkemi, $\mathcal{A}$-shapes and their derivatives, In *13th ACM Symposium on Computational Geometry*, pages 367–369, June 1997

[16] G. Taubin and J. Rossignac. Geometric compression through topological surgery. *Research Report RC20340*, IBM, 1996.

Whole-genome Comparative Annotation and Regulatory Motif Discovery in Multiple Yeast Species

Manolis Kamvysselis[1,2], Nick Patterson[1], Bruce Birren[1], Bonnie Berger[2,3,5], Eric Lander[1,4,5]

manoli@mit.edu, nickp@genome.wi.mit.edu, birren@wi.mit.edu, bab@mit.edu, lander@wi.mit.edu

(1) MIT/Whitehead Institute Center for Genome Research, 320 Charles St., Cambridge MA 02139

(2) MIT Lab for Computer Science, 200 Technology Square, Cambridge MA 02139

(3) MIT Department of Mathematics, 77 Massachusetts Ave, Cambridge MA 02139

(4) MIT Department of Biology, 31 Ames St, Cambridge MA 02139

(5) Corresponding author

ABSTRACT

In [13] we reported the genome sequences of *S. paradoxus*, *S. mikatae* and *S. bayanus* and compared these three yeast species to their close relative, *S. cerevisiae*. Genome-wide comparative analysis allowed the identification of functionally important sequences, both coding and non-coding. In this companion paper we describe the mathematical and algorithmic results underpinning the analysis of these genomes.

We developed methods for the automatic comparative annotation of the four species and the determination of orthologous genes and intergenic regions. The algorithms enabled the automatic identification of orthologs for more than 90% of genes despite the large number of duplicated genes in the yeast genome, and the discovery of recent gene family expansions and genome rearrangements. We also developed a test to validate computationally predicted protein-coding genes based on their patterns of nucleotide conservation. The method has high specificity and sensitivity, and enabled us to revisit the current annotation of *S.cerevisiae* with important biological implications.

We developed statistical methods for the systematic de-novo identification of regulatory motifs. Without making use of co-regulated gene sets, we discovered virtually all previously known DNA regulatory motifs as well as several noteworthy novel motifs. With the additional use of gene ontology information, expression clusters and transcription factor binding profiles, we assigned candidate functions to the novel motifs discovered.

Our results demonstrate that entirely automatic genome-wide annotation, gene validation, and discovery of regulatory motifs is possible. Our findings are validated by the extensive experimental knowledge in yeast, confirming their applicability to other genomes.

Categories and Subject Descriptors

J.3 [Life and medical sciences]: Biology and Genetics

General Terms: Algorithms.

Keywords: Computational biology, Comparative genomics, Genome annotation, Regulatory motif discovery.

RECOMB '03, April 10-13, 2003, Berlin, Germany.

1. INTRODUCTION

With the availability of complete sequences for a number of model organisms, comparative analysis becomes an invaluable tool for understanding genomes. Complete genomes allow for global views and multiple genomes increase predictive power.

In [13] we used a comparative genomics approach to systematically discover the full set of conserved genes and regulatory elements in yeast. We sequenced and assembled three novel yeast species, *S.paradoxus*, *S.mikatae* and *S.bayanus* and compared them to their close relative *S. cerevisiae*. The work represented the first genome-wide comparison of four complete eukaryotic genomes. This paper focuses on the mathematical and algorithmic developments underpinning the work.

First, we describe our methods for resolving the gene correspondence between each of the newly sequenced species and *S. cerevisiae* to identify orthologous regions and validate predicted protein-coding genes. We then describe our methods to identify conserved intergenic sequence elements within these regions and to cluster them into a small number of regulatory motifs.

The gene correspondence method presented here was used for the automatic annotation of the three newly sequenced species, and correctly identified unambiguous orthologs for more than 90% of protein coding genes. It also correctly identified the evolutionary events that separate the four species, discerning segmental duplications and gene loss, while correctly resolving genes that duplicated before the divergence of the species compared.

The methods for regulatory motif discovery presented here do not rely on previous knowledge of co-regulated sets of genes, and in that way differ from the current literature on computational motif discovery. The motifs discovered include most previously published regulatory motifs, adding confidence to our method. Moreover, a number of novel motifs are discovered that appear near functionally related genes. We have used the extensive experimental knowledge in yeast to validate our results, thus confirming that the methods presented here are applicable to other species.

1.1 Comparative annotation: graph separation

The first issue in comparative genomics is determining the correct correspondence of functional elements across the species compared. We decided to use predicted protein coding genes as genomic anchors in order to align and compare the species. Resolving the correspondence between ~6000 predicted genes in each species requires an algorithm for comparative annotation that

accounts for gene duplication and loss, and ensures that the 1:1 matches established are true orthologs.

Previously described algorithms for comparing gene sets have been widely used for various purposes, but they were not applicable to the problem at hand. Best Bidirectional Hits (BBH) [6, 7] looks for gene pairs that are best matches of each other and marks them as orthologs. In the case of a recent gene duplication, only one of the duplicated genes will be marked as the ortholog without signaling the presence of additional homologs. Clusters of Orthologous Genes (COG) [22, 23] goes a step further and allows many-to-many orthologous matches. It is able to capture gene duplication events when both copies of a duplicated gene have the same best hit in two other species that are themselves orthologous. It still suffers though from having slight changes in similarity influence a hard decision of a single best match. Moreover, since *Saccharomyces* underwent a whole-genome duplication event [14] before the divergence of the species compared, individual COGs currently contain both copies of each duplicated pair of genes in a single cluster of orthology, and hence was not applicable in our pairwise comparative annotation.

The comparative annotation algorithm we developed has features that make it useful in many applications. It compares two genomes at a time, and hence can be applied at any range of evolutionary distances, without requiring a balanced phylogenetic tree. Moreover, at its core, it represents the best match of every gene as a set of genes instead of a single best hit, which makes it more robust to slight differences in sequence similarity. Also, it groups the genes into progressively smaller subsets, retaining ambiguities until later in the pipeline when more information becomes available. It progressively refines the synteny map of conserved gene order while resolving ambiguities, one task helping the other. When it terminates, it returns the one-to-one orthologous pairs resolved, as well as sets of genes whose correspondence remains ambiguous in a small number of homology groups.

We applied this algorithm to automatically annotate the assemblies of the three species of yeast. Our Python implementation terminated within minutes for any of the pairwise comparisons. It successfully resolved the graph of sequence similarities between the four species, and found important biological implications in the resulting graph structure. More than 90% of genes were connected in a one-to-one correspondence, and groups of homologous proteins were isolated in small subgraphs. These contain expanding gene families that are often found in rapidly recombining regions near the telomeres, and genes involved in environmental adaptation, such as sugar transport and cell surface adhesion [13]. Not surprisingly, transposon proteins formed the largest homology groups.

This algorithm has also been applied to species at much larger evolutionary distances, with very successful results (Kamvysselis and Lander, unpublished). Despite hundreds of rearrangements and duplicated genes separating *S.cerevisiae* and *K.yarowii*, it successfully uncovered the correct gene correspondence between the two species that are more than 100 million years apart.

Finally, the algorithm works well with unfinished genomes. By working with sets of genes instead of one-to-one matches, this algorithm correctly groups in a single orthologous set all portions of genes that are interrupted by sequence gaps and split in two or multiple contigs. A best bi-directional hit would match only the longest portion and leave part of a gene unmatched. Finally, since synteny blocks are only built on one-to-one unambiguous matches, the algorithm is robust to sequence contamination. A contaminating contig will have no unambiguous matches (since all features will also be present in genuine contigs from the species), and hence will never be used to build a synteny block. This has allowed the true orthologs to be determined and the contaminating sequences to be marked as paralogs.

This algorithm provides a good solution to comparative genome annotation, works well at a range of evolutionary distances, and is robust to sequencing artifacts of unfinished genomes.

1.2 Motif discovery: signal from noise

Having accounted for the evolutionary events that gave rise to the gene sets in each species, we can align orthologous genes and intergenic regions and use the multiple alignments to discover conserved features, and in particular regulatory motifs. This amounts to extracting small sequence signals hidden within largely non-functional intergenic sequences. This problem is difficult in a single genome where the signal-to-noise ratio is very small.

Traditional methods for regulatory motif discovery have addressed the signal-to-noise problem by focusing on small subsets of co-regulated genes whose promoter regions are enriched in regulatory motifs. A number of elegant algorithms have been developed to search for subtle sequence signals within unaligned sequences, pioneered by Lawrence and coworkers [15], and made popular in programs such as AlignACE [11, 20, 24], MEME [10] or BioProspector [17]. More recent work has presented additional statistical methods for motif discovery using phylogenetic footprinting [3, 12, 18, 26]. Computational methods have also been developed for finding groups of possibly co-regulated genes that share similar expression profiles in a number of experimental conditions [8]. Additional experimental methods to find co-regulated genes include genome-wide discovery of promoter regions bound by a tagged transcription factor in chromatin IP experiments [16, 21], proteins found in the same protein complex obtained by MS [9] and proteins involved in the same genetically defined pathway [19]. Together, these experiments have allowed the elucidation of a large number of regulatory motifs in yeast [28] that have been categorized in promoter databases [27, 29].

Known regulatory motifs are short and sometimes degenerate, and hence appear frequently throughout the genome, often by chance alone, other times with a functional role. Phylogenetic footprinting has been used to distinguish between functional and non-functional instances, by observing alignments of orthologous promoters across multiple genomes [4]. The functional sites are constrained to contain the motifs since their change disrupts regulation which is detrimental to the organism, whereas non-functional sites are free to change and accumulate mutations.

The use of comparative information thus provides additional information that can help us separate signal from noise. This, together with a genome-wide view of the complete set of aligned orthologous intergenic regions, allows us to approach motif discovery at the genome-wide level. We are no longer constrained to observing subsets of co-regulated genes, but can search for regulatory motifs in all 6000 intergenic regions simultaneously for those sequences that are preferentially

conserved. We can then provide a global view of regulatory sequences that is not constrained by the experimental conditions generated in the laboratory, but instead captures the entire evolutionary history since the divergence of the species compared.

Our motif discovery strategy consists of an exhaustive enumeration and testing of short sequence patterns to find unusually conserved motif cores, followed by a motif refinement and collapsing step that ultimately produces a small number of full motifs. We used three different genome-wide statistics of non-random conservation to select motif cores from a large exhaustive set of short sequence patterns. We extended these cores with correlated surrounding bases that are frequently conserved, and collapsed them hierarchically based on sequence similarity and genome-wide co-occurrence. The final list of 72 genome-wide motifs includes most previously published regulatory motifs, as well as additional motifs that correlate strongly with experimental data.

Our results provide a global view of functionally important regulatory motifs, and provides an important link between protein interaction networks, clusters of gene expression, and transcription binding profiles towards understanding the dynamic nature of the cell and the complexity of regulatory interactions.

2. COMPARATIVE ANNOTATION

The first step to comparative genomics is understanding the correspondence between genes and other functional features across the species compared. Each species is under selective pressure to conserve the sequence of functionally important regions. We can begin to understand these pressures by observing the patterns of change in the sequence of orthologous regions.

In presence of gene duplication however, some of the evolutionary constraints a region is under are relieved, and uniform models of evolution no longer capture the underlying selection for these sites. Hence, before any type of motif discovery, we needed to identify unambiguously all orthologous sequences across the four genomes as a guide to our subsequent work.

We used genes as discrete genomic anchors to construct a large-scale alignment. The anchors were then used to construct a nucleotide-level alignment of genes and flanking intergenic regions. With the full assemblies of the yeast species available, we predicted all Open Reading Frames (ORFs) of at least 50 amino acids in each of the newly sequenced species, and compared the predicted proteins to the annotated proteins of S. cerevisiae using protein BLAST [1]. Since every predicted protein typically matched multiple *S.cerevisiae* genes, we first had to resolve the resulting ambiguities.

We formulated the problem of genome-wide gene correspondence in a graph-theoretic framework. We represented the similarities between the genes as a bipartite graph connecting genes between two species (Figure 1). We weighted every edge connecting two genes by the sequence similarity between the two genes, and the overall length of the match. We separated this graph into progressively smaller subgraphs until the only remaining matches connected true orthologs. To achieve this separation, we eliminated edges that are sub-optimal in a series of steps. As a pre-processing step, we eliminated all edges that are not within 20% of the maximum-weight edge incident to each node. We then separated the resulting graph into connected components, and built blocks of conserved gene order (synteny) when neighboring genes in one species had one-to-one matches to neighboring genes in the other species. We used these blocks of conserved synteny to resolve additional ambiguities by preferentially keeping syntenic edges incident to a node, and eliminating its non-syntenic edges. We finally separated out subgraphs that were connected to the remaining edges by solely non-maximal edges as described in the Best Unambiguous Subsets (BUS) algorithm. When the set of edges for each node was no further reducible, we output the connected components of the final graph as the orthology groups between the two species. We finally marked the isolated genes as paralogs of their best match.

2.1. Initial pruning of sub-optimal matches

Let **G** be a weighted bipartite graph describing the similarities between two sets of genes **X** and **Y** in the two species compared (Figure 1, top left panel). Every edge e=(x,y) in E that connects nodes $x \in X$ and $y \in Y$ was weighted by the total number of amino acid similarities in BLAST hits between genes x and y. When multiple BLAST hits connected x to y, we summed the non-overlapping portions of these hits to obtain the total weight of the corresponding edge. We constructed graph **M** as the directed version of **G** by replacing every undirected edge **e**=(**x**,**y**) by two directed edges (**x**,**y**) and (**y**,**x**) with the same weight as **e** in the undirected graph (Figure 1, top right panel). This allowed us to rank edges incident from a node, and construct subsets of **M** that contain only the top matches out of every node.

This step drastically reduced the overall graph connectivity by simply eliminating all out-edges that are not near optimal for the node they are incident from. We defined $\mathbf{M_{80}}$ as the subset of **M** containing for every node only the outgoing edges that are at least 80% of the best outgoing edge. This was mainly a preprocessing

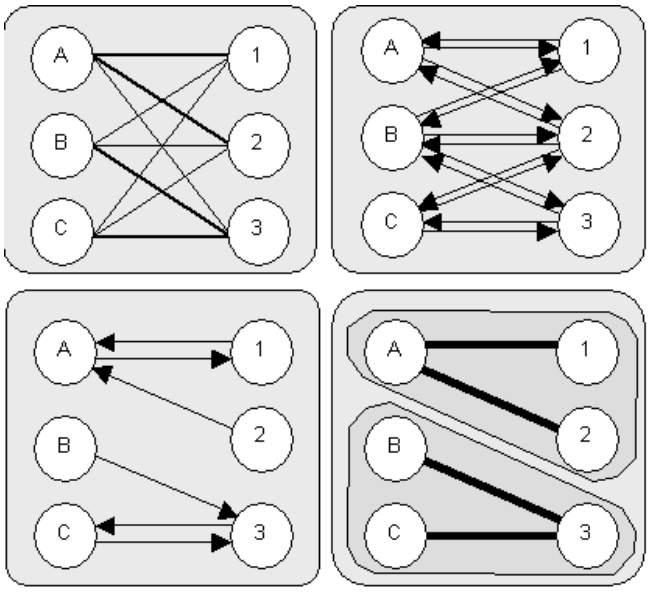

Figure 1. Overview of graph separation. We construct a bipartite graph based on the blast hits. We consider both forward and reverse matches for near-optimality based on synteny and sequence similarity. Sub-optimal matches are progressively eliminated simplifying the graph. We return the connected components of the undirected simplified graph.

step that eliminated matches that were clearly non-optimal. Virtually all matches eliminated at this stage were due to protein domain similarity between distantly related proteins of the same super-family or proteins of similar function but whose separation well-precedes the divergence of the species. Selecting a match threshold relative to the best edge ensured that the algorithm performs at a range of evolutionary distances. After each stage, we separated the resulting subgraph into connected components of the undirected graph (Figure 1, bottom right panel).

2.2. Blocks of conserved synteny

The initial pruning step created numerous two-cycle subgraphs (unambiguous one-to-one matches) between proteins that do not have closely related paralogs. We used these to construct blocks of conserved synteny based on the physical distance between consecutive matched genes, and preferentially kept edges that connect additional genes within the block of conserved gene order. Edges connecting these genes to genes outside the blocks were then ignored, as unlikely to represent orthologous relationships. Without imposing an ordering on the scaffolds or the chromosomes, we associated every gene x with a fixed position (s, start) within the assembly, and every gene y with a fixed position (chromosome, start) within *S. cerevisiae*. If two one-to-one unambiguous matches (x1, y1) and (x2, y2) were such that x1 was physically near x2, and y1 was physically near y2, we constructed a synteny block B=({x1, x2},{y1,y2}). Thereafter, for a gene x3 that was proximal to {x1, x2}, if an outgoing edge (x3, y3) existed such that y3 was proximal to {y1,y2}, we ignored other outgoing edges (x3, y') if y' was not proximal to {y1,y2}.

Without this step, duplicated genes in the yeast species compared remained in two-by-two homology groups, especially for the large number of ribosomal genes that are nearly identical to one another. We found this step to play a greater role as evolutionary distances between the species compared became larger, and sequence similarity was no longer sufficient to resolve all the ambiguities. We only considered synteny blocks that had a minimum of three genes before using them for resolving ambiguities, to prevent being misled by rearrangements of isolated genes. We set the maximum distance ***d*** for considering two neighboring genes as proximal to 20kb, which corresponds to roughly 10 genes. This parameter should match the estimated density of syntenic anchors. If many genomic rearrangements have occurred since the separation of the species, or if the scaffolds of the assembly are short, the syntenic segments will be shorter and setting *d* to larger values might hurt the performance. On the other hand if the number of unambiguous genes is too small at the beginning of this step, the genes used as anchors will be sparse, and no synteny blocks will be possible for small values of ***d***.

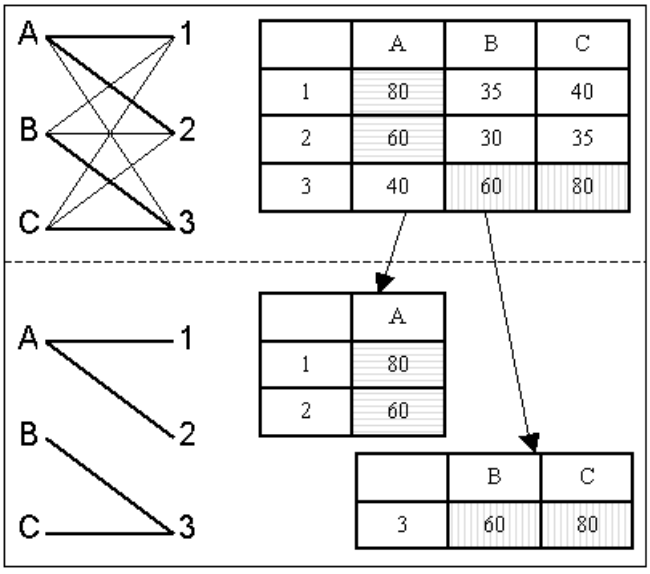

Figure 2. Best Unambiguous Subsets (BUS). A BUS is a set of genes that can be isolated from a homology group while preserving all potentially orthologous matches. Given the similarity matrix above and no synteny information, two such sets are (A,1,2) and (B,C,3).

2.3. Best Unambiguous Subsets (BUS)

To resolve additional orthologs, we extended the notion of a best bi-directional hit for sets of genes instead of individual genes. Moreover, we only constructed such a best subset when no gene outside the subset had its best match within the subset, hence when the best bi-directional subset was unambiguous. We defined a Best Unambiguous Subset (BUS) of the nodes of $\mathbf{X} \cup \mathbf{S}$, to be a subset **S** of genes, such that $\forall x: x \in S \Leftrightarrow best(x) \subseteq S$, where best(x) are the nodes incident to the maximum weight edges from x. We then constructed $\mathbf{M_{100}}$, following the notation above, namely the subset of **M** that contains only best matches out of a node. Note that multiple best matches were possible based on our definition. To construct a BUS, we started with the subset of nodes in any cycle in M_{100}. We augmented the subset by following forward and reverse best edges, that is including additional nodes if their best match was within the subset, or if they were the best match of a node in the subset. This ensured that separating a subgroup did not leave any node orphan, and did not remove the strictly best match of any node. When no additional nodes needed to be added, the BUS condition was met.

Figure 2 shows a toy example of a similarity matrix. Genes A, B, and C in one genome are connected in a complete bipartite graph to genes 1, 2 and 3 in another genome (ignoring for now synteny information). The sequence similarity between each pair is given in the matrix, and corresponds to the edge weight connecting the two genes in the bipartite graph. The set (A,1,2) forms a BUS, since the best matches of A, 1, and 2 are all within the set, and none of them represents the best match of a gene outside the set. Hence, the edges connecting (A,1,2) can be isolated as a subgraph without removing any orthologous relationships, and edges (B,1), (B,2), (C,1), (C,2), (A,3) can be ignored as non-orthologous. Similarly (B,C,3) forms a BUS. The resulting bipartite graph is shown. A BUS can be alternatively defined as a connected component of the undirected version of M_{100} (Figure 1, bottom panels).

This part of the algorithm allowed us to resolve the remaining orthologs, mostly due to subtelomeric gene family expansions, small duplications, and other genes that did not benefit from synteny information. In genomes with many rearrangements, or assemblies with low sequence coverage, which do not allow long-range synteny to be established, this part of the algorithm will play a crucial role. We have experimented running only BUS without the original pruning and synteny steps, and the results were satisfactory. More than 80% of ambiguities were resolved, and the remaining matches corresponded to duplicated ribosomal

proteins and other gene pairs that are virtually unchanged since their duplication. The algorithm was slower, due to the large initial connectivity of the graph, but a large overall separation was obtained. Figure 3 compares the dotplot of *S. paradoxus* and *S.cerevisiae* with and without the use of synteny. Every point represents a match, the x coordinate denoting the position in the *S.paradoxus* assembly, and the y coordinate denoting the position in the *S.cerevisiae* genome, with all chromosomes put end-to-end. Lighter dots represent homology containing more than 15 genes (typically transposable elements) and circles represent smaller homology groups (rapidly changing protein families that are often found near the telomeres). The darker dots represent unambiguous 1-to-1 matches, and the boxes represent synteny blocks.

2.4. Validating predicted protein-coding genes

Once we have resolved the pairwise species comparisons to *S.cerevisiae*, we build multiple alignments of both genes and flanking intergenic regions using CLUSTALW [25]. We can then

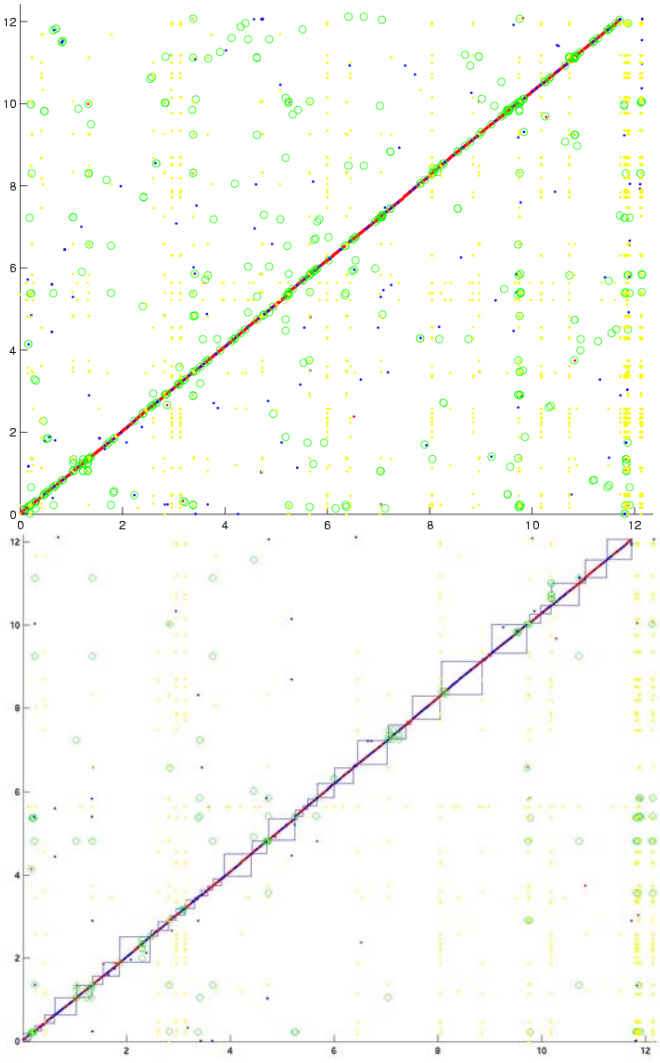

Figure 3. The effect of using synteny. Blocks of conserved gene order (blue squares) help resolve additional ambiguities.

observe the different patterns of nucleotide change in genes and intergenic regions. We find radically different types of conservation. Intergenic regions typically show short stretches between 8 and 10 bases of near-perfect conservation, surrounded by non-conserved bases, rich in isolated gaps. Protein-coding genes on the other hand are much more uniform in their conservation, and typically differ in the largely-degenerate third-codon position. Importantly, gaps are rare and when they do occur, they either happen in multiples of three, or are compensated by proximal gaps that restore the reading frame. This pressure for reading frame conservation can be used to discriminate protein-coding from intergenic regions, simply based on the pattern of gaps in the alignment.

To measure frame conservation between two aligned sequences, we label each non-gap nucleotide of the first sequence as 1, 2 or 3 cycling in order and starting at 1. We label the second sequence similarly, but once for every starting frame offset. We then simply count the percentage of aligned nucleotides that contain the same label for each of the three offsets. The offset with the maximum number of in-frame nucleotides is selected. To evaluate the frame conservation of a complete ORF, we average the percentages obtained in overlapping windows of 100bp. We obtain an average of 44% for intergenic regions (we should expect 33% at random), and an average of more than 99% for protein-coding genes. We applied a simple cutoff for each species, and tested all named S.cerevisiae ORFs, and as a control three hundred intergenic regions. We found that only 1% of intergenic regions pass the test, and less than 0.5% of named ORFs are rejected. The rejected ORFs show weak biological evidence and probably do not correspond to real genes [13].

Hence, comparative analysis can complement the primary sequence of a species and provide general rules for gene discovery that do not rely solely on known splicing signals for gene discovery. In the availability of comparative sequence information, this test provides a nice complement to programs such as GENSCAN that only look for signals in primary sequence, judging the predictions in the eye of evolution. The test presented can be used to test the validity of predicted genes in a wide range of sequenced species, even in absence of biological experimentation or known splicing signals. Even in well-studied species, this test can be used to discover additional genes that may not follow the typical rules of translation due to non-standard splicing signals, stop-codon read-through, post-transcriptional RNA editing, varying codon composition, or simply sequencing errors.

Thus, in a fully automated fashion, we have used comparative genomics to discover orthologs for virtually all protein-coding ORFs, and construct multiple alignments across the entire genome. We have used these alignments to judge the validity of protein coding genes. We now turn to the discovery of conserved regulatory motifs within the aligned intergenic regions.

3. REGULATORY MOTIF DISCOVERY

The traditional method for computational discovery of regulatory motifs has been to search within sets of co-regulated genes for enriched intergenic sequence patterns. We have undertaken a genome-wide discovery approach that should be applicable without previous knowledge of co-regulated sets. This approach is possible because the signal-to-noise ratio can be increased by comparing multiple species. Since mutations in transcription

factor binding sites may disrupt regulation, we expect regulatory motifs to be more strongly conserved than non-functional sequences that are free to diverge. Indeed, in four-way alignments of orthologous intergenic regions we observe that experimentally determined transcription factor binding sites correlate strongly with islands of sequence conservation. Moreover, the sequences of known regulatory motifs show a stronger genome-wide conservation as summed over all intergenic regions, as compared to random control patterns of the same degeneracy. Motivated by these results, we will search for motifs that show a strong genome-wide conservation.

We first exhaustively enumerated and tested the conservation of short sequence patterns to find unusually conserved motif cores. We then refine and combine these cores to construct full motifs.

3.1. Discovery of motif cores

We first enumerated all motif sequences of length 6, separated by a central gap between 0 and 21 nucleotides (mini-motifs). Each gap size consists of 2080 motifs, considering a motif and its reverse palindrome as the same motif. This results in a total of 45760 distinct mini-motifs. We assume that the large majority of these show a random conservation. We then look for those sequences that are unusually conserved as compared to a random population of mini-motifs. We use three different conservation tests.

3.1.1: Intergenic conservation (INT)

We first searched for motifs that show a significant conservation in all intergenic regions. For every mini-motif, we counted the number of perfectly conserved intergenic instances in all four species (**ic**), and the total number of intergenic instances in *S.cerevisiae* (**i**). The two counts seem linearly related for the large majority of patterns, which can be attributed to a basal level of conservation $\mathbf{r_i}$ given the total evolutionary distance that separates the four species compared. We estimate the typical ratio r_i as the log-average of non-outlier instances of ***ic/i*** within a control set of motifs. We then calculate for every motif the binomial probability p_i of observing ***ic*** successes out of ***i*** trials, given parameter r_i. We then assign a z-score S_i to every motif as the number of standard deviations away from the mean of a normal distribution that correspond to tail area p_i. This score is positive if the motif is conserved more frequently than random, and negative if the motif is diverged more frequently than random. We find that the distribution of scores is symmetric around zero for the vast majority of motifs. The right tail of the distribution however extends much further than the left tail, containing 1190 motifs more than 5 sigma away from the mean, as compared to 25 motifs for the left tail. By comparing the two counts, we estimate that 94% of these 1190 motifs are non-random in their conservation enrichment.

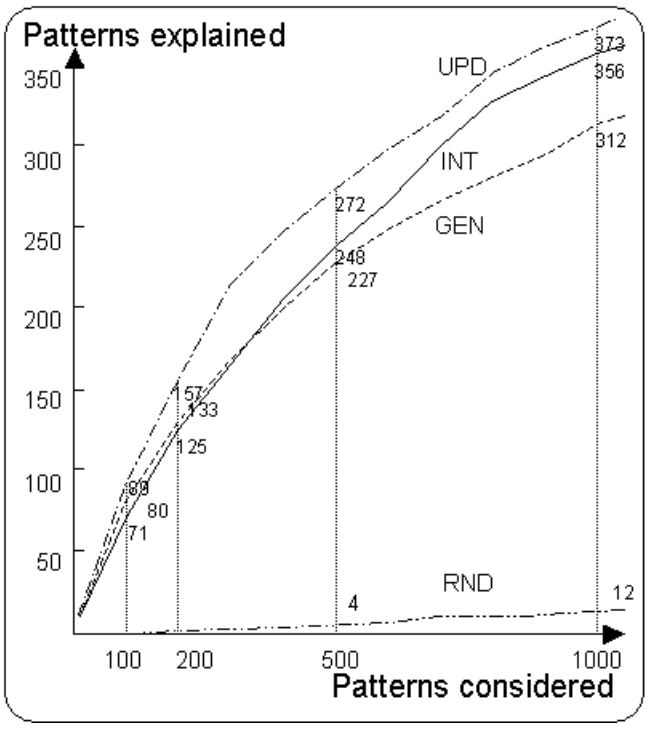

Figure 4. Selecting motif cores using three different tests. Patterns selected by one of the three tests (INT,GEN,UPD) correlate with function 90 times more frequently than randomly chosen motifs (RND).

3.1.2: Intergenic vs. coding conservation (GEN)

We then searched for motifs that are preferentially conserved in intergenic regions, as compared to coding regions. In addition to ***ic*** and ***i*** (see previous section), we counted the number of conserved coding instances ***gc***, and the number of total coding instances ***g***, for every mini-motif. We then compared the proportion of motif instances that were in intergenic regions for both conserved instances and total instances, namely ***a=ic/(ic+gc)*** and ***b=i/(i+g)***. On average, 25% of all motif instances are found in intergenic regions, which account for roughly ¼ of the yeast genome. However, only 10% of conserved motif instances appear in intergenic regions, since nucleotides in genes are more strongly conserved. For a population of motifs of similar GC-content, the ratio ***f =a/b*** remains constant. For a given motif, we calculate the enrichment in the proportion of intergenic instances, as the binomial probability of seeing at least ***ic*** successes, given ***ic+gc*** trials, given the probability ***p*** of success. To estimate ***p***, we use the proportion of total intergenic instances for that motif ***i/(i+g)***, and corrected by the log-average ***f*** of control motifs. We then score this motif by the standard deviations away from the mean of a standard normal distribution that correspond to this probability. The distribution of scores is again centered around zero for most motifs, but shows a heavier right tail. At 5 sigma, 1110 motifs are on the right tail, as compared to 39 motifs on the left tail. Hence at this cutoff, we expect 97% of motifs to be non-random.

3.1.3: Upstream vs. downstream conservation (UPD)

We finally searched for motifs that are conserved differently in upstream regions and downstream regions. We defined upstream-only intergenic regions as divergent promoters that are upstream of both flanking ORFs, and downstream-only regions as convergent 3' intergenic regions that are downstream of both flanking ORFs. We then counted ***uc*** and ***u***, the conserved and total counts in upstream-only regions, and similarly ***dc*** and ***d*** in downstream-only regions. Although upstream-only regions account for twice the total length of downstream-only regions, they show the same level of conservation, and the two ratios ***uc/u*** and ***dc/d*** are both similar to ***ic/i*** for the large majority of motifs. To detect a specificity in the upstream vs. downstream

conservation of a motif, we use a chi-square contingency test on the four counts (uc,u,dc,d). We find 1089 mini-motifs with a chi-square value of 10.83 or greater, which corresponds to a p-value of .001. We thus expect to see roughly 46 of the 45760 motifs with such a score by chance alone, and hence estimate that 96% of the 1089 mini-motifs chosen to be non-random.

3.1.4: Motifs found show category enrichment

The mini-motifs selected are indeed enriched in regulatory sequences. Many of them are at the core of well-known motifs such as Abf1, Reb1, Cbf1, Mbp1. Moreover, their conserved instances are enriched in functionally related genes. We calculated the hypergeometric enrichment score for each of these motifs against 358 functional categories, consisting of 146 sets of genes co-bound in chromatin immunoprecipitation experiments[16], 120 sets of GO molecular processes as annotated in SGD[2, 5], and 92 clusters of genes that are coordinately expressed [8]. We found that more than a third of these motifs show a significant enrichment (hypergeometric score of 10^{-5} or stronger). If we compare this result with that of a random collection of 1000 motifs, we find that only 1% show a category enrichment.

Figure 4 shows the number of motifs that show a significant category enrichment score for increasingly larger sets of top-ranked motifs in each test (INT,GEN,UPD), as compared to a random sorting (RND). From the top 100 motifs of each test, 71, 80, and 89 are explained by at least one category, as compared to only 1 for random motifs. This trend continues for the top 200, 500 and 1000 motifs. Naturally, the categories chosen here do not capture but a small fraction of the wealth of transcriptionally controlled molecular processes a cell coordinates, and hence we should not expect all motifs to show a category correlation. However, with respect to functional categories, our search shows a 90-fold enrichment in explained motifs as compared to random.

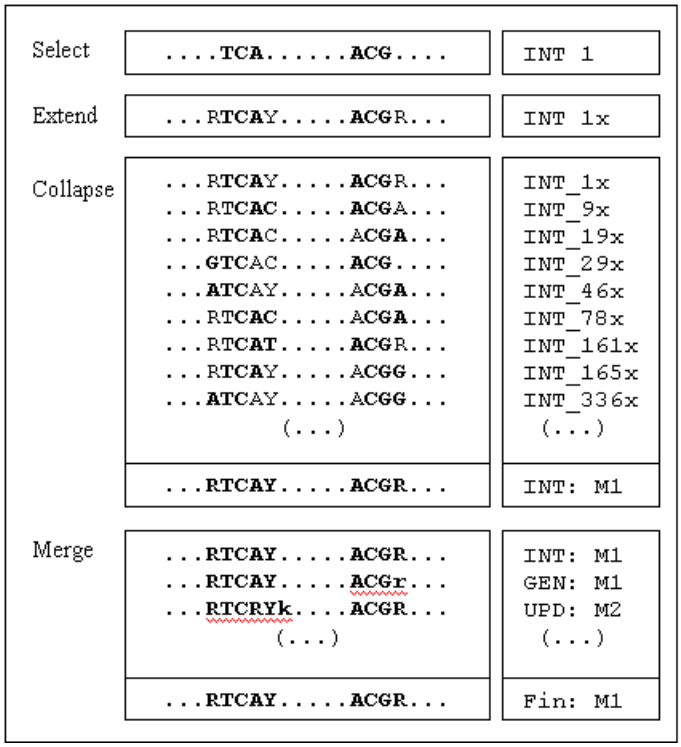

Figure 5. Overview of genome-wide motif discovery. We select motif cores by one of three tests, extend them to include additional conserved bases, and collapse together motifs with similar extension. We then merge motifs across multiple tests based on their co-occurrence in the same intergenic regions.

3.2. Constructing full motifs

We extend each of these mini-motifs by searching for surrounding bases that are preferentially conserved when the motif is conserved. We extend the motif iterative, one base at a time, by choosing, amidst the neighborhood of all conserved instances of the motif the base that maximally discriminates these from the neighborhood of non-conserved instances. The added base can be any of the fourteen degenerate symbols of the IUB code (A, C, G, T, S, W, R, Y, M, K, B, D, H, V). When no such symbol separates the conserved instances, the extension terminates. Figure 5 shows the top-scoring mini-motif found in the first test (INT_1), and the corresponding extension (INT_1x).

Many mini-motifs will have the same or similar extensions, and we group these based on sequence similarity. The similarity between two profiles is measured as the number of bits in common in the best ungapped alignment of the two profiles, divided by the number of bits contained in the profile with fewer bits. Based on the pairwise motif similarity matrix, we cluster the motifs hierarchically, until an average 70% similarity within a group is reached. This collapses the 1190 extended motifs discovered in test1 (INT) into 332 unique patterns, the 1110 motifs from test2 into 269, and the 1089 motifs from test 3 into 285 distinct patterns. The first 9 members of a cluster containing ABF1-like motifs from test1 are shown in figure 5, with mini-motif cores shown in bold, and the corresponding consensus INT_M1.

Finally, we merge motifs that co-occur in the same intergenic regions (Figure 5). The same motif will frequently be discovered across tests, or even multiple times within a test with slightly different sequences. These variations may prevent the sequence-based clustering from detecting an overlap, but the motifs will still typically occur in the same intergenic regions. To detect further overlaps, we compute a co-occurrence score between the conserved intergenic regions of each pair of collapsed motifs, and construct a consensus for the resulting group. We iterate this collapsing based on the newly constructed consensus and obtained fewer than 200 distinct motifs, of which 71 show a strong genome-wide conservation as compared to motifs of similar degeneracy.

These contain 30 known motifs, of which 28 correlate with functional categories, and an additional 41 'novel' motifs of which 61% correlate with at least one category (see [13]).

3.3. Category-based motif discovery

We further applied our motif discovery methods within functional categories. To select mini-motifs, we counted the conserved instances within the category (IN), and the conserved instances outside the category (OUT). We estimated the ratio IN/(IN+OUT) that we should expect for the category, based on the entire population of mini-motifs. We then calculated the significance of an observed enrichment as the binomial

probability of observing IN successes out of IN+OUT trials given the probability of success p. We assign a z-score to each mini-motif, as described in the genome-wide search, and similarly extended and collapsed the significant mini-motifs.

From the 106 profiled factors, 42 recognize a well-characterized motif. Of these however, only 25 show an actual enrichment in the published motif within the regions bound. In the remaining cases, the published motif may be incorrect or the ChIP experiment may be incorrect. For these 25 factors, we compared the published motif to the motif we discovered using our method, as well as the motif discovered by MEME and reported in Lee et al.

We identified short and concise motifs for all 25 factors, all of which agreed with the published consensus. On the contrary, the patterns produced by MEME typically contain additional bases that obscure the real binding site. By comparing multiple species, the signal therefore becomes stronger. It allows the search to focus on the conserved bases, eliminating most of the noise.

Table 1 summarizes the results. For each factor, we show the published motif, the hypergeometric enrichment score of the motif within the category (Hyper), the motif discovered by MEME and a quality assessment, the motif discovered by our method, as well as the corresponding category-based score and a quality assessment, and finally the comparison of our method to MEME. The performance of MEME degrades for less enriched motifs, but we consistently find the correct motif.

We then applied our methods to the complete set of 358 categories and discovered a total of 183 significant motifs. 109 categories gave rise to at least one motif, 46 gave rise to at least two motifs, and 16 gave rise to 3 motifs or more. The category-based motifs found are frequently shared across categories. After collapsing category-based motifs by sequence similarity, we obtain only 51 distinct motifs.

This overlap of the motifs discovered across categories is certainly to be expected between functionally related categories such as the chromatin IP experiment for Gcn4, the expression cluster of genes involved in amino acid biosynthesis, as well as the GO annotations for amino acid biosynthesis, all of which are enriched in the Gcn4 motif, the master regulator of amino acid metabolism.

Table 1: Category-based motif discovery. By searching for motifs that are both enriched in the category and evolutionarily conserved across the four species, we increase our sensitivity and specificity in category-based regulatory motif discovery. Here we compare known regulatory motifs to those discovered by MEME in a single genome and the ones we discover in conserved bases.

	Name	Motif	Hyper	MEME	Quality1	Our method	Score	Quality2	Comparison
1	**ABF1**	RTCRYnnnnnACG	91.4	**TRTCAY**T-Y--**ACG**RA	✓	**RTCAC_____ACG**A	14.6	✓	same
2	**GCN4**	ATGACTCAT	47.8	**TGAGTCA**Y	✓	**RTGACTCA**	10.9	✓	same
3	**REB1**	CCGGGTAA	44.7	**SCGGGTAA**Y	✓	**CCGGGTAA**C	8.7	✓	same
4	**MCM1a**	TTWCCnwwwwrGGAA	35.9	**TTTCC-AAW-RGGAA**A	✓	**TCC_____GGA**	4.4	✓	same
5	**RAP1**	ACACCCATACATTT	30.0	TTW**ACAYCCRTACAY-Y**	✓	**ACCCA.ACA**	8.7	✓	same
6	**Cbf1**	RTCACRTG	24.2	TR**GTCACGTG**	✓	**GTCACGTG**	10	✓	same
7	**FKH2**	TTGTTTACST	20.7	**TTGTTTAC**-TWTT	✓	**TGTTTTAC..TT**	8.3	✓	same
8	**SWI4**	CRCGAAAA	19.9	CSMRR**CGCGAAAA**	✓	CAA**CRCGAAAA**	8.1	✓	same
9	**MBP1**	ACGCGTnA	19.6	G-RR-A-**ACGCGT-R**	~	A**ACGCGT**CG	9.5	✓	better (+)
10	**STE12**	RTGAAACA	17.8	GSAASRR-**TGA**TR**A**WGYA		**YTGAAACA**	12.2	✓	better (+)
11	**Gal4**	CGGnnnnnnnnnnnCCG	16.1	**CGG**M---CW-Y--**CCG**	~	**CGG_____CCGA**	7.8	✓	better (+)
12	**SWI6**	ACGCGT	15.6	**WCGCGT**CGCGTY-C	✓	**ACGCGT**	7.4	✓	same
13	**PHO4**	CACGTG	14.2	TTGTA**CACTT**YGTTT		CG**CACGTG**	4.6	✓	better (+)
14	**HSF1**	TTCTAGAA	14.1	TY**TTC**YA**GAA--TTC**Y	✓	G**TTC**TA**GAA__TTC__**G	9.6	✓	same
15	**Dig1**	RTGAAACA	13.6	CCY**TG-AY**TTCW-CTTC		**TGAAACR**	11.8	✓	better (+)
16	**INO4**	CATGTGAAat	13.4	G..G**CATGTGAAAA**	✓	G...**CATGTGAA**	6.8	✓	same
17	**FKH1**	TTGTTTACST	13.2	C**YTRTTTAY**-WTT	✓	**TGTTTAC**	6.5	✓	same
18	**Leu3**	CCGGNNCCGG	13.1	G**CCGG**TMM**CGS**YC--	✓	**CCGG___CGG**	6.6	✓	better (+)
19	**Bas1**	TGACTC	10.2	CS-CCAATGK--CS		**TGACTCT**A	9.5	✓	better (+)
20	**SWI5**	KGCTGR	9.2	CACACACACACACACACA		T**GCTGG**	6.1	✓	better (+)
21	**HAP4**	TnRTTGGT	8.5	YCT**-ATTSG**-C-GS	~	**TGATTGGT**	6.4	✓	better (+)
22	**RLM1**	CTAWWWWTAG	8.4	A-CTSGAAGAAATGCGGT		**CTA..TTTAG**	4.7	✓	better (+)
23	**INO2**	CATGTGAAat	7.4	G**CATGTGRAAA**	✓	**CATGTG**	4.4	✓	same
24	**MET31**	AAACTGTGGC	7.0	GCACGTGATS		**TGTGGC**	5.8	✓	same
25	**ACE2**	GCTGGT	5.2	GTGTGTGTGTGTGTG		T**GCTGGT**	7.4	✓	better (+)

More surprisingly however, different transcription factors often share the same binding specificity, and the same motif appears in multiple expression clusters and functional categories. For example, Cbf1, Met4, and Met31 share a motif, and so do Hsf1, Msn2 and Msn4; Fkh1 and Fkh2; Fhl1 and Rap1; Ste12 and Dig1; Swi5 and Ace2; Swi6, Swi4, Ash1 and Mbp1. Also, a single motif involved in environmental stress response is found repeatedly in numerous expression clusters, and in functional categories ranging from secretion, cell organization and biogenesis, transcription, ribosome biogenesis and rRNA processing.

Hence, the set of regulatory motifs that are specific to the categories analyzed seems limited. Only a small minority of the transcription factors probed show specificity to a concise sequence. This may be due to the cooperative nature of binding that hides the actual sequence elements used in each region. The expression clusters we have used, although constructed over an impressive array of experiments, are still limited to the relatively few experimental conditions generated in the lab. Finally, the functional categories we used are limited to the few well-characterized processes in yeast, and the molecular function of more than 3000 ORFs remains unknown.

Moreover, category-based computational identification of regulatory elements can be hampered by the fact that motifs are shared across categories. No category will be enriched in a single motif, and no motif will be enriched in a single category. By discovering in an unbiased way the complete set of conserved sequence elements, as well as their target intergenic regions, we will have the building blocks to subsequent analyses of regulatory interaction networks. Thus, a genome-wide approach is a new and powerful paradigm to understanding the dictionary of regulatory motifs.

4. CONCLUSION

Our results show that comparative analysis with closely related species can be invaluable in annotating a genome. It reveals the way different regions change and the constraints they face, providing clues as to their use. Even in a genome as compact as that of *S.cerevisiae*, where genes are easily detectable and rarely spliced, much remains to be learned about the gene content. We found that a large number of the annotated ORFs are dubious, adjusted the boundaries of hundreds of genes, and discovered more than 50 novel ORFs and 40 novel introns. Moreover, our comparisons have enabled a glimpse into the dynamic nature of gene regulation and co-regulated genes by discovering most known regulatory motifs as well as a number of novel motifs. The signals for these discoveries are present within the primary sequence of *S.cerevisiae*, but represent only a small fraction of the genome. Under the lens of evolutionary conservation, these signals stand out from the non-conserved noise. Hence, in studying any one genome, comparative analysis of closely related species can provide the basis for a global understanding of all functional elements.

5. ACKNOWLEDGEMENTS

We would like to acknowledge the continuous help of the SGD curators and in particular Mike Cherry, Kara Dolinski and Dianna Fisk. We thank Tony Lee, Nicola Rinaldi, Rick Young, Julia Zeitlinger for discussions and providing pre-publication chromatin immunoprecipitation data. We thank Mike Eisen and Audrey Gasch for discussions and providing pre-publication clusters of expression data. We thank Jon Butler, Sarah Calvo, Matt Endrizzi, James Galagan, David Jaffe, Joseph Lehar, Li-Jun Ma and all the people at the MIT/Whitehead Institute Center for Genome Research for their help and discussions. We thank Ziv Bar-Joseph, John Barnett, Tim Danford, David Gifford and Tommi Jaakkola in the MIT Lab for Computer Science for their help and discussions. We thank Gerry Fink, Ernest Fraenkel, Ben Gordon, Trey Ideker, Sue Lindquist and Owen Ozier in the Whitehead Institute for their help and discussions.

6. REFERENCES

1. Altschul, S.F., Gish, W., Miller, W., Myers, E.W. and Lipman, D.J. Basic local alignment search tool. *J Mol Biol*, *215* (3). 403-410.
2. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M. and Sherlock, G. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, *25* (1). 25-29.
3. Blanchette, M., Schwikowski, B. and Tompa, M. Algorithms for phylogenetic footprinting. *J Comput Biol*, *9* (2). 211-223.
4. Cliften, P.F., Hillier, L.W., Fulton, L., Graves, T., Miner, T., Gish, W.R., Waterston, R.H. and Johnston, M. Surveying Saccharomyces genomes to identify functional elements by comparative DNA sequence analysis. *Genome Res*, *11* (7). 1175-1186.
5. Dwight, S.S., Harris, M.A., Dolinski, K., Ball, C.A., Binkley, G., Christie, K.R., Fisk, D.G., Issel-Tarver, L., Schroeder, M., Sherlock, G., Sethuraman, A., Weng, S., Botstein, D. and Cherry, J.M. Saccharomyces Genome Database (SGD) provides secondary gene annotation using the Gene Ontology (GO). *Nucleic Acids Res*, *30* (1). 69-72.
6. Fitch, W.M. Distinguishing homologous from analogous proteins. *Syst Zool*, *19* (2). 99-113.
7. Fitch, W.M. Uses for evolutionary trees. *Philos Trans R Soc Lond B Biol Sci*, *349* (1327). 93-102.
8. Gasch, A.P. and Eisen, M.B. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biol*, *3* (11). RESEARCH0059.
9. Gavin, A.C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J.M., Michon, A.M., Cruciat, C.M., Remor, M., Hofert, C., Schelder, M., Brajenovic, M., Ruffner, H., Merino, A., Klein, K., Hudak, M., Dickson, D., Rudi, T., Gnau, V., Bauch, A., Bastuck, S., Huhse, B., Leutwein, C., Heurtier, M.A., Copley, R.R., Edelmann, A., Querfurth, E., Rybin, V., Drewes, G., Raida, M., Bouwmeester, T., Bork, P., Seraphin, B., Kuster, B., Neubauer, G. and Superti-Furga, G. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, *415* (6868). 141-147.
10. Grundy, W.N., Bailey, T.L., Elkan, C.P. and Baker, M.E. Meta-MEME: motif-based hidden Markov models of protein families. *Comput Appl Biosci*, *13* (4). 397-406.
11. Hughes, J.D., Estep, P.W., Tavazoie, S. and Church, G.M. Computational identification of cis-regulatory elements

associated with groups of functionally related genes in Saccharomyces cerevisiae. *J Mol Biol*, *296* (5). 1205-1214.

12. Jiao, K., Nau, J.J., Cool, M., Gray, W.M., Fassler, J.S. and Malone, R.E. Phylogenetic footprinting reveals multiple regulatory elements involved in control of the meiotic recombination gene, REC102. *Yeast*, *19* (2). 99-114.
13. Kamvysselis, M., Patterson, N., Edrizzi, M., Birren, B. and Lander, E.S. submitted.
14. Keogh, R.S., Seoighe, C. and Wolfe, K.H. Evolution of gene order and chromosome number in Saccharomyces, Kluyveromyces and related fungi. *Yeast*, *14* (5). 443-457.
15. Lawrence, C.E., Altschul, S.F., Boguski, M.S., Liu, J.S., Neuwald, A.F. and Wootton, J.C. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, *262* (5131). 208-214.
16. Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., Zeitlinger, J., Jennings, E.G., Murray, H.L., Gordon, D.B., Ren, B., Wyrick, J.J., Tagne, J.B., Volkert, T.L., Fraenkel, E., Gifford, D.K. and Young, R.A. Transcriptional regulatory networks in Saccharomyces cerevisiae. *Science*, *298* (5594). 799-804.
17. Liu, X., Brutlag, D.L. and Liu, J.S. BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pac Symp Biocomput*. 127-138.
18. McCue, L., Thompson, W., Carmack, C., Ryan, M.P., Liu, J.S., Derbyshire, V. and Lawrence, C.E. Phylogenetic footprinting of transcription factor binding sites in proteobacterial genomes. *Nucleic Acids Res*, *29* (3). 774-782.
19. Mewes, H.W., Frishman, D., Guldener, U., Mannhaupt, G., Mayer, K., Mokrejs, M., Morgenstern, B., Munsterkotter, M., Rudd, S. and Weil, B. MIPS: a database for genomes and protein sequences. *Nucleic Acids Res*, *30* (1). 31-34.
20. Roth, F.P., Hughes, J.D., Estep, P.W. and Church, G.M. Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nat Biotechnol*, *16* (10). 939-945.
21. Simon, I., Barnett, J., Hannett, N., Harbison, C.T., Rinaldi, N.J., Volkert, T.L., Wyrick, J.J., Zeitlinger, J., Gifford, D.K., Jaakkola, T.S. and Young, R.A. Serial regulation of transcriptional regulators in the yeast cell cycle. *Cell*, *106* (6). 697-708.
22. Tatusov, R.L., Koonin, E.V. and Lipman, D.J. A genomic perspective on protein families. *Science*, *278* (5338). 631-637.
23. Tatusov, R.L., Natale, D.A., Garkavtsev, I.V., Tatusova, T.A., Shankavaram, U.T., Rao, B.S., Kiryutin, B., Galperin, M.Y., Fedorova, N.D. and Koonin, E.V. The COG database: new developments in phylogenetic classification of proteins from complete genomes. *Nucleic Acids Res*, *29* (1). 22-28.
24. Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J. and Church, G.M. Systematic determination of genetic network architecture. *Nat Genet*, *22* (3). 281-285.
25. Thompson, J.D., Higgins, D.G. and Gibson, T.J. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, *22* (22). 4673-4680.
26. Tompa, M. Identifying functional elements by comparative DNA sequence analysis. *Genome Res*, *11* (7). 1143-1144.
27. Wingender, E., Chen, X., Fricke, E., Geffers, R., Hehl, R., Liebich, I., Krull, M., Matys, V., Michael, H., Ohnhauser, R., Pruss, M., Schacherer, F., Thiele, S. and Urbach, S. The TRANSFAC system on gene expression regulation. *Nucleic Acids Res*, *29* (1). 281-283.
28. Zhang, M.Q. Promoter analysis of co-regulated genes in the yeast genome. *Comput Chem*, *23* (3-4). 233-250.
29. Zhu, J. and Zhang, M.Q. SCPD: a promoter database of the yeast Saccharomyces cerevisiae. *Bioinformatics*, *15* (7-8). 607-611.

Phylogenetically and Spatially Conserved Word Pairs Associated with Gene Expression Changes in Yeasts

Derek Y. Chiang
Dept of Molecular and Cell Biology, UC Berkeley
dchiang@ocf.berkeley.edu

Alan M. Moses
Graduate Group in Biophysics, UC Berkeley
amoses@ocf.berkeley.edu

Manolis Kamvysselis
MIT Laboratory of Computer Science
manolis@mit.edu

Eric S. Lander
MIT/Whitehead Institute Center for Genome Research
lander@genome.wi.mit.edu

Michael B. Eisen
Dept of Molecular and Cell Biology, UC Berkeley
Division of Genome Sciences, Lawrence Berkeley National Lab
mbeisen@lbl.gov

ABSTRACT

Background. Transcriptional regulation in eukaryotes is often multifactorial, involving multiple transcription factors binding to the same transcription control region (*e.g.*, upstream activating sequences and enhancers), and to understand the regulatory content of eukaryotic genomes it is necessary to consider the co-occurrence and spatial relationships of individual binding sites. The identification of sequences conserved among related species (often known as phylogenetic footprinting) has been successfully used to identify individual transcription factor binding sites. Here, we extend this concept of functional conservation to higher-order features of transcription control regions involved in the multifactorial control of gene expression.

Results. We used the genome sequences of four yeast species of the genus *Saccharomyces* to identify sequences potentially involved in multifactorial control of gene expression. We found 1,117 potential regulatory "templates": pairs of hexameric sequences that are jointly conserved in transcription regulatory regions and also exhibit non-random relative spacing. Many of the individual sequences in these templates correspond to known transcription factor binding sites, and the sets of genes containing a particular template in their transcription control regions tend to be differentially expressed in conditions where the corresponding transcription factors are known to be active.

Conclusions. The incorporation of both joint conservation and spacing constraints of sequence pairs predicts groups of target genes that were specific for common patterns of gene expression. Our work suggests that positional information, especially the relative spacing between transcription factor binding sites, may represent a common organizing principle of transcription control regions.

RECOMB'03, April 10-13, 2003, Berlin, Germany.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Biology and genetics

General Terms

Algorithms

Keywords

Phylogenetic footprinting, comparative genomics, multifactorial regulation, transcription regulation, promoter structure

1. INTRODUCTION

All organisms have evolved intricate signaling networks that sense and respond to their environment. At a cellular level, the activation of one or more signaling networks often leads to coordinated changes in gene expression, via the regulated activity and binding of transcription factors to transcription control regions (TCR's) of genes (e.g. enhancers and upstream activating sequences). In yeast and most other eukaryotes, the transcriptional regulation of individual genes is often multifactorial, as multiple transcription factors may bind to a single TCR [1] [2] [3]. In some cases, multiple transcription factors bind to a TCR and act independently of one another to alter gene expression in response to distinct cellular cues [4]; in other examples, multiple factors bind and/or act cooperatively to modulate gene expression via direct or indirect physical interactions with each other [5] [6] [7].

The challenges in understanding how regulatory information is encoded in genomes include both the identification of regulatory sequences in TCR's, and the elucidation of the constraints on productive multifactorial regulation. Many experiments have shown that specific pairs of factors must be bound near each other in order to act cooperatively [8] [9] [10], and it is on these spatial constraints that we focus here.

Previous computational work has been devoted to identifying putative transcription factor binding sites. A plethora of computational methods has been developed to find over-represented sequences in a subset of genes believed to contain a common transcription factor binding site (reviewed in [11]). The

rapid pace of genome sequencing has enabled a complementary approach – phylogenetic footprinting (reviewed in [12] [13]) – that recognizes that the conservation of sequences across related organisms often reflects evolutionary selection for their presence in TCR's. Several algorithms have been developed to perform phylogenetic footprinting analyses systematically [14] [15] [16].

After compiling a collection of putative binding sites, associations can be made between various binding site assortments and gene expression. Some recent approaches include Boolean logic [17], regression methods [18] [19] [20], spatial clustering [21], and multiple binding site matrix classifiers [22] [23] [24]. Spatial information on the relative locations of binding sites is ignored in all but the last two classes of approaches. Yet even these methods, which often search for fixed arrangements among the individual binding sites, may miss permutations of binding sites within TCR's that may still be bound and regulated by their corresponding transcription factors.

The primary aim of this work was to incorporate positional information and phylogenetic footprinting to identify sequence motifs that may regulate gene expression. Consequently, we expanded the focus of phylogenetic footprinting from the conservation of contiguous sequences to higher-order features of TCR's, namely the spatial organization of individual binding sites. Since transcription factors participating in multifactorial regulation may require physical proximity among their binding sites, we searched for groups of conserved sequences that were more closely spaced in TCR's than expected. We refer to these spatially organized sequences as conserved word templates. As a proof of principle, we started with the simplest example of such templates: pairs of conserved 6-bp words. Conservation was assessed using the genome sequences of three additional *Saccharomyces* species, which were chosen to be sequenced in order to elucidate regulatory sequences conserved among these closely related species [25]. To exploit this comparative genome data, we have devised a method that systematically tested sequence pairs for joint conservation across genomes and close spacing within individual TCR's. Since genes regulated by the same set of transcription factors often display similar gene expression patterns in certain experimental conditions, we identified conserved word pair templates whose gene targets were associated with common changes in gene expression. We adopted a group-by-sequence approach to first identify genes that contained the word pair templates and then to test for significant associations with expression levels of the identified genes [26]. Significant associations between conserved word pair templates and specific gene expression changes, the prevalence of known transcription factor binding sites, and the enrichment for common functional roles among gene groups, suggest that conserved word pair templates comprise sequences important for multifactorial regulation in yeast.

2. CONSERVED WORD PAIR TEMPLATE ALGORITHM

2.1 Overview

We present a method to find conserved higher-order sequence templates from related *Saccharomyces* genomes. Our method incorporates sequential statistical tests, with each step focusing on a distinct property of conserved sequence templates. The simplest instances of sequence templates involve word pairs and their relative spacing. First, word pairs that show enriched conservation as a unit were identified using a chi-square test for independence. Next, the relative spacing of conserved word pairs was assessed using a permutation test. Finally, those conserved word pairs with close spacing were verified for functional importance by testing for gene expression differences between matching genes and the rest of the genome. The output for this algorithm is a P × C data matrix, whose entries correspond to the strength of association with differential gene expression, i.e. the K-S significance level (see §2.5). Note that P is the number of significant conserved word pairs, and C is the number of gene expression conditions.

2.2 Datasets

Whole-genome shotgun sequencing of *Saccharomyces bayanus*, *Saccharomyces mikatae*, and *Saccharomyces paradoxus* has been previously described [25]. All of these organisms are highly related to *Saccharomyces cerevisiae*, as they are grouped within the *sensu stricto* branch of the *Saccharomyces* genus [47]. Intergenic regions were aligned using CLUSTALW as described [25] and are available from the Saccharomyces Genome Database [43]. A total of 4101 CLUSTALW alignments were analyzed. These alignments were filtered for orthologs in at least 3 genomes.

Gene expression measurements were obtained from the Stanford Microarray Database [48] and Rosetta [34]. The main experimental types among the 342 conditions examined include diauxic shift [27], cell cycle [29] [30], environmental stress response [28], DNA damage [31] [32], low phosphate [33], cadmium (N. Ogawa and P. O. Brown, unpublished data), and inhibition of ergosterol biosynthesis [34]. This data has been log-transformed (base 2), and each experimental condition has been median normalized.

2.3 Dependent Conservation of Word Pairs

To assess whether two words were co-conserved in the same intergenic regions, a chi-square test of independence was systematically conducted for all possible words of length six. We defined a word to include a 6-bp sequence and its reverse complement. Define a transcription control region (TCR) for a gene as the 600 base pairs upstream of its translation start site. TCR's shared between divergently transcribed genes less than 600 bp long were only counted once. A word was labeled conserved in a TCR if all six bases were identical among three or more genomes in the CLUSTALW alignment. For each word pair (W, V) whose overlap was less than 4, a contingency table $\boldsymbol{C}_{wv}$ was constructed. In this table, $\boldsymbol{C}_{wv}$ = # TCR($\boldsymbol{I}_w \cap \boldsymbol{I}_v$), where $\boldsymbol{I}_w$, $\boldsymbol{I}_v$ are indicator variables for the presence of each conserved word in a TCR. TCR's shared between divergently transcribed genes less than 600 bp long were only counted once. The expected counts $\boldsymbol{E}_{wv}$ were obtained from an independence assumption, *i.e.* the product of the individual word conservation probabilities, multiplied by the total number of TCR's. Thus the chi-square statistic with Yates continuity correction was computed according to the definition:

$$\chi^2_{wv} = \sum_{I_w=0}^{1} \sum_{I_v=0}^{1} \frac{(|C_{wv} - E_{wv}| - \frac{1}{2})^2}{E_{wv}} \qquad (1)$$

2.4 Spatial Proximity of Word Pairs

The second requirement for a conserved sequence template involved constraints on spatial arrangements between individual words. Any method that evaluates spacing distributions between word pairs must take into account positional biases that may be present for individual words (A. M. Moses, unpublished results). We used a permutation test to evaluate the significance of the average minimum distance, excluding overlaps, between conserved word pairs. By permuting the TCR labels for one of the words, but not the word positions themselves, we retained the positional biases of individual words within intergenic regions. Within any given TCR t, define $p_t(W) = \{p_t^1(W), \ldots, p_t^j(W)\}$ as a vector of positions in *S. cerevisiae* where word W is conserved. Suppose that words W and V were jointly conserved in TCR's $T_1 \ldots T_N$. Then the average minimum distance, $\overline{D}$, can be computed as:

$$\overline{D}_{wv} = \frac{1}{T}\sum_{t=1}^{T} \min_{j,k} \left| p_t^j(W) - p_t^k(V) \right| \qquad (2)$$

We used a permutation test to generate an empirical null distribution of $\overline{D}$ for all word pairs with $N \geq 10$. After randomly permuting the labels t for the position vectors of word V, a permutation test statistic, $\overline{D}^*$, can be calculated as above. By repeating this resampling procedure R times, an empirical null distribution $\overline{D}_{null} = \{\overline{D}^{*1}, \ldots, \overline{D}^{*R}\}$ can be obtained. The significance of the observed average minimum distance, $\overline{D}$, in the N promoters was calculated as its quantile in the empirical null distribution $\overline{D}_{null}$. We set an upper bound of $R = 10^6$, but stopped permutations early if 20 or more values in $\overline{D}_{null}$ were found less than $\overline{D}$.

Correction for multiple testing involved control of the proportion of false positives using a False Discovery Rate method [1]. This method has increased power over Bonferroni-type methods. Permutation quantiles for all N word pairs tested were sorted in non-decreasing order: $q_1 \leq \ldots \leq q_N$. Let $k = \max\left(i : q_i < \frac{0.05 i}{N} \right)$. Then the first k word pairs in the ordering had a corrected significance level of $q < 0.05$, *i.e.* the rate of false positives is approximately 5%.

2.5 Association between Template-Specified Gene Groups and Gene Expression Changes

So far we have identified word pairs with two properties: dependent conservation and spatial proximity among all TCR's in the whole genome. These word pairs can be viewed as sequence-based rules for selecting a subset of genes based on the conservation of an element of TCR architecture. In this stage, we would like to evaluate the transcriptional information associated with these rules by assaying for gene expression changes among genes that match these sequence constraints.

For each gene expression condition c in our dataset, $c \in \{1, \ldots, 342\}$, we tested the null hypothesis that a gene subset $G_{wv} \subseteq G$ selected by a conserved word pair (w, v) had the same distribution of gene expression ratios (E_{wv}^c) as the entire genome (E^c). The alternate hypothesis stated that the two gene expression distributions were significantly different. Any gene was an element of G_o if its corresponding TCR conserved both sequences in the word pair. Since the size N_o of gene subsets may be small and the distributions may not be normally distributed, we used the nonparametric Kolmogorov-Smirnov (K-S) test. The test statistic K compares the cumulative distribution functions F_{wv}^c and F^c corresponding to E_{wv}^c and E^c by the formula $K = \max_x \left| F_{wv}^c(x) - F^c(x) \right|$. The significance level of an observed value K^* can be obtained using a numerical approximation [51].

A gene subset determined by a word pair was deemed to have significantly different expression if its K-S p-value was less than a certain threshold. To correct for multiple testing, this threshold was established by controlling the False Discovery Rate. The significance levels p_i from each K-S test were ordered in ascending order. Let N represent the total number of K-S tests performed, i.e. the number of jointly conserved, closely spaced word pairs times the number of gene expression experiments). If k was the largest i such that $p_i < i\alpha / N$, then the first k word pairs in the ordering were deemed to have a significance level of $p < \alpha$.

We ensured that the K-S p-value for the conserved word pair subset G_o was more significant than subsets G_w or G_v comprised of only one conserved word by computing K for E_w^c vs. E_v^c, as well as for E_w^c vs. E^c. The marginal improvement of the joint word pair was defined as: K (F_o^c vs. F^c) – max(K (F_w^c vs. F^c), K (F_v^c vs. F^c)).

3. RESULTS

3.1 Identification of conserved word pair templates

We initialized our word list using all 2080 words of length six, treating a given word and its reverse complement as identical. For each TCR (consisting up to 600 bp upstream of an open reading frame), a word was labeled conserved if all six bases were identical in at least three of the four *Saccharomyces* genomes, based on the CLUSTALW alignment of that TCR. To systematically test whether words were conserved more often in the same intergenic regions of the *Saccharomyces* genomes than expected by independent conservation, a chi-square test was performed on all possible pairwise combinations of words (see §2.3). Pairs of words that overlapped each other by more than three nucleotides were excluded. A significant proportion of word pairs showed dependent conservation: among the 2.16 million word pairs tested, 8452 of them (~0.4%) had conservation c2 scores greater than 31.1. This threshold corresponds to a probability of 0.05 for obtaining one or more false positives after a Bonferroni correction for multiple testing.

Next, we selected word pairs that displayed closer physical spacing in intergenic regions than expected by chance. As a metric for the closeness in relative spacing between word pairs, the average minimum distance between two words in *S. cerevisiae*, $\overline{\ }$, was calculated based on the genes whose TCR's conserved both words. If two non-overlapping words were closely

spaced in all TCR's, we should find $\overline{D}$ to be smaller than expected by chance. This spacing was assessed using a permutation test by selecting the set of genes that contained a conserved word pair and then randomizing the assignment of one of the words to the genes containing that word (see §2.4). By permuting the TCR labels for one of the words, but not the word positions themselves, we retained the positional biases of individual words within intergenic regions.

After correcting for multiple testing, a total of 1117 out of 8452 word pairs (~13%) had significantly small values (FDR $q < 0.05$) for $\overline{D}$ (Figure 1). As a negative control, we also assayed a sample of word pairs that did not show dependent conservation (conservation $\chi^2 < 1$), yet were jointly conserved in at least 10 TCR's. Only 161 out of 42801 (~0.4%) random word pairs with non-dependent conservation ($\chi^2 < 1$) showed significantly small values for $\overline{D}$. Figure 2 illustrates the distributions of $\overline{D}$ for conserved word pair templates, jointly conserved word pairs, and randomly conserved word pairs. The medians of these distance distributions were 100 nucleotides, 116.5 nucleotides and 132 nucleotides, respectively. Notably, the median $\overline{D}$ for template pairs was significantly smaller ($p < 0.05$) than the median $\overline{D}$ for randomly conserved pairs. These results indicate that many of the word pairs that were conserved in the same intergenic regions of multiple *Saccharomyces* genomes also exhibited closer spacing in TCR's.

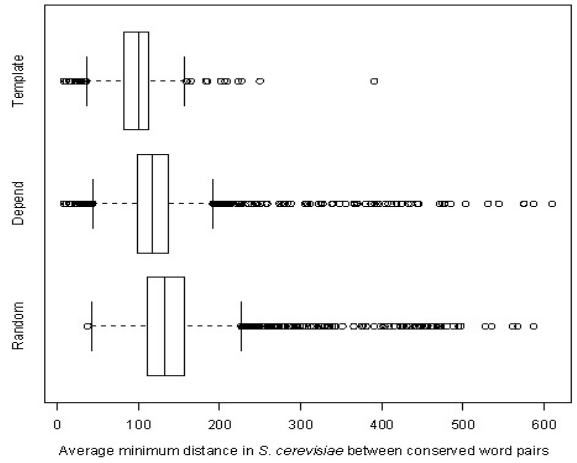

Figure 1) Word pairs in conserved word pair templates are closely spaced in *S. cerevisiae*

Template denotes closely spaced and jointly conserved word pairs ($\chi^2 > 31.1$, spacing $q < 0.05$, $N = 1117$). *Depend* denotes dependently conserved word pairs ($\chi^2 > 31.1$, $N = 8452$) and includes all of the word pairs in the template category. *Random* denotes a sample of randomly conserved word pairs ($\chi^2 < 1$, $N = 4667$). For each category, the distribution of average minimum distances is represented by a box-and-whisker plot.

3.2 Conserved word pair templates were significantly associated with gene expression

Our method identified conserved word pair templates that were statistically significant with respect to both co-conservation in multiple genomes and close spacing in *S. cerevisiae* TCR's. To evaluate the regulatory information in these templates, we assessed the statistical association between gene groups that shared a template and changes in gene expression. Similar to other group-by-sequence approaches for finding regulatory sequences, we expect that gene subsets defined by common TCR sequence rules should have gene expression patterns that are similar under conditions where the transcription factors are active, yet are different from the average expression of genes in the genome [26].

To assess the association between conserved word pair templates and differentially expressed genes, we identified gene subsets that contain both conserved words in the template within their TCR's and observed their expression patterns in S. cerevisiae in publicly available datasets ([27] to [34], see §2.5). We then conducted Kolmogorov-Smirnov (K-S) tests to evaluate for differential gene expression between each gene subset and the whole genome. A P × C matrix was computed: each conserved word pair in P was assigned a K-S p-value for each experimental condition observed in C. (see §2.5). Entries in this matrix (K-S p-values) were filtered out if the K-S *p*-value: (1) did not meet the threshold for multiple testing; or (2) was less than 10 times more significant than the K-S *p*-value for a gene subset associated with either word alone (see §2.5). The latter criterion discounts gene expression changes that are due predominantly to the action of a single transcription factor.

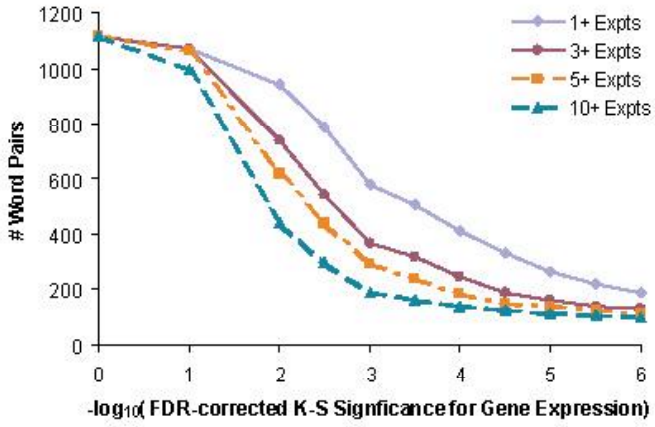

Figure 2) Total number of conserved word pair template associations at different K-S significance values

The horizontal axis shows different multiple testing-corrected significance levels for the K-S test (see §2.5). The number of closely spaced word pairs meeting this cutoff for different minimum numbers of expression conditions is shown on the vertical axis. Word pairs were also filtered for an improvement of 10× over the K-S significance from any single word.

Figure 2 displays the number of conserved word pair templates that were significantly associated with gene expression changes, for varying significance levels of the K-S test, which have been corrected for multiple testing (see §2.5). Each line indicates the number of gene subsets that were significant in a different minimum number of experimental conditions. Several hundred closely spaced word pairs were significantly associated with differential gene expression. For example, 293 word pairs met an FDR-corrected significance threshold of $p < 10^{-3}$ for 5 or more experimental conditions.

3.3 Many identified sequences represent known transcription factor binding sites

Conserved word pair templates that were most strongly associated with gene expression changes also agreed with prior experiments on transcription factors [35]. In all analyses described below, we used a set of 339 word pairs that had significant associations with gene expression changes at an FDR-corrected multiple testing threshold of 0.005 for 10 or more experiments. For visualization purposes, we organized the P × C matrix by hierarchically clustering the K-S p-values for the 339 word pairs.

Hierarchical clustering of this output matrix identified groups of word pairs with similar K-S p-values in specific subsets of experimental conditions (Figure 4). In many cases, the word pairs that clustered together also comprised overlapping hexamer sequences, suggesting that some of the hexamers in different pairs may represent a larger, somewhat variable sequence (Table 1). For example, group #13 in Figure 4 includes 8 word pairs. In each of these word pairs, one of the component words – such as TCACGT, GCACGT or CACGTG – matched part of the Cbf1p or Pho4p binding sites. The other component word in each pair – such as AACTGT, ACTGTG, CTGTGG, TGTGGC or GTGGCT – represented part of the known Met31/32p binding site (AAACTGTGG). Therefore, genes whose TCR's contained any word pair within this group likely contained a conserved Cbf1p or Pho4p binding site, along with a conserved Met31/32p binding site, and the distances between the conserved sites in these genes were also smaller than expected by chance. These results agree with the known interaction of Cbf1p and Met31/32p for the regulation of genes involved in sulfur utilization (see Discussion).

Table 1 shows a partial list of the 13 most significant groups of consensus sequences, which were assembled by joining adjacent word pairs in the clustered output matrix with overlapping sequences. Many of these consensus sequences matched transcription factor binding sites that had been biochemically verified. Several pairs of transcription factors, denoted by boldface in Table 1, were not previously known to participate in multifactorial regulation. Three of these pairs included new putative transcription factor binding sites. In group 8, the word ACAGCC is found in a template with the GATA motif. In group 9, the word CGGGCC is found in a template with the binding site for the stress-induced transcription factor, Msn2/4p. In group 2, one of the words in each word pair was the binding site for Swi4/6p, which regulates the expression of cell-cycle dependent genes. The other word in each pair was an invariant CGCCAA, which is highly similar to, though distinct from, the characterized Swi4/6 binding site CRCGAAA [35].

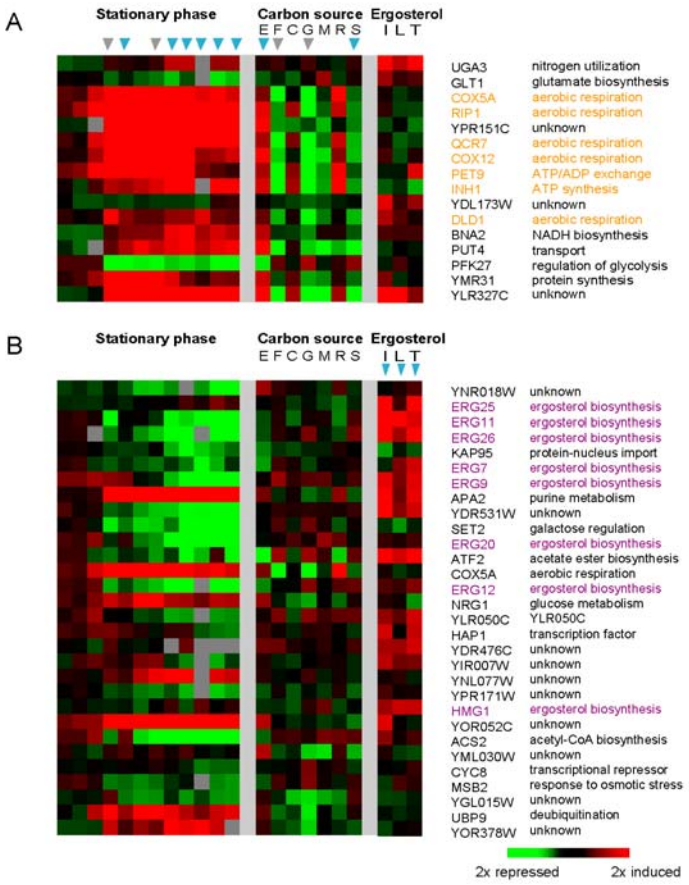

Figure 3) Multifactorial regulation of Hap1p target genes

Gene expression patterns are shown for genes whose TCR's contain binding sites for: (A) Hap1p (CCGATA) and Hap2/3/4/5p (CCAATC); or (B) Hap1p (CCGATA) and Ecm22p/Upc2p (TCGTTT). The genes are listed in ascending order of minimum distance between the two conserved words in the corresponding TCR of *S. cerevisiae*. Each row represents a given gene's expression pattern under the conditions shown in each column: progression into stationary phase (2 h, 4 h, 8 h, 12 h, 1 day, 2 days, 3 days, 5 days, 7 days, 13 days, 22 days, 28 days of growth) [28]; steady-state growth on different carbon sources: ethanol (E), fructose (F), galactose (C), glucose (G), mannose (M), raffinose (R) and sucrose (S) [28]; and growth in the presence of drugs that inhibit ergosterol biosynthesis: itraconazole (I), lovastatin (L) and terbinafine (T) [34]. A red color indicates that the gene's expression was induced under those conditions, while a green color indicates that the gene was repressed under those conditions; black indicates no detectible change in expression, and grey indicates missing data. Gene names highlighted in orange (A) or in purple (B) correspond to genes whose products are involved in respiration and ergosterol biosynthesis, respectively. Arrows above the columns indicate conditions in which the displayed gene groups show significant gene expression changes according to the Kolmogorov-Smirnov test, after False Discovery Rate correction for multiple testing at a *p*-value of 0.005 (blue) or 0.01 (grey).

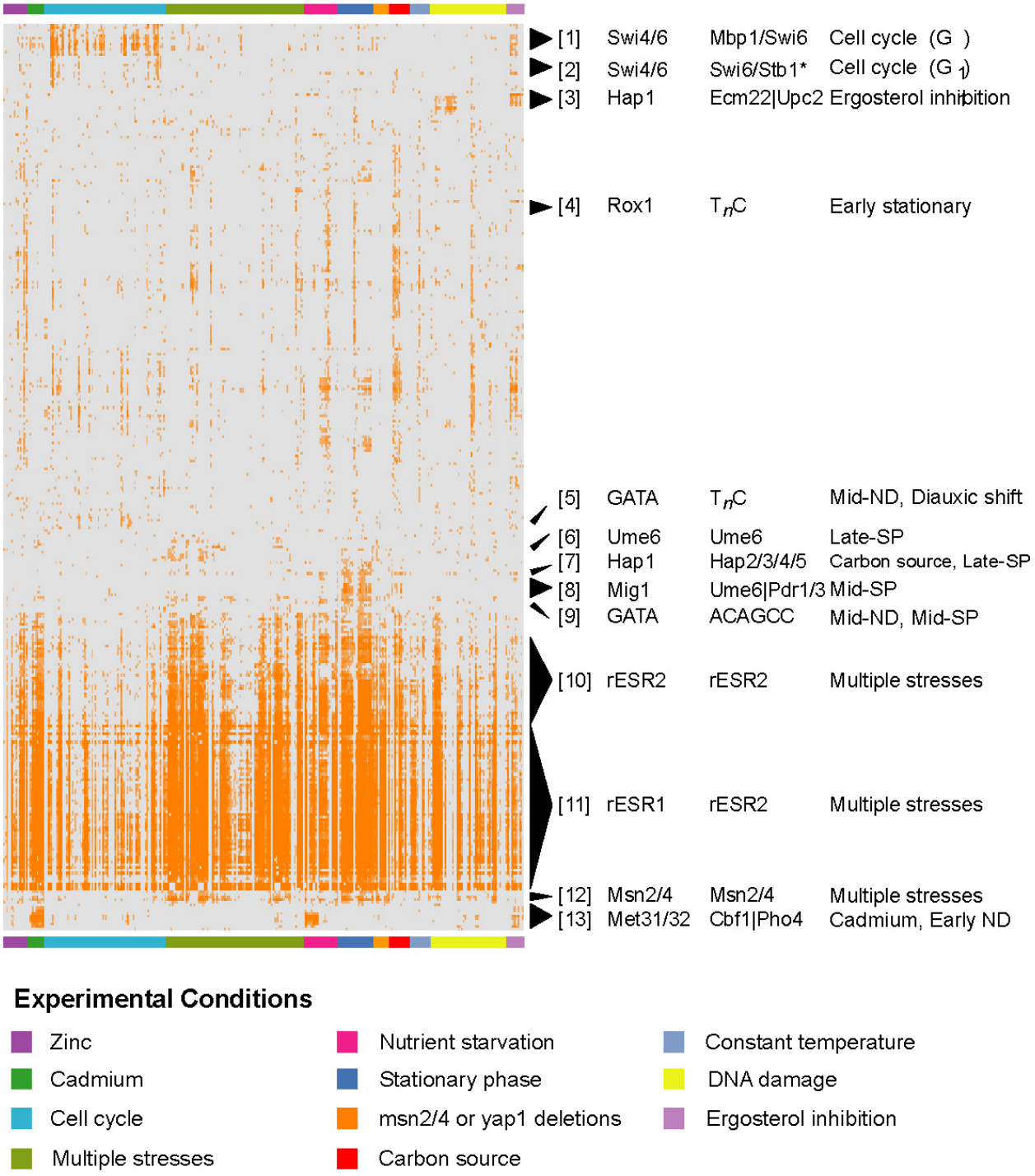

Figure 4) Specific patterns of gene expression changes associated with templates

The $P \times C$ matrix of K-S p-values was hierarchically clustered by rows and visualized with TreeView (http://rana.lbl.gov). Each row corresponds to a conserved word pair template, and each column represents a single gene expression experiment. The experimental conditions are indicated by the color bar above and below the figure, according to the key shown below. The value in each cell corresponds to the K-S p-value of gene expression changes in each condition (column) for a group of genes that contain the conserved word pair template (row) in their TCR's. An orange color denotes a K-S p-value below the FDR critical value of 0.005 for multiple testing, while grey represents values that were not significant. Word pairs that failed to meet a False Discovery Rate critical value of 0.005 for multiple testing in 10 or more experiments are not shown. Some of the most significant conserved word pair associations are labeled and annotated in Tables 1 and 2. Abbreviations for experimental conditions include: ND (nitrogen depletion), SP (stationary phase).

Table 1) Gene expression associations for most significant groups of word pairs

The output $\boldsymbol{P} \times \boldsymbol{C}$ matrix of word pairs ($\boldsymbol{P}$) that were significantly associated ($p < 0.005$) with at least 10 or more environmental conditions ($\boldsymbol{C}$) was ordered using hierarchical clustering. Numbers correspond to groups of overlapping word pairs indicated in Figure 4. Boldface denotes sequence pairs whose involvement in multifactorial regulation has not been previously reported. Consensus sequences were assembled from groups of word pairs that were found in adjacent rows in the ordering. Residues are shown in bold if it is contained in at least two hexamers. Numbers denote the groups that are indicated in Figure 4. IUPAC codes used: K (G or T); M (A or C); R (A or G); S (C or G); W (A or T).

	Most significant word pair in consensus group					
	Conserved Word Pairs (Consensus of overlapping words)	Known transcription factors or motifs	Conservation (χ^2, p-val via Bonferroni)	Avg. min dist $\overline{\boldsymbol{D}}$	# TCR	Expression conditions with significant gene subsets (FDR significance)
1	RCGAAA, RACGCG,	Swi4/6, Swi6/Mbp1	83.0 (7×10^{-13})	68.1	36	Cell cycle, G1 phase (10^{-6})
2*	**CACGAAA, CGCCAA**	**Swi4/6, Stb1 (putative)**	**55.6 (2×10^{-7})**	**78.2**	**25**	**Cell cycle, G1 phase (10^{-4})**
3*	**CCGATA, TC[GT]TTT**	**Hap1, Ecm22 | Upc2**	**36.2 (0.004)**	**81.7**	**30**	**Ergosterol inhibition (10^{-4}) MMS (DNA damage) (10^{-3})**
4*	**GATAAG, TTCTTT**	**GATA, T_nC**	**36.0 (0.005)**	**100.5**	**88**	**Nitrogen depletion 8h (10^{-5})**
5	GGCTAA CGGCGG	Ume6, Ume6	179.2 (2×10^{-34})	81.9	15	Late stationary phase (10^{-4})
6	CCGATA, CCAATC	Hap1, Hap2/3/4/5	35.0 (0.007)	88.6	16	Stationary phase (10^{-4}) Ethanol (10^{-4})
7	ACCCCA, CCGCCG	Mig1, Ume6 | Pdr1/3	66.7 (7×10^{-10})	70.5	16	Stationary phase (10^{-6}) Ethanol (10^{-5})
8*	**GATAAG, ACAGCC**	**GATA, Novel**	**39.5 (0.004)**	**75.5**	**21**	**Nitrogen depletion 8,12h (10^{-5}) Stationary phase 10h-2d (10^{-4})**
9*	**AAGGGG, CGGGCC**	**Msn2/4, Novel**	**33.4 (0.016)**	**79.6**	**14**	**Elutriation 2d, 4d, 6d (10^{-5}) Stationary phase 10h-3d (0.005)**
10	ANTGAAA, GAAAAWT	rESR2 (Overlap)	96.9 (2×10^{-16})	96.8	68	Repressed in multiple environmental stresses (10^{-6})
11	G[AC]GATGAG TGAAAATTTT	rESR1 motif, rESR2 motif	240.6 (10^{-49})	41.6	183	Repressed in multiple environmental stresses (10^{-6})
12	AWAAGG, AGGGG	Msn2/4 (Overlap)	94.7 (5×10^{-16})	99.0	29	Multiple stresses (10^{-3})
13	ACTGTGGC, [GT]CACGTG	Met31/32, Cbf1 | Pho4	47.5 (2×10^{-5})	43.5	22	Amino acid starv. (10^{-6}) Nitrogen depletion (10^{-6}) Cadmium (10^{-6})

Recent chromatin immunoprecipitation experiments suggested that this sequence may represent the binding site for Stb1, a transcription factor that binds Swi6 in vitro and is implicated in cell cycle regulation [36]. This sequence was found in several genes adjacent to intergenic regions bound by Stb1 *in vivo* [37].

Some groups of genes with shared word pair templates were enriched for known targets of transcription factors. Genes with a conserved half-site for the Hap1p transcription factor, as well as a conserved Hap2/3/4/5p binding site, in their TCR's were significantly associated with induction in late stationary phase (Figure 3A). In addition, many of these genes were more highly expressed in growth medium containing ethanol, relative to other carbon sources (Figure 3A). Many of these genes encode aerobic respiration enzymes, which are required for the switch from fermentation to respiration [38] [39]. Indeed, both the Hap1p transcription factor and the Hap2/3/4/5p transcription factor complex are known to regulate the expression of these genes in response to heme and/or oxygen availability and carbon source, respectively. By contrast, gene groups with both a conserved Hap1p binding site and a conserved Ecm22p or Upc2p binding site in their TCR's were only significantly associated with induction in the presence of a drug that inhibited ergosterol biosynthesis (Figure 3B). This group of 30 genes contained 8 ergosterol biosynthesis genes; this proportion represented an enrichment compared to the rest of the genome. The transcription factors Ecm22p and Upc2p have been shown to induce the expression of ergosterol biosynthesis genes in response to low intracellular concentrations of ergosterol, while Hap1p is known to regulate the expression of these genes according to the availability of heme and oxygen which are required for the pathway (see Discussion) [40]. Note that the gene groups shown in Figure 3 and Figure 3 showed significant gene expression changes in different sets of environmental conditions.

4. DISCUSSION

This work describes two principles for analyzing combinations of regulatory sequences. First, sequence conservation among closely related yeast species was used to find sequences that were more likely to be functionally important. Secondly, a template approach that considered joint positional distributions of word pairs increased the specificity of gene expression predictions using sequence-based rules. We have demonstrated that higher-order sequence features within TCR's were conserved across multiple *Saccharomyces* genomes. Closely spaced and jointly conserved word pairs were also more likely to be associated with gene expression changes. A large proportion of words contained in templates matched known transcription factor binding sites, and some of the uncharacterized words may represent novel regulatory sequences. In many cases, associations between templates and gene expression changes were significant in conditions when the corresponding transcription factors are known to be active. In addition, groups of genes that co-conserved both words in a template often were enriched for common functional roles. These results suggest that conserved word pair templates, which were discovered strictly based on higher-order properties of sequence conservation, also carry biological relevance.

Conserved word pair templates may be classified under several distinct classes of regulatory elements in TCR's. One possible interpretation of templates is that closely spaced sequence pairs may promote direct or indirect interactions between transcription factors by increasing the local concentrations of the individual factors. For example, the proximity of Cbf1p and Met31/32p binding sites may promote interaction between these factors in recruiting their common transcriptional activators, Met4 and Met28. Experimental studies on the TCR's of *MET3* and *MET28* have demonstrated that the binding of Cbf1p enhances the DNA binding affinity of Met31/32p [41]. Indeed, biochemical experiments suggest that all of these proteins interact at the TCR's of some sulfur utilization genes [41].

Another possible regulatory scheme for conserved, closely-spaced word pairs is that individual sequences found in templates may correspond to binding sites for transcription factors that bind independently under the same or separate conditions. The Hap1p and Hap2/3/4/5p transcription factors, whose binding sites were identified in a template, represent an example of multifactorial regulation in response to different environmental stimuli [42]. In some cases, templates could discern genes that shared binding sites for one transcription factor, but were differentially expressed under certain sets of conditions. Genes that conserved both Hap1p and Hap2/3/4/5p binding sites in their TCR's included genes encoding mitochondrial enzymes, as well as respiration proteins, that showed significant induction during growth in stationary phase and ethanol (Figure 3A). By contrast, genes that conserved both the Hap1p and Upc2p/Ecm22p binding sites in their TCR's were enriched for genes encoding ergosterol biosynthesis enzymes. Unlike the genes encoding mitochondrial enzymes, these genes showed no expression changes in response to different carbon sources, yet they were significantly induced under treatment with drugs that inhibit ergosterol biosynthesis: itraconazole, lovastatin and terbinafine (Figure 3B) [34]. A biochemical link between these two enzyme categories may explain their common regulation via Hap1p: the protein products of these genes all require the cofactor heme, whose intracellular levels are sensed by Hap1p [43] [44]. Our results suggest that Hap1p controls the expression of all of these genes in response to heme and/or oxygen levels. The expression of genes encoding mitochondrial and respiration enzymes may be controlled by Hap2/3/4p in response to nonfermentable carbon sources, whereas the expression of ergosterol biosynthesis genes may be regulated by Ecm22p and/or Upc2p in response to ergosterol levels. Whether these factors act cooperatively with, or independently of, Hap1p will require further biochemical investigation to elucidate.

Close spacing between word pairs may be important for reasons other than the promotion of transcription factor interactions. Different regions of TCR's at varying windows away from translation start sites may be more competent at recruiting or inhibiting RNA polymerase. These differences may be influenced by nucleosome accessibility, chromatin structure, or DNA physical properties, which can be correlated with local A/T content (see [45] for references). Notably, we have also found that the relative proportions of A and T nucleotides vary considerably within the 200 bp closest to translation start sites (A. M. Moses, M. B. Eisen and Audrey Gasch, unpublished results). Low-complexity words that contained 4 or more A's or T's could be found in many templates (denoted by TnC in Figure 4 and Table 1); these words may serve as surrogates for a distance window from translation start. Transcription factor binding sites that are closely spaced to these low-complexity words may be found in more transcriptionally competent regions of TCR's.

Since transcription factor binding sites often contain degenerate positions that reflect specificity, a key limitation of our approach is the use of exact words [11]. The known binding sites listed in Table 1 correspond to transcription factors with high sequence specificities. Since other known binding sites are poorly modeled by exact words (in that they bind sequences with relaxed specificity at certain positions in their binding sites), our method has failed to include them in conserved word pair templates. In addition, our method currently requires sequence identity for a word to be labeled as conserved. This strict requirement omits binding sites that may retain their function, despite mutations in degenerate positions that may have little impact on transcription factor binding.

The consideration of joint conservation and close spacing has provided insights into how TCR organization may influence the multifactorial regulation of gene expression in *Saccharomyces cerevisiae*. These criteria were motivated by experimental studies on the positional organization of individual binding sites within TCR's, with the hypothesis that this underlying architecture would be functionally conserved. Even more complicated higher-order sequence rules are apparent in the organization of cis-regulatory modules in *Drosophila melanogaster* [46]. Nevertheless, a common organizational theme of the TCR's in both of these organisms is the importance of relative spacing between transcription factor binding sites. The discovery of additional principles for TCR organization will further advance our understanding of how regulatory information is encoded in genome sequences.

5. ACKNOWLEDGEMENTS

We thank Audrey Gasch and Justin Fay for their critical reading of the manuscript and insightful suggestions during the development of this work; Peter Bickel and John Storey for their statistical advice; and Nabuo Ogawa and Patrick Brown for sharing gene expression data on cadmium conditions prior to publication. D.Y.C. is a Howard Hughes Medical Institute Predoctoral Fellow, and M.B.E. is a Pew Scholar in the Biomedical Sciences. This work was conducted under the US Department of Energy contract No. ED-AC03-76SF00098.

6. REFERENCES

[1] Composite regulatory elements: structure, function and classification. [http://www.gene-regulation.com/pub/databases/transcompel/compel.html]

[2] Wolberger C: Multiprotein-DNA complexes in transcriptional regulation. *Annu Rev Biophys Biomol Struct* 1999, 28:29-56.

[3] Kel OV, Romaschenko AG, Kel AE, Wingender E, Kolchanov NA: A compilation of composite regulatory elements affecting gene-transcription in vertebrates. *Nucleic Acids Res* 1995, 23:4097-4103.

[4] Gasch AP: The environmental stress response: a common yeast response to diverse environmental stresses. In: *Yeast Stress Responses* Edited by S Hohmann, WH Mager, vol. 1. pp. 11-70. Berlin: Springer; 2003: 11-70.

[5] Mead J, Bruning AR, Gill MK, Steiner AM, Acton TB, Vershon AK: Interactions of the Mcm1 MADS box protein with cofactors that regulate mating in yeast. *Mol Cell Biol* 2002, 22:4607-4621.

[6] Bhoite LT, Allen JM, Garcia E, Thomas LR, Gregory ID, Voth WP, Whelihan K, Rolfes RJ, Stillman DJ: Mutations in the Pho2 (Bas2) transcription factor that differentially affect activation with its partner proteins Bas1, Pho4, and Swi5. *J Biol Chem* 2002, 277:37612-37618.

[7] Verger A, Duterque-Coquillaud M: When Ets transcription factors meet their partners. *Bioessays* 2002, 24:362-370.

[8] Ambrosetti DC, Basilico C, Dailey L: Synergistic activation of the fibroblast growth factor 4 enhancer by Sox2 and Oct-3 depends on protein-protein interactions facilitated by a specific spatial arrangement of factor binding sites. *Mol Cell Biol* 1997, 17:6321-6329.

[9] Ludwig MZ, Patel NH, Kreitman M: Functional analysis of eve stripe 2 enhancer evolution in Drosophila: rules governing conservation and change. *Development* 1998, 125:949-958.

[10] Liu Z, Little JW: The spacing between binding sites controls the mode of cooperative DNA-protein interactions: implications for evolution of regulatory circuitry. *J Mol Biol* 1998, 278:331-338.

[11] Stormo GD: DNA binding sites: representation and discovery. *Bioinformatics* 2000, 16:16-23.

[12] Duret L, Bucher P: Searching for regulatory elements in human noncoding sequences. *Curr Opin Struct Biol* 1997, 7:399-406.

[13] Pennacchio LA, Rubin EM: Genomic strategies to identify mammalian regulatory sequences. *Nat Rev Genet* 2001, 2:100-109.

[14] Blanchette M, Tompa M: Discovery of regulatory elements by a computational method for phylogenetic footprinting. *Genome Res*. 2002, 12:739-748.

[15] Loots GG, Ovcharenko I, Pachter L, Dubchak I, Rubin EM: rVista for comparative sequence-based discovery of functional transcription factor binding sites. *Genome Res* 2002, 12:832-839.

[16] Schwartz S, Zhang Z, Frazer KA, Smit A, Riemer C, Bouck J, Gibbs R, Hardison R, Miller W: PipMaker--a web server for aligning two genomic DNA sequences. *Genome Res* 2000, 10:577-586.

[17] Pilpel Y, Sudarsanam P, Church GM: Identifying regulatory networks by combinatorial analysis of promoter elements. *Nat. Genet.* 2001, 29:153-159.

[18] Bussemaker HJ, Li H, Siggia ED: Regulatory element detection using correlation with expression. *Nat. Genet.* 2001, 27:167-171.

[19] Keles S, van der Laan M, Eisen MB: Identification of regulatory elements using a feature selection method. *Bioinformatics* 2002, 18:1167-1175.

[20] Wang W, Cherry JM, Botstein D, Li H: A systematic approach to reconstructing transcription networks in Saccharomyces cerevisiae. *Proc Natl Acad Sci USA* 2002, 99:16893-16898.

[21] Wagner A: Genes regulated cooperatively by one or more transcription factors and their identification in whole eukaryotic genomes. *Bioinformatics* 1999, 15:776-784.

[22] Klingenhoff A, Frech K, Quandt K, Werner T: Functional promoter modules can be detected by formal models independent of overall nucleotide sequence similarity. *Bioinformatics* 1999, 15:180-186.

[23] Pavlidis P, Furey TS, Liberto M, Haussler D, Grundy WN: Promoter region-based classification of genes. *Proceedings of the Pacific Symposium on Biocomputing* 2001, 6:151-164.

[24] Kel-Margoulis OV, Ivanova TG, Wingender E, Kel AE: Automatic annotation of genomic regulatory sequences by searching for composite clusters. *Pac Symp Biocomput* 2002, 7:187-198.

[25] Kamvysselis M, Patterson N, Birren B, Berger B, Lander ES: Whole-genome comparative annotation and regulatory motif discovery in multiple yeast species. *Proceedings of the 7th International Conference on Research in Computational Molecular Biology 2003*, 7.

[26] Chiang DY, Brown PO, Eisen MB: Visualizing associations between genome sequences and gene expression data using genome-mean expression profiles. *Bioinformatics* 2001, 17 Suppl 1:S49-S55.

[27] DeRisi JL, Iyer VR, Brown PO: Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science* 1997, 278:680-686.

[28] Gasch AP, Spellman PT, Kao CM, Carmel-Harel O, Eisen MB, Storz G, Botstein D, Brown PO: Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell* 2000, 11:4241-4257.

[29] Spellman PT, Sherlock G, Zhang MQ, Iyer VR, Anders K, Eisen MB, Brown PO, Botstein D, Futcher B: Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. *Mol Biol Cell* 1998, 9:3273-3297.

[30] Cho RJ, Campbell MJ, Winzeler EA, Steinmetz L, Conway A, Wodicka L, Wolfsberg TG, Gabrielian AE, Landsman D, Lockhart DJ, et al: A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell.* 1998, 2:65-73.

[31] Gasch AP, Huang M, Metzner S, Botstein D, Elledge SJ, Brown PO: Genomic expression responses to DNA-damaging agents and the regulatory role of the yeast ATR homolog Mec1p. *Mol Biol Cell* 2001, 12:2987-3003.

[32] Lee SE, Pellecioli A, Demeter J, Vaze MP, Gasch AP, Malkova A, Brown PO, Botstein D, Stearns T, Foiani M, et al: In: *Biological Responses to DNA Damage*, vol. 65. pp. 303-314. Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press; 2000: 303-314.

[33] Ogawa N, DeRisi J, Brown PO: New components of a system for phosphate accumulation and polyphosphate metabolism in Saccharomyces cerevisiae revealed by genomic expression analysis. *Mol. Biol. Cell* 2000, 11:4309-4321.

[34] Hughes TR, Marton MJ, Jones AR, Roberts CJ, Stoughton R, Armour CD, Bennett HA, Coffey E, Dai HY, He YDD, et al: Functional discovery via a compendium of expression profiles. *Cell* 2000, 102:109-126.

[35] Zhu J, Zhang MQ: SCPD: a promoter database of the yeast Saccharomyces cerevisiae. *Bioinformatics* 1999, 15:607-611.

[36] Ho Y, Costanzo M, Moore L, Kobayashi R, Andrews BJ: Regulation of transcription at the Saccharomyces cerevisiae start transition by Stb1, a Swi6-binding protein. *Mol Cell Biol* 1999, 19:5267-5278.

[37] Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, Hannett NM, Harbison CT, Thompson CM, Simon I, et al: Transcriptional regulatory networks in Saccharomyces cerevisiae. *Science* 2002, 298:799-804.

[38] Gancedo JM: Yeast carbon catabolite repression. *Microbiol Mol Biol Rev* 1998, 62:334-361.

[39] Kwast KE, Burke PV, Poyton RO: Oxygen sensing and the transcriptional regulation of oxygen-responsive genes in yeast. *J Exp Biol* 1998, 201:1177-1195.

[40] Vik A, Rine J: Upc2p and Ecm22p, dual regulators of sterol biosynthesis in Saccharomyces cerevisiae. *Mol Cell Biol* 2001, 21:6395-6405.

[41] Blaiseau PL, Thomas D: Multiple transcriptional activation complexes tether the yeast activator Met4 to DNA. *EMBO J* 1998, 17:6327-6336.

[42] Guarente L, Lalonde B, Gifford P, Alani E: Distinctly regulated tandem upstream activation sites mediate catabolite repression of the CYC1 gene of *S. cerevisiae*. *Cell* 1984, 36:503-511.

[43] Burke PV, Poyton RO: Structure/function of oxygen-regulated isoforms in cytochrome c oxidase. *J Exp Biol* 1998, 201:1163-1175.

[44] Lees ND, Skaggs B, Kirsch DR, Bard M: Cloning of the late genes in the ergosterol biosynthetic pathway of Saccharomyces cerevisiae--a review. *Lipids* 1995, 30:221-226.

[45] Liao GC, Rehm EJ, Rubin GM: Insertion site preferences of the P transposable element in *Drosophila melanogaster*. *Proc Natl Acad Sci USA* 2000, 97:3347-3351.

[46] Berman BP, Nibu Y, Pfeiffer BD, Tomancek P, Celniker SE, Levine M, Rubin GM, Eisen MB: Exploiting transcription factor binding site clustering to identify cis-regulatory modules involved in pattern formation in the *Drosophila* genome. *Proc. Natl. Acad. Sci. USA* 2002, 99:757-762.

[47] Cliften PF, Hillier LW, Fulton L, Graves T, Miner T, Gish WR, Waterston RH, Johnston M: Surveying Saccharomyces genomes to identify functional elements by comparative DNA sequence analysis. *Genome Res*. 2001, 11:1175-1186.

[48] Saccharomyces Genome Database. [http://genome-www.stanford.edu/Saccharomyces/]

[49] Sherlock G, Hernandez-Boussard T, Kasarskis A, Binkley G, Matese JC, Dwight SS, Kaloper M, Weng S, Jin H, Ball CA, et al: The Stanford Microarray Database. *Nucleic Acids Res* 2001, 29:152-155.

[50] Benjamini Y, Hochberg Y: Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Statist. Soc. B* 1995, 57:289-300.

[51] Press WH, Teukolsky SA, Vertterling WT, Flannery BP: *Numerical Recipes in C*, Second Edition. Cambridge: Cambridge University Press; 1992.

Network Motif Discovery Using Subgraph Enumeration and Symmetry-Breaking

Joshua A. Grochow and Manolis Kellis

Computer Science and AI Laboratory, M.I.T.
Broad Institute of M.I.T. and Harvard
joshuag@cs.uchicago.edu (current affiliation), manoli@mit.edu

Abstract. The study of biological networks and network motifs can yield significant new insights into systems biology. Previous methods of discovering network motifs – network-centric subgraph enumeration and sampling – have been limited to motifs of 6 to 8 nodes, revealing only the smallest network components. New methods are necessary to identify larger network sub-structures and functional motifs.
Here we present a novel algorithm for discovering large network motifs that achieves these goals, based on a novel symmetry-breaking technique, which eliminates repeated isomorphism testing, leading to an exponential speed-up over previous methods. This technique is made possible by reversing the traditional network-based search at the heart of the algorithm to a motif-based search, which also eliminates the need to store all motifs of a given size and enables parallelization and scaling. Additionally, our method enables us to study the clustering properties of discovered motifs, revealing even larger network elements.
We apply this algorithm to the protein-protein interaction network and transcription regulatory network of *S. cerevisiae*, and discover several large network motifs, which were previously inaccessible to existing methods, including a 29-node cluster of 15-node motifs corresponding to the key transcription machinery of *S. cerevisiae*.

1 Introduction

1.1 Network Motifs

In the past decade, new technologies have enabled the observation and study of networks of thousands and millions of nodes, such as social networks, computer networks, and, notably, *biological networks*, including protein-protein interaction networks [4–6], genetic regulatory networks [12, 18], and metabolic networks [7]. In order to extract meaningful information from these vast and sometimes noisy datasets, it is necessary to develop methods of computational analysis that are both efficient and robust to errors in the underlying data.

Network motifs – patterns of connectivity that occur significantly more frequently than expected – were introduced by Milo *et al.* [18] and provide one such robust property of biological networks. Network motifs also provide an important tool for understanding the modularity and the large-scale structure of networks

[8, 13, 20, 25]. The importance of network motifs as information-processing modules has been modeled theoretically [12, 21] and verified experimentally [8, 13, 20, 25]. Network motifs also have numerous other applications: they have been used to classify networks into "superfamilies" [17], they have been used in combination with machine learning techniques to determine the most appropriate network model for a given real-world network [16], and they have been used to determine which properties to use in parsimony models of phylogeny [19].

Unfortunately, all of these applications are hampered by the limited size of motifs discoverable by current methods. Exact counting methods have only been reported to find motifs up to 4 nodes [18] and motif generalizations up to 6 nodes [10]. Subgraph sampling methods have found motifs up to 7 [9] and 8 nodes [1, 16]. The statistical measures developed by Ziv *et al.* [26] are an important step towards larger network structures, but unfortunately lack a one-to-one correspondence with subgraphs, making them potentially difficult to interpret. Motif generalizations [10] are another important step towards these goals, although current methods are still limited to finding motif generalizations of only 6 nodes.

This current size limitation leaves many fundamental questions unanswered, and significant additional insight could be gained by exploring larger subgraphs and finding larger motifs. [1, 10]. We should not expect *a priori* that the building blocks of complex networks are as small as 4 nodes, or that the largest significant structures and pathways contain only 8 nodes. What are the fundamental building blocks? How do they combine to form larger structures? [1, 10] Do networks which share the same building blocks also share the same combinations of these blocks? [10] How can larger structures be used to distinguish between networks of different types, or between proposed models for a given network? [1]

In this paper, we present a new approach for discovering network motifs. The heart of our algorithm exhaustively assesses the significance of **a single query subgraph** as a potential motif. This can then be applied to all subgraphs of a given size to emulate the behavior of previous exhaustive algorithms, but with an exponential speed-up due to a **novel symmetry-breaking technique** (which is not feasible with previous methods). The symmetry-breaking technique also allows us to write instances of a subgraph to disk as they are found, **further eliminating limitations due to memory usage**. We are thus able to find motifs of up to 15 nodes, to find all instances of subgraphs of 31 nodes, and potentially even larger subgraphs. Although this work is motivated by biological networks, and this paper focuses on the protein-protein interaction (PPI) network and the transcription network of *S. cerevisiae*, our methods are applicable to any network – directed or undirected – and thus to many different fields, even outside the realm of biology.

In this section, we review previous work and give an overview of our algorithm, outlining several novel techniques which apply both to our approach and to previous approaches. In Sec. 2 we present our algorithm in detail. In Sec. 3 we present benchmarks comparing our approach to previous approaches. Additionally, we present data as to the effectiveness of the resulting improvements as

applied to both the transcription and PPI networks of *S. cerevisiae*. In Sec. 4, we present some of the larger subgraphs we have discovered. Finally, in Sec. 5 we discuss the significance of these contributions for the understanding of networks in general.

1.2 Limitations of Network-Centric Approaches for Motif Discovery

Two basic techniques have been proposed for identifying network motifs: exact counting [18] and subgraph sampling [1, 9, 16]. These methods attempt to determine the significance of all or many subgraphs of a given size by comparing their frequency in a given network to their frequency in a random ensemble of networks with similar properties to the original. To determine which subgraphs are motifs, subgraph sampling [1] is an effective and efficient approach, and has been used to evaluate the significance of larger subgraphs than can be evaluated by the exact counting method.

Most methods for finding DNA *sequence motifs* scan or sample a sequence pattern from a genome. Similarly, previous techniques for finding network motifs scan or sample subgraphs from a network, and count the number of occurrences of each subgraph encountered. (This process is then repeated for each network in a random ensemble resembling the initial network, and the counts are compared.) For the discovery of DNA sequence motifs, this general methodology is very efficient, because sequence motifs can be efficiently hashed based on their content. Thus a single linear scan of the genome suffices to exhaustively count all possible substrings of a given size, *regardless of the size of the substrings*.

In contrast, for the discovery of network motifs, enumerating all subgraphs of a given size is in general *exponential in the number of nodes of the subgraphs*. Additionally, there is no known efficient algorithm that correctly identifies two graphs as isomorphic or not. (The *graph isomorphism problem* is not known to be either in P or to be NP-complete.) This intrinsic difference in complexity between discovering *sequence* motifs and discovering *network* motifs makes traditional network-scanning methodologies inefficient for network motif discovery.

1.3 Distinguishing Features of the New Algorithm

To avoid these limitations of the traditional network-centric approaches, we have taken a motif-centric approach which has several attractive features, outlined here. Features 1-3 are specific to motif-centric methods, while features 4 and 5 can also benefit traditional network-centric methods.

(1) Searching for a single query graph. To avoid the increased complexity of subgraph enumeration (in the absence of an appropriate hashing scheme) our algorithm works by exhaustively searching for the instances of a *single query graph* in a network. (To find all motifs of a given size we couple this search with subgraph enumeration, using McKay's `geng` and `directg` tools [15]). Even though the *subgraph isomorphism problem* – finding a given graph as a subgraph of a larger network – is known to be NP-complete, several algorithmic improvements

enable this search to be carried out effectively in practice, even for subgraphs up to 31 nodes (and potentially even more).

(2) Mapping instead of enumerating. Rather than enumerating all connected subgraphs of a given size and testing to see whether each is isomorphic to the query graph, our algorithm attempts to map the query graph onto the network in all possible ways. We developed this technique for subgraph isomorphism independently, but subsequently identified a prior use [23].

(3) Taking advantage of subgraph symmetries. We introduce a technique that *avoids spending time finding a subgraph more than once* due to its symmetries. This technique improves the speed of our method by a factor exponential in the size of the query subgraph (Table 1). Moreover, since each instance is discovered exactly once, our algorithm can write instances to disk as they are found, greatly improving memory usage.

(4) Improved isomorphism testing. Our isomorphism test takes into account the degree of each node, and the degrees of each node's neighbors, leading to marked improvements over current motif-finding algorithms, which use exhaustive graph isomorphism tests.

(5) Subgraph hashing. When examining all subgraphs of a given size we hash the graphs based on their degree sequences, which leads to a significant improvement in the number of isomorphism tests needed. In a *directed* network, we group the query graphs based on their *undirected* isomorphism types, find all instances, and then go back to the directed network and divide these instances into their directed isomorphism types.

2 Description of the Algorithm

For clarity, we first present the basic mapping algorithm for subgraph isomorphism, without taking into account the symmetries of the query graph. In Sec. 2.2 we incorporate our symmetry-breaking technique into the algorithm. In the pseudo-code, we identify statements used solely for symmetry-breaking by enclosing them in square brackets. Finally, In Sec. 2.3 we incorporate our technique into two new methods of finding motifs.

Throughout this section, G will denote the network being searched and H will denote the query subgraph. We say that a node g of G can *support* a node h of H if we cannot rule out a subgraph isomorphism from H into G which maps h to g based on the degrees of h and g and the degrees of their neighbors. (Other constraints could also be used here, but these two proved effective and simple to implement.) This notion of support is used to exclude inconsistent candidate maps during the backtracking search.

2.1 Finding a Given Subgraph (Subgraph Isomorphism)

We start by presenting the algorithm without symmetry-breaking. Note that symmetry-breaking is not required for correctness of the algorithm.

FindSubgraphInstances(H,G):
Finds all instances of query graph H in network G

Start with an empty set of instances.
[Find Aut(H). Let H_E be the equivalence representatives of H.]
[Find symmetry-breaking conditions C for H given H_E and Aut(H).]
Order the nodes of G by increasing degree and
 then by increasing neighbor degree sequence.
For each node g of G
 For each node h of H [H_E] such that g can support h
  Let f be the partial map associating $f(h) = g$.
  Find all isomorphic extensions of f [up to symmetry]
   i.e. call IsomorphicExtensions(f,H,G[,$C(h)$]).
  Add the images of these maps to the set of all instances.
 Remove g from G.
Return the set of all instances.

FindSubgraphInstances includes the *images* of the maps in the list of instances, thus merging all maps which differ only by a symmetry of H. (Without symmetry-breaking, the algorithm spends additional time finding several distinct maps to a single subgraph.)

IsomorphicExtensions is a backtracking search to find all isomorphisms from H into G. As is standard in backtracking searches, the algorithm uses the most constrained neighbor to eliminate maps that cannot be isomorphisms: that is, the neighbor of the already-mapped nodes which is likely to have the fewest possible nodes it can be mapped to. First we select the nodes with the most already-mapped neighbors, and amongst those we select the nodes with the highest degree and largest neighbor degree sequence.

For each call to IsomorphicExtensions, f is extended by a single node. Each time an extension is made, the algorithm ensures that the newly mapped node is appropriately connected to the already-mapped nodes. Thus when IsomorphicExtensions returns a map, it is guaranteed to be an isomorphism.

We have effectively pushed the isomorphism testing of previous exhaustive methods into IsomorphicExtensions, which allows the isomorphism test to abort early. The ability to abort early when finding instances of a particular query graph presents significant savings over previous methods.

2.2 Exploiting Subgraph Symmetries to Speed Up the Search

Due to symmetries, a given subgraph of G may be mapped to a given query graph H multiple times. For example, the subgraph in Fig. 1 can be mapped to the same 6 nodes in 8 different ways. Thus a simple mapping-based search for a query graph will find each instance of the query graph as many times as the graph has symmetries. To avoid this, we compute and enforce several symmetry-breaking conditions, which ensure that there is a *unique* map from the query graph H to each instance of H in G, so that our search only spends time finding each instance once.

IsomorphicExtensions**(f,H,G[,C(h)]:**
Finds all isomorphic extensions of partial map $f : H \to G$ [satisfying C(h)]

Start with an empty list of isomorphisms.
Let D be the domain of f.
If $D = H$, return a list consisting solely of f. (Or write to disk.)
Let m be the most constrained neighbor of any $d \in D$
(constrained by degree, neighbors mapped, etc.)
For each neighbor n of $f(D)$
If there is a neighbor $d \in D$ of m such that n is *not* neighbors with $f(d)$,
or if there is a *non*-neighbor $d \in D$ of m such that n *is* neighbors with $f(d)$
[or if assigning $f(m) = n$ would violate a symmetry-breaking condition in $C(h)$],
then continue with the next n.
Otherwise, let $f' = f$ on D, and $f'(m) = n$.
Find all isomorphic extensions of f'.
Append these maps to the list of isomorphisms.
Return the list of isomorphisms.

The symmetries of a graph H are known as automorphisms (self-isomorphisms), and the group of automorphisms of H is denoted $\mathrm{Aut}(H)$. For a set A of automorphisms, two nodes are said to be "A-equivalent" if there is some automorphism in A which maps one to the other, or simply "equivalent" if $A = \mathrm{Aut}(H)$. We denote the A-equivalence of two nodes n_1, n_2 by $n_1 \sim_A n_2$. This equivalence relation partitions the nodes of H into equivalence classes. Since starting a map from two equivalent nodes is unnecessary and wasteful, FindSubgraphInstances uses a set consisting of one representative from each equivalence class of H.

The symmetry-breaking conditions are based on labellings of the nodes of H by integers, represented as maps from $H \to \mathbf{Z}$. Let $\ell : G \to \mathbf{Z}$ be a labelling of the nodes of G by *distinct* integers. Then each map $f : H \to G$ generates a labelling $L : H \to \mathbf{Z}$, given by $L(n) = \ell(f(n))$ for nodes $n \in H$. Thus, conditions on labellings of H translate into restraints on maps from H into G.

Given a set of conditions C, we say an automorphism α *preserves the conditions* C if, given a labelling L_1 of H which satisfies C, the corresponding labelling $L_2 : H \to \mathbf{Z}$ given by $L_2(n) = L_1(\alpha(n))$ also satisfies C. We are thus searching for conditions C such that the only automorphism which preserves C is the identity. This ensures there will be exactly one map from H onto each of its instances in G which satisfies the conditions.

To find these conditions, we pick an $\mathrm{Aut}(H)$-equivalence class $\{n_0, \ldots, n_k\}$ of nodes of H, and we impose the condition $L(n_0) < \mathrm{MIN}(L(n_1), \ldots, L(n_k))$. Any automorphism must send n_0 to one of the n_i, since these are all of the nodes equivalent to n_0. But to preserve this condition, an automorphism must send n_0 to itself. Then we continue recursively, replacing $\mathrm{Aut}(H)$ with the set A of automorphisms which send n_0 to itself. For example, see Fig. 1.

Because FindSubgraphInstances starts with a particular node, we can consider that node already fixed. (Note that the version of FindSubgraphInstances which uses symmetry-breaking only iterates over a set of equivalence

class representatives, and not over all nodes of H.) Thus for each representative used by FindSubgraphInstances, SymmetryConditions must generate a series of symmetry-breaking conditions which start by fixing that node.

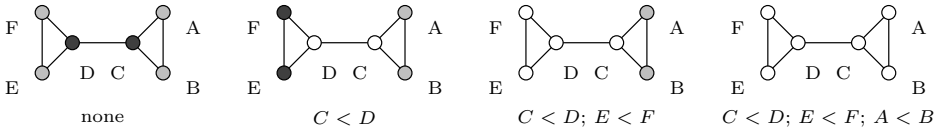

Fig. 1. Finding conditions that will break all the symmetries of a 6-node graph. White nodes are fixed by any automorphism preserving the indicated conditions, and other nodes are shaded according to their equivalence class under the automorphisms which preserve the indicated conditions.

To find the automorphisms of H, we use IsomorphicExtensions *without* symmetry-breaking, which returns an exhaustive list of all isomorphisms from H to itself. To find the automorphisms which fix a node or a set of nodes, the algorithm filters this list in a single pass.

Finding the automorphisms of a graph is thought to be computationally expensive[1], but in practice we have found this is far from the bottleneck in motif-finding algorithms. We were able to *exhaustively* find the automorphisms of all 11,117 8-node undirected graphs in under 30 seconds on a standard laptop, and McKay's tools [14] can find all the automorphisms of very large graphs very rapidly (e.g. some graphs with thousands of nodes and millions of automorphisms, in less than one second on a standard laptop).

SymmetryConditions:
Finds symmetry-breaking conditions for H given H_E, $\mathbf{Aut}(H)$

Let M be an empty map from equivalence representatives to sets of conditions.
For each $n \in H_E$
 Let C be an empty set of conditions.
 $n' \leftarrow n$, and $A \leftarrow \mathrm{Aut}(H)$.
 Do until $|A| = 1$:
  Add "$\textsc{label}(n') < \textsc{min}\{\textsc{label}(m) | m \sim_A n' \text{ and } m \neq n'\}$" to C.
  $A \leftarrow \{f \in A | f(n') = n'\}$.
  Find the largest A-equivalence class E.
  Pick $n' \in E$ arbitrarily.
 Let $M(n) = C$.
Return M.

[1] Finding graph automorphisms is at least as hard as determining if two graphs are isomorphic. Like the graph isomorphism problem, the graph automorphism problem is not known to be either in P or to be NP-complete.

2.3 Subgraph Evaluation and Network Motif Discovery

To find network motifs we enumerate candidate subgraphs H (exhaustively or by sampling), and evaluate each candidate based on its instances.

Evaluating candidate subgraphs. We find all instances of a query graph H in the network G, as well as in a random ensemble of networks with the same degree distribution and same distribution of 3-node subgraphs as G.[2] We evaluate the overrepresentation of the query graph H based on the z-score of its abundance in G against the distribution of its abundance in the random ensemble, as in [18, 21].

Exhaustive subgraph enumeration. Our method can be used to find all instances of subgraphs of a given size, similar to previous exhaustive methods. To do this, we generate all non-isomorphic graphs of a particular size using McKay's `geng` and `directg` tools [15]. Then for each graph, we evaluate its significance as above.

Subgraph sampling. Our method can also be used in combination with subgraph sampling. We sample connected subgraphs (usually relatively large, compared to previous network motifs: 10, 15, or 20 nodes) by picking a node at random, and taking a random walk until we have as many nodes as desired [16]. Then we assess the significance of this subgraph as above.

Sampling subgraphs to find anti-motifs. Some studies have also considered *anti*-motifs: subgraphs which are significantly *under*represented compared to randomized versions of the network. To use a sampling method to find anti-motifs, it might be more fruitful to sample initial subgraphs from the random ensemble rather than the network being studied. Anti-motifs will be more prevalent in the ensemble than in the target network, and thus are more likely to be discovered by sampling from the ensemble.

3 Results and Evaluation

We applied our algorithm to the PPI network (1379 nodes, 2493 edges) [4] and transcription network (685 nodes, 1052 edges) [2] of *S. cerevisiae* and compared its performance to previous methods of motif disccovery.

Comparison with previous methods: time. We compare the time requirements of our method to those of Milo *et al.* [18] (Fig. 2). We make this comparison on the undirected PPI network of *S. cerevisiae* [4], by exhaustively counting subgraphs up to 7 nodes.

We implemented both our algorithm and two versions of the Milo *et al.* algorithm [18]: both as originally presented [18], and also by additionally hashing subgraphs by their degree sequence (Sec. 1.3). Fig. 2 shows that our algorithm

[2] Although Shen-Orr *et al.*[21] use a model in which the distribution of $(n-1)$-node subgraphs is preserved when looking for n-node motifs, they only applied this to the case $n = 4$, and we have found it computationally infeasible to preserve this distribution for $n > 4$. Nonetheless, we have found it fruitful to preserve the distribution of 3-node subgraphs, regardless of n.

provides an *exponential* improvement in time, even compared to the modified version of the previous algorithm [18].

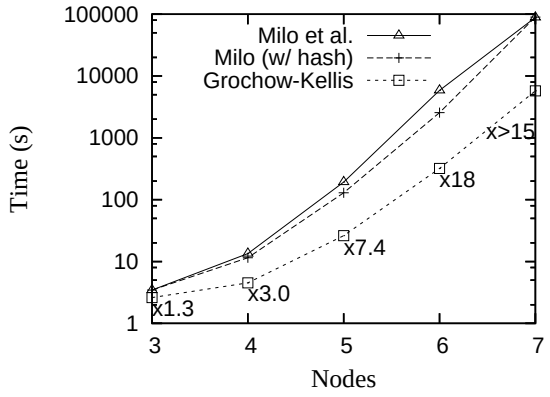

Fig. 2. The runtimes of the original algorithm of Milo *et al.* [18], an improved version of their algorithm, and our new algorithm, as applied to the undirected PPI network of *S. cerevisiae* [4]. The speed-up from the original algorithm of Milo *et al.* [18] to our algorithm is indicated. (Note: the values for 7 nodes for the two variants of Milo *et al.*'s algorithm are underestimates: the program ran out of memory before finishing.)

Comparison with previous methods: space. Our method gains an exponential memory advantage over previous exhaustive methods by not keeping a list of previously visited subgraphs. In the previous exact counting method [18], a list of the subgraphs encountered at each node is necessary in order to avoid duplication, even when the instances of the subgraphs are not desired as output. Thus the space required by the previous method is proportional to the number of subgraphs of a given size going through a given node, which can be exponential in the size of the subgraphs. Because our method does not need to keep such a list, its asymptotic memory requirements are determined by the maximum depth of recurion of IsomorphicExtensions, which is linear in the size of the query graph. Our method thus uses exponentially less space than previous exhaustive methods.

Disk usage. Furthermore, our algorithm uses no more memory to find a list of all instances than to simply count the instances. Since each instance is encountered exactly once, it can be written to disk and removed from active memory as soon as it is encountered, using effectively no additional memory.

Improvement due to symmetry-breaking. The main reason for these improvements is our novel symmetry-breaking technique. Symmetry-breaking ensures that each instance is discovered exactly once, so our method does not have to check a list of the subgraphs previously encountered at a node in order to avoid duplicate counting, while the previous method of exact counting does. Ta-

ble 1 quantifies this contribution explicitly, showing that the average number of automorphisms of graphs weighted by their occurences in the PPI network and regulatory network of yeast – i.e. the savings gained by symmetry-breaking – appears to grow exponentially.

Table 1. The number of subgraphs encountered by our algorithm with and without symmetry-breaking (including multiple encounters for the version without symmetry-breaking). The improvement factor is exactly the average number of automorphisms of subgraphs of the associated size.

	Undirected PPI Network			Directed Regulatory Network		
Nodes	Total Subgraphs Searched	With Symmetry-Breaking	Improvement	Total Subgraphs Searched	With Symmetry-Breaking	Improvement
3	3.7×10^4	1.1×10^4	$\times3.13$	2.6×10^4	1.3×10^4	$\times2.02$
4	4.0×10^5	7.0×10^4	$\times5.77$	9.7×10^5	1.8×10^5	$\times5.41$
5	4.4×10^6	4.1×10^5	$\times10.9$	4.4×10^7	2.5×10^6	$\times18.0$
6	5.1×10^7	2.3×10^6	$\times22.2$	2.3×10^9	3.2×10^7	$\times73.3$
7	5.7×10^8	1.2×10^7	$\times46.3$	1.3×10^{11}	4.0×10^8	$\times334$
8	6.4×10^9	6.6×10^7	$\times96.2$	—	—	—

4 Discovered Motifs and Their Biological Significance

Discovered motifs. We exhaustively evaluated all candidate motifs and anti-motifs up to 7 nodes in the PPI network of *S. cerevisiae*[4] (1379 nodes, 2493 edges). We used a random ensemble of networks with the same degree distribution and the same distribution of 3-node subgraphs as the PPI network.[3] The most significant subgraphs tend to be motifs rather than anti-motifs: using a z-score cutoff of 4.0, only 3 of the 54 significant subgraphs of size at most 7 were anti-motifs. Two of the motifs were trees, and the most dense motif had 18 edges. Most of the significant graphs were of moderate density: the mean number of edges for 7-node motifs and anti-motifs is 11.49 ± 2.89.

Large discovered motifs. We also discovered larger motifs by first sampling connected subgraphs from the PPI network of *S. cerevisiae*, and then assessing their significance using our method. We sampled approximately 100 connected subgraphs of 15 and 20 nodes, and found several motifs. One such 15-node motif (Fig. 3) represents a common connectivity pattern found within the transcriptional machinery of *S. cerevisiae* (see discussion below).

Clustering of discovered motifs and larger network structures. We noted that almost all of the larger subgraphs we evaluated have large numbers of overlapping instances, which become apparent since our method reports all network instances of a discovered motif. To quantify this property, we developed

[3] See footnote 2.

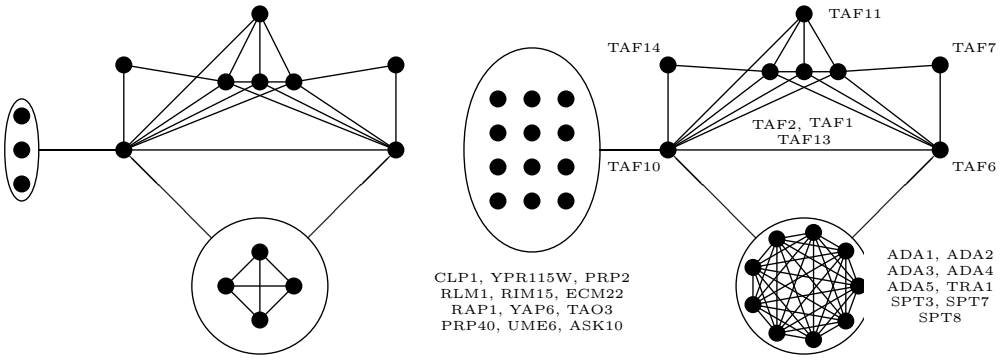

Fig. 3. A motif of 15 nodes and 34 edges (left). An edge from a group of nodes to a node n indicates that each node in the group is connected to n. This motif appears 27,720 times in the PPI network of *S. cerevisiae*[4], and does not appear at all in random ensembles which preserve the degree distribution and the distribution of 3-node subgraphs. All 27,720 instances are clustered into a total of 29 nodes (right), corresponding to the cellular transcription machinery.

a subgraph clustering score, based on the number of subgraph instances overlapping a given node, averaged over all nodes in any subgraph instance. We applied this score to evaluate the clustering properties of all discovered motifs, and we found that indeed some of the most abundant motifs show striking clustering properties.

The clustered instances frequently reveal important larger network structures. For example, the 15-node motif of Fig. 3 occurs 27,720 times in a single sub-network of 29 nodes, part of the core transcription machinery of *S. cerevisiae*. This includes a complete 11-node graph (including the two central hubs) corresponding to the SAGA complex, and consisting almost entirely of chromatin modification and histone acetylation factors an 8-node core (shared by all instances of the 15-node motif) corresponding to the TFIID complex, and 12 attachments, which are known activators and suppressors of these two complexes [11]. Similarly, the subgraph of 20 nodes shown in Fig. 4 occurs 5,020 times in a total of 31 nodes, enriched in cell-cycle regulation.

The role of combinatorial effects. The extreme clustering properties of the most abundant motifs appear to result from combinatorial connectivity patterns prevalent in larger network structures. For example, all 27,720 instances of the 15-node motif in Fig. 3 result by choosing 3 attachments from the left and 4 attachments from the the bottom of Fig. 3 ($\binom{12}{3}\binom{9}{4} = 27,720$), and similarly for the 5,020 instances of the 20-node subgraph in Fig. 4. Additionally, in the random ensemble, these combinatorially appearing structures occur either thousands of times, or not at all – they almost never occur just a few or a few hundred times.

Although motif clustering has previously been observed [3] and demonstrated analytically [24], previous motifs studied do not have enough nodes to exhibit the extreme combinatorial clustering we observed for large subgraphs (at least 15 nodes). The magnitude of this combinatorial clustering effect brings into question

the current definition of network motif, when applied to larger structures. We propose that additional statistics, either alone or in combination, might be well-suited to identify larger meaningful network structures: our subgraph clustering score, the total number of nodes covered by all instances of the query graph, the total number of edges, and the weighting of the number of nodes/edges based on the number of overlapping instances. All of these statistics can be easily calculated using our algorithm, since it finds and stores all motif instances, and these will be the subject of future studies.

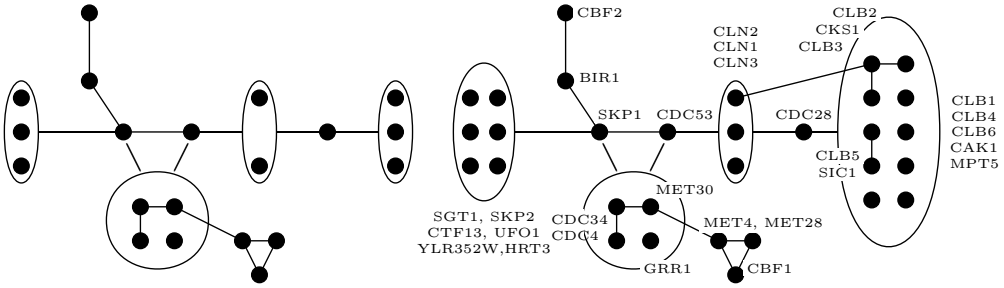

Fig. 4. A subgraph of 20 nodes and 27 edges (left). An edge from a group of nodes to a node n indicates that each node in the group is connected to n. This subgraph appears 5,020 times in the PPI network of *S. cerevisiae* [4]. All 5,020 instances are clustered into a total of 31 nodes (right), enriched in cell-cycle regulation.

5 Discussion

We presented a novel approach to the discovery of network motifs, based on a solution to the subgraph isomorphism problem that uses a new symmetry-breaking technique, an improved isomorphism test, and hashing based on degree sequences. Several of the techniques presented in our algorithm can also be used in previous algorithms, and lead to significant improvements.

We implemented our algorithm and used it to find significant structures of 15 and 20 nodes in the PPI network and the regulatory network of *S. cerevisiae*, where previous methods had been limited to motifs of 6 and 8 nodes. Using our approach to motif-finding, we re-discovered the cellular transcription machinery – as a 29-node cluster of 15-node motifs – based solely on the structure of the protein interaction network.

Previous methods of motif discovery were network-centric, and could therefore not take advantage of subgraph symmetries. By using a motif-centric algorithm instead, we are able to use symmetry-breaking to get an exponential improvement.

5.1 Applications and Advantages of the New Method

(1) Finding larger motifs. Our improvements have enabled the exhaustive discovery of motifs up to 7 nodes. To find even larger motifs, we sample a connected subgraph as in [16], and then find *all* its instances and assess its significance using our method. This technique has enabled us to find motifs up to 15 nodes and examine subgraphs up to 31 nodes.

(2) Querying a particular subgraph. Our method can be used to query whether a particular subgraph is significant, whereas previous methods can only do this by examining all subgraphs of the same size, which quickly becomes prohibitive for even moderate sizes. This technique could be used to explore *in silico* the prevalence of a subgraph of interest, identified experimentally (e.g. known pathways), computationally (e.g. motif generalizations [10]), or genetically.

(3) Exploring motif clustering. Because our algorithm finds all instances of a given subgraph, it can be used to explore how these instances cluster together to form larger structures. For example, after finding a 15-node motif, we were able to determine that all of its 27,720 instances clustered in 29 nodes (Fig. 3).

(4) Time and space. Our method, applied to all subgraphs of a given size, takes exponentially less time than previous methods, even when we implement the previous method with our hashing scheme (Sec. 3). Additionally, there are essentially no space limitations on our method: since each instance is found exactly once due to our symmetry-breaking technique, it can be written to disk and removed from active memory as soon as it is found.

(5) Parallelization. Our method is more easily parallelizable than previous motif-finding methods, since each subgraph can be counted on a separate processor. We have found this attribute to be very useful, and we believe other researchers will as well, as cluster computing becomes commonplace in the computational biology community.

5.2 Clustering Properties of Large Subgraphs and Motifs

We revealed that larger subgraphs tend to cluster together *combinatorially* – that is, all instances share a significant core of nodes, and each instance represents a choice of attachments to these core nodes. This combinatorial clustering brings into question the relevance of the standard definition of network motif for large subgraphs of 15 nodes or more. We proposed several statistics which may be more appropriate in this domain.

Finally, we mention that the statistics of Ziv *et al.* [26] may not suffer from these combinatorial effects. The main drawback of these statistics is their lack of one-to-one correspondence with subgraphs. In combination with our algorithm, however, the large subgraphs encompassed by these statistics could be further explored, allowing for a clearer interpretation of the most significant statistics.

Moving forward, we expect the network motifs and methodology presented here will open a window into the large structures and global organization of biological and other networks.

Acknowledgements. The authors would like to thank Pouya Kheradpour, Mike Lin, Matt Rasmussen, Alex Stark, and Radek Szklarczyk (all at the M.I.T. Computer Science and AI Laboratory) for many useful and interesting discussions.

All algorithms were implemented in Java using the Java Universal Networks and Graphs (JUNG) framework [22]. Our software is available on request. McKay's `geng` and `directg` tools [15] were used to enumerate all graphs of a given size. Much of the computational work was carried out on the compute cluster at the Broad Institute of M.I.T. and Harvard.

This work was supported in part by startup funds from Professor Kellis.

References

1. K. Baskerville and M. Paczuski. Subgraph ensembles and motif discovery using a new heuristic for graph isomorphism, 2006. arxiv.org:q-bio/0606023.
2. M. C. Costanzo, M. E. Crawford, J. E. Hirschman, J. E. Kranz, P. Olsen, L. S. Robertson, M. S. Skrzypek, B. R. Braun, K. L. Hopkins, P. Kondu, C. Lengieza, J. E. Lew-Smith, M. Tillberg, and J. I. Garrels. Ypd(tm), pombepd(tm), and wormpd(tm): model organism volumes of the bioknowledge(tm) library, an integrated resource for protein information. *Nucleic Acids Res.*, 29:75–79, 2001.
3. R. Dobrin, Q. K. Beg, A.-L. Barabási, and Z. N. Oltvai. Aggregation of topological motifs in the *Escherichia coli* transcriptional regulatory network. *BMC Bioinformatics*, 5:10, Jan 2004.
4. J.-D. J. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. M. Walhout, M. E. Cusick, F. P. Roth, and M. Vidal. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430(6995):88–93, Jul 2004.
5. A. Jaimovich, G. Elidan, H. Margalit, and N. Friedman. Towards an integrated protein-protein interaction network: a relational markov network approach. *J. Comp. Bio.*, 13:145–164, 2006.
6. H. Jeong, S. Mason, A.-L. Barabási, and Z. N. Oltvai. Centrality and lethality of protein networks. *Nature*, 411, 2001.
7. H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. The large-scale organization of metabolic networks. *Nature*, 407, 2000.
8. S. Kalir, J. McClure, K. Pabbaraju, C. Southward, M. Ronen, S. Leibler, M. G. Surette, and U. Alon. Ordering genes in a flagella pathway by analysis of expression kinetics from living bacteria. *Science*, 292(5524):2080–2083, Jun 2001.
9. N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, 20(11):1746–1758, Jul 2004. Evaluation Studies.
10. N. Kashtan, S. Itzkovitz, R. Milo, and U. Alon. Topological generalizations of network motifs. *Phys. Rev. E*, 70:031909, 2004.
11. T. I. Lee and R. A. Young. Transcription of eukaryotic protein-coding genes. *Annu. Rev. Genet.*, 34:77–137, 2000.
12. S. Mangan and U. Alon. Structure and function of the feed-forward loop network motif. *PNAS*, 100(21):11980–11985, Oct 2003.
13. S. Mangan, A. Zaslaver, and U. Alon. The coherent feedforward loop serves as a sign-sensitive delay element in transcription networks. *J. Mol. Biol.*, 334(2):197–204, Nov 2003.

14. B. D. McKay. Practical graph isomorphism. In *Proceedings of the Tenth Manitoba Conference on Numerical Mathematics and Computing, Vol. I (Winnipeg, Man., 1980)*, volume 30, pages 45–87, 1981. http://cs.anu.edu.au/∼bdm/nauty/.
15. B. D. McKay. Isomorph-free exhaustive generation. *J. Algorithms*, 26:306–324, 1998.
16. M. Middendorf, E. Ziv, and Chris H. Wiggins. Inferring network mechanisms: the Drosophila melanogaster protein interaction network. *PNAS*, 102(9):3192–3197, Mar 2005.
17. R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, and U. Alon. Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–1542, Mar 2004.
18. R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, Oct 2002.
19. T. M. Przytycka. An important connection between network motifs and parsimony models. In *RECOMB 2006*, pages 321–335, 2006.
20. M. Ronen, R. Rosenberg, B. I. Shraiman, and U. Alon. Assigning numbers to the arrows: parameterizing a gene regulation network by using accurate expression kinetics. *Proc Natl Acad Sci U S A*, 99(16):10555–10560, Aug 2002.
21. S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of Escherichia coli. *Nature Genetics*, 31(1):64–68, May 2002.
22. JUNG Framework Development Team. Jung: The java universal network/graph framework, 2005.
23. J. R. Ullman. An algorithm for subgraph isomorphism. *J. Assoc. Comp. Mach.*, 23(1):31–42, Jan 1976.
24. A. Vazquez, R. Dobrin, D. Sergi, J.-P. Eckmann, Z. N. Oltvai, and A.-L. Barabasi. The topological relationship between the large-scale attributes and local interaction patterns of complex networks. *PNAS*, 101(52):17940–17945, Dec 2004.
25. A. Zaslaver, A. E. Mayo, R. Rosenberg, P. Bashkin, H. Sberro, M. Tsalyuk, M. G. Surette, and U. Alon. Just-in-time transcription program in metabolic pathways. *Nature Genetics*, 36(5):486–491, May 2004.
26. E. Ziv, R. Koytcheff, M. Middendorf, and C. Wiggins. Systematic identification of statistically significant network measures. *Phys. Rev. E*, 71:016110, 2005.

Information-Theoretic Inference of Gene Networks Using Backward Elimination

Patrick E. Meyer[1,2,3], **Daniel Marbach**[1,2], **Sushmita Roy**[1,2] **and Manolis Kellis**[1,2]
[1]Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, MA, USA
[2]Broad Institute of MIT and Harvard, MA, USA
[3]Machine Learning Group, Université Libre de Bruxelles, Brussels, Belgium

Abstract— *Unraveling transcriptional regulatory networks is essential for understanding and predicting cellular responses in different developmental and environmental contexts. Information-theoretic methods of network inference have been shown to produce high-quality reconstructions because of their ability to infer both linear and non-linear dependencies between regulators and targets. In this paper, we introduce MRNETB an improved version of the previous information-theoretic algorithm, MRNET, which has competitive performance with state-of-the-art algorithms.*

MRNET infers a network by using a forward selection strategy to identify a maximally-independent set of neighbors for every variable. However, a known limitation of algorithms based on forward selection is that the quality of the selected subset strongly depends on the first variable selected. In this paper, we present MRNETB, an improved version of MRNET that overcomes this limitation by using a backward selection strategy followed by a sequential replacement. Our new variable selection procedure can be implemented with the same computational cost as the forward selection strategy.

MRNETB was benchmarked against MRNET and two other information-theoretic algorithms, CLR and ARACNE. Our benchmark comprised 15 datasets generated from two regulatory network simulators, 10 of which are from the DREAM4 challenge, which was recently used to compare over 30 network inference methods. To assess stability of our results, each method was implemented with two estimators of mutual information. Our results show that MRNETB has significantly better performance than MRNET, irrespective of the mutual information estimation method. MRNETB also performs comparably to CLR and significantly better than ARACNE indicating that our new variable selection strategy can successfully infer high-quality networks.

Keywords: mutual information, systems biology, network inference

1. Introduction

Transcriptional regulatory networks are networks of transcription factor (TF) proteins and their target genes, where each edge represents the regulatory activity of each TF on a target gene. These networks summarize the underlying regulatory circuitry by which precise, condition-specific patterns of gene expression are generated under different cellular contexts. Understanding these networks can not only provide insights into how cells function, but can lead to effective drug delivery for treating human diseases by specific network-informed intervention.

Network inference methods (also called *reverse engineering* methods) attempt to reconstruct the transcriptional network of a cell from gene expression data [1]. However, this is a challenging task because of the large amount of experimental and biological noise in expression data, and because of the high dimensionality and combinatorial nature of the problem [25]. Notwithstanding, the bioinformatics community has developed several successful network inference techniques in the last decade [8]. In particular, information-theoretic methods for network inference have been shown to scale to datasets with several thousand genes and a limited amount of samples [7], [13], [15].

In this paper, we present MRNET Backward (MRNETB), an improved version of the previous network inference method called MRNET (Minimum Redundancy NETwork) [15]. MRNET infers a network of interactions between TFs and target genes by using a forward selection strategy to identify a maximally independent set of neighbors for every variable. However, methods based on forward selection, including MRNET, rely on the correct choice of the first neighbor and suffer in performance if the first neighbor is chosen incorrectly. MRNETB overcomes this limitation by implementing a combination of a backward elimination and a sequential replacement procedure. Importantly, we implement our new neighbor selection procedure at the same computational cost as forward selection.

Our proposed technique is benchmarked against MRNET and two other state-of-the-art information-theoretic algorithms, namely CLR (Context Likelihood of Relatedness) [7] and ARACNE (Algorithm for the Reconstruction of Accurate Cellular Networks) [13]. Our benchmark comprised fifteen datasets generated from two regulatory network simulators. Ten out of the fifteen datasets come from the DREAM4 challenge, a competition devoted to compare the performance of network inference methods [11]. To assess stability of our results, each method was implemented

with two different estimators of mutual information. Our results show that irrespective of the mutual information estimation used, MRNETB performs better than MRNET and ARACNE and performs comparably against CLR. Our new strategy thus successfully infers high-quality networks, without incurring additional computational costs.

The rest of the paper is organized as follows: Section 2 provides background on information-theoretic methods of network inference and the two network inference methods, CLR and ARACNE. Section 3 describes MRNET and our new approach, MRNETB. Section 4 describes the experimental framework and results and the conclusions are summarized in Section 5.

2. Mutual Information Network Inference

Mutual information network inference methods comprise a subcategory of network inference methods, which infer regulatory interactions between genes based on pairwise mutual information [15]. As a first step, these methods require the computation of the mutual information matrix (MIM), a square matrix whose mim_{ij} element is given by the mutual information between X_i and X_j:

$$\text{mim}_{ij} = I(X_i; X_j) \tag{1}$$

where X_i and X_j are random variables denoting the expression levels of genes i and j, respectively.

The main advantages of mutual information networks for inference of transcriptional regulatory networks are:

- The computational complexity is affordable. This results from the fact that only $\binom{n}{2}$ calls of mutual information, based on bivariate probability distributions, are required to compute the MIM [13].
- The number of required samples is rather low, since only bivariate distribution are to be estimated [17].

In the following, we first review two state-of-the-art network inference methods based on pairwise mutual information. We proceed by describing two commonly used mutual information estimators.

2.1 Mutual Information Estimation

Two popular ways of computing mutual information are: (1) discretizing variables with an equal frequency binning (so that marginal distributions are uniform), and (2) assuming normally distributed variables. We will now describe these two procedures.

2.1.1 Uniformly distributed discrete variables and empirical estimation

If X_i is a continuous random variable taking values between a and b, the interval $[a, b]$ can be discretized by partitioning it into $|\mathcal{X}_i|$ subintervals, called *bins*, where the symbol $\mathcal{X}_i$ denotes the bin vector. We use $\#(x_{i_k})$ to denote the number of data points in the kth bin of variable X_i, the symbol $m = \sum_k \#(x_{i_k})$ to denote the total number of samples and the symbol $\hat{p}(x_{i_k}) = \frac{\#(x_{i_k})}{m}$ to denote the empirical probability.

One possible binning strategy is to divide the interval $[a, b]$ of X_i such that each subinterval contains the same number of data points. It follows that subinterval sizes are typically different. This binning strategy is often referred as *equal frequency binning*. The number of subintervals should be chosen so that all bins contain a significant number of samples. Here, we set $|\mathcal{X}_i| = \sqrt{m}$, which is a common choice [28]. The entropy of X_i is $H(X_i) = \log |\mathcal{X}_i|$, since its distribution is uniform across the bins. As a result, the "empirical" or "plug-in" estimation of the mutual information becomes

$$\begin{array}{c} I(X_i; X_j) = \log(|\mathcal{X}_i||\mathcal{X}_j|) \\ + \sum_{x_{i_k} \in \mathcal{X}_i} \sum_{x_{j_l} \in \mathcal{X}_j} \hat{p}(x_{i_k}, x_{j_l}) \log \hat{p}(x_{i_k}, x_{j_l}) \end{array} \tag{2}$$

It should be noted that this entropy estimation tends to underestimate the true value of entropy, i.e. $E[H(p)] \leq H(E[p])$ [18].

2.1.2 Normally distributed continuous variables

Let X be a multivariate Gaussian random variable with mean μ and covariance C. The probability density function of X is

$$f(X) = \frac{1}{\sqrt{(2\pi)^n |C|}} \exp^{(-\frac{1}{2}(x-\mu)^T C^{-1}(x-\mu))}. \tag{3}$$

The mutual information between two Gaussian random variables, X_i and X_j, is then given by

$$I(X_i, X_j) = \frac{1}{2} \log \left(\frac{\sigma_{ii}\sigma_{jj}}{|C|} \right) \tag{4}$$

$$= -\frac{1}{2} \log(1 - \rho^2). \tag{5}$$

where $|C|$ is the determinant of the covariance matrix and ρ is the Pearson's correlation [9].

The Spearman rank correlation was shown to have better empirical performance for network inference than Pearson's correlation [17].

2.2 Context Likelihood of Relatedness (CLR)

The CLR algorithm [7] is an extension of the relevance network approach [4]. The latter approach has been introduced for gene clustering and successfully applied to infer relationships between RNA expression and chemotherapeutic susceptibility [3]. The relevance networks approach consists of inferring a network in which a pair of genes $\{X_i, X_j\}$ are linked by an edge if the mutual information $I(X_i; X_j)$ is larger than a given threshold θ. The complexity of the method is $O(n^2)$ since all pairwise interactions are considered.

The CLR algorithm derives a score from the empirical distribution of the mutual information for each pair of genes. In particular, instead of considering the information $I(X_i; X_j)$ between genes X_i and X_j, it estimates a score $w_{ij} = \sqrt{z_i^2 + z_j^2}$, where

$$z_i = \max\left(0, \frac{I(X_i; X_j) - \mu_i}{\sigma_i}\right) \quad (6)$$

The parameters μ_i and σ_i are the mean and the standard deviation of the empirical distribution of the mutual information values $I(X_i, X_k)$ of X_i with all other variables X_k ($k = 1, \ldots, n$). The CLR algorithm has a complexity in $O(n^2)$. It was successfully applied to decipher the *E. coli* transcriptional regulatory network [7].

2.3 Algorithm for Reconstruction of Accurate Cellular Networks (ARACNE)

ARACNE [13] is based on the Data Processing Inequality, which states that, if gene X_i interacts with gene X_j through gene X_k, then

$$I(X_i; X_j) \leq \min\left(I(X_i; X_k), I(X_j; X_k)\right)$$

ARACNE first assigns to each pair of nodes a weight equal to their mutual information. Then, all edges for which $I(X_i; X_j) < \theta$ are removed, where θ is a given threshold. Eventually, the weakest edge of each triplet is interpreted as an indirect interaction and is removed.

An extension of ARACNE removes the weakest edge only if the difference between the two lowest weights lies above a threshold η. Hence, η allows the number of pruned edges to be tunable.

If the network is a tree including only pairwise interactions, the method guarantees the reconstruction of the original network, once it is provided with the exact MIM [13]. ARACNE's complexity is $O(n^3)$ since the algorithm considers all triplets of genes. ARACNE successfully recovered components of transcriptional regulatory networks in mammalian cells and was shown to have favorable performance compared to Bayesian networks and relevance networks [13].

3. Minimum Redundancy Networks

MRNET infers a network by using a variable selection procedure called Maximum Relevance Minimum Redundancy (MRMR) for every random variable $X_j \in X$, [23], [19]. Assume X_j is the variable for which we need to select the predictor variables. The MRMR methods ranks a set $X_{S_j} \subseteq \{X \setminus X_j\}$ of the predictor variables according to the difference between the mutual information of $X_i \in X_{S_j}$ with X_j (the relevance) and the average mutual information with the selected variables in X_{S_j} (the redundancy). The rationale is that direct interactions should be ranked before indirect interactions (i.e. the ones with redundant information with the direct ones) by the method.

The MRMR method has been introduced together with a forward selection that starts by selecting the variable X_k that has the highest mutual information with the target X_j. The second selected variable X_i will be the one having a high information $I(X_i; X_j)$ with the target and at the same time a low information $I(X_i; X_k)$ with the previously selected variable. In the following steps, given a set X_{S_j} of selected variables, X_{S_j} is updated by choosing the variable, X_i^{MRMR}, that maximizes the MRMR score s_i:

$$\begin{gathered} X_i^{MRMR} = \arg\max_{X_i \in X_{-(i,j)}} (s_i) \\ s_i = I(X_i; X_j) - \frac{1}{|S_j|} \sum_{k \in S_j} I(X_i; X_k) \end{gathered} \quad (7)$$

At each step of the algorithm, the selected variable thus represents a trade-off between relevance and redundancy.

The generic principle of MRNET consists of identifying a subset X_{S_j} (for each variable X_j, $j = 1, 2.., n$) whose variables have maximal pairwise relevance with X_j and, at the same time, maximal pairwise independence among them. More formally,

$$\begin{gathered} X_{S_j} = \arg\max_{X_{S_j} \in X_{-j}} (u - r) \\ u = \frac{1}{|S_j|} \sum_{i \in S_j} I(X_i; X_j) \\ r = \frac{2}{|S_j|(|S_j|-1)} \sum_{i,k>i \in S_j} I(X_i; X_k) \end{gathered} \quad (8)$$

The network inference approach MRNET consists of repeating this selection procedure for each target gene $X_j \in X$. For each pair $\{X_i, X_j\}$, MRNET returns two (not necessarily equal) scores s_i and s_j according to (7). The score of the pair $\{X_i, X_j\}$ is defined as the maximum between s_i and s_j. A specific network can then be inferred by only keeping edges whose score lies above a given threshold θ (as in CLR). Thus, the algorithm infers an edge between X_i and X_j either when X_i is a well-ranked predictor of X_j ($s_i > \theta$), or when X_j is a well-ranked predictor of X_i ($s_j > \theta$). In practice, the selection of variables stops when the average redundancy term $\frac{1}{|S_j|} \sum_{k \in S_j} I(X_i; X_k)$ exceeds the relevance term $I(X_i; X_j)$.

MRNET has complexity of $O(n^3)$ since each variable selection is $O(n^2)$. MRNET was shown to have similar or better performance than alternative information-theoretic network inference methods [15], [10], [21], [17].

3.1 MRNET Backward (MRNETB)

In this section, we introduce MRNETB, an improved version of MRNET. This new algorithm overcomes a major limitation of MRNET coming from the use of forward selection as the subset search strategy. A known limitation of algorithms based on forward selection, including MRNET, is that the quality of the selected subset strongly depends on the first variable selected. If the first selected variable, i.e. the variable with the highest mutual information with the target variable X_j, is not a true neighbor of X_j, then minimizing

redundancy (or equivalently maximizing independency) with that wrongly selected variable is not desirable.

The optimisation problem of MRNET (8) is a binary quadratic optimization problem. Backward elimination combined with a sequential search is known to perform well on binary quadratic problems [2]. The backward elimination method starts with a set containing all the variables and then selects the variable X_i whose removal induces the highest increase of the objective function till the stopping criterion is fulfilled (i.e. a relevance term $\frac{1}{|S_j|}\sum_{i\in S_j} I(X_i;X_j)$ higher than redundancy term $\frac{2}{|S_j|(|S_j|-1)}\sum_{i,k>i\in S_j} I(X_i;X_k)$). The procedure is enhanced by an iterative sequential replacement which, at each step, swaps the status of a selected and a non selected variable such that the largest increase in the objective function is achieved. The sequential replacement is stopped when no further improvement is met.

Forward selection, backward elimination, and sequential replacement all have an algorithmic complexity of $O(n^2)$ [14]. In other words, the network built using a backward elimination followed by sequential replacement has the same asymptotic computational cost as the one based on a forward selection strategy. Backward elimination has been previously shown to outperform forward selection in variable selection [6]. In the next section, we show that the backward selection strategy also improves the performance in network inference using mutual information.

We have made MRNETB available in the latest version (2.5) of our bioconductor open-source package called minet [16].

4. Experiments

We compare the performance of our new approach MRNETB with the three existing network inference methods described above (ARACNE, CLR and MRNET) using a framework based on artificial (simulated) regulatory networks (Fig 1). In simulation, the ground truth is known and inferred networks can be systematically evaluated (which is typically not possible in vivo) [11].

The framework is composed of the following four steps:

1) Produce artificial gene expression data from networks of known structure
2) Compute the MIM using two different mutual information estimators for each produced dataset, namely the empirical and the Spearman based estimator.
3) Infer the network from each computed MIM
4) Assess the quality of the inferred networks using the area under the precision-recall (PR) curve.

4.1 Metrics

The area under the PR curve (AUPRC) is a standard metric used in the network inference community [20]. The PR curve plots precision (pre) against recall (rec) for different values of a threshold determining which pairs qualify as a predicted edge [22]. Precision, pre, is given by

$$pre = \frac{tp}{tp + fp} \tag{9}$$

Precision measures the fraction of real edges (tp) among the ones classified as positive ($tp + fp$). Recall, rec, also called true positive rate (tpr), is given by

$$rec = tpr = \frac{tp}{tp + fn} \tag{10}$$

It denotes the fraction of real edges that are correctly inferred.

In a network inference setting, the PR curve illustrates the trade-off between eliminating many low confidence edges by using a high threshold (low recall, high precision) versus keeping many arcs (high recall) with doubt on their significance (low precision). These quantities depend on the threshold chosen. A good network inference method maximizes both precision and recall.

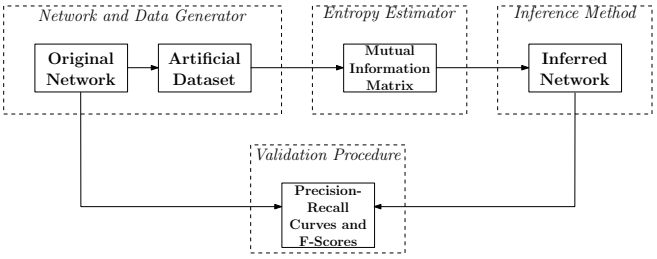

Fig. 1: Experimental framework used to assess network reconstruction quality

4.2 Datasets

We compared the different network inference methods on fifteen datasets (Table 1). These datasets come from two different generators, namely Syntren [27] and GeneNetWeaver (GNW) [12], [11]. Ten out of these fifteen datasets are from the DREAM4 challenge, which was recently used to compare over 30 network inference methods. Briefly, five of the DREAM4 datasets are from the so-called multifactorial challenge. These five datasets have been obtained by applying multifactorial perturbations to the original network. The other five datasets contain time courses capturing the response of the network to a perturbation, followed by gradual return to steady state upon removal of the perturbation. For a detailed description of these datasets, refer to the DREAM project website (http://wiki.c2b2.columbia.edu/dream/).

4.3 Results

We compared MRNETB against three state-of-the-art information theoretic methods for network inference (Tables 2 and 3) using the area under the PR curve (AUPRC) on the benchmark datasets, as described in the previous section. MRNETB has significantly higher AUPRC than both

Dataset	Source	Variables	Samples
syntren1	Syntren - E.Coli	300	100
syntren2	Syntren - E.Coli	300	100
syntren3	Syntren - E.Coli	300	100
syntren4	Syntren - E.Coli	300	100
syntren5	Syntren - E.Coli	300	100
multifact1	Dream4-Multifact1	100	100
multifact2	Dream4-Multifact2	100	100
multifact3	Dream4-Multifact3	100	100
multifact4	Dream4-Multifact4	100	100
multifact5	Dream4-Multifact5	100	100
dream1	Dream4-TS1	100	210
dream2	Dream4-TS2	100	210
dream3	Dream4-TS3	100	210
dream4	Dream4-TS4	100	210
dream5	Dream4-TS5	100	210

Table 1: Benchmark Datasets

MRNET and ARACNE (p-values were computed using a paired Wilcoxon test on the AUPRC, see Table 4), suggesting that MRNETB outperforms these methods. MRNETB outperforms CLR on the DREAM time series datasets, and is outperformed by CLR on the Syntren datasets, suggesting that MRNETB and CLR have at par performance. Note that the results for ARACNE are over-pessimistic because the method typically produces networks with high precision and low recall. As a result, for lower thresholds the area under ARACNE's PR-curve is much smaller than for the other methods.

The results are similar for both estimators of mutual information (i.e. empirical and Spearman based estimation). However, network inference based on Spearman's correlation has higher performance than inference based on empirical estimation of mutual information.

It is worth noting that MRNETB and CLR would have ranked third and fourth in the multifactorial DREAM4 challenge.

	CLR	ARACNE	MRNET	MRNETB
syntren1	**0.12**	0.04	0.09	0.11
syntren2	**0.12**	0.02	0.08	0.10
syntren3	**0.12**	0.02	0.08	0.10
syntren4	**0.11**	0.02	0.07	0.09
syntren5	**0.12**	0.02	0.08	0.10
multifact1	0.12	0.12	0.11	**0.13**
multifact2	0.12	0.11	0.11	**0.13**
multifact3	**0.23**	0.19	0.21	**0.23**
multifact4	**0.18**	0.14	0.14	**0.18**
multifact5	0.16	0.15	0.16	**0.17**
dream1	**0.11**	0.05	0.08	0.10
dream2	**0.11**	0.08	0.10	**0.11**
dream3	**0.18**	0.10	0.12	**0.18**
dream4	**0.14**	0.08	0.10	0.13
dream5	0.13	0.09	0.12	**0.14**
avg	0.138	0.082	0.11	0.133

Table 2: AUPRC Scores for the Empirical Estimation of Mutual Information. Best scores are in bold.

	CLR	ARACNE	MRNET	MRNETB
syntren1	**0.13**	0.04	0.10	0.12
syntren2	**0.13**	0.03	0.10	0.12
syntren3	**0.14**	0.04	0.11	0.12
syntren4	**0.13**	0.04	0.10	0.12
syntren5	**0.14**	0.03	0.09	0.12
multifact1	**0.22**	0.18	0.21	0.21
multifact2	0.21	0.16	0.22	**0.24**
multifact3	0.34	0.35	**0.38**	0.32
multifact4	0.33	0.29	**0.34**	0.31
multifact5	0.32	0.32	**0.36**	0.30
dream1	**0.09**	0.05	0.08	**0.09**
dream2	0.10	0.08	0.11	**0.12**
dream3	0.18	0.11	0.18	**0.21**
dream4	**0.12**	0.07	0.11	**0.12**
dream5	0.14	0.11	**0.17**	0.15
avg	0.181	0.126	0.177	0.178

Table 3: AUPRC Scores for the Squared Spearman Rho Correlation. Best scores are in bold.

MRNETB Losses/Ties/Wins	CLR	ARACNE	MRNET
Total	16/6/8	2/0/28	4/0/26
P-val MRNETB vs	CLR	ARACNE	MRNET
Total	0.11	5e-6	0.01

Table 4: Losses/ties/wins of MRNETB vs each method and p-values obtained with a paired Wilcoxon test.

5. Conclusion

This paper introduces MRNETB, a new information-theoretic method for inferring gene regulatory networks using gene expression data. Similar to other state-of-the-art information-theoretic methods, MRNETB relies on pairwise mutual information. As a result, this method can tackle datasets with large number of variables and low number of samples. An appealing aspect of the new method is the use of a better and more robust search for identifying a maximally informative subset of variables than a classic forward selection, without incurring additional computational cost. We compared MRNETB against three other approaches on fifteen datasets and the experimental results show that the proposed technique is competitive. Further research will focus on using MRNETB within an integrative framework to assess the relative information contribution of different data sources for improving network reconstruction.

References

[1] A. Ambesi-Impiombato and D. di Bernardo. Computational biology and drug discovery: from single-target to network drugs. *Current Bioinformatics*, 2006.

[2] A. Billionnet and F. Calmels. Linear programming for the 0-1 quadratic knapsack problem. *European Journal of Operational Research*, 92:310–325, 1996.

[3] A. J. Butte, P. Tamayo, D. Slonim, T.R. Golub, and I.S. Kohane. Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. *Proceedings of the National Academy of Sciences*, 97(22):12182–12186, 2000.

[4] A. J. Butte and I. S. Kohane. Mutual information relevance networks: Functional genomic clustering using pairwise entropy measurments. *Pacific Symposium on Biocomputing*, 5:415–426, 2000.

[5] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley, New York, 1990.

[6] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research* **3** (2003) 1157–1182

[7] J.J. Faith, B. Hayete, J.T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J.J. Collins, and T.S. Gardner. Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biology*, 5, 2007.

[8] T. S. Gardner and J. Faith. Reverse-engineering transcription control networks. *Physics of Life Reviews 2*, 2005.

[9] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall International, 1999.

[10] F. M. Lopes, D. C. Martins, and R. M. Cesar. Comparative study of grns inference methods based on feature selection by mutual information. In *IEEE International Workshop on Genomic Signal Processing and Statistics*, 2009.

[11] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky. Revealing strengths and weaknesses of methods for gene network inference. *PNAS*, 107(14):6286-6291, 2010.

[12] D. Marbach, T. Schaffter, C. Mattiussi, and D. Floreano. Generating realistic in silico gene networks for performance assessment of reverse engineering methods. *Journal of Computational Biology*, 16(2):229-239, 2009.

[13] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Dalla Favera, and A. Califano. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7, 2006.

[14] P. Merz and B. Freisleben. Greedy and local search heuristics for unconstrained binary quadratic programming. *Journal of Heuristics*, 8(2):1381–1231, 2002.

[15] P. E. Meyer, K. Kontos, F. Lafitte, and G. Bontempi. Information-theoretic inference of large transcriptional regulatory networks. *EURASIP Journal on Bioinformatics and Systems Biology*, Special Issue on Information-Theoretic Methods for Bioinformatics, 2007.

[16] P. E. Meyer, F. Lafitte, and G. Bontempi. Minet: An open source r/bioconductor package for mutual information based network inference. *BMC Bioinformatics*, 2008.

[17] C. Olsen, P. E. Meyer, and G. Bontempi. On the impact of entropy estimation on transcriptional regulatory network inference based on mutual information. *EURASIP Journal on Bioinformatics and Systems Biology*, 2009.

[18] L. Paninski. Estimation of entropy and mutual information. *Neural Computation*, 15(6):1191–1253, 2003.

[19] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.

[20] R. J. Prill, D. Marbach, J. Saez-Rodriguez, P. K. Sorger, L. G. Alexopoulos, X. Xue, N. D. Clarke, G. Altan-Bonnet, and G. Stolovitzky. Towards a rigorous assessment of systems biology models: the DREAM3 challenges. *PLoS ONE*, 5(2):e9202, 2010.

[21] T. Shimamura, S. Imoto, R. Yamaguchi, A. Fujita, M. Nagasaki, and S. Miyano. Recursive regularization for inferring gene networks from time-course gene expression profiles. *BMC Systems Biology*, 3, 2009.

[22] M. Sokolova, N. Japkowicz, and S. Szpakowicz. Beyond accuracy, f-score and roc: a family of discriminant measures for performance evaluation. In *Proceedings of the AAAI'06 workshop on Evaluation Methods for Machine Learning*, 2006.

[23] G. D. Tourassi, E. D. Frederick, M. K. Markey, and Jr. C. E. Floyd. Application of the mutual information criterion for feature selection in computer-aided diagnosis. *Medical Physics*, 28(12):2394–2402, 2001.

[24] M. J. van de Vijver, Y. D. He, L. J. van't Veer, H. Dai, A. A. Hart, D. W. Voskuil, G. J. Schreiber, J. L. Peterse, C. Roberts, M. J. Marton, M. Parrish, D. Atsma, A. Witteveen, A. Glas, L. Delahaye, T. van der Velde T, H. Bartelink H, S. Rodenhuis, E. T. Rutgers, S. H. Friend, and R. Bernards. A gene-expression signature as a predictor of survival in breast cancer. *New England Journal of Medecine*, 347, 2002.

[25] E. P. van Someren, L. F. A. Wessels, E. Backer, and M. J. T. Reinders. Genetic network modeling. *Pharmacogenomics*, 3(4):507–525, 2002.

[26] L. J. van 't Veer, H. Dai, M. J. van de Vijver, Y. D. He, A. A. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, and S. H. Friend. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 406, 2002.

[27] T. Van den Bulcke, K. Van Leemput, B. Naudts, P. van Remortel, H. Ma, A. Verschoren, B. De Moor, and K. Marchal. Syntren: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC Bioinformatics*, 7(1):43, 2006.

[28] Y. Yang and G. I. Webb. Discretization for naive-bayes learning: managing discretization bias and variance. Technical Report 2003/131 School of Computer Science and Software Engineering, Monash University, 2003.

[29] N. V. Zhukov and S. A. Tjulandin. Targeted therapy in the treatment of solid tumors: Practice contradicts theory. *Biochemistry Moscow*, 2008.

Short Specialist Review

Gene finding using multiple related species: a classification approach

Manolis Kellis
MIT Broad Institute for Biomedical Research, Cambridge, MA, USA

1. Introduction

Ideally, we should be able to systematically discover all the functional genes in a newly sequenced genome from its sequence alone. Computational discovery methods rely both on the direct signals used by the cell to guide transcription, splicing, and translation, and also on indirect signals such as evolutionary conservation. In this paper, we summarize the principles of a classification-based approach for systematic gene identification, on the basis of comparative sequence information from multiple, closely related species. We first frame gene identification as a classification problem of distinguishing real genes from spurious gene predictions. We then present the Reading Frame Conservation (RFC) test, a new computational method implementing such a classification approach, on the basis of the patterns of nucleotide changes in the alignment of orthologous regions. We finally summarize our results of applying this method to reannotate the yeast genome, and the challenges of using related methods to discover all functional genes in the human genome.

2. Defining real genes

What is a real gene? This question is relatively easy to answer for those genes that are abundantly expressed, encode well-characterized proteins, and whose disruption affects a specific function in the organism. Beyond these, the distinction is much more subtle between functional genes and spurious gene predictions.

Experimentally, the definition of a functional gene comes largely as accumulating evidence of its usage, including gene expression (Velculescu, 1997), protein fragments (Rezaul *et al*., 2005), biochemical function (Jackman *et al*., 2003), protein interactions (Bai and Elledge, 1997), or the effect of its disruption (McAlister and Holland, 1982). It is important to note, however, that absence of experimental evidence does not imply that a gene annotation is spurious: a real

gene may be missing experimental evidence because it is not used in the particular conditions surveyed. Conversely, any individual report of gene usage could be due to experimental noise, cross-hybridization, or chance transcription events due to a basal level of intergenic transcription. Thus, experimental evidence alone is insufficient to distinguish between real genes and spurious gene predictions.

Computationally, the processes of transcription, splicing, and translation can be thought of as a series of decisions taken by the cell, based on signals in the genome. These signals include distant enhancer elements, regulatory elements surrounding the transcription start site, splicing enhancer and repressor signals in the message, and translation signals in the sequence or structure of mature mRNAs (Fairbrother *et al*., 2002; Wang, 2004; Thanaraj and Robinson, 2000; Yeo and Burge, 2004). The subset of these signals that we currently understand is insufficient to specify the set of known genes. Thus, in addition to the *direct* signals used by the cell, gene identification methods (Burge and Karlin, 1997; Majoros *et al*., 2004; Kulp *et al*., 1996; Krogh, 1997; Henderson *et al*., 1997; Stanke and Waack, 2003; Salzberg *et al*., 1998) routinely identify genes using additional *indirect* signals (reviewed in Fickett and Tung, 1992), which are not available to the cell, although they are generally good indicators of protein-coding genes. These include the frequency of each codon in protein-coding regions, the overall length of the translated protein product, and importantly, the evolutionary conservation of protein sequences across related species.

Evolutionary conservation is perhaps the strongest indicator that a predicted gene is functional. A gene that confers even a slight evolutionary advantage can be conserved across millions of years, regardless of the rarity of its usage. Hence, even if experimental methods fail to detect the usage of a gene in a given set of experimental conditions, evolutionary methods are able to detect *indirect* evidence of its usage, by observing the pressure to preserve its function over millions of years.

3. Gene finding as a classification problem

The challenge of using evolutionary conservation to define genes lies in the ability to reject spurious genes. Although it is generally well understood that genes conserved across large evolutionary distances are functional, lack of conservation is generally attributed to evolutionary divergence, thus providing no evidence toward either accepting or rejecting a gene prediction.

Yet, the comparison of related genomes contains information much richer than simply the presence or absence of a protein in a genome. By working with closely related genomes, we can define regions of conserved gene order, or *synteny blocks*, which span several genes, and potentially entire chromosomes. Defining conserved synteny blocks allows us to construct global alignments, spanning both well-conserved regions and regions of low conservation. In particular, these give us access to full nucleotide alignments for all predicted genes and for all intergenic regions within orthologous segments.

With the availability of orthologous alignments for both protein-coding and non-coding regions, we can study their distinct properties and build a classifier between the two types of region. The simplest and most commonly used such classifier

observes the overall level of nucleotide conservation in the alignment, and selects regions of high conservation that are likely to be functional. Other classifiers may observe more subtle signals, such as the number of amino acid changes per nucleotide substitution (Hurst, 2002), the frequency of insertions and deletions, the periodicity of mutations, and so on. By working with properties of the nucleotide alignment, rather than protein alignment, we are able to apply it uniformly to evaluate both genes and intergenic regions in the same test.

One particular conservation property unique to protein-coding segments is the pressure to preserve the reading frame of translation. Since protein sequences are translated every three nucleotides, the length of insertions and deletions (indels) is largely constrained to remain a multiple of three, thus preserving the frame of translation. Within coding regions, indels that disrupt the frame of translation are excluded, or compensated with nearby indels that restore the reading frame. In noncoding regions, the length of indels does not have this constraint, and short spacing changes are tolerated.

To evaluate this property quantitatively, we developed the RFC test. This test evaluates the pressure to preserve the reading frame in a fully aligned interval, by measuring the portion of nucleotides in this interval for which the reading frame has been locally conserved (Kellis *et al*., 2004). The RFC test provides a classifier between coding and noncoding regions that is completely independent of the traditional signals used in gene identification. It does not rely on start, stop, or splicing signals, nor does it rely on the conservation of protein sequence. It can, therefore, be combined with existing gene-finding tools and provide a highly informative score for any interval considered.

4. Reannotation of yeast

A classification approach is particularly well suited for the yeast genome. The general scarcity of introns makes it hard to rely on splicing signals to discover genes. Thus, the annotation of *Saccharomyces cerevisiae* has traditionally relied solely on the length of predicted proteins to annotate genes, resulting in 6062 annotated open reading frames (ORFs), which potentially encode proteins of at least 100 amino acids (aa). Additionally, a tentative functional annotation has been inferred for as many as 3966 of these ORFs, on the basis of classical genetic experiments and systematic genome-wide studies of gene expression, deletion phenotype, and protein–protein interaction. Together, the interval-based annotation and the large set of well-known genes make it possible to apply the RFC test systematically to evaluate the functional significance of the remaining ORFs.

To apply the test, we constructed multiple sequence alignments for every ORF and every intergenic region across four closely related species. We sequenced and assembled the genomes of *S. paradoxus*, *S. mikatae*, *and S. bayanus* (Kellis *et al*., 2003), and defined genome-wide synteny blocks with *S. cerevisiae*, on the basis of discrete anchors provided by unique protein blast hits (Kellis *et al*., 2004). Within these synteny blocks, we constructed global alignments of orthologous genes and intergenic regions across the four species using CLUSTALW (Higgins and Sharp, 1988), and systematically evaluated each alignment. We compared *S. cerevisiae* to

each species in turn, and every comparison cast a vote based on its overall RFC and a species-specific cutoff. A decision was then reached for each gene by tallying the votes from all comparisons.

We evaluated the sensitivity and specificity of the approach based on the 3966 genes with functional annotations ("known genes"), and 340 randomly chosen intergenic sequences with lengths similar to the annotated ORFs. The RFC test correctly accepted 3951 known genes (99.6%) and rejected only 15 known genes; upon manual inspection, these 15 are indeed likely to be spurious (most lack experimental evidence, and deletion phenotypes of the rest are likely due to their overlap with the promoter of other known genes). The method also correctly rejected 326 intergenic regions (96%), accepting only 14 intergenic regions (of which 10 appear to define short ORFs or extend annotated ORFs, suggesting that at most 1% of true intergenic regions failed to be rejected by the RFC test). In summary, the RFC test shows a very strong discrimination between genes and intergenic regions, with sensitivity and specificity values greater than 99%.

We then applied the RFC test to all previously annotated genes, leading to a major revisiting of the yeast gene set. For ORFs with lengths greater than 100 aa, our analysis accepted 5538 ORFs and rejected 503 (of which 376 were immediately rejected, 105 were rejected with additional criteria, and 32 were merged with neighboring ORFs); the classifier abstained from making a decision in 20 cases. The rejected ORFs show an abundance of frame-shifting indels across their entire length, in-frame stop codons, and low conservation of protein sequence, in addition to the low RFC score; their length distribution and atypical codon usage additionally suggest that they are likely occurring by chance (Goffeau, 1996; Dujon, 1994; Sharp and Li, 1987); furthermore, previous systematic experimentation showed no compelling evidence that these may encode a functional gene. Thus, it appears that more than 500 previously annotated ORFs in the yeast genome are spurious predictions.

Below the 100 amino acid cutoff, no previous systematic annotation or experimentation was available. Thus, to validate our method, we compared the results of the RFC test with an independent metric: the proportion of the *S. cerevisiae* ORF that was also free of stop codons in the other three species. Between 50 and 99 aa, we found that the method is still reliable and reports 43 candidate new genes at that length. As ORFs smaller than 50 aa were tested, we found that the specificity of the test decreased, since small intervals tend to be devoid of indels by chance rather than by presence of selective pressure. Thus, additional constraints, and additional species, will be needed to discover genes reliably at such short lengths.

In addition to the discovery of genes themselves, the comparative analysis allowed us to refine gene boundaries. Once an interval was determined to be under selective pressure for RFC, the boundaries of that interval were adjusted on the basis of the conservation of start/stop/splice signals and the boundaries over which the reading frame is conserved. This led to a large-scale reannotation of gene structure, which affected hundreds of genes (146 start codon changes, 67 intron changes, 32 merges of consecutive ORFs, and 45 changes of ORF ends). It is worth noting that in 134 cases, the inferred boundary changes pinpointed sequencing errors in the primary sequence of *S. cerevisiae*, ~50 of which were tested and

corrected by resequencing. These boundary changes reveal the true location of promoter regions, new protein domains in elongated ORFs, and previously overlooked functional relationships in the case of merged ORFs

In summary, the RFC test was able to reliably distinguish between genes and intergenic regions, leading to a systematic reannotation of the yeast genome. The comparative analysis led to the rejection of more than 500 previously annotated genes, and to the discovery of many novel genes. The results agree with similar comparative analyses carried out from a number of yeast species (Cliften *et al*., 2003; Blandin, 2000; Wood *et al*., 2001; Brachat, 2003). In addition, it allowed us to refine the gene structure of hundreds of genes, adjusting start and stop boundaries, merging consecutive ORFs, and discovering many new introns. Moreover, by using multiple species, the signals leading to gene identification were powerful enough to pinpoint sequencing errors in any individual species, including *S. cerevisiae*.

5. Implications for the human genome

The challenge of gene finding in the human genome is far greater than in yeast, due to the vastly larger intergenic regions, numerous and large introns, small exons, and alternatively spliced genes. Most approaches to gene finding in higher eukaryotes have relied on Hidden Markov Models (Burge and Karlin, 1997; Majoros *et al*., 2004; Kulp *et al*., 1996; Krogh, 1997; Henderson *et al*., 1997; Stanke and Waack, 2003), which inherently emphasize the importance of exon chaining and rely on knowledge of the expected length distributions for both exons and introns. These have been recently extended to use sequences from multiple species (Yeh *et al*., 2001; Siepel and Haussler, 2004a; Siepel and Haussler, 2004b; Batzoglou *et al*., 2000; Korf *et al*., 2001; Dewey *et al*., 2004; Rinner and Morgenstern, 2002; Parra, 2003; Meyer and Durbin, 2002; Novichkov *et al*., 2001). These approaches have limitations however, in cases of alternative splicing (Brett, 2000; Mironov *et al*., 1999; Cawley and Pachter, 2003), differences in splicing between species (Nurtdinov *et al*., 2003), widely varying exon and intron lengths, and noncanonical splice sites.

An exon-based classification approach to gene finding can help overcome these limitations. By exhaustively enumerating and testing all candidate protein-coding intervals, classification approaches make exon detection independent of chaining. Splice site models can be applied to each species independently and then compared with each other, which is much stronger than the evidence in any one species alone. The intervals can be then tested on the basis of discriminating variables that distinguish genes from noncoding regions (Fickett and Tung, 1992; Goldman and Yang, 1994). New discriminating variables can be defined as alignments of each type of region are systematically compared, and these can be combined and weighted, leveraging traditional machine-learning techniques for feature selection and classification (Moore and Lake, 2003). Once relevant intervals have been identified, their boundaries can be adjusted for optimal chaining into complete genes. In particular, chaining can also leverage an inferred frame of translation for each exon, based on the higher mutation rate observed in largely degenerate third codon positions (Hurst, 2002). The exon-chaining step produces full gene models,

and is able to cope with alternative splicing and missing exons, since it is not constrained to a single optimal path through the exons.

By systematically observing alignment properties of large sequence regions, we can build new rigorous approaches for sequence analysis. These are widely applicable beyond coding exons, and similar classification-based approaches can be used to distinguish CpG islands, 5′- and 3′-untranslated regions, promoter regions, regulatory islands, and other functional elements. Through the lens of evolutionary selection, our ability to directly interpret genomes is revolutionized. Coupled with systematic experimentation and validation, these analyses can lead to a systematic catalog of functional elements in the human genome, forming the future foundations of biomedical research.

References

Bai C and Elledge SJ (1997) Gene identification using the yeast two-hybrid system. *Methods in Enzymology*, **283**, 141–156.

Batzoglou S, Pachter L, Mesirov JP, Berger B and Lander ES (2000) Human and mouse gene structure: Comparative analysis and application to exon prediction. *Genome Research*, **10**, 950–958.

Blandin G, Durrens P, Tekaia F, Aigle M, Bolotin-Fukuhara M, Bon E, Casaregola S, de Montigny J, Gaillardin C, Lepingle A, *et al*. (2000) Genomic exploration of the hemiascomycetous yeasts: 4. The genome of *Saccharomyces cerevisiae* revisited. *FEBS Letters*, **487**, 31–36.

Brachat S, Dietrich FS, Voegeli S, Zhang Z, Stuart L, Lerch A, Gates K, Gaffney T and Philippsen P (2003) Reinvestigation of the *Saccharomyces cerevisiae* genome annotation by comparison to the genome of a related fungus: *Ashbya gossypii*. *Genome Biology*, **4**, R45.

Brett D, Hanke J, Lehmann G, Haase S, Delbruck S, Krueger S, Reich J and Bork P (2000) EST comparison indicates 38% of human mRNAs contain possible alternative splice forms. *FEBS Letters*, **474**, 83–86.

Burge C and Karlin S (1997) Prediction of complete gene structures in human genomic DNA. *Journal of Molecular Biology*, **268**, 78–94.

Cawley SL and Pachter L (2003) HMM sampling and applications to gene finding and alternative splicing. *Bioinformatics*, **19**(Suppl 2), II36–II41.

Cliften P, Sudarsanam P, Desikan A, Fulton L, Fulton B, Majors J, Waterston R, Cohen BA and Johnston M (2003) Finding functional features in *Saccharomyces* genomes by phylogenetic footprinting. *Science*, **301**, 71–76.

Dewey C, Wu JQ, Cawley S, Alexandersson M, Gibbs R and Pachter L (2004) Accurate identification of novel human genes through simultaneous gene prediction in human, mouse, and rat. *Genome Research*, **14**, 661–664.

Dujon B, Alexandraki D, Andre B, Ansorge W, Baladron V, Ballesta JP, Banrevi A, Bolle PA, Bolotin-Fukuhara M and Bossier P (1994) Complete DNA sequence of yeast chromosome XI. *Nature*, **369**, 371–378.

Fairbrother WG, Yeh RF, Sharp PA and Burge CB (2002) Predictive identification of exonic splicing enhancers in human genes. *Science*, **297**, 1007–1013.

Fickett JW and Tung CS (1992) Assessment of protein coding measures. *Nucleic Acids Research*, **20**, 6441–6450.

Goffeau A, Barrell BG, Bussey H, Davis RW, Dujon B, Feldmann H, Galibert F, Hoheisel JD, Jacq C, Johnston M, *et al*. (1996) Life with 6000 genes. *Science*, **274**, 546, 563–567.

Goldman N and Yang Z (1994) A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Molecular Biology and Evolution*, **11**, 725–736.

Henderson J, Salzberg S and Fasman KH (1997) Finding genes in DNA with a hidden Markov model. *Journal of Computational Biology*, **4**, 127–141.

Higgins DG and Sharp PM (1988) CLUSTAL: A package for performing multiple sequence alignment on a microcomputer. *Gene*, **73**, 237–244.

Hurst LD (2002) The Ka/Ks ratio: Diagnosing the form of sequence evolution. *Trends in Genetics*, **18**, 486.

Jackman JE, Montange RK, Malik HS and Phizicky EM (2003) Identification of the yeast gene encoding the tRNA m1G methyltransferase responsible for modification at position 9. *RNA*, **9**, 574–585.

Kellis M, Patterson N, Birren B, Berger B and Lander ES (2004) Methods in comparative genomics: Genome correspondence, gene identification and regulatory motif discovery. *Journal of Computational Biology*, **11**, 319–355.

Kellis M, Patterson N, Endrizzi M, Birren B and Lander ES (2003) Sequencing and comparison of yeast species to identify genes and regulatory elements. *Nature*, **423**, 241–254.

Korf I, Flicek P, Duan D and Brent MR (2001) Integrating genomic homology into gene structure prediction. *Bioinformatics*, **17**(Suppl 1), S140–S148.

Krogh A (1997) Two methods for improving performance of an HMM and their application for gene finding. *Proceedings/International Conference on Intelligent Systems for Molecular Biology*, **5**, 179–186.

Kulp D, Haussler D, Reese MG and Eeckman FH (1996) A generalized hidden Markov model for the recognition of human genes in DNA. *Proceedings/International Conference on Intelligent Systems for Molecular Biology*, **4**, 134–142.

Majoros WH, Pertea M and Salzberg SL (2004) TigrScan and GlimmerHMM: Two open source ab initio eukaryotic gene-finders. *Bioinformatics*, **20**, 2878–2879.

McAlister L and Holland MJ (1982) Targeted deletion of a yeast enolase structural gene. Identification and isolation of yeast enolase isozymes. *The Journal of Biological Chemistry*, **257**, 7181–7188.

Meyer IM and Durbin R (2002) Comparative *ab initio* prediction of gene structures using pair HMMs. *Bioinformatics*, **18**, 1309–1318.

Mironov AA, Fickett JW and Gelfand MS (1999) Frequent alternative splicing of human genes. *Genome Research*, **9**, 1288–1293.

Moore JE and Lake JA (2003) Gene structure prediction in syntenic DNA segments. *Nucleic Acids Research*, **31**, 7271–7279.

Novichkov PS, Gelfand MS and Mironov AA (2001) Gene recognition in eukaryotic DNA by comparison of genomic sequences. *Bioinformatics*, **17**, 1011–1018.

Nurtdinov RN, Artamonova II, Mironov AA and Gelfand MS (2003) Low conservation of alternative splicing patterns in the human and mouse genomes. *Human Molecular Genetics*, **12**, 1313–1320.

Parra G, Agarwal P, Abril JF, Wiehe T, Fickett JW and Guigo R (2003) Comparative gene prediction in human and mouse. *Genome Research*, **13**, 108–117.

Rezaul K, Wu L, Mayya V, Hwang SI and Han DK (2005) A systematic characterization of mitochondrial proteome from human T leukemia cells. *Molecular and Cellular Proteomics*, **4**, 169–181.

Rinner O and Morgenstern B (2002) AGenDA: Gene prediction by comparative sequence analysis. *In Silico Biology*, **2**, 195–205.

Salzberg S, Delcher AL, Fasman KH and Henderson J (1998) A decision tree system for finding genes in DNA. *Journal of Computational Biology*, **5**, 667–680.

Sharp PM and Li WH (1987) The codon Adaptation Index–a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic Acids Research*, **15**, 1281–1295.

Siepel A and Haussler D (2004a) Combining phylogenetic and hidden Markov models in biosequence analysis. *Journal of Computational Biology*, **11**, 413–428.

Siepel A and Haussler D (2004b) Phylogenetic estimation of context-dependent substitution rates by maximum likelihood. *Molecular Biology and Evolution*, **21**, 468–488.

Stanke M and Waack S (2003) Gene prediction with a hidden Markov model and a new intron submodel. *Bioinformatics*, **19**(Suppl 2), II215–II225.

Thanaraj TA and Robinson AJ (2000) Prediction of exact boundaries of exons. *Briefings in Bioinformatic*, **1**, 343–356.

Velculescu VE, Zhang L, Zhou W, Vogelstein J, Basrai MA, Bassett DE Jr, Hieter P, Vogelstein B and Kinzler KW (1997) Characterization of the yeast transcriptome. *Cell*, **88**, 243–251.

Wang Z, Rolish ME, Yeo G, Tung V, Mawson M and Burge CB (2004) Systematic identification and analysis of exonic splicing silencers. *Cell*, **119**, 831–845.

Wood V, Rutherford KM, Ivens A, Rajandream M-A and Barrell B (2001) A re-annotation of the *Saccaromyces cerevisiae* genome. *Comparative and Functional Genomics*, **2**, 143–154.

Yeh RF, Lim LP and Burge CB (2001) Computational inference of homologous gene structures in the human genome. *Genome Research*, **11**, 803–816.

Yeo G and Burge CB (2004) Maximum entropy modeling of short sequence motifs with applications to RNA splicing signals. *Journal of Computational Biology*, **11**, 377–394.

A Phylogenomic Approach to the Evolutionary Dynamics of Gene Duplication in Birds

Chris L. Organ[1*], Matt Rasmussen[2*], Maude W. Baldwin[1], Manolis Kellis[2], Scott V. Edwards[1]

[1]*Museum of Comparative Zoology, Harvard University, 26 Oxford Street, Cambridge, MA 02138.* [2]*Massachusetts Institute of Technology, Center for Genome Research, 320 Charles Street, Cambridge, MA 02141*

*These authors contributed equally

Corresponding author contact information:
Chris Organ
Museum of Comparative Zoology
Department of Organismic and Evolutionary Biology
26 Oxford Street
Harvard University
Cambridge, MA 02138
Tel (617) 496-9389
Fax (617) 495-5667

Keywords: gene duplication, phylogenomics, comparative genomics, birds

Running title: The Evolutionary Dynamics of Gene Duplication in Birds

Abstract: Gene duplication is a fundamental aspect of genome evolution that produces large and small gene families of (usually) related function. We perform a phylogenomic analysis of gene duplication in the chicken (*Gallus gallus*) to characterize the dynamics and evolution of gene duplication on the evolutionary line to birds. In *Gallus*, the distribution of the number of paralogs per gene family is heavily skewed towards small families. This finding is in accord with other studies that find gene family size typically follows a power-law distribution in animals, a pattern thought to be produced by differential rates of pseudogenization among families. We also test for within-family evolutionary rate variation in *Gallus*, finding that the vast majority of gene families exhibit substantial rate variation among lineages. This rate variation probably stems from two sources: natural deviations in the clock as commonly found, for example, in phylogenetic analyses of different species; and bursts of adaptive evolution among newly evolved gene family members. The age of gene duplications in *Gallus* are distributed exponentially, with most duplications occurring recently, a pattern consistent with analyses on other eukaryotes. Taken together, these results begin to reveal the dynamics of gene family evolution in birds, the most speciose group of living amniotes, though whole genome data are required from more bird and reptile species to fully understand patterns of gene gain and loss in this group.

Introduction

New genes are thought to primarily arise through a process of gene duplication. Genes that are homologous as a result of divergence across lineages via speciation are said to be orthologous, whereas genes that are homologous as a result of gene duplication are paralogous (Li 2006). Paralogous genes that are functionally redundant and selectively nearly neutral can result in one copy being mutated into a functionless sequence called a pseudogene, or they can be deleted altogether. On the other hand, some duplicated genes can be beneficial from their time of origin because of dosage effects (Kondrashov, Rogozin et al. 2002) and may ultimately be important for speciation. In Passeriformes (perching birds) this may be the case for growth hormone (GH) paralogs, which have undergone differential selection since their divergence (Yuri, Kimball et al. 2008). In a process called subfunctionalization, each paralog adopts partial function of their ancestral gene (Nowak, Boerlijst et al. 1997; Lynch and Force 2000). Changes in gene expression immediately following duplication as a result of subfunctionalization appear to be common (Gu, Nicolae et al. 2002). Duplicated genes can also diverge to produce novel functions in a process known as neofunctionalization (Zhang 2003). For example, some duplicated members of the *RNaseA* gene superfamily in primates evolved a novel antibacterial function that was not present in the common ancestral gene or its descendants (Zhang, Rosenburg et al. 1998). Through the acquisition of novel functions, gene duplication plays a vital role in generating diversity at both molecular (Eirin-Lopez, Gonzalez-Tizon et al. 2004) and organismal (Hittinger and Carroll 2007) levels and can increase the phenotypic complexity and diversity of animals (Ohno 1970; Maniatis and Tasic 2002). Protein coding genes, regulatory genes, and RNA non-coding genes are all subject to gene duplication. Gene duplication occurs in all three major domains of life; Bacteria, Archaea, and Eukarya (Zhang 2003), and comparative analyses have suggested an average origin rate of new gene duplicates on the order of 1 per gene per 100 million years (Lynch and Conery 2000).

The ubiquity of gene duplication and its power to generate material on which selection may act is an especially interesting topic in birds because of their uniquely structured genomes. Birds have smaller genomes than any other amniote group (Gregory 2002), but they also have a substantially reduced density of active repetitive elements (Shedlock 2006), and segmental duplications and pseudogenes (Hillier, Miller et al. 2004). Birds may also have fewer protein-coding genes in their genomes than mammals, with roughly 18,000 in chicken compared with approximately 22,000 in human (Hillier, Miller et al. 2004). By comparing the genomes of chicken with those of human and puffer fish (*Fugu*), the difference in gene count can be explained by significantly less lineage-specific gene duplication along the lineage of birds. For example, gene duplication by retroposition appears to be rare in birds compared with mammals, which have roughly 300 times the number of retrotransposed gene duplicates than in chicken (over 15,000 compared with 51) (Hillier, Miller et al. 2004). Of the 51 duplicates detected in the chicken genome by Hillier, Miller et al. (2004), 36 appear to be pseudogenes. This is probably due in part to the abundance of long interspersed nuclear elements (LINEs) in mammal genomes, which are likely responsible for the reverse transcription of retrotransposed gene duplicates. Chicken repeat 1 (CR1) is the dominate active transposable element in chicken and other archosaurs (Shedlock et al. 2007), a feature that may account for the lack of retrotransposed gene duplicates due to this element's inability to copy polyadenylated mRNA (Haasa, Grabowskia et al. 2001; Hillier, Miller et al. 2004).

The whole-genome sequencing of the chicken (*Gallus gallus*) provided an unprecedented window into the architecture of bird genomes (Hillier, Miller et al. 2004). This study revealed a number of important details of gene duplication. For example, most of the expansions in gene families within chicken are associated with the immune system and host defense against parasites (Ota and Nei 1994; Nei et al. 1997). The chicken genome project found an expansion of the Scavenger Receptor Cysteine-Rich (SRCR) domain (Hillier, Miller et al. 2004), a highly conserved protein module involved in the innate immune system (Sarrias, Grønlund et al. 2004). Linked with the major histocompatibility complex (MHC) class I gene cluster in humans are certain olfactory receptors which also underwent a lineage-specific expansion within the chicken. These olfactory receptor genes, related to two orthologs in human (*OR5U1* and *OR5BF1*), appear to have expanded within birds relatively recently to constitute the majority of the over 200 olfactory receptor genes in the chicken genome. The expansions of these genes may be part of the genetic mechanism linking the immune and olfactory systems with mate choice, kin recognition, and social interactions in birds (Zelano and Edwards 2002). Other gene expansions were also likely important for key innovations in birds, such as expansions in the keratin gene family, which are the proteinaceous building blocks of feathers and therefore vital for thermoregulation, sexual display, and flight. However, even in the absence of selection (most gene duplicates are thought to be pseudogenized during evolution) they may still play a critical role in creating postmating reproductive barriers that aid in speciation (Lynch and Conery 2000).

Age of gene duplications within lineages is also of fundamental interest in the study of gene family evolution. Analysis of animal, plant and fungal genomes has shown that the majority of duplications are recent, because most duplicated genes are thought to be pseudogenized shortly after the duplication event (Lynch and Conery 2000). It is currently unknown if the small, streamlined genomes of birds deviate from this pattern, given the fewer number of total genes, paucity of transposable elements, and highly recombinant microchromosomes found in chicken.

Here we explore the evolution of chicken gene families within the larger context of amniote evolution. A specific analytical strategy called phylogenomics (Eisen 1998) attempts to systematically reconstruct the phylogeny of gene families across multiple complete genomes so as to deduce putative protein homologies and functions as well as the specific gene duplications and losses responsible for gene family diversity. The phylogenomic approach to multigene family evolution has shown recent success in many different clades, such as vertebrates (Zmasek and Eddy 2002; Storm and Sonnhammer 2003; Li, Coghlan et al. 2006; Huerta-Cepas, Dopazo et al. 2007), and 16 fungi species (Wapinski, Pfeffer et al. 2007). Recently Rasmussen and Kellis (2007) deduced a pattern of substitution rates within and among *Drosophila* and yeast species by using machine-learning to compare the gene trees in multigene families to the presumed species tree. In addition, Heger and Ponting (2007) developed a comprehensive database of orthologous and paralogous gene sets for several clades, including amniotes. Here we follow a complementary approach to characterize the dynamics and evolution of gene duplication within birds. Whereas many previous approaches to recognizing paralogs use reciprocal BLAST hits and pairwise comparison of orthologs, we apply Ensembl phylogenies and a variety of phylogenetic and comparative analyses within a clade of two dozen sequenced amniote species to analyze the dynamics of gene duplication in the complete chicken (*Gallus gallus*) genome.

Computational Approach

For our analysis, we obtained 25,363 gene trees from Ensembl's (v50) gene tree database. Ensembl's gene trees were produced with their custom pipeline of computational analysis. Initially, family clusters were defined by single-linkage clustering of best reciprocal BLAST hits between amino acid translations of all genes within the Ensembl database (Hubbard, Aken et al. 2007). Each cluster was then processed by Ensembl to create a multiple amino acid alignment using the alignment algorithm MUSCLE (Edgar 2004) and a maximum likelihood phylogenetic tree using PHYML (Guindon and Gascuel 2003). Our database of gene trees assembled from Ensembl contains 611,441 genes from 39 species, including 25 mammals, *Gallus gallus*, *Xenopus tropicalis*, five fish, six invertebrates, and *Saccharomyces cerevisiae*. There are 17,487 annotated chicken (*Gallus gallus*) genes in the Ensembl database, 13,649 (78%) of which belong to one of the 7,785 gene trees.

To study gene duplication along the bird lineage, we identified bird/mammalian duplications in each gene tree (Figure 1 and Figure 2). This event is represented as a node that is parental to one clade of chicken-specific genes and another clade of mammalian-only genes. Although we describe the chicken paralogs as chicken-specific, we do so only because chicken is the only bird in our analysis; presumably, sampling of additional genomes within Reptilia would reveal many of these genes and gene duplication to be shared with other birds and non-avian reptiles. If duplications occurred prior to the common ancestor of birds and mammals, a single family may contain multiple gene divergences. In addition, some of these gene duplications can be partially obscured when there is complete gene loss in either the bird or mammalian lineage. We were able to identify 12,094 unambiguous amniote duplications (using *Gallus* and *Homo*). For each duplication, we isolated the subtree of amniote genes rooted at the amniote ancestor for further analysis. Outgroup species (fish; *Fugu*) were used to help position the subtree root. We expect this protocol to yield chicken gene families that differ from those circumscribed, for example, solely by reciprocal BLAST hits; sometimes our definition of chicken gene families include genes not detected by non-phylogenetic methods, whereas in other instances our approach will miss some genes that manual inspection and curation would have revealed. On the other hand, our approach has the advantage of being objective and repeatable, and can easily be extended to study the dynamics of gene duplication in other taxa.

Dynamics of Chicken-Specific Gene Duplication

We found that the distribution of the number of paralogs per chicken gene family was heavily skewed towards small families (Figure 3). Whereas nearly 30 gene families had three members, only six families had more than nine. These figures focus only on those gene families that duplicated after the divergence of chicken from the amniote ancestor (by comparing *Gallus* with *Homo*). Thus very ancient gene families – many potentially with large numbers of family members – do not figure into this calculation. Still, we were surprised by the small number of very large (> 20) gene families in the chicken. Gene family size has been found to follow a power-law distribution in animals, a pattern thought to be produced by differential rates of pseudogenization among families (Huynen and van Nimwegen 1998; Hughes and Liberles 2008). Moreover, birth-death models and purifying selection appear to account for much of the conservation seen within lineage-specific paralogs (Ota and Nei 1994; Piontkivska, Rooney et al. 2002; Piontkivska and Nei 2003; Eirin-Lopez, Gonzalez-Tizon et al. 2004).

Substitution rates within amniote lineages are known to be quite variable, and we expect similar rate variation among paralogous members of chicken gene families. We estimated dates of divergence for each node in our chicken multigene family trees by using a penalized likelihood model, as implemented in the r8s rate analysis program (Sanderson 2003), and by using 310 million years as an estimate for the *Gallus*/*Homo* common ancestor (Benton and Donoghue 2007). The penalized likelihood model finds the optimal trade-off between maximizing the likelihood of a Poisson process for nucleotide substitution and a penalty term for rate variation between neighboring branches. The weighting of these two terms is determined by a coefficient λ, such that higher values of λ greatly penalize rate variation in favor of a clock-like model, and lower values allow a large amount of rate variation. Using a cross-validation procedure (Sanderson 2003), we determined the optimal choice of λ from the possible values $10^{-2} - 10^{3}$. Within our amniote gene trees we found that over 54 % of them were best explained by a λ value of 10^{-2}, indicating that the vast majority of gene families exhibit substantial rate variation among lineages (Figure 4). This rate variation probably stems from two sources: natural deviations in the clock as commonly found, for example, in phylogenetic analyses of different species; and bursts of adaptive evolution among newly evolved gene family members. Under the first hypothesis, it might be expected that larger gene families would exhibit more rate variation than small ones, and that the incidence of rate variation would increase with family size. However, we did not find this trend when we regressed λ on gene family size (Figure 4). Many of the gene families best explained by a log λ value of -2 showed levels of divergence that suggests duplication since the Cenozoic era, especially since the Neogene period. For this reason, we suspect that much of the rate variation among gene family members may in fact be due to adaptive bursts, because generation time effects among different lineages of birds are only expected to influence rate variation for those gene families that duplicated prior to the chicken's divergence from other lineages. Of the other categories of substitution rate variation among gene family members, the class best explained by $\lambda = \log(3)$ was the next most common. Substitution rates among gene family members in this category are fairly clock-like.

The age of gene duplications in chicken are distributed exponentially, with most duplications occurring recently (Figure 5). This pattern is consistent with previous analyses (Lynch and Conery 2000) and suggests that, assuming a relatively constant rate of gene duplication, most genes are pseudogenized or eliminated from the genome soon after duplication. This pattern could also suggest that concerted evolution might be more common among chicken gene families than in other groups. For example, concerted evolution among major histocompatibility complex (MHC) gene copies in birds is thought to occur more frequently, and over a shorter time scale, than in mammals (Hess and Edwards 2002), and the phylogenetic scale over which MHC orthologs can be identified in birds is probably much smaller than in mammals.

Examples of families with chicken–specific duplications

Gene family composition is shaped both by gene gain and loss, yet as other researchers have noted (Furlong 2005), gene family expansion is easier to detect, especially when annotation is not complete and gaps still remain in recent genome builds. We examined several gene families containing lineage-specific expansions in the chicken, using the amniotic subtree rooted at the chicken-human divergence. In Table 1, we describe the dynamics of five representative families: Toll-like receptors, hemoglobin, ovalbumin-related serpins, four sub-families of olfactory receptors, and keratin. These families were selected for their variety in size, age and

function, and because the annotation and family membership could be at least partially cross-validated with recent studies.

Toll-like receptors

Temperley and colleagues (2008) describe the evolutionary history and chromosomal location of chicken toll-like receptors (TLRs), a family that is part of the innate immune system and is characterized by an ancient, highly conserved pathogen-recognition domain that triggers an inflammatory response. These authors discovered that while chickens and humans both have ten receptors, only four genes in chicken maintain one-to-one orthology with mammalian genes; much gain and loss has occurred in every lineage. In the chicken, a duplication event estimated at 67 million years ago (mya) gave rise to *TLR2A* and *TLR2B*, orthologs to the single *TLR2* in mammals. Three other genes, two that duplicated in tandem (*TLR1LA* and *TLR1LB*, estimated duplication time 147 mya), as well as *TLR15* have no mammalian counterpart. Other mammalian members have been pseudogenized or fully lost in chicken. Using our automated phylogenetic approach, we are able to analyze one of these chicken-specific expansions: the duplication event that gave rise to *TLR2A* and *TLR2B* in chickens. We obtained nearly the same estimate of duplication time (67 mya). As with majority of these five families that we looked at in detail, there was considerable rate variation, as log λ was -2. The amount of sequence evolution (point mutations) in the Toll-like receptor family is greater than 50.8% of sequence evolution in other chicken gene families. We could not analyze the other duplication in chicken, as our dataset from Ensembl was missing one of the chicken-specific genes (*TLR1LB*). *TLR15*, another gene unique to birds, had no mammalian ortholog, so it also was not included in our amniote subtree.

Ovalbumin-related serpins

Another gene family with documented chicken-specific expansions are the ov-serpins, also called the ovalbumin-related serpins, or clade B serpins. Benarafa and O'Donnell (2005) examine the phylogenetic relationship between the chicken members (some of which function as egg-white storage proteins) and their mammalian counterparts (involved in diverse roles like embryogenesis, inflammation regulation, and angiogenesis). The initial duplication is thought to have occurred very early in the vertebrate lineage; and, like TLRs, the family is also marked by recent lineage-specific expansions and losses. Chickens have 10 members and humans have 13 members. Three genes in chicken-- *ovalbumin*, and ovalbumin-like genes *X* and *Y* -- are paralogs and lack a human ortholog. Another gene, with a single human ortholog, seems to have duplicated to produce the chicken genes *Serpinb10* and *MENT* (Mature Erythrocyte Nuclear Termination state-specific protein). The remaining family members from chicken each have single human orthologs. Among the two subfamilies with chicken-specific expansions, rate variation is substantial (log $\lambda = -2$). Moreover, the amount of sequence evolution (point mutations) in the subfamily containing *ovalbumin* and ovalbumin-like genes *X* and *Y* is greater than 69.6% of sequence evolution in other chicken gene families that have duplicated since the chicken-mammalian split while one of the other ovalbumin subfamilies is similarly divergent (77.6%; *Serpinb10* and *MENT*).

Hemoglobin

Metabolic rate is an important trait that governs many organismal characters, from growth strategies to sustained physical activity. In amniotes, an elevated metabolism

(endothermy) has only evolved within two extant groups, birds and mammals, although paleontologists suspect that many extinct dinosaurian lineages possessed endothermy (de Ricqlès, Padian et al. 2001; Horner, Padian et al. 2001). Whereas the typical mammalian and avian condition is homeothermy (roughly constant body temperature), some birds, such as swifts, hummingbirds, and nightjars are facultatively poikilothermic, a condition in which their usually elevated body temperature can vary over a wider range than that seen in mammals. The hemoglobin multigene family is closely associated with metabolism and the respiratory system.

Hemoglobin, a multi-domain protein, has rapidly diversified with vertebrate lineages (Gribaldo, Casane et al. 2003). For example, α-globin underwent rapid duplication and deletion in mammals (Hoffmann, Opazo et al. 2008). Based on an analysis of the platypus genome, which incorporated information from flanking loci, a recent model (Patel, Cooper et al. 2008) proposes that the β-globin paralogs arose from a single transposition in the amniote ancestor followed by independent duplication in birds and mammals. From our dataset, the β-globin paralogs β^H, ρ and ε appear as a chicken-specific expansion, consistent with this model. Rate diversity is high in this family as well and sequence evolution (point mutations) in β-globin paralogs is similar to that seen in the Toll-like receptors family (greater than 50.8% of sequence evolution in other chicken gene families).

Olfactory Receptors

Olfaction has recently gained much recognition as an important sensory modality for birds (Nevitt, Losekoot et al. 2008; O'Dwyer, Ackerman et al. 2008; Steiger, Fidler et al. 2008); Historically, birds were assumed to communicate primarily via the visual or auditory systems, but behavioral and genomic data suggest that chemosensory perception plays a larger role. The chicken genome paper remarked on the surprisingly large group of chicken-specific olfactory receptors: 218 genes were identified, representing an avian expansion orthologous to the human receptors *OR5U1* and *OR5BF1*. Our bioinformatic approach detected 196 genes belonging to this subfamily, the discrepancy is perhaps in part due to different genome builds. Similar to recent work (Lagerström, Hellström et al. 2006), we also identified other families of olfactory receptors with small expansions in chicken (two to three genes in our analysis). These include a cluster associated with the previously identified *COR1-6* genes (Chicken Olfactory Receptor genes) on chromosome 5, a second cluster on chromosome 10, related to *COR7*, and a third cluster on chromosome 1. Interestingly, the subfamilies have differing amount of sequence evolution ranging from 26.7% (family containing *COR7)* to 88.4% (family containing *COR1-6).* The latter gene family also had a more clock-like rate than other families, indicating that it is among the oldest gene duplications (238 mya) in chicken and, like the ovalbumin clade B serpins (MENT, serpinb10) and the large olfactory receptor family, arose in non-avian reptilian ancestors (see Table 1 and Figure 5).

Keratin

The evolution of feathers in theropod dinosaurs was a major innovation that likely provided insulation for metabolically active animals, ornamentation for display, and in one lineage transformed arms into wings (Ji, Currie et al. 1998; Zhang and Zhou 2000; Currie and Chen 2001; Norell, Ji et al. 2002; Sawyer and Knapp 2003). β-keratins are the basic structural elements of feathers and therefore a gene family vital for the success of birds. In the publication of the first genome draft, the International Chicken Genome Sequencing Consortium (2004)

noted the large expansion of the avian-specific keratin gene family, estimated at around 150 members. This avian keratin family, which encodes proteins forming feathers and scales (Sawyer, Glenn et al. 2000), is functionally and evolutionarily distinct from the mammalian hair-specific alpha- keratin, which, like another main component of hair, the keratin-associated proteins (KRTAP) have no members in the chicken genome (Wu, Irwin et al. 2008). Within non-avian reptiles, β-keratins likely duplicated by retrotransposition, resulting in the loss of introns in some paralogs; in birds all paralogs have lost introns. Unequal crossover is also thought to expand and contract keratin gene arrays in birds, resulting in a tandem organization of multiple paralogs (Toni, Dalla Valle et al. 2007). We find that the amount of sequence evolution in keratin is very high (98% larger than other chicken gene families), likely owing to the absence of other reptilian comparisons, but also to the adaptive significance of these proteins within birds.

Prospectus and Conclusion

Reptilia, including birds and non-avian reptiles, is the sister group of mammals, and as such holds an important phylogenetic position for shedding light on patterns of gene duplication in amniotes. Reptiles are arguably more diverse than mammals in many traits; with ~17,000 species (~10,000 in birds and 7,000 non-avian reptiles) they are substantially more species-rich than mammals (~5,000 species), and possess a greater diversity of sex chromosome and sex determination systems (Organ and Janes 2008). The chicken genome is currently the sole member of Reptilia with a draft genome, and as such provides the only point of comparison of genome dynamics between mammals and their sister group. A greater understanding of genome and multigene family dynamics in mammals will undoubtedly require greater genome sampling and characterization in Reptilia.

Gene duplication and the families they produce are vital for generating the thread with which evolution weaves new adaptations and species. We have developed a pipeline for phylogenomic analysis of gene duplication in the chicken lineage, but our approach can be applied easily to any particular clade of interest. Our approach rests on the assumption that gene orthology and paralogy are best identified through phylogenetic analysis, and we delimit chicken-specific gene duplications by an approach (Figure 1) that combines initial identification and collection of gene copies across many vertebrates that show significant sequence similarity in Ensembl, followed by phylogenetic analysis of these gene sets; identification of particular nodes in these gene trees that correspond to gene duplications, in our case the mammal-bird divergence; identification of those gene clusters that diversify from these particular nodes; and statistical analysis of the collected gene trees. Many of the duplications we have identified here as “chicken-specific” in fact will be found to have duplicated in ancestors of the chicken, since orthologs of many chicken genes will no doubt be discovered in other reptile genomes as they emerge. Nonetheless, using an approximate time scale (Figure 5) we can predict which chicken gene duplicates might be found in upcoming reptilian genome projects based on their estimated timing of duplication relative to the divergence times of species whose genomes are being compared. Our approach has the advantage of providing an objective means of identifying chicken-specific gene duplications, but of course when conducted on a genome-wide scale, it will miss some gene family members that manual curation will identify; we have illustrated this with some specific examples. The loss of detail for some gene families is offset by the ability to study genome-wide distributions of multigene family dynamics; both approaches are required to provide an informed view of the dynamics of multigene family evolution in birds and relatives.

Phylogenomic approaches such as those presented here have only just begun to provide a window into the dynamics and importance of gene duplication within organisms. For example, non-protein coding RNA paralogs are dispersed throughout the chicken genome; this, along with an unusual paucity of non-protein coding RNA pseudogenes, suggests that they may not undergo the same processes of duplication (unequal crossover and retrotransposition) that characterize protein coding genes (Hillier, Miller et al. 2004). Currently available data are insufficient to address this and other hypotheses because as of the time of this writing the genome of only one reptile species has been sequenced. But progress is quickly being made with the imminent release of the zebra finch (*Taeniopygia guttata*) and anole lizard (*Anolis carolinensis*) genomes. An increase in the number of genomes will permit more detailed quantitative comparison of the evolutionary dynamics of gene duplication in amniotes and other lineages, and will help clarify the role of these gene duplications in organismal diversification.

References

Benarafa, C. and E. Remold-O'Donnell (2005). "The ovalbumin serpins revisited: Perspective from the chicken genome of clade B serpin evolution in vertebrates." Proceedings of the National Academy of Sciences of the United States of America 102(32): 11367-11372.

Benton, M. J. and P. C. J. Donoghue (2007). "Paleontological evidence to date the tree of life." Molecular Biology and Evolution 24(1): 26–53.

Currie, P. J. and P.-J. Chen (2001). "Anatomy of Sinosauropteryx prima from Liaoning, northeastern China." Canadian Journal of Earth Sciences 38(12): 1705-1727.

de Ricqlès, A. J., K. Padian, et al. (2001). The bone histology of basal birds in phylogenetic and ontogenetic perspectives. New Perspectives on the Origin and Early Evolution of Birds: Proceedings of the International Symposium in Honor of John H. Ostrom. J. Gauthier and L. F. Gall. New Haven, CT, Peabody Museum of Natural History: 411-426.

Edgar, R. C. (2004). "MUSCLE: multiple sequence alignment with high accuracy and high throughput." Nucleic Acids Research 32: 1792-1797.

Eirin-Lopez, J. M., A. M. Gonzalez-Tizon, et al. (2004). "Birth-and-death evolution with strong purifying selection in the histone H1 multigene family and the origin of orphon H1 genes." Molecular Biology and Evolution 21(10): 1992-2003.

Eisen, J. A. (1998). "Phylogenomics: improving functional predictions for uncharacterized genes by evolutionary analysis." Genome Research 8: 163-167.

Furlong, R. F. (2005). "Insights into vertebrate evolution from the chicken genome sequence." Genome Biology 6(2).

Gregory, T. R. (2002). "A bird's-eye view of the C-value enigma: genome size, cell size, and metabolic rate in the class Aves." Evolution 56(1): 121-130.

Gribaldo, S., D. Casane, et al. (2003). "Functional divergence prediction from evolutionary analysis: A case study of vertebrate hemoglobin." Molecular Biology and Evolution 20(11): 1754-1759.

Gu, Z., D. Nicolae, et al. (2002). "Rapid divergence in expression between duplicate genes inferred from microarray data." Trends in Genetics 18(12): 609-613.

Guindon, S. and O. Gascuel (2003). "A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood." Systematic Biology 52(5): 696-704.

Haasa, N. B., J. M. Grabowskia, et al. (2001). "Subfamilies of CR1 non-LTR retrotransposons have different 50 UTR sequences but are otherwise conserved." Gene 265: 175-183.

Heger, A. and C. P. Ponting (2007). "Evolutionary rate analyses of orthologs and paralogs from 12 *Drosophila* genomes." Genome Research 17(12): 1837-1849.

Hess, C. M. and S. V. Edwards (2002). "The evolution of the major histocompatibility complex in birds." Bioscience 52(5): 423-431.

Hillier, L. W., W. Miller, et al. (2004). "Sequence and comparative analysis of the chicken genome provide unique perspectives on vertebrate evolution." Nature 432(7018): 695-716.
Hittinger, C. T. and S. B. Carroll (2007). "Gene duplication and the adaptive evolution of a classic genetic switch." Nature 449: 677-681.
Hoffmann, F. G., J. C. Opazo, et al. (2008). "Rapid rates of lineage-specific gene duplication and deletion in the alpha-globin gene family." Molecular Biology and Evolution 25(3): 591-602.
Horner, J. R., K. Padian, et al. (2001). "Comparative osteology of some embryonic and perinatal archosaurs: developmental and behavioral implications for dinosaurs." Paleobiology 27(1): 39-58.
Hubbard, T. J. P., B. L. Aken, et al. (2007). "Ensembl 2007." Nucleic Acids Research 35(Database issue): D610-D617.
Huerta-Cepas, J., H. Dopazo, et al. (2007). "The human phylome." Genome Biology 8: R109.
Hughes, T. and D. A. Liberles (2008). "The power-law distribution of gene family size is driven by the pseudogenisation rate's heterogeneity between gene families." Gene 414(1-2): 85-94.
Huynen, M. A. and E. van Nimwegen (1998). "The frequency distribution of gene family sizes in complete genomes." Molecular Biology and Evolution 15(5): 583-589.
Ji, Q., P. J. Currie, et al. (1998). "Two feathered dinosaurs from northeastern China." Nature 393: 753-761.
Kondrashov, F. A., I. B. Rogozin, et al. (2002). "Selection in the evolution of gene duplications." Genome Biology 3(2): 1-9.
Lagerström, M. C., A. R. Hellström, et al. (2006). "The G protein - Coupled receptor subset of the chicken genome." Plos Computational Biology 2(6): 493-507.
Li, H., A. Coghlan, et al. (2006). "TreeFam: a curated database of phylogenetic trees of animal gene families." Nucleic Acids Research 34: D572-D580.
Li, W.-H. (2006). Molecular evolution. Sunderland, MA, Sinauer Associates.
Lynch, M. and J. S. Conery (2000). "The evolutionary fate and consequences of duplicate genes." Science 290: 1151-1155.
Lynch, M. and A. Force (2000). "The probability of duplicate-gene preservation by subfunctionalization." Genetics 154: 459-473.
Maniatis, T. and B. Tasic (2002). "Alternative pre-mRNA splicing and proteome expansion in metazoans." Nature 418(6894): 236-243.
Nevitt, G. A., M. Losekoot, et al. (2008). "Evidence for olfactory search in wandering albatross, *Diomedea exulans*." Proceedings of the National Academy of Sciences 105(12): 4576-4581.
Norell, M. A., Q. Ji, et al. (2002). "'Modern' feathers on a non-avian dinosaur." Nature 416: 36-37.
Nowak, M. A., M. C. Boerlijst, et al. (1997). "Evolution of genetic redundancy." Nature 388(6638): 167-171.
O'Dwyer, T. W., A. L. Ackerman, et al. (2008). "Examining the development of individual recognition in a burrow-nesting procellariiform, the Leach's storm-petrel." Journal of Experimental Biology 211(3): 337-340.
Ohno, S. (1970). Evolution by gene duplication. Heidelberg, Germany, Springer-Verlag.
Organ, C. L. and D. E. Janes (2008). "Evolution of sex chromosomes in Sauropsida." Integrative and Comparative Biology 48(4): 512-519.
Ota, T. and M. Nei (1994). "Divergent Evolution and Evolution by the Birth-and-Death Process in the Immunoglobulin V-H Gene Family." Molecular Biology and Evolution 11(3): 469-482.
Patel, V. S., S. J. Cooper, et al. (2008). "Platypus globin genes and flanking loci suggest a new insertional model for beta-globin evolution in birds and mammals." BMC Biology 6(34).
Piontkivska, H. and M. Nei (2003). "Birth-and-death evolution in primate MHC class I genes: Divergence time estimates." Molecular Biology and Evolution 20(4): 601-609.
Piontkivska, H., A. P. Rooney, et al. (2002). "Purifying selection and birth-and-death evolution in the histone H4 gene family." Molecular Biology and Evolution 19(5): 689-697.

Rasmussen, M. D. and M. Kellis (2007). "Accurate gene-tree reconstruction by learning gene- and species-specific substitution rates across multiple complete genomes." Genome Research 17(12): 1932-1942.
Sanderson, M. J. (2003). "r8s: inferring absolute rates of molecular evolution and divergence times in the absence of a molecular clock." Bioinformatics 19(2): 301-302.
Sarrias, M. R., J. Grønlund, et al. (2004). "The Scavenger Receptor Cysteine-Rich (SRCR) domain: an ancient and highly conserved protein module of the innate immune system." Critical Reviews in Immunology 24(1): 1-37.
Sawyer, R. H., T. Glenn, et al. (2000). "The expression of beta (β) keratins in the epidermal appendages of reptiles and birds." American Zoologist 40(4): 530-539.
Sawyer, R. H. and L. W. Knapp (2003). "Avian skin development and the evolutionary origin of feathers." Journal of Experimental Zoology Part B-Molecular and Developmental Evolution 298B(1): 57-72.
Shedlock, A. M. (2006). "Phylogenomic investigation of CR1 LINE diversity in reptiles." Systematic Biology 55(6): 902-911.
Steiger, S. S., A. E. Fidler, et al. (2008). "Avian olfactory receptor gene repertoires: evidence for a well-developed sense of smell in birds? ." Proceedings of the Royal Society (B) 275(1649): 2309-2317.
Storm, C. E. V. and E. L. L. Sonnhammer (2003). "Comprehensive analysis of orthologous protein domains using the HOPS database." Genome Research 13: 2353-2362.
Temperley, N. D., S. Berlin, et al. (2008). "Evolution of the chicken Toll-like receptor gene family: A story of gene gain and gene loss." BMC Genomics 9(62).
Toni, M., L. Dalla Valle, et al. (2007). "Hard (Beta-) keratins in the epidermis of reptiles: composition, sequence, and molecular organization " Journal of Proteome Research 6(9): 3377 -3392.
Wapinski, I., A. Pfeffer, et al. (2007). "Natural history and evolutionary principles of gene duplication in fungi." Nature 449: 54-61.
Wu, D. D., D. M. Irwin, et al. (2008). "Molecular evolution of the keratin associated protein gene family in mammals, role in the evolution of mammalian hair." BMC Evolutionary Biology 8(241).
Yuri, T., R. T. Kimball, et al. (2008). "Duplication of accelerated evolution and growth hormone gene in Passerine birds." Molecular Biology and Evolution 25(2): 352-361.
Zelano, B. and S. V. Edwards (2002). "An Mhc component to kin recognition and mate choice in birds: predictions, progress, and prospects " American Naturalist 160: S225 -S237.
Zhang, F. and Z. Zhou (2000). "A primitive enantiornithine bird and the origin of feathers." Science 290(5498): 1955-1959.
Zhang, J. (2003). "Evolution by gene duplication: an update." Trends in Genetics 18(6): 292-298.
Zhang, J., H. F. Rosenburg, et al. (1998). "Positive Darwinian selection after gene duplication in primate ribonuclease genes." Proceedings of the National Academy of Sciences 98: 3708–3713.
Zmasek, C. M. and S. R. Eddy (2002). "RIO: analyzing proteomes by automated phylogenomics using resampled inference of orthologs." BMC Bioinformatics 3: 14.

Tables

Table 1: Summary of properties discussed in the text for specific gene families in the amniote-bird lineage. Number of amniote paralogs is defined as the number of genes across all species in the amniote tree (25 mammals and chicken; see Ensembl for exact species). Mean branch length is the average path length (in substitutions/site) between chicken paralogs (tips) and their common ancestor (the first chicken duplication). Across all families this length is on average 0.235 substitutions/site. Estimated duplication time is the time of first chicken duplication in millions of years. Sequence divergence before the duplication in chicken is given in substitutions/site between the root of the amniote tree (*Gallus/Homo* common ancestor) and the first chicken duplication. λ is a molecular rate variation parameter (low values are highly variable rates; high values are clock-like).

Family name	Number of amniote paralogs	Number of chicken paralogs	Mean branch length from tips to duplication in chicken	Estimated duplication time (mya)	Sequence divergence before chicken duplication after amniote divergence	Log λ
Toll-like-receptors (*TLR2A* and *TLR2B*)	18	2	0.075	67	0.260	-2
Ovalbumin B serpins (gene *X*, gene *Y*, and *ovalbumin*)	73	3	0.141	107	0.227	-2
Ovalbumin B serpins (*MENT*, *serpinb10*)	24	2	0.183	203	0.094	-2
Hemoglobin β chain (β^H,ρ, ε)	69	3	0.075	122	0.086	-2
Olfactory receptors (orthologous to *OR5U1* and *OR5BF1* in *Homo*)	511	196	0.205	281	0.320	-2*
Olfactory receptors (related to *COR7*)	12	3	0.014	15	0.394	-2
Olfactory receptors (related to *COR 1-6*)	60	4	0.275	238	0.084	2
Olfactory receptors (small cluster on chromosome 1)	207	3	0.049	30	0.421	-2
β Keratin	117	117	0.635	NA	NA	NA

* This λ is calculated from *Homo* and *Gallus* for computational limitations due to large family size.

NA = these statistics cannot be computed because of chicken-only expansions.

Figures

Figure 1: Diagram showing the phylogenetic pattern that gene duplication takes in multi-species comparisons.

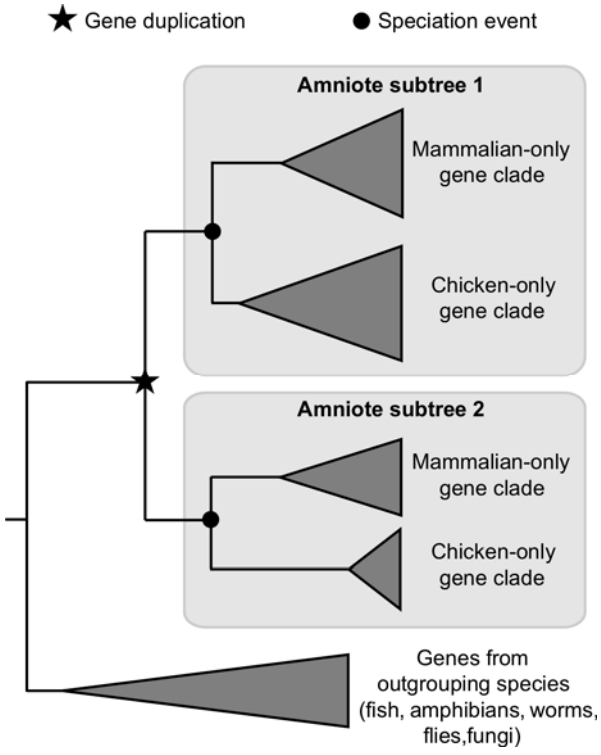

Figure 2: The β-keratin gene family tree as defined by single-linkage clustering of best reciprocal BLAST hits between amino acid translations within the Ensembl database. This is an example of chicken gene family tree used in various analyses throughout this chapter. The tip names are designated as Ensembl
proteins.

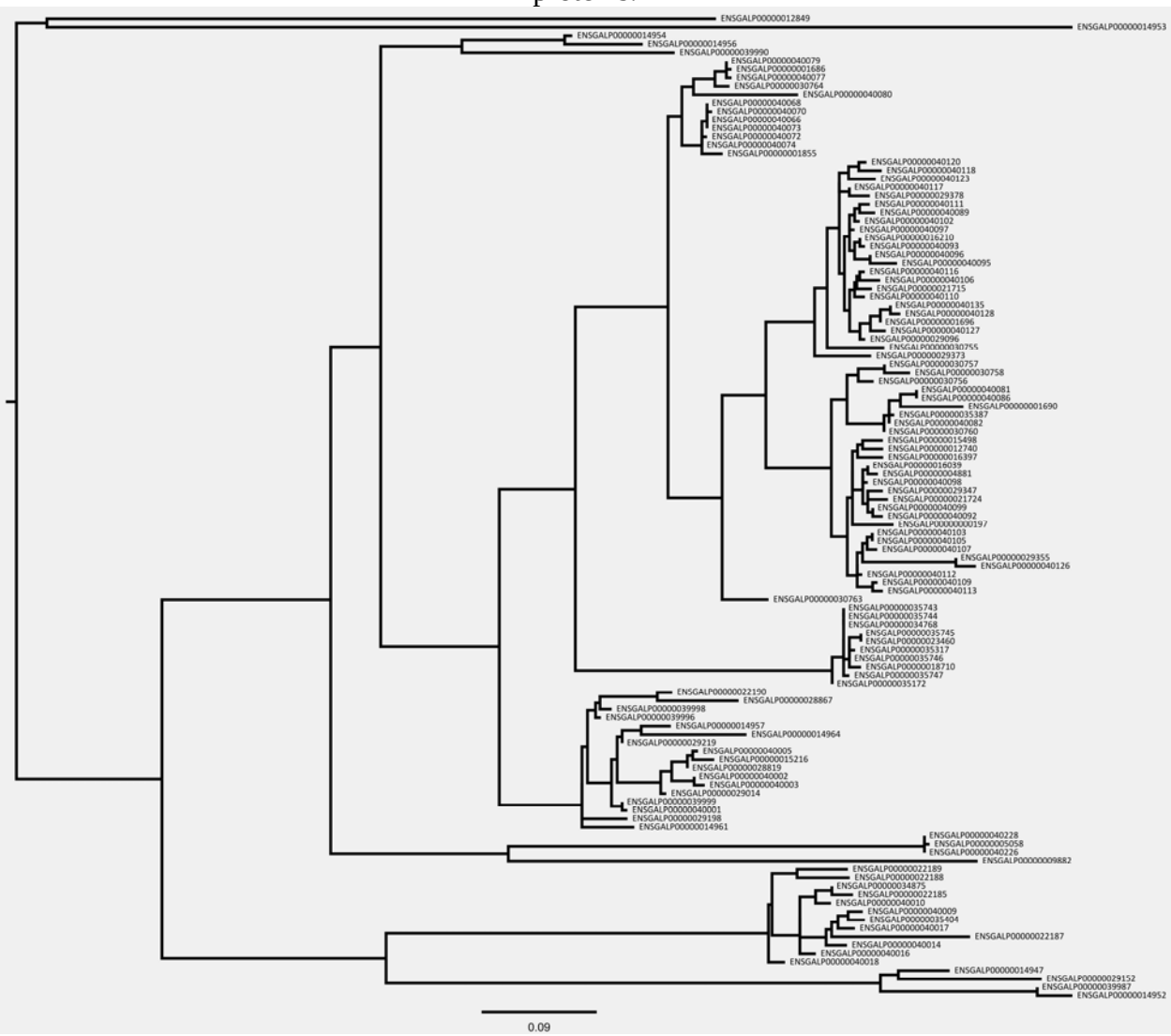

Figure 3: Count of gene families vs. the number of chicken genes in amniote subtrees.

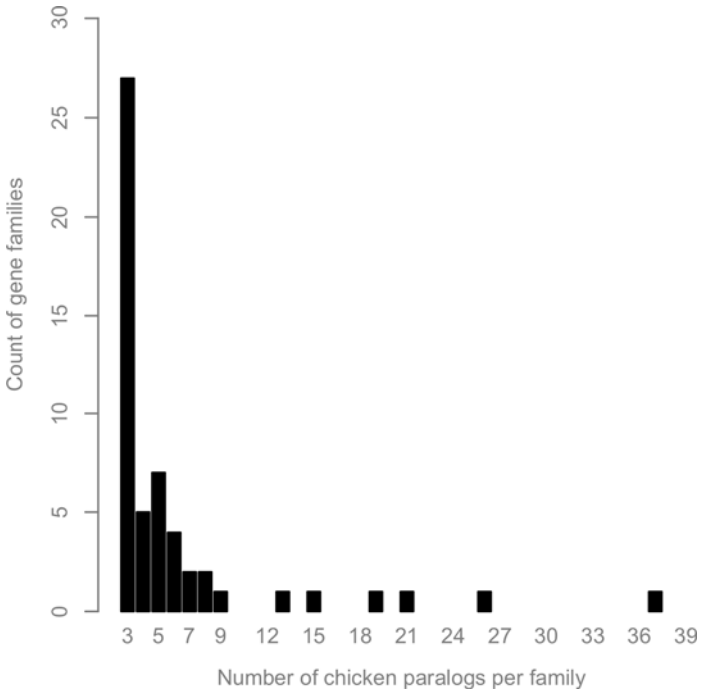

Figure 4: Branch length rate variation (molecular clock; λ) for amniote gene trees (25 mammals and chicken). **A**, the distribution of branch length rate variation binned in different λ values. Low values of λ represent highly variable rates and high values of λ represent clock-like rates. **B**, the best fit (by likelihood criterion) λ by tree size. r^2 relating tree size with log λ is equal to 0.0002843 (p-value = 0.08, n = 6,901). Values of λ were dithered for display purposes to illustrate density.

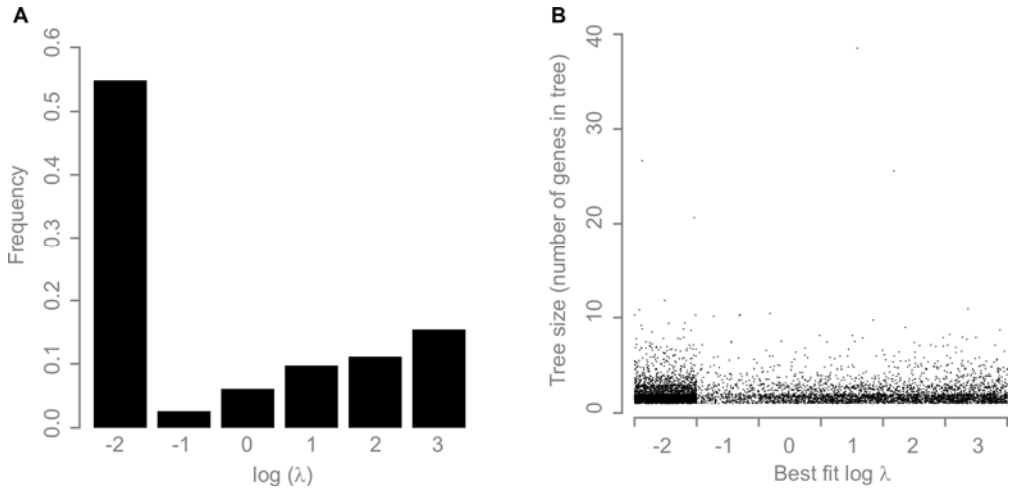

Figure 5: Depth of paralogs on the lineage leading to *Gallus gallus* that evolved after the divergence between *Gallus* and *Homo*. Pivotal events during the evolution of this lineage are noted on the figure. Pg stands for Paleogene and Ng stands for Neogene.