The UMAP Journal

Vol. 31, No. 3

Subscription Rates for 2010 Calendar Year: Volume 31

Institutional Web Membership (Web Only)

Institutional Web Memberships do not provide print materials. Web memberships allow members to search our online catalog, download COMAP print materials, and reproduce them for classroom use.

(Domestic) #3030 $467 (Outside U.S.) #3030 $467

Institutional Membership (Print Only)

Institutional Memberships receive print copies of *The UMAP Journal* quarterly, our annual CD collection UMAP Modules, *Tools for Teaching*, and our organizational newsletter *Consortium*.

(Domestic) #3040 $312 (Outside U.S.) #3041 $351

Institutional Plus Membership (Print Plus Web)

Institutional Plus Memberships receive print copies of the quarterly issues of *The UMAP Journal*, our annual CD collection UMAP Modules, *Tools for Teaching*, our organizational newsletter *Consortium*, and online membership that allows members to search our online catalog, download COMAP print materials, and reproduce them for classroom use.

(Domestic) #3070 $615 (Outside U.S.) #3071 $659

For individual membership options visit
www.comap.com for more information.

To order, send a check or money order to COMAP, or call toll-free
1-800-77-COMAP (1-800-772-6627).

The UMAP Journal is published quarterly by the Consortium for Mathematics and Its Applications (COMAP), Inc., Suite 3B, 175 Middlesex Tpke., Bedford, MA, 01730, in cooperation with the American Mathematical Association of Two-Year Colleges (AMATYC), the Mathematical Association of America (MAA), the National Council of Teachers of Mathematics (NCTM), the American Statistical Association (ASA), the Society for Industrial and Applied Mathematics (SIAM), and The Institute for Operations Research and the Management Sciences (INFORMS). The Journal acquaints readers with a wide variety of professional applications of the mathematical sciences and provides a forum for the discussion of new directions in mathematical education (ISSN 0197-3622).

Periodical rate postage paid at Boston, MA and at additional mailing offices.

Send address changes to: info@comap.com
COMAP, Inc., Suite 3B, 175 Middlesex Tpke., Bedford, MA, 01730


COMAP

Vol. 31, No. 3 2010

Table of Contents

Guest Editorial

Articles

BioMath Module

Guest Editorial

The Bio-Math Connection

Margaret Cozzens
The Center for Discrete Mathematics and Theoretical Computer Science (DIMACS)
Rutgers University
96 Frelinghuysen Road, Room 419
Piscataway, NJ 08854–8018
midgec@dimacs.rutgers.edu

Introduction

Biology as a discipline is a small component of school learning from elementary school through high school and is most often the course taken in college to satisfy a science requirement. In contrast, mathematics has always had the luxury and responsibility of being recognized as a major fundamental part of all school learning, indeed one of the three Rs. Early on, the relationships between mathematics and the physical sciences have been appreciated and often have been used as a reason to study mathematics and its applications to the physical sciences. However, the interplay between mathematics and the biological sciences was understood by only a few.

The Change

All this has changed! Increasingly, many biological phenomena are being viewed as involving the processing of information, which ultimately involves using the mathematical/information sciences ("mathematics" in the larger sense). They have played an important role in major biological accomplishments (e.g., in sequencing the human genome) and are fundamentally important in the rapidly expanding "digital biology." Use of mathematics increasingly appears in papers and books in all areas of biology, while biological ideas inspire new concepts and methods in mathematics itself. More and more, students are exposed to interplay between the mathematical and biological sciences.

The UMAP Journal 31 (3) (2010) 185–188.

Despite the fact that contemporary biology is inextricably linked to mathematical concepts, in high school both subjects are usually taught in strict isolation; very little has been done to develop the interconnections in the high school curriculum. Introducing high school students to these interconnections enhances the study of both disciplines:

- Students interested in biology realize the importance of understanding modern mathematics and computer science.
- New horizons are opened for those who find mathematics interesting but wonder how it might be useful.
- Rapidly developing opportunities for further study are revealed and new career opportunities are suggested.

To achieve these ends, it is critical to have curricular materials that highlight the interconnections and are readily available for teachers to use.

BMC

Recognizing this need, the National Science Foundation provided grants to DIMACS in partnership with COMAP and Colorado State University for the Bio-Math Connection (BMC) project. BMC provides an opportunity for high school teachers, writers, researchers, and others to get in on the ground floor of developing innovative classroom materials. The principal goal is to provide teachers with curricular materials that highlight the interconnections; a secondary goal is to help teachers use these materials and understand the interface between the two disciplines. Thus, the main product of this project is 24 teaching/learning Modules, including teacher support materials, that can be adapted for use in a variety of grade levels in either biology or mathematics classes (or both). A book containing some of the Modules is intended for a one-semester senior-level non-calculus-based course that will satisfy part of state requirements for a fourth year of mathematics or science. Fifteen Modules are complete now. To write such materials requires first answering some important questions:

- What mathematics and what biology does a student need to know to advance to college and/or work at the interface of biology and mathematics?
- How do teachers add more mathematics to a very full biology curriculum, and how can biological applications be added to a very full mathematics curriculum?
- How do mathematics teachers learn enough biology to incorporate biological applications in their courses, and how do biology teachers learn enough mathematics to incorporate more mathematics in their biology courses?

- Does the sequencing of biology and mathematics courses at the high school level "work," so that what is learned in one course can be used in the other course? For example, if Algebra 1 and Biology are taught in different years, is that a problem?

Discussions of these questions preceded and informed the writing of the Modules and continues today.

Teachers are involved in the production, testing, and dissemination of the BMC Modules—teachers are critically important in determining what teachers need to be able to teach the Modules.

Teachers to field test Modules first met for a one-week summer workshop with a lead lecturer, other experts, and lead teachers. Each of the three summer workshops treated three Modules. For the last six Modules and all new ones, teachers will not have the benefit of such workshops, but support materials will be provided through teacher guides and an online collaborative learning environment.

Not only have exciting, challenging materials at the interface of biology and mathematics been developed for high school mathematics and biology classes, but much has been learned along the way:

- First and foremost, the solid, multipronged on-going evaluation program has been extremely important for success of the BMC project.
- Teachers are eager for new materials and eager to participate in groundbreaking workshops and activities. Providing teachers with support materials is best done through week-long summer workshops; but we can provide support systems online, utilizing what we have already learned from the workshops and making teachers who are expert in a Module available online for real-time discussion.
- Perhaps most surprising is how extraordinarily responsive students are to learning mathematics and biology together. One might have expected this in AP courses in biology or mathematics, or even in 11th or 12th grade classes, but it is true in classes at all levels (8th through 12th), in all parts of the country, in urban, rural, and suburban schools. For example, in an alternative high school participating in the field-test program, the teachers reported unusually high student interest and enthusiasm; students were "very proud to be a part of this special group" using the BMC Module. The teachers related:

  > During the week of teaching this Module, attendance was unusually high in our classes. They wanted to be here (in school) and in our class. Discussion was very good, and we did not have to force them to do the work. There was a friendly competitiveness in how to do the algorithm and who got it right. They were focused in the classroom, which is very atypical. Not a lot of redirecting had to happen. It was fun week to be teaching.

The first 15 Modules are all in almost final form:

- Computational Biology:
  - Spider Silk
  - Genetic Inversions (reprinted in this issue)
  - Evolution by Substitution
  - Microarrays
  - Forensics
- Epidemiology:
  - Imperfect Testing
  - Competition in Disease Evolution
  - Modeling Disease Outbreaks
  - Genetic Epidemiology
- Ecology:
  - Food Webs
  - Home Range Analysis
  - Ecological Footprints
  - Drawing Lines—Voronoi Diagrams
  - Habitat Formation—Help, I'm Surrounded by Squirrels
  - Evolutionarily Stable Strategies

Nine more Modules are to be written over the next two years.

A Role for You

- **We are looking for biomath topics** that go beyond ecology, computational biology, and epidemiology, yet are accessible to high school students without a calculus background.
- **We are seeking writers** for BioMath Modules: mathematicians, biologists, mathematics and science educators, and high school teachers in both fields.
- Since each Module goes through a final technical review process after field testing by both a mathematician and a biologist, **we are looking for reviewers** for Modules (those listed above and subsequent ones).
- **Finished materials are free for you to use; contact COMAP for details.**

About the Author

Margaret ("Midge") Cozzens is a Professor at DIMACS at Rutgers University, a former director at the NSF, a former high school teacher, and currently director of the Bio-Math Connection project.

Curve Interpolation and Coding Theory

Darren Glass
Mathematics Dept.
Gettysburg College
Gettysburg, PA 17325–1486
dglass@gettysburg.edu

Introduction

Whether it is downloading files from the Internet, having conversations between cell phones, or sending information from a laptop to a printer, we often want to transmit data in situations where we need to worry about interference from other signals that may cause errors in the transmission.

The branch of mathematics known as coding theory is dedicated to finding ways to tell when there are errors in transmission and, when possible, how to correct those errors. The goal of coding theory is to build as much redundancy as possible into a message without greatly increasing its length. Much of coding theory uses deep mathematics to achieve this end, but a surprising amount of work follows from the following fact of Euclidean geometry, which is known by schoolchildren:

Two points determine a unique line.

More accurately, we will use the more sophisticated version of this fact that says:

Fact: *Any n points with different x-coordinates determine a unique $(n-1)$st-degree polynomial $y = a_{n-1}x^{n-1} + \ldots + a_1x + a_0$.*

In the following section, we discuss why this fact is true and mention a couple of approaches to proving it. We then discuss how Reed and Solomon [1960] used it to create a family of error-correcting codes. In our opinion, Reed-Solomon codes are an underappreciated application of mathematics that is both extremely useful and very accessible to any student who has made it through the opening weeks of a linear algebra course.

The UMAP Journal 31 (3) (2010) 189–196.

Polynomial Interpolation

That through two points a line can be drawn is Postulate 1 of Euclid; that the line is unique, we know from analytic geometry applied to Euclidean geometry. But why is it that three points with distinct x-coordinates determine a unique quadratic equation?

We start by sketching one possible answer: In drawing a parabola, you could choose any two x-values to be the zeros. For example, let's choose the values $x_1 = -1$ and $x_2 = 3$. As you can see in **Figure 1**, many different parabolas have these two zeros. In fact, for any constant a, the curve $y = a(x+1)(x-3)$ passes through the two points $(-1, 0)$ and $(3, 0)$.

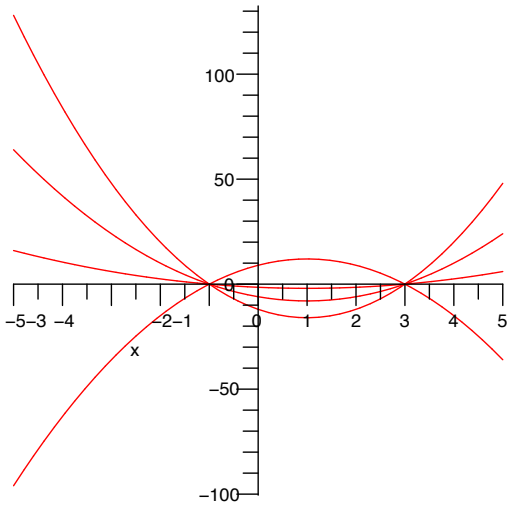


Figure 1. Many parabolas through three specified points.

According to the **Fact** cited above on p. 189, we can choose one more point so as to obtain a unique quadratic equation. Since we did not choose $x = 0$ as either of the zeros, we could choose the y-intercept as the third point. In particular, we could specify that the curve pass through $(0, 3)$, finding then that the unique quadratic equation passing through these three points is $y = -x^2 + 2x + 3$. (As an interesting aside, think about what would happen if we chose $(0, 0)$ as the y-intercept.)

This approach of determining a unique curve by picking points based on their x-coordinates generalizes to situations where we wish to have a parabola that does not have two distinct zeros or where we specify more than three points (and therefore are considering polynomials of higher degree). However, writing down a careful general proof of the **Fact**, using analytic geometry, can be tedious.

A different approach to proving the **Fact** in general, which many students see in a course in linear algebra, is to show that as long as $x_1, \ldots, x_n$ are distinct, the following *Vandermonde matrix* is invertible:

$$\begin{bmatrix} 1 & x_1 & {x_1}^2 & \cdots & {x_1}^{n-1} \\ 1 & x_2 & {x_2}^2 & \cdots & {x_2}^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & {x_n}^2 & \cdots & {x_n}^{n-1} \end{bmatrix}$$

Showing that this matrix is invertible is an exercise in many linear algebra texts [Lay 2003, Ch. 2; Shifrin and Adams 2002, Sect. 1.6; Leon 1980, Sect. 1.4]. From the invertibility, it follows that for any choice of $y_1, \ldots, y_n$, the following linear system has a unique solution $a_1, \ldots, a_n$:

$$\begin{bmatrix} 1 & x_1 & {x_1}^2 & \cdots & {x_1}^{n-1} \\ 1 & x_2 & {x_2}^2 & \cdots & {x_2}^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & {x_n}^2 & \cdots & {x_n}^{n-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}.$$

The uniqueness of the a_is yields a unique polynomial

$$P(x) = a_{n-1}x^{n-1} + \ldots + a_1 x + a_0$$

of degree at most $n - 1$ that passes through the desired points (x_i, y_i).

In addition to being useful in curve interpolation, the Vandermonde matrix also shows up in discrete Fourier transforms, representation theory of the symmetric group, and many other places in mathematics.

Finding the a_is can be done quickly via *Lagrange interpolation*, which defines the polynomial by the following formula:

$$P(x) = \sum_{j=1}^{n} y_j \prod_{i \neq j} \frac{x - x_i}{x_j - x_i}.$$

Coding Theory

The goal of coding theory is to build enough redundancy into a message so that errors can be corrected while keeping the message reasonably short.

For example, let's say that you want to transmit the message 8675309 but you are worried that the line is too noisy, so that errors are likely to occur. One naïve approach would be simply to transmit the message *twice*. If the message received is $8679300/8695329$, then one can easily tell that errors have occurred because the two transmissions don't exactly match up. For example, we can be sure that there is an error in the third position in (at least) one of the two transmissions. Unfortunately, we have no way of knowing which message is correct in this position.

The slightly less naïve approach of transmitting the message *three times* may allow the recipient to correct errors. Assume that the message received

is $8679300/8695329/8775309$. Decode by majority rule: If two or all three of the transmissions agree in a position, assume that they are correct, since the odds of the same error occurring in the same place twice will be small.

You have probably hit on the next generalization: If you want to be surer that the message gets through correctly, repeat it more times! This will certainly work, but at the cost of increasing the transmission length, taking up more bandwidth (or cellphone minutes).

However, there are many other approaches to error correction; many use sophisticated mathematics in complicated ways. An interested beginner might start investigating in Gallian [1993] or Roman [1997].

Reed-Solomon Codes

Reed-Solomon codes require only simple polynomial interpolation to correct errors more efficiently than the naïve approach above.

Example. Assume that we want to communicate the message $(1, -2, -1)$. We first encode this message as the coefficients of a polynomial: $f(x) = x^2 - 2x - 1$. Next, we compute the value of this polynomial at a set of prescribed points and transmit those values. For our example, we assume that the pre-chosen points are $x = 0, 1, 2, 3, 4$; so we compute $f(0) = -1$, $f(1) = -2$, $f(2) = -1$, $f(3) = 2$, and $f(4) = 7$ and transmit the message $C_f = (-1, -2, -1, 2, 7)$.

If there are no errors, the recipient of the message can use Lagrange interpolation or other methods to discover the unique quadratic through the points $(0, -1)$, $(1, -2)$, $(2, -1)$, $(3, 2)$, and $(4, 7)$ and recover the intended message $(1, -2, -1)$ as its coefficients.

To see the advantage of this method, note what will happen if one of the values in the message is received incorrectly. For example, maybe the message received is $C = (-1, -2, 1, 2, 7)$.

If we plot the resulting five points $(0, -1)$, $(1, -2)$, $(2, 1)$, $(3, 2)$, and $(4, 7)$, we can easily detect that there has been an error in transmission, because the points no longer lie on a parabola (**Figure 2**). Moreover, we can *correct* the error by finding a polynomial that fits as many of the points as possible.

If we fit the first three points to a quadratic polynomial (using, for example, Lagrange interpolation), we get $f_1(x) = 2x^2 - 3x - 1$, which misses the last two points (**Figure 3**).

Similarly, if we fit the middle three points to a quadratic polynomial, we get $f_2(x) = -x^2 + 6x - 7$, which misses the other two points.

However, if we fit the first, second, and fourth points to a quadratic polynomial, we get $f(x) = x^2 - 2x - 1$, which also passes through the fifth point! And this is the best that we can do, since all five points do not lie on a quadratic.

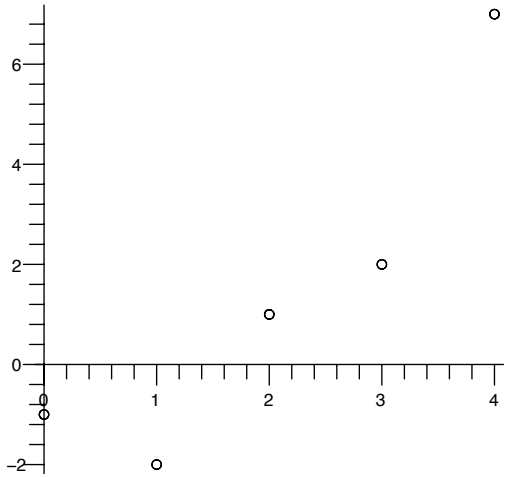


Figure 2. Because no parabola fits, there is an error.

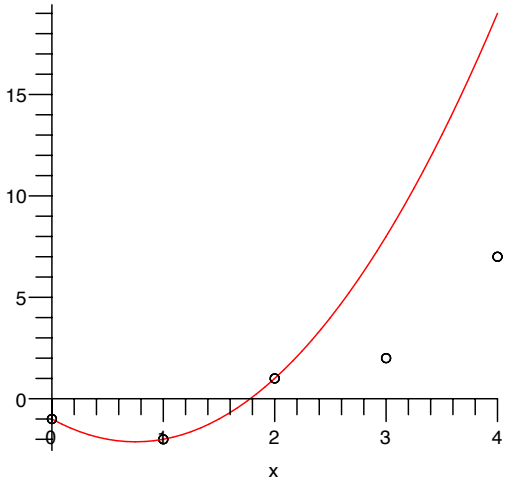


Figure 3. A parabola that fits the first three points does not fit the last two.

Moreover, because any three points lie on a unique quadratic curve, there is no quadratic that passes through any other set of four of the points. Thus, this process recovers the initial message. Comparing this example to the naïve approach of sending the message twice, here we send a shorter message (five characters instead of six) and can correct an error instead of just identifying it. This capability gives some indication of the strength of Reed-Solomon codes.

More generally, suppose that we have a message of length k but the technical capacity and time to send a transmission of length n. The Reed-Solomon approach begins by fixing n numbers $x_1, \ldots, x_n$. An n-tuple is a *k-valid codeword* if it can be generated by evaluating a polynomial of degree strictly less than k at these n predetermined points.

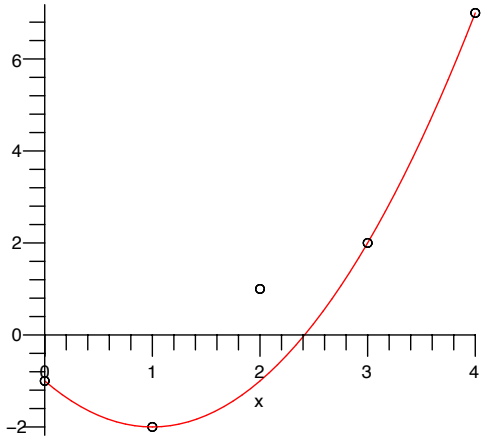


Figure 3. A parabola that fits the first, second, and fourth points also passes through the fifth point—but not through the third.

Example. If $n = 4$, $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, and $x_4 = 4$, then

- $(-2, -1, 0, 1)$ is a 2-valid codeword, because it is $\big(f(1), f(2), f(3), f(4)\big)$ where $f(x) = x - 3$ is a linear equation; but
- $(0, 0, 0, 1)$ is not a 2-valid codeword, because there is no linear or constant equation with $f(1) = f(2) = f(3) = 0$ and $f(4) = 1$.

To send our message, we construct a polynomial $g(x)$ of degree $k - 1$. We evaluate the polynomial at n distinct predetermined points $x_1, \ldots, x_n$ and transmit the message

$$\big(g(x_1), \ldots, g(x_n)\big).$$

Because any two k-valid n-tuples agree in at most $k - 1$ coordinates—if two polynomials of degree less than k agree in k points, then they are the same polynomial!—the n-tuples must disagree in at least $n - (k - 1)$ of the n preselected points. Therefore, changing fewer than $n - k + 1$ of the values in the n-tuple will result in an invalid codeword, and *we can thus detect up to $n - k$ errors*.

To see this in an explicit example, try changing any two of the values in the 2-valid 4-tuple $(-2, -1, 0, 1)$—you will see that the resulting 4-tuple is invalid! Moreover, if you change only one of the values, I can correct it by finding the unique line through the other three points (I can do this even if I do not know which point you changed, since only one set of three of the four points will be collinear!). In this manner,

Reed-Solomon codes can correct up to $\left\lfloor \frac{n-k}{2} \right\rfloor$ errors.

Hamming Distance

One way to reconceptualize the situation is to view each k-valid n-tuple as a point in n-dimensional space. However, instead of measuring the distance between two points in the normal Euclidean way, we define a new distance measure, the number of coordinates in which they disagree, called the *Hamming distance*, after Richard Hamming, who is often credited as the father of coding theory.

In this conceptualization, two k-valid points obtained from polynomials as on p. 194 must be at least $n - k + 1$ apart in Hamming distance. Therefore, from any given point x, there is at most one k-valid point whose distance is at most $\lfloor \frac{n-k}{2} \rfloor$ from x. Indeed, if there were two k-valid points y and z whose distances were less than or equal to $\lfloor \frac{n-k}{2} \rfloor$, then the triangle inequality would tell us that the distance between y and z was at most $2\lfloor \frac{n-k}{2} \rfloor$, which is strictly less than $n - k + 1$.

In particular, if we start at a k-valid point and make changes in up to $\lfloor \frac{n-k}{2} \rfloor$ positions, our original point will be the only k-valid point within a radius of $\lfloor \frac{n-k}{2} \rfloor$. Therefore, we can correct up to this number of errors by moving to the unique k-valid point that is closest to the received codeword.

Generalizations

This approach can be generalized further to find other sets of valid codewords that work nicely, known as AG-codes, which were first developed by Goppa [1981]. The curve-interpolation approach starts with a polynomial of given degree, defined everywhere on the x-axis and with a single pole of fixed order at $x = \infty$. More-general AG-codes look at functions on other curves with prescribed zeros together with points (poles) where the function is allowed to be undefined but where the pole has a fixed order. The general idea is that by evaluating these functions at n different points on the curve, you get a codeword of length n. There may be other valid codewords, depending on the genus of the curve and the set of prescribed zeros and points where the function is not defined. Depending on the choices of curve and functions, there may also be efficiencies (or a lack thereof) in encoding and decoding. Fully understanding this approach involves learning some algebraic geometry as well as some analysis. An excellent introduction is Walker [2000].

There are other approaches to constructing codes. In fact, many people would say that the *real* goal of coding theory is to define large sets of points in n-dimensional space so that they are efficiently packed, with a large minimum distance but a small total volume. Sphere packing is a long-studied problem with many other applications. For a full discussion, see Pfender and Ziegler [2004].

References

Gallian, Joseph. 1993. How computers can read and correct ID numbers. *Math Horizons* (Winter 1993): 14–15.

Goppa, V. 1981. Codes on algebraic curves. *Doklady Akademii Nauk SSSR* 259 (6): 1289–1290.

Lay, David. 2003. *Linear Algebra and Its Applications*. 3rd ed. Reading, MA: Addison-Wesley.

Leon S. 1980. *Linear Algebra with Applications*. New York: Macmillan.

Pfender, Florian, and Günter M. Ziegler. 2004. Kissing numbers, sphere packings, and some unexpected proofs. *Notices of the American Mathematical Society* 51 (8): 873–883.
`http://www.ams.org/notices/200408/fea-pfender.pdf` .

Reed, I., and G. Solomon. 1960. Polynomial codes over certain finite fields. *SIAM Journal* 8 (2): 300–304.

Roman, S. 1997. *Introduction to Coding and Information Theory*. New York: Springer-Verlag.

Shifrin, T., and M. Adams. 2002. *Linear Algebra: A Geometric Approach*. New York: W.H. Freeman.

Walker, J. 2000. *Codes and Curves.* IAS/Park City Mathematical Subseries. Providence, RI: American Mathematical Society, and Princeton, NJ: Institute for Advanced Study.

About the Author

Darren Glass received his B.A. from Rice University, with a double major in mathematics and mathematical economic analysis. He then went to the University of Pennsylvania, where he studied arithmetic geometry and received both an M.A. and a Ph.D. After several years as an NSF-VIGRE postdoc at Columbia University, Glass joined the faculty at Gettysburg College in 2005. His primary research interests are in the fields of number theory and algebraic geometry, with an eye towards applications in cryptography and coding theory. In his spare time, he enjoys spending time with his toddler son, watching baseball, and cooking Mexican food.

Cutoffs and Thresholds in the Democratic Primaries

Michael A. Jones
Mathematical Reviews
416 Fourth Street
Ann Arbor, MI 48103
maj@ams.org

Jennifer M. Wilson
The Eugene Lang College
The New School for Liberal Arts
New York, NY 10011
wilsonj@newschool.edu

Introduction

The 2008 Democratic Primary season was one of the most riveting in recent history in part because of the lengthy contest between New York Senator Hillary Clinton and Illinois Senator Barack Obama. Of the eight candidates who mounted national campaigns, only 3 received delegates during the primary season, and by "Super Tuesday," February 5, 2008 (which for 2008 was dubbed "Super Duper Tuesday" because 24 states held primaries or caucuses), all but the top 2 candidates had been eliminated. What caused this rapid narrowing of the field? The answer lies in part with the apportionment process used to assign delegates to presidential candidates.

Apportionment is best known for determining the number of seats that each state receives in the U.S. House of Representatives (see Eisner [1992] and Malkevitch [2000]). Less well known is that apportionment also plays a vital role in the Democratic Party Presidential Primary. The Democratic Delegate Selection Rules (**Table 1**) stipulate that delegates be awarded based on the candidate's share of the popular vote at local district levels, as well as at the state level (as described in [Geist, Jones and Wilson 2010]). While the details vary among the states, the rules specify that in each contest Hamilton's method is to be used to apportion the delegates, and that only

The UMAP Journal 31 (3) (2010) 197–214.

candidates receiving at least 15% of the popular vote are eligible to receive delegates.

We focus on the relationship between apportionment methods and arbitrary cutoffs such as 15%. A larger cutoff can contribute to the faster elimination of candidates by denying delegates to weakly supported candidates. This feature has the most impact in the early stages of the primary process, for example, in the first primary of the season, which historically takes place in New Hampshire. The Democratic Party's choice of 15% represents a political compromise between a wider, more-inclusive field and a faster and more-decisive consensus—the latter is exemplified by the Republican Party's "winner take all" primary process.

Table 1.
Democratic Delegate Selection Rules
[Democratic Party of the United States 2006, 15 (Section 13, Part D)].

Steps	Action
1	Tabulate the percentage of the vote that each presidential preference (including uncommitted status) receives in the congressional district to three decimals.
2	Retabulate the percentage of the vote to three decimals, received by each presidential preference excluding the votes of presidential preferences whose percentage in Step 1 falls below 15%.
3	Multiply the number of delegates to be allocated by the percentage received by each presidential preference.
4	Delegates shall be allocated to each presidential preference based on the whole numbers which result from the multiplication in Step 3.
5	Remaining delegates, if any, shall be awarded in order of the highest fractional remainders in Step 3.

We examine the consequences of using a cutoff by analyzing the minimal support that is necessary and sufficient for a candidate to receive a delegate.

In any apportionment method, the number of delegates a candidate receives depends not only on his or her share of the popular vote but also on the distribution of votes among the other candidates. In fact, three situations can occur:

- A candidate's popular vote may be insufficient to earn a delegate regardless of how the popular vote is split.
- A candidate's popular vote may be large enough to earn a delegate if the popular vote is split in a particular way.
- A candidate's popular vote is sufficient to earn a delegate regardless of how the vote is split.

The boundaries among the three cases can be described as solutions to two optimization problems:

- the *threshold of inclusion*, which represents the minimal popular support *necessary* for a candidate to receive a delegate; and
- the *threshold of exclusion*, which represents the minimal popular support *sufficient* for a candidate to receive a delegate.

Thresholds of inclusion and exclusion have been studied previously in more generality in the context of *proportional representation systems* of government, in which each party receives a number of seats in parliament based on its share of the popular vote. Lijphart and Gibberd [1977] and Palomares and Ramírez [2003] determine the minimal and maximal bounds for a party's level of support to receive a fixed number of seats, using Hamilton's method or using one of several divisor methods.

Our interest is in special cases of their results in which the thresholds relate to a candidate's first delegate. We look at how Hamilton's method was applied in the 2004 New Hampshire Democratic Primary (because the 2004 primary exhibited behavior not present in 2008). We then evaluate thresholds using Hamilton's method in a 3-candidate race, developing intuition by analyzing the geometry of a simplex. We compare these results to what happens under a 15% cutoff. We then look at thresholds of inclusion and exclusion for two well-known methods of apportionment, due to Thomas Jefferson and Daniel Webster, and relate them to those of Hamilton and to cutoffs, including the 15% used in the Democratic Primary.

Hamilton's Method and Thresholds of Inclusion and Exclusion

The Democratic Party selects its nominee through a series of state caucuses and primaries, which culminates in a final vote at the Democratic National Convention. Although every state primary adheres to the Democratic Delegate Selection Rules, the number of delegates and their organization into districts is unique to each state. The district delegates are awarded to candidates based on the number of votes that the candidates receive in the district. Additional delegates—the at-large and pledged "Party Leaders and Elected Officials" (PLEOs) delegates—are awarded according to the number of votes received at the state level. Details can be found in the Delegate Selection Plan documents found on each state's Democratic Party website. A candidate must receive a majority of the 4,234 delegates at the convention to become the nominee.

The role of the cutoff is particularly important in the early states of the Democratic Primary, when a strong showing is crucial for candidates to raise money and to remain viable. To illustrate the use of Hamilton's method and how the cutoff can affect the outcome, we consider the 2004 New Hampshire Democratic Primary.

Example 1 (2004 New Hampshire Primary). New Hampshire is divided into two districts, each with 7 delegates. Of the 23 Democratic candidates who received votes in New Hampshire's First District, the top 5 vote-getters are listed in **Table 2**.

Table 2.
Allocating the 7 delegates in New Hampshire's First District with a 15% cutoff.

Candidates	Votes	Percentage	Adjusted Percentage	Quota	Initial App't	Final App't
Clark	13,304	12.685				
Dean	24,268	23.138	36.883	2.582	2	3
Edwards	13,091	12.481				
Kerry	41,530	39.596	63.117	4.418	4	4
Lieberman	10,404	9.9195				
Others	2,287	2.1805				
Total	104,884	100	100	7.000	6	7

The columns of the table demonstrate the steps outlined in the Democratic Delegate Selection Rules of **Table 1**.

- Following Step 1, the first column contains the percentage of the popular vote that the 5 top candidates earned, as well as the remaining 18 candidates.
- As described in Step 2, all candidates who receive less than 15% of the popular vote are eliminated (including General Clark and Senators Edwards and Lieberman), and the percentages of the remaining candidates (Governor Dean and Senator Kerry) are adjusted to represent their percentage of the vote after all votes for candidates that did not meet the cutoff are discarded.
- Steps 3 to 5 comprise what is usually referred to as Hamilton's method.
  - In Step 3, the remaining candidates' adjusted percentages are multiplied by 7 to calculate their *quotas*—the proportion of the delegates the candidate should receive given his or her adjusted percentage of the vote.
  - In Step 4, the quotas are rounded down to the nearest integer, giving an initial apportionment of 2 and 4 to Governor Dean and Senator Kerry, respectively.
  - In Step 5, the remaining delegate is awarded to the candidate with the largest remainder—Governor Dean, because $0.582 > 0.418$.

The 15% cutoff had an effect on the outcome because only the top 2 candidates obtained enough votes to receive a delegate.

Table 3 applies Hamilton's method to apportion the 7 delegates without the 15% cutoff. After assigning initial delegates according to the value of candidates' quotas rounded down to the nearest integer, the remaining 4 delegates are awarded in succession to Clark, Edwards, Kerry, and Lieberman. The 15% cutoff prevented allocating 1 delegate each to Clark, Edwards, and Lieberman. □

Table 3.
Allocating the 7 delegates in New Hampshire's First District without a cutoff.

Candidates	Votes	Percentage	Quota	Initial App't	Final App't
Clark	13,304	12.685	0.888	0	1
Dean	24,268	23.138	1.620	1	1
Edwards	13,091	12.481	0.874	0	1
Kerry	41,530	39.596	2.772	2	3
Lieberman	10,404	9.9195	0.694	0	1
Others	2,287	2.1805	0.153	0	0
Total	104,884	100	7.000	3	7

The outcome from the 2004 New Hampshire Primary raises a number of questions about the relationship between the cutoff and the level of support necessary for a less-popular candidate to receive a delegate with or without a cutoff. To investigate this relationship, we begin by formally defining the two thresholds described in the introduction.

Suppose that P votes are cast for n candidates and that each candidate i gains a share P_i of the votes, where $P_i \geq 0$ and $\sum_{i \leq n} P_i = P$. Further, suppose that H delegates are to be apportioned among the candidates based on the P_is; H is used because the district size takes the place of the house size in a more traditional apportionment context.

An *apportionment method* assigns a nonnegative integer number of delegates h_i to each candidate i such that $\sum_{i=1}^{n} h_i = H$. To allow for the possibility of ties at some step in the apportionment process, an apportionment method is formally described as a set-valued function that maps the popular vote distribution to a set of possible apportionments

$$F(p_1, p_2, \ldots, p_n) \subset \{(h_1, h_2, \ldots, h_n) \;\Big|\; \sum_{i=1}^{n} h_i = H \text{ and } h_i \geq 0 \text{ for all } i\},$$

where $p_i = P_i/P$ is the percentage of the vote received by candidate i. The values of the h_is depend on the particular method of apportionment. (See Balinski and Young [2001] for the definitive work on apportionment methods.) It is usually assumed that if $F(p_1, \ldots, p_n)$ is not a singleton, then an apportionment is selected from the set by either an a priori rule or randomly. In practice, such ties occur infrequently.

To aid in the presentation, we define the thresholds of inclusion and exclusion in terms of the percentage of the vote for candidate n to *possibly*, and to *assuredly*, respectively, receive a delegate. The thresholds of inclusion and exclusion are defined, for convenience, in terms of candidate n, because apportionment methods are symmetric with respect to permutations of the p_is.

For a specific apportionment method, the *threshold of inclusion* T_I is

$$T_I = \inf\{p_n | \text{ there exist } p_1, \ldots, p_{n-1} \text{ such that } h_n \neq 0 \text{ for some } (h_1, \ldots, h_n) \in F(p_1, \ldots, p_n)\}.$$

The value T_I is the smallest portion of the vote for which candidate n *may* receive a delegate.

Similarly, the *threshold of exclusion* T_E is

$$T_E = \inf\{p_n | \text{ for all } p_1, \ldots, p_{n-1} \text{ there exists } (h_1, \ldots, h_n) \in F(p_1, \ldots, p_n) \text{ such that } h_n \neq 0\}.$$

The value T_E is the smallest portion of the vote for which candidate n *always* receives a delegate.

In general, the values of T_I and T_E depend on the number of delegates H, the number of candidates n, and the method of apportionment.

The relationship between the nth candidate's percentage of the popular vote and the cutoff c is partially described by T_I and T_E.

- If $\max\{p_n, c\} < T_I$, then candidate n would never receive a delegate, even if the candidate's vote exceeded the cutoff.
- If $T_I \leq p_n < c < T_E$, then there are instances (distributions of the other candidates' percentages of votes) for which candidate n would have received a delegate had the cutoff been lower.
- If $T_E \leq p_n < c$, then candidate n would have assuredly received a delegate under the apportionment method had c been less than or equal to p_n.

In the next section, T_I and T_E are viewed geometrically for Hamilton's method and 3 candidates; the analysis motivates the general results for any number of candidates.

The Geometry of Thresholds under Hamilton's Method and Three Candidates

Hamilton's method, proposed by and named for the American statesman Alexander Hamilton, is described in Steps 3 to 5 in **Table 1** and can be formalized as follows.

Each candidate's *quota* $q_i = Hp_i$ is calculated to determine the proportion of the delegates that each candidate should receive. Then the quota is rounded down to give candidate i an initial apportionment of $\lfloor Hp_i \rfloor$ delegates. At this point, not all of the delegates have been awarded unless Hp_i is an integer for all i. The remaining delegates

$$H - \sum_{i=1}^{n} \lfloor Hp_i \rfloor$$

are awarded, one delegate each to the candidates with the highest fractional remainders $Hp_i - \lfloor Hp_i \rfloor$, with no candidate receiving more than one additional delegate. If there are candidates tied for the largest fractional remainder, the apportionment may not be unique and in this case it is equal to the set of apportionments in which the delegate is awarded to one of the tied candidates.

To develop intuition about Hamilton's method and the values of T_I and T_E when there are 3 candidates, we follow in the footsteps of Lucas [1983], Balinski and Young [2001], and Bradberry [1992] and use the geometry of the simplex. We visualize the set of possible quotas for 3 candidates as a 2-simplex $\{(q_1, q_2, q_3) \mid q_1 + q_2 + q_3 = H; q_i \geq 0 \text{ for all } i\}$, which is the intersection of the plane $x + y + z = H$ and the positive octant. Under a particular apportionment method, the simplex can be broken into regions for which the apportionment is fixed at one (or more) 3-tuples (h_1, h_2, h_3). This is described in more detail for 5 delegates in the following example.

Example 2 (Geometric perspective of Hamilton's method for 3 candidates and 5 delegates). In **Figure 1**, we illustrate Hamilton's apportionment for $H = 5$. The simplex is divided into hexagonal regions in which each quota vector (q_1, q_2, q_3) is mapped to the apportionment (h_1, h_2, h_3) corresponding to the coordinates of the center of the hexagonal region in which the quota vector lies. Ties occur on the boundaries of the hexagonal regions.

For example, consider the apportionments shown in **Table 4** when:

1. the candidates receive 75%, 15%, and 10% of the vote respectively; and
2. the candidates receive 70%, 20%, and 10% of the vote.

As expected, in case 1) the quota vector $\otimes = (3.75, 0.75, 0.5)$ lies in the region mapped to the apportionment $(4, 1, 0)$, and in case 2) the quota vector $\square = (3.5, 1, 0.5)$ falls on the boundary of the apportionment regions for $(4, 1, 0)$ and $(3, 1, 1)$.

The quantities T_I and T_E can also be determined in **Figure 1** by examining the regions in which candidate 3 does not receive a delegate. Since the value of the q_3 coordinate of each point is a function of the height in the simplex, the smallest quota for which candidate 3 can receive a delegate is denoted by q_* and occurs at each of the points

Table 4.
Examples of Hamilton's method for 5 delegates.

Candidate	Case 1 Percent	Quota $q_i = 5p_i$	Initial App't	Final App't	Case 2 Percent	Quota $q_i = 5p_i$	Initial App't	Final App't
A	0.75	3.75	3	4	0.70	3.50	3	3 or 4
B	0.15	0.75	0	1	0.20	1.00	1	1
C	0.10	0.50	0	0	0.10	0.50	0	1 or 0

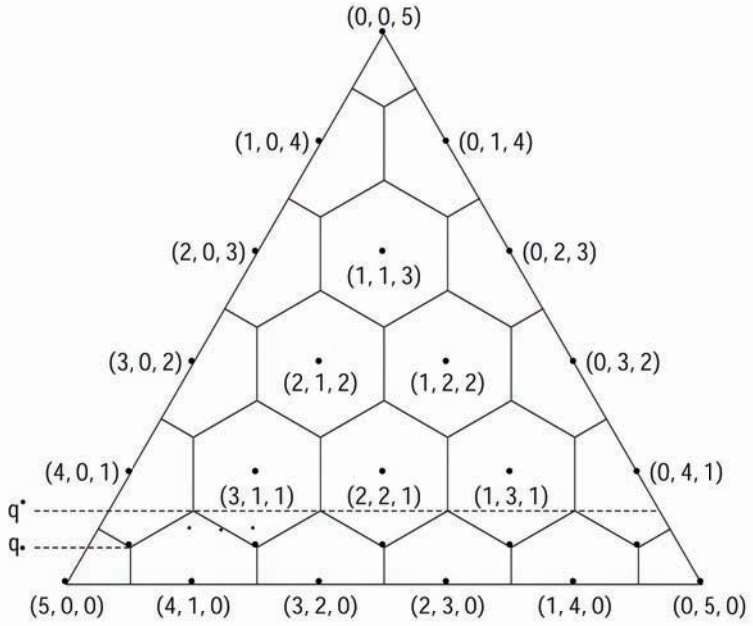


Figure 1. Hamilton apportionment for $n = 3$ and $H = 5$ with minimal quotas for candidate 3 to possibly (q_*) and to assuredly (q^*) receive a delegate.

marked by $\circ$ in the figure. From the geometry of the $30°/60°$ right triangle, it is possible to determine the coordinates of these points as (from left to right in **Figure 1**):

$$\left(\frac{13}{3}, \frac{1}{3}, \frac{1}{3}\right), \left(\frac{10}{3}, \frac{4}{3}, \frac{1}{3}\right), \left(\frac{7}{3}, \frac{7}{3}, \frac{1}{3}\right), \left(\frac{4}{3}, \frac{10}{3}, \frac{1}{3}\right), \left(\frac{1}{3}, \frac{13}{3}, \frac{1}{3}\right).$$

In each instance, we have $q_* = q_3 = 1/3$. Since $q_3 = 5p_3$, it follows that $p_3 = 1/15$. Hence, the smallest value of p_3 for which candidate 3 can receive a delegate is $T_I = 1/15$.

Geometry can also be used to determine the smallest quota for which candidate 3 is assured one of the 5 delegates. This is given by q^*, which is related to the height of the lowest line lying entirely in the regions in which candidate 3 gets at least one delegate. Elementary calculations show that $q^* = 5T_E = 2/3$ or $T_E = 2/15$.

To summarize, for the case without a cutoff, candidate 3 is assured

not to receive a delegate if $p_3 < T_I = 1/15$ and is assured to receive a delegate if $p_3 \geq T_E = 2/15$. For p_3 satisfying $T_I < p_3 < T_E$, candidate 3 may receive a delegate, depending on how the remaining vote is divided between candidates 1 and 2. This is illustrated in **Figure 1** by the points

$$\otimes = \left(\frac{15}{4}, \frac{3}{4}, \frac{2}{4}\right) = (3.75, 0.75, 0.5),$$
$$\oplus = \left(\frac{13}{4}, \frac{5}{4}, \frac{2}{4}\right) = (3.25, 1.25, 0.5).$$

Both correspond to $p_3 = 1/10$, which lies between T_I and T_E. Yet in the former, candidate 3 does not receive a delegate, since the resulting apportionment is $(4, 1, 0)$, while in the latter, candidate 3 does receive a delegate, since the resulting apportionment is $(3, 1, 1)$. □

Further insights into how Hamilton's method works can be found by subdividing the hexagons to reflect the process by which the apportionment occurs: the initial apportionment, and the sequence in which the additional delegates are awarded to the candidates. When there are 3 candidates, $H - \sum_i \lfloor p_i H \rfloor \leq 2$. Hence, the initial apportionment may fail to allocate 1 or 2 delegates. **Figure 2** shows the portion of the simplex corresponding to the region $q_1 \geq q_2 \geq q_3$ and illustrates the manner in which candidate 3 receives a delegate.

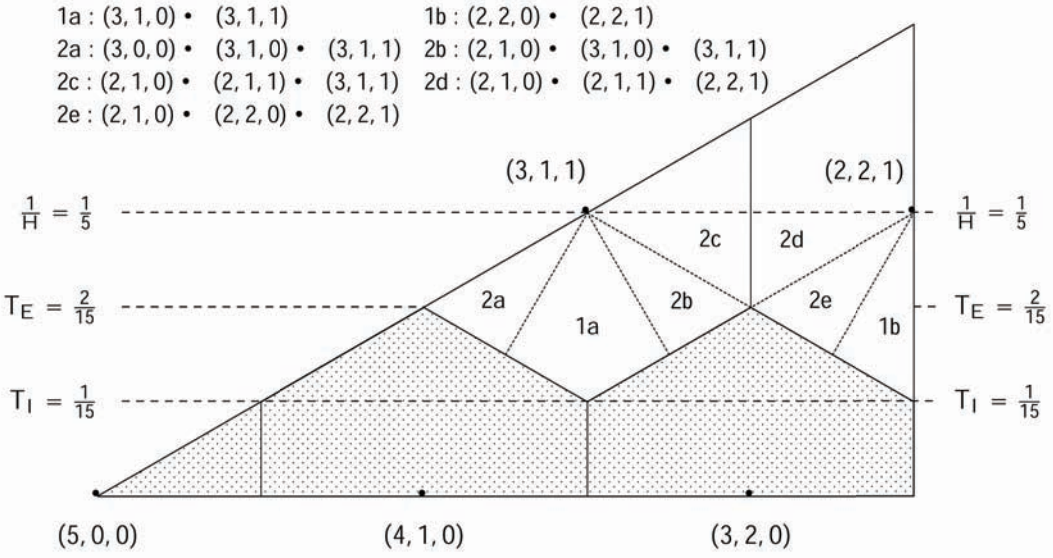


Figure 2. The process of apportionment under Hamilton's method for 5 candidates as it is related geometrically to the 2-simplex where $q_1 \geq q_2 \geq q_3$.

Because both thresholds involve minimizing p_3, we need consider only situations in which candidate 3 does not receive an initial delegate but

gains a delegate through being awarded one of the additional delegates. Hence, we can assume that $q_3 < 1$ or $p_3 < 1/5$. Of particular importance are the regions $1a$ and $1b$ in **Figure 2**, in which candidate 3 receives the only additional delegate to be awarded. These regions include the minimum quota for which candidate 3 can be awarded a delegate; this occurs at the tips of the blade-like regions where $p_3 = T_I$.

Minimal Requirements for Representation

When there are more than 3 candidates, the geometry of the simplex in higher dimensions is unwieldy and more-general arguments can be used to determine T_E and T_I directly. The following proposition is a special case of results in [Lijphart and Gibberd 1977].

Proposition 1 (Lijphart and Gibberd 1977) *Under Hamilton's method,*

$$T_I = \frac{1}{nH} \quad \textit{and} \quad T_E = \begin{cases} \dfrac{1}{H+1}, & H \le n-1; \\ \dfrac{n-1}{nH}, & H > n-1 \end{cases}$$

The 15% Cutoff

Table 5 demonstrates how T_I and T_E compare with the 15% cutoff stipulated by the Democratic Party rules for low values of H and n, which is the case for most primary elections at the district level.

Table 5.
Bimatrix entries T_I and T_E for low values of H and n.

$H \backslash n$	3		4		5		6	
2	0.167	0.333	0.125	0.333	0.100	0.333	0.083	0.333
3	0.111	0.222	0.083	0.250	0.067	0.250	0.056	0.250
4	0.083	0.167	0.063	0.188	0.050	0.200	0.042	0.200
5	0.067	0.133	0.050	0.150	0.040	0.160	0.033	0.167
6	0.056	0.111	0.042	0.125	0.033	0.133	0.028	0.139

Hamilton's method is also used to apportion delegates at the state level. The statewide popular vote is used and H is much larger. The threshold of inclusion is $T_I \le 0.15$ for all $H > 2$; hence there is a possibility under these circumstances that a candidate would be denied a delegate by the 15%

cutoff rule. Additionally, $T_E \leq 0.15$ for all but a small number of n and H. In particular, this is true if $n = 3$ and $H \geq 5$, if $n = 4$ and $H \geq 5$, if $n = 5$ and $H \geq 6$, and for all $n \geq 6$ and $H \geq n$. Hence, in these circumstances, a candidate receiving a level of support $T_E \leq p_n < 0.15$ would be denied a guaranteed delegate under the Democratic Delegate Selection Rules.

In the very early stages of the primary season, frequently there are more candidates than there are delegates, as in the 2004 New Hampshire primary, when 23 Democratic candidates competed for 7 delegates in each district race. In such a case, the threshold of exclusion $T_E = 1/(H + 1)$ does not depend on the number n and is less than 15 % as long as $H > 6$. However, with such a large number of candidates, it seems unlikely that a candidate's level of support will lie between T_E and 15%. We return to the relationship between a cutoff and thresholds, after determining the thresholds for other apportionment methods.

General Divisor Methods

In an invited address at the January 2008 Joint Mathematics Meetings in San Diego, Paul H. Edelman [2008] classified all apportionment methods into two classes: Hamilton's method and everything else (divisor methods). In this section, we consider the values of T_I and T_E for Jefferson's and Webster's methods, methods proposed by two other famous Americans, Thomas Jefferson and Daniel Webster.

The methods of Hill-Huntington and Adams, two other well-known divisor methods, have the property that each candidate receiving a positive number of votes automatically receives at least one delegate. Hence the thresholds of inclusion and exclusion for both of those methods are 0 (as long as there are sufficient delegates to go around).

Divisor methods are so-named because they are based on a *critical divisor* that roughly represents the total number of votes that each delegate represents. Under Jefferson's method, for instance, each candidate's vote total is divided by the critical divisor and the result is rounded down. If the rounded quotients sum to the total number of delegates to be apportioned, then each candidate receives the number of delegates equal to her rounded quotient. Otherwise, the divisor is adjusted until the quotients, after rounding, sum to the correct total. In general, the critical divisor will not be unique. We illustrate Jefferson's method without the 15% cutoff for the results from the 2004 New Hampshire First District.

Example 3 (Jefferson's method without cutoff). There were 104,884 votes cast and 7 delegates to be apportioned. Thus the critical divisor should be roughly $104,884/7 = 14,983$. However, as seen in **Table 6**, the divisor must be adjusted so that the quotients, after rounding down, sum to 7. In the table, we use a value of 12,000; in fact any value between 10,405 and 12,134 suffices. □

Table 6.
2004 delegate apportionment in New Hampshire's First District under Jefferson's method.

Candidates	**Votes**	$P_j/14,983$	**Rounded**	$P_j/12,000$	**Rounded**
Clark	13,304	0.88794	0	1.1087	1
Dean	24,268	1.6197	1	2.0223	2
Edwards	13,091	0.87372	0	1.0909	1
Kerry	41,530	2.7718	2	3.4608	3
Lieberman	10,404	0.69439	0	0.86700	0
Others	2,287	0.15264	0	0.19058	0
Total	104,884	7.0002	3	8.7403	7

Divisor methods differ in how the quotients are rounded. In general, a divisor method corresponds to a rounding rule f that assigns to each positive integer h a value $f(h) \in [h-1, h]$ such that there do not exist positive integers a and b for which $f(a) = a - 1$ and $f(b) = b$. The value $f(h)$ acts as a bifurcation point for which numbers between $h-1$ and $f(h)$ are rounded down to $h-1$ and numbers between $f(h)$ and h are rounded up to h. A candidate j with total level of support P_j is given h_j delegates if there exists a divisor $x > 0$ such that P_j/x is in the interval $[f(h_j), f(h_j+1)]$ for each j, where $\sum_{j=1}^{n} h_j = H$. For convenience, we define $f(0) = 0$; thus a candidate i receives no delegates if $0 \le P_j/x \le f(1)$. If $P_j/x = f(h)$ for some integer h, then candidate j may be awarded either h or $h-1$ delegates. If $f(1) = 0$, then by convention each candidate with positive share of the vote receives at least one delegate as long as $H \ge n$. If $H < n$, then the H candidates with the largest shares of the vote are each given a delegate.

The above definition is equivalent to finding a value of $x > 0$ so that

$$\frac{P_i}{f(h_i+1)} \le x \le \frac{P_j}{f(h_j)}$$

for all i and j. Hence the apportionment is $(h_1, \ldots, h_n)$ if and only if

$$\max_{h_j \ge 0} \frac{P_j}{f(h_j+1)} \le \min_{h_i > 0} \frac{P_i}{f(h_i)} \qquad \textbf{(1)}$$

where we define division by 0 (!) so that $P_i/0 < P_j/0$ if $P_i < P_j$. The apportionment and the condition in **(1)** is unchanged if each P_j is replaced by the corresponding $p_j = P_j/P$ (although the x will be different). Eisner [1992] provides an accessible introduction to divisor methods.

The rounding rules for some well-known divisor methods are shown in **Table 7**. Webster's method uses $f(h) = h - 1/2$, which amounts to what many think of as "regular rounding," in which values with decimal pieces below 0.5 are rounded down to the nearest integer and values with decimal

pieces above 0.5 are rounded up to the next integer. The functions in the right column satisfy $f(1) = 0$. For these apportionments, the threshold of inclusion is 0 and the threshold of exclusion is 0 if $H \geq n$ and $1/(H+1)$ for $H < n$.

Table 7.
Rounding functions for some well-known divisor methods.

Method	**Rounding Function**	**Method**	**Rounding Function**
Jefferson	$f(h) = h$	Adams	$f(h) = h - 1$
Webster	$f(h) = h - 1/2$	Hill-Huntington	$f(h) = \sqrt{(h-1)h}$

We end this description of divisor methods by considering what would have happened in the First District in the 2004 New Hampshire Primary if Jefferson's method had been used in conjunction with a 15% cutoff. The results are shown in **Table 8**. For ease of calculation, we use percentages p_j rather than P_j; thus, the critical divisor is scaled down proportionately so that $x = 12.300$.

Table 8.
New Hampshire's First District under Jefferson's method with a 15% cutoff.

Candidates	**Percentages**	**Adjusted Percentages**	p_j/x $= p_j/12.300$	**App't**
Clark	12.685			
Dean	23.138	36.883	2.999	2
Edwards	12.481			
Kerry	39.596	63.117	5.131	5
Lieberman	9.9195			
Others	2.1805			
Total	100	100	7	7

In general, expressions for the thresholds of inclusion and exclusion for divisor methods depend on the rounding functions and the values of n and H. The following propositions are special cases of results proved in Palomares and Ramírez [2003]. They consider the more-general thresholds that are the necessary and sufficient conditions for a candidate to receive h delegates and prove that these thresholds are the solutions to certain optimization problems, of which the specific versions for $h = 1$ are given below. Jones and Wilson [2010] solve the optimization problems for divisor methods of apportionment when the rounding function satisfies a discretized notion of concavity or convexity. That work allows general thresholds to be evaluated for classes of divisor methods, such as those based on power means; these classes include the named divisor methods of Adams, Jefferson, Webster, Hill-Huntington, and Dean.

Proposition 2 (Palomares and Ramírez 2003) *For a divisor method of apportionment with rounding function $f(h)$ where $f(1) \neq 0$, the threshold of inclusion is*

$$T_I = \min \frac{f(1)}{f(1) + \sum_{i<n} f(h_i + 1)},$$

where the minimum is taken over all nonnegative integers $h_1, \ldots, h_{n-1}$ such that $\sum_{i<n} h_i = H - 1$; and the threshold of exclusion is

$$T_E = \max \frac{f(1)}{f(1) + \sum_{i<n} f(h_i)},$$

where the maximum is taken over all nonnegative integers $h_1, \ldots, h_{n-1}$ such that $\sum_{i<n} h_i = H$.

For any divisor method satisfying $f(1) = 0$, such as the Hill-Huntington and Adams methods, T_I and T_E both reduce to 0 in **Proposition 2**.

Moreover, the rounding functions for Jefferson's and Webster's methods have the form $f(h) = h + C$, where C is a constant; hence, the values of T_I and T_E can be easily determined by substituting explicit expressions for the sums $\sum f(h_i)$ and $\sum f(h_i + 1)$. The following corollary is again a special case of a more-general result in Palomares and Ramírez [2003].

Corollary 3 (Palomares and Ramírez 2003) *Under Jefferson's method,*

$$T_I = \frac{1}{H + n - 1} \qquad \textit{and} \qquad T_E = \frac{1}{H + 1}.$$

Under Webster's method,

$$T_I = \frac{1}{2H + n - 2} \qquad \textit{and} \qquad T_E = \begin{cases} \dfrac{1}{H+1}, & H \leq n-1 \\ \dfrac{1}{2H - n + 2}, & H > n - 1. \end{cases}$$

Comparing Thresholds and Cutoffs

Because cutoffs affect apportionment methods differently, we compare T_I and T_E for different apportionment methods, beginning with the cases for which there are 2 or 3 candidates. To avoid confusion, $T_I(H)$ and $T_E(H)$ represent T_I and T_E for Hamilton's method; similarly, J and W represent apportionment by Jefferson's and Webster's methods.

When there are 2 candidates, the value of p_2 that allows candidate 2 possibly to receive a delegate and the value that assures that candidate 2

receives a delegate are the same; call this threshold $T = T_I = T_E$. Substituting $n = 2$ into the threshold equations, it follows that

$$T(H) = T(W) = \frac{1}{2H} < T(J) = \frac{1}{H+1}$$

for all $H > 1$. It is not surprising that $T(H) = T(W)$, since Hamilton's and Webster's methods are equivalent when there are 2 candidates. A similar ordering of T_I and T_E values for 3 candidates depends on H and is illustrated in **Figure 3**.

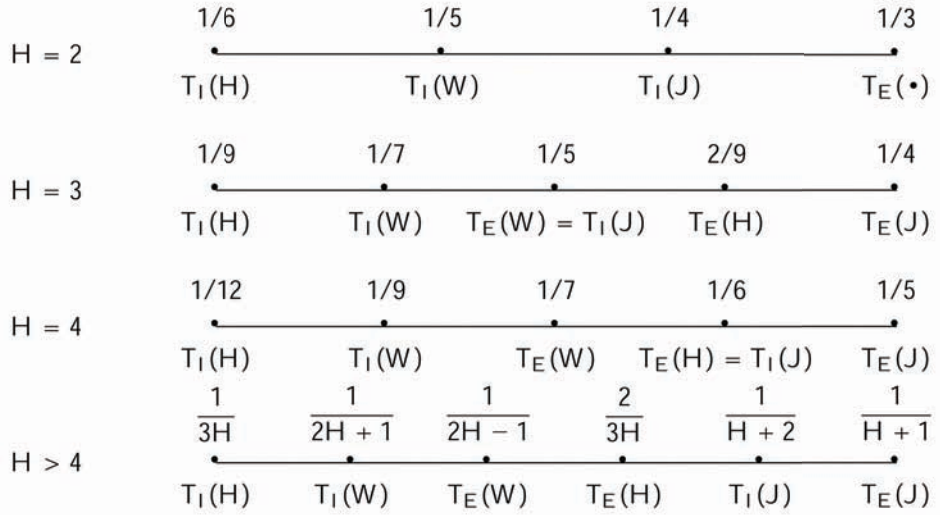


Figure 3. For $n = 3$ candidates, the not-to-scale number lines give the rank ordering of T_I and T_E for Hamilton's (H), Jefferson's (J), and Webster's (W) methods.

To relate this comparison to cutoffs, note that $T_I(H) < T_I(W) < 15\%$ for all $H > 2$ and $T_I(H) < T_I(W) < T_I(J) < 15\%$ for all $H > 4$, hence a 15% cutoff is more likely to deny a possible delegate under Hamilton's method than under Webster's or Jefferson's method. Alternatively,

$$T_E(W) < T_E(H) < T_E(J) < 15\%$$

for all $H > 5$, hence a 15% cutoff is most likely to deny an assured delegate under Webster's method. Additional scenarios can be considered by determining where 15% appears in the different number lines in **Figure 3**.

When there are more than 3 candidates, as there often are in the early stages of the primary season, these results can be generalized. Straightforward algebraic calculations show the following relationships between different methods' thresholds of inclusion and exclusion.

Proposition 4 *For $n \geq 3$ and $H > 1$, the thresholds of inclusion satisfy*

$$T_I(H) < T_I(W) < T_I(J).$$

Proposition 5 *For $n \geq 3$, the thresholds of exclusion satisfy*

$$\frac{1}{H+1} = T_E(W) = T_E(H) = T_E(J) \text{ if } H \leq n-1, \quad \text{and}$$
$$T_E(W) < T_E(H) < T_E(J) \text{ if } H > n-1.$$

Similar algebraic calculations also yield relationships among thresholds of inclusion and exclusion for different methods as given in the following two propositions, summarized in **Figure 4**.

Proposition 6 *For $n \geq 3$, $T_I(J) > T_E(W)$ if and only if $H > 2n-3$, and equality holds if $H = 2n-3$.*

Proposition 7 *For $n \geq 3$, $T_I(J) > T_E(H)$ if and only if $H > (n-1)^2$, and equality holds if $H = (n-1)^2$.*

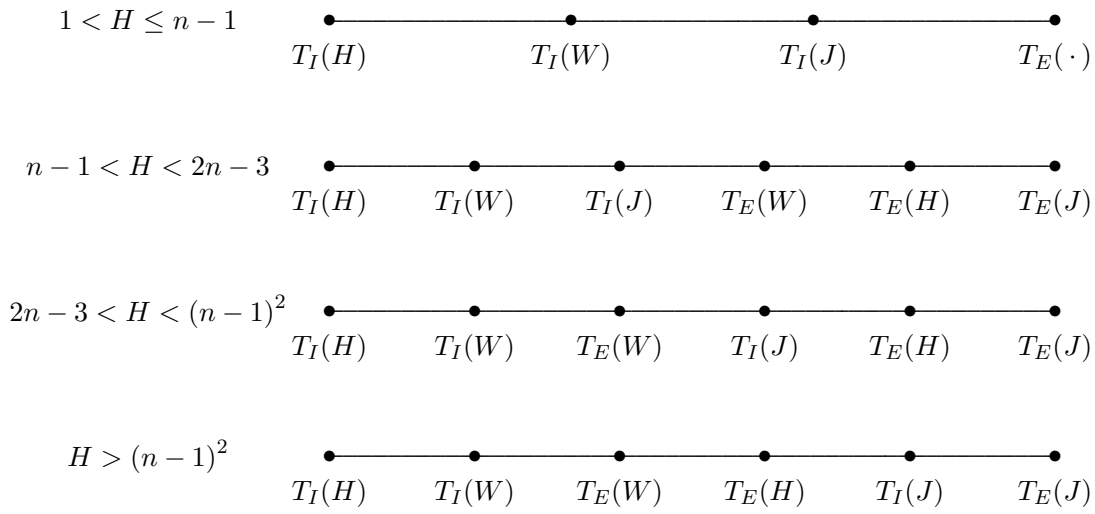


Figure 4. Not-to-scale number lines indicating the rank ordering of T_I and T_E for Hamilton's (H), Jefferson's (J), and Webster's (W) methods for $n \geq 3$. When $H = 2n-3$, we have $T_I(J) = T_E(W)$; and when $H = (n-1)^2$, we have $T_I(J) = T_E(H)$.

The observations regarding the ordering of the thresholds of inclusion and the ordering of the thresholds of exclusion for 3 candidates also hold for more candidates. In particular, if n is large relative to H, as in New Hampshire, then $T_E(\cdot) < 15\%$ for all apportionment methods. Other consequences for particular values of n and H, as well as the effect of choosing a different cutoff c, follow from an analysis of the different number lines in **Figure 4**.

The threshold of inclusion acts as a natural cutoff by stipulating a level of support necessary for a candidate to receive a delegate. Apportionment methods with higher thresholds of inclusion, such as Jefferson's method, tend to favor candidates with higher levels of support; this agrees with the

historical bias of Jefferson's method toward larger states in apportioning representatives in the U.S. House of Representatives [Balinski and Young 2001]. In fact, if $H > (n-1)^2$, both $T_E(W)$ and $T_E(H)$ are less than $T_I(J)$, showing that there is an interval of support in which a candidate may be guaranteed a delegate under Hamilton's and Webster's methods and yet not qualify for a delegate under any distribution of the other candidates' support under Jefferson's method.

The Democratic Party's choice of Hamilton's method is interesting, since Hamilton's method has the lowest threshold of inclusion among all those compared in **Figure 4**. Perhaps that is why a formal cutoff was added to the Delegate Selection Rules. The purpose of the primary season is both to vet the candidates and to help narrow the nominees' views on policy matters important to the party's constituents. A higher cutoff may eliminate candidates quickly while stifling minority views; a lower cutoff may encourage more dialogue at the expense of a more protracted campaign. Choosing the right cutoff is thus a matter of finding a balance between these extremes.

References

Balinski, Michel L., and H. Peyton Young. 2001. *Fair Representation: Meeting the Ideal of One Man, One Vote*. 2nd edition. Washington, DC: Brookings Institution Press.

Bradberry, Brent A. 1992. A geometric view of some apportionment paradoxes. *Mathematics Magazine* 65 (February) (1): 3–17.

Democratic Party of the United States. 2006. Delegate Selection Rules for the 2008 Democratic National Convention. `http://s3.amazonaws.com/apache.3cdn.net/de68e7b6dfa0743217_hwm6bhyc4.pdf`.

Edelman, Paul H. 2008. Mathematics and the law: The apportionment of the House of Representatives. Invited address. Joint Mathematics Meetings, San Diego, CA.

Eisner, Milton P. 1982. Methods of Congressional apportionment. UMAP Modules in Undergraduate Mathematics and Its Applications: Module 620. Lexington, MA: COMAP.

Geist, Kristi A., Michael A. Jones, and Jennifer M. Wilson. 2010. Apportionment in the Democratic primary process. *Mathematics Teacher* 104 (3) (October): 214–220.

Jones, Michael A., and Jennifer M. Wilson. 2010. Evaluation of thresholds for power mean-based and other divisor methods of apportionment. *Mathematical Social Sciences* 59 (3): 343–348.

Lijphart, Arend, and Robert W. Gibberd. 1977. Thresholds and payoffs in list system of proportional representation. *European Journal of Political Research* 5: 219–244.

Lucas, William F. 1983. The apportionment problem. In *Political and Related Modules*, Modules in Applied Mathematics, Volume 2, edited by Steven J. Brams, William F. Lucas, and Philip D. Straffin, Jr., Chapter 14, 358–396. New York: Springer-Verlag.

Malkevitch, Joseph. 2000. Fair legislative representation. *The UMAP Journal* 21(1): 55–71.

Palomares, Antonio, and Victoriano Ramírez. 2003. Thresholds of the divisor methods. *Numerical Algorithms* 34: 405–415.

Acknowledgments

This paper was presented in March 2009 at the Public Choice Society meetings and appeared on the associated Website. This paper benefited from comments from Bernie Grofman, Jack Nagel, and Tommy Ratliff.

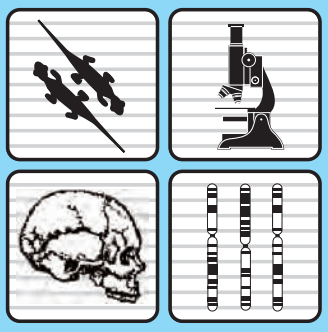
About the Authors

Michael A. Jones is an Associate Editor at *Mathematical Reviews* in Ann Arbor, MI. His research and teaching interests often center on mathematics applied to the social sciences. He has taught the content of this article as part of a two-week course (The Mathematics of Decisions, Elections, and Games) for high school students in the Michigan Math and Science Scholars program at the University of Michigan.

Jennifer Wilson is an Assistant Professor of Mathematics at Eugene Lang College of the New School for Liberal Arts, where she teaches game theory, voting theory, and other mathematical applications to the social sciences. She became interested in the Democratic Party primary while on sabbatical in the spring of 2008 when every morning's newspaper revealed new numbers to the mounting delegate count.

BioMath

Genetic Inversion



Teacher Edition

This BioMath Module, with permission to reproduce copies for classroom use, is available free of charge from COMAP to teachers. Both student and teacher versions are originally in color and at 8.5″ × 11″ page size. Please inquire for permission by phone at (800) 77–COMAP or by email at `info@comap.com`. Both versions are necessarily reproduced here in the printed issue at reduced size and in black and white. The student version is on pp. 1–19, followed by the teacher version on pp. 20–58.

The teacher version does not subsume the student version but includes

- an overview of the topics, prerequisites, target audience, time needed, standards addressed, and goals;
- a detailed timeline for lessons;
- a glossary;
- detailed plans for the lessons and instructions for the activities; and
- answer keys for the lesson activities, homework assignments, and small group activities.

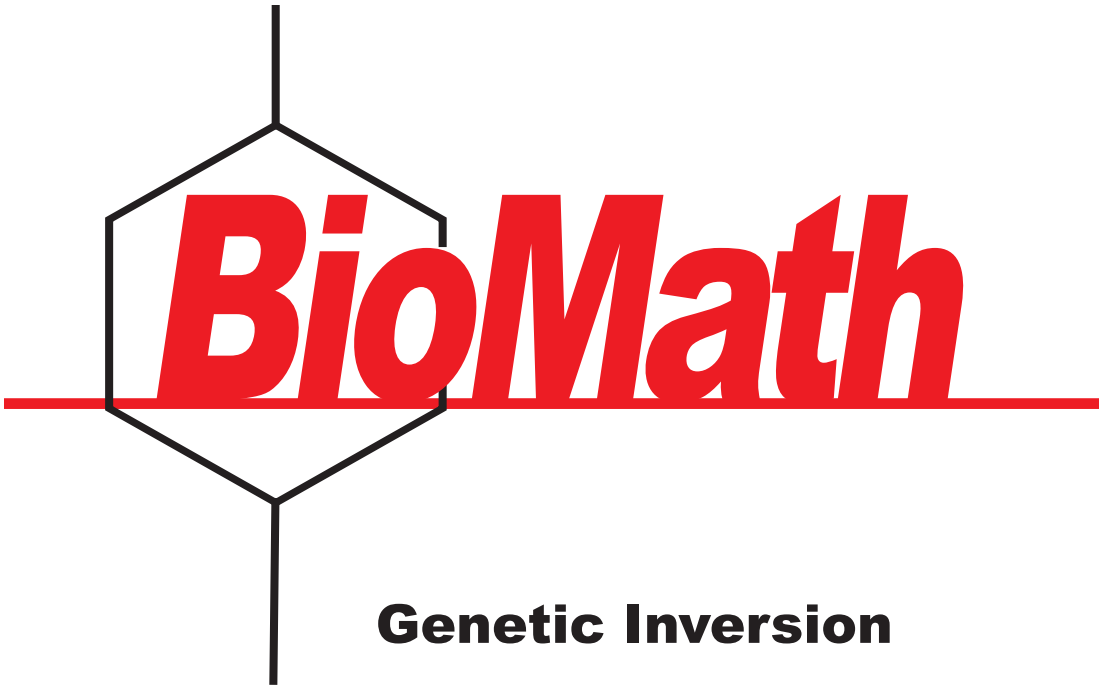
This—and other BioMath Modules under development—arose from a conference in April, 2005, held at the Center for Discrete Mathematics and Theoretical Computer Science (DIMACS) at Rutgers University, organized by Fred Roberts and Margaret Cozzens (both now at DIMACS at Rutgers University). The conference explored methods to establish connections between mathematics and biology, bringing together those who have tried it, those who have made it work on the undergraduate level, and those who know how to get new programs into the schools.

From that conference arose the NSF-sponsored Bio-Math Connection (BMC). It develops innovative classroom materials that highlight connections between mathematics and biology and helps teachers use the materials at various grade levels, in either biology or mathematics classes (or both).

The main BMC products are 24 teaching/learning Modules (including this one), together with a book (containing some of the Modules) intended for a one-semester senior-level non-calculus-based course that will satisfy part of state requirements for a fourth year of mathematics or science. These materials are developed primarily for an audience of high school students, but with little adaptation they are suitable for college students too.

BMC is seeking topics, writers, reviewers, and users for further Modules. If you are interested, please contact Prof. Cozzens at `midgec@dimacs.rutgers.edu`.

The UMAP Journal 31 (3) (2010) 215–276.



Genetic Inversion

Celeste Young
Tom Fleetwood
Paul Kehle

With contributions from:
Patrick Carney
Val DeBillis
Kathy Erickson
Patrick Flynn
Brian VanGorden

Funded by the National Science Foundation,
Proposal No. ESI-06-28091

This material was prepared with the support of the National Science Foundation. However, any opinions, findings, conclusions, and/or recommendations herein are those of the authors and do not necessarily reflect the views of the NSF.

COMAP, Inc.
175 Middlesex Turnpike
Suite 3B
Bedford, MA 01730

Lesson 1

Problem Statement

In this module, we will work on the problem of determining how closely related two different species of animals are. Because the genes located on the chromosomes of animals can be represented mathematically as sequences of numbers, we will focus on the related question of how closely related two numerical sequences are.

For example, most people would say that the first two sequences of numbers at left below are much more similar than the two sequences at right below. How can we measure how closely related two given sequences are? What is the significance of this question for biology and for mathematics? These are the questions we explore in this module.

1 3 2 4 5 6	3 4 6 5 2 1
1 2 3 4 5 6	1 2 3 4 5 6

A *subsequence* is a sequence of two or more adjacent items that are usually a smaller portion of a larger sequence; however, sometimes the entire sequence might be considered a subsequence of itself.

An *inversion* is the reversing of a subsequence within a sequence, or the reversing of the entire sequence.

Example: Given the sequence 3 2 1 5 6 4, we could invert the subsequence 3 2 1 to produce the new sequence 1 2 3 5 6 4. Next, we could invert the subsequence 5 6 to produce the new sequence 1 2 3 6 5 4. Finally, we could invert the subsequence 6 5 4 to produce the identity (or normal) sequence 1 2 3 4 5 6. We just used three inversions to change 3 2 1 5 6 4 into 1 2 3 4 5 6.

Lesson 1

Activity 1

Playing a game to gain insights into the similarity of sequences

With your partner, decide who goes first. Player A gives Player B a random sequence of six numbered cards (Ace through 6, or 1 through 6). Player B must use subsequence inversions to change the random arrangement of cards into the normal (or identity) ordering of 1 2 3 4 5 6. The number of subsequence inversions used by Player B is Player B's score for the first round. Then Player B gives Player A a random sequence of the six cards, and Player A changes the sequence into the identity ordering by using sequential inversions. The number of inversions used is Player's A score. After 3 rounds of play, each player adds up his or her score. The lowest score wins. Now play a few more times but with sequences of different lengths (choose lengths between 3 and 10).

In the table below, keep track of the lengths of your sequences and the numbers of inversions you use. Pay attention to the least and greatest numbers of inversions different pairs of sequences need to transform one sequence of the pair into the other sequence of the pair.

Length of sequences	Pairs of sequences	Numbers of inversions required

Alternate Activity 1

Lesson 1

Reverso! Playing a game to gain insights into the similarity of sequences

Studying the similarity of two sequences isn't limited to numbers. *Reverso* is an interactive computer game based on sequences of colors. The goal is to change one sequence of colors into another sequence called the identity (or target) sequence.

Reverso requires Java to be installed, usually as part of a Web browser.

Instructions: After launching the applet, minimize other windows so that you can see the work space window and the initial window asking you how long of a sequence you want to work with (between 6 and 14). Begin with a small number, such as 6. You begin with two rows of colored tiles. The top row is the identity (ID, or target sequence). The second row is the sequence you need to transform into the ID sequence. You are to click on two tiles that mark the beginning and end of the subsequence you want to invert. X's appear on the two tiles, if you make a mistake, clicking again on a tile removes its X. Once you have two X's, click on "Go!" Another row of tiles will appear that results from making the subsequence inversion you specified. The goal is to make as few inversions as possible while transforming the sequence of colors into the ID sequence. When you succeed, a message appears telling you how many swaps (or inversions) you made.

You can replay the game with the same sequences if you think you can complete the transformation in fewer inversions or with a new pair.

In the table below, keep track of the lengths of your sequences and the numbers of inversions you use. Pay attention to the least and greatest numbers of inversions different pairs of sequences need to transform one sequence of the pair into the other sequence of the pair.

Length of sequences	Pairs of sequences	Numbers of inversions required

Lesson 1

Activity 2

Biology Background Review

Answer the following questions by using the reading on the following pages. If you want more information, consult one or more of the references listed below.

1. What is a gene? What does a gene do?

2. What is the term used when a gene mutates by rotating 180°?

3. What are the three possible outcomes of a mutation?

4. Why is a mutation in a sex cell more significant than a mutation in any other type of cell?

5. Explain the relation between **centromere** location and evolution.

Biology textbooks.

The library or the Internet.

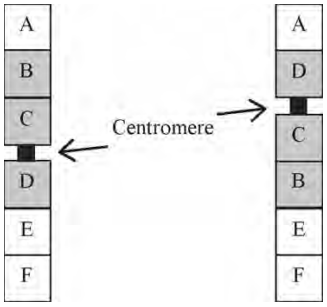
http://www.dnai.org/c/index.html

http://cstaff.hinsdale86.org/~kgabric/DIMACS/DimacsIntro.htm

Activity 2

Cells are controlled by the genetic information contained within their **DNA**. DNA is coiled and packed into a structure called a **chromosome**. A segment of the DNA/chromosome that codes for a particular **protein** is a gene. The chromosome has many genes lined up in a particular order. Another piece of the chromosome is called the **centromere**. A centromere is visualized as a constriction observed in mitotic chromosomes. The location of the centromere can be useful in determining if an inversion has occurred. **Mutations** can occur that change the order of the genes and the centromere along a chromosome. These "chromosomal mutations" include insertions, duplications, deletions, translocations and inversions. This module is only interested in inversions. An inversion occurs when a single gene or a group of genes detach from the DNA strand, rotate 180°, and reattach to the strand.

Figure 1.



Source: www.msu.edu/~lattasch/inversion.html

The diagrams above show a section of a chromosome (with letters representing arbitrary regions containing many genes), including the centromere that has been inverted, thus moving the centromere. Any genes that were within this section of the chromosome have now been moved to a new position with the exception of the potential gene at the 'pivot point.' Illustrated a different way with numbers, let us start with the gene sequence 123456. If we invert the strip '234' of the chromosome then we have the new alignment 143256. If the centromere was within the 234 segment then it will move positions, otherwise no change in centromere location will be noted. Note that genes outside the inversion area remain unchanged in their global location on the chromosome but may now be next to a gene they were not adjacent to before. In the above example, gene 1 started next to gene 2. After the inversion of 234, gene 1 is now adjacent to gene 4, even though gene 1 was not involved in the inversion.

The change in location of the centromere can indicate that an inversion has occurred. If you know the original chromosome has the centromere in the center and, after DNA replication, a copy has the centromere in a different location you can deduce that an mutation event has occurred. Again, this can be seen in **Figure 1.**

Figure 2 below shows all the different human chromosomes—note the centromere location is quite different in any given chromosome. The fact that it is not in the center is suggestive that inversions occurred throughout evolution to shift its position, and therefore the relative gene positions, which led to the human species.

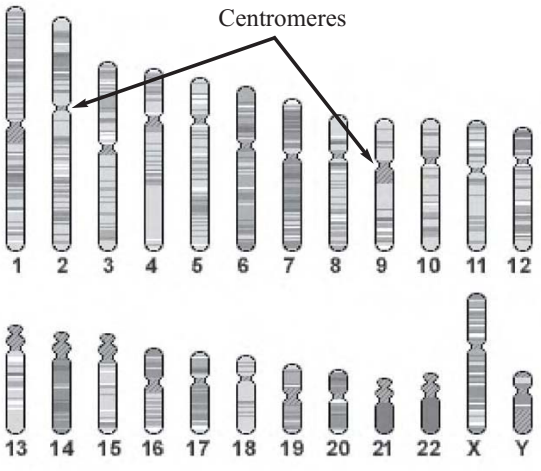


Figure 2.

Source: www.ncbi.nlm.nih.gov/Class/MLACourse/Original8Hour/Genetics/chromosome_ideograms.gif

Activity 2

Chromosomal inversions can occur in any cell in the human body (or in any living creature) but the most important events occur in sex cells (sperm or egg cells). These changes exhibit their full effect on any offspring formed from these inverted sperm or egg chromosomes. In other words, an inversion that occurs in a skin cell only affects that skin cell and its descendents, but the other trillion cells in your body remain normal. However, if the inversion occurs in a sperm or egg cell, when the cells combine to form a single cell zygote every new cell created in that offspring will contain that inverted sequence.

Chromosomal inversion events can result in 3 basic outcomes for the offspring:

Advantageous - perhaps due to the activation of a gene that was previously inhibited due to its position on the chromosome or deactivation of a gene that was previously active.

> i.e. In an arctic environment, if the inversion deactivates a gene that codes for black fur then it causes the animal to have white fur thus blending better with its environment.

Disadvantageous – same as above but either the environment was different so the color change was not preferred or the inactivated gene coded for a protein that is essential for the survival of the organism. Without the essential protein the animal will not survive. Therefore disadvantageous changes can result in an offspring that does not survive or an offspring with less chance to survive in that particular environment.

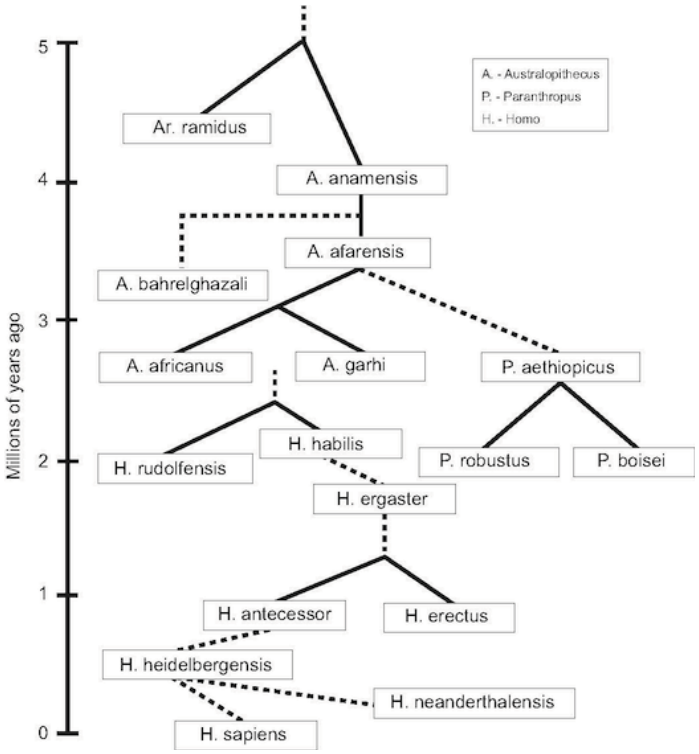
> i.e. hemophilia, a disease where an individual continues to bleed when injured, can be caused by genetic inversions on the x-chromosome. Another example of a disease that is caused by an inversion on the X chromosome is Complete Androgen Insensitivity Syndrome (CAIS). This is where a person's cells do not respond to testosterone (a type of androgen hormone) so the person has the external anatomy of a female, but the XY chromosomes of a male. The mutation on the X chromosome disrupts the gene that codes for androgen receptors, so the person does not make the receptor to cause the cells to react to the presence of testosterone.

No effect – this could occur for 2 reasons

> a. The activation or deactivation of a gene was not important to the organism's survival.
>
> b. The inversion caused no change in the gene activity – the genes are simply in a new location but work exactly the same as before the inversion.

If you are interested, ask your teacher for a copy of the ***Joint Genome Institute: Sequencing Targets and Associated Diseases*** that lists diseases that have been mapped to mutations of chromosome 5, 16 and 19. Not all of these diseases are caused by an inversion mutation but, nonetheless, the image is an excellent visual to demonstrate the importance of mutations and genetic study.

Ancestor Homework



1. Which organism is the oldest?

2. Which two are more closely related: *H. rudolfensis* and *H. erectus*, or *P. robustus* and *P. boisei?*

3. Who is the direct ancestor of *P. boisei*? Of *Au. Garhi*?

4. What does it mean if a species has no descendants?

5. What do we call the type of diagram above?

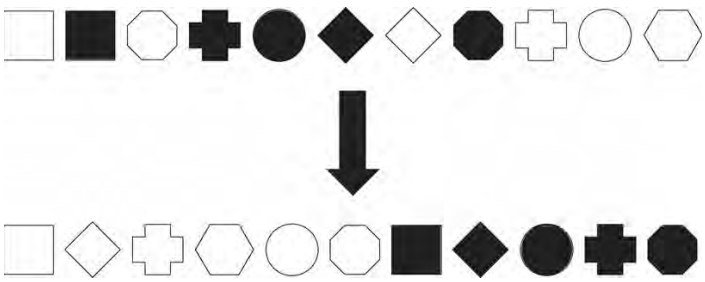
6. How do you think decisions were made regarding where to place the species?

7. What other ways could they have compared these species to place them on the tree? What way do you think would give the most accurate diagram?

Homework 2

Inversion Puzzle

Convert the top sequence into the bottom identity sequence using only inversions.



1. How many inversions did you need to make?

2. Do you think anyone could make the conversion with fewer inversions? Why or why not?

Lesson 2

Activity 1

Algorithm Activity

Definition: An **algorithm** is a step-by-step set of rules used to solve a given problem in a finite number of steps. Following an algorithm is often more successful and less frustrating than using trial and error. Try to follow the algorithm.

Following a Simple Origami Algorithm

If not already done, cut a circle out of a piece of paper.

Draw a diameter of the circle and label the endpoints A and B.

Fold the paper so that point A touches the center of the circle.

Fold the paper along the chord formed by point B and the point where the previous fold meets the circumference (on either side).

Repeat step 4 from point B to the point where the previous fold meets the circumference on the other side.

Fold one vertex of the triangle to what was the center of the circle.

Repeat step 6 for the other two vertices of the triangle.

Notice that this algorithm is very general. It will work the same way on a circle of any size. The algorithm wouldn't be very useful if it had to be changed for circles of different sizes. Good algorithms are general, efficient, and precise—and they always work, if followed correctly!

Writing an Algorithm

Now that you have followed an algorithm, you will create one for someone else to follow. Take another circle of paper and fold it into some new shape of your own design. As you fold the shape, write down precise directions for each step so that someone else could follow them and make the same shape that you made—without knowing what it is ahead of time!

After your algorithm is complete, exchange algorithms with a partner—but do NOT let your partner see your completed shape! Your partner must follow your algorithm as best as possible WITHOUT seeing your shape, and WITHOUT any help or hints from you. When you are each done following each other's algorithms, compare shapes and discuss how successful you were at both writing and following your algorithms.

Homework 1

Designing an Inversion Algorithm

Write an algorithm to transform any given sequence into its identity sequence. Your algorithm should work on sequences of any length and transform them into the sequence:

1 2 3 4 5 … *n* (*n* is the largest number in the given sequence.)

Make sure your algorithm is general, precise, and most of all, make sure it works!

Lesson 3

Activity

Small Group Discussion of Inversion Algorithms

1. In your group, exchange algorithms with someone else and apply them to the three initial sequences given below. Record each intermediate sequence. Every algorithm should transform each sequence into the identity sequence (1 2 3 4 5 6).

A) 2 3 4 5 6 1 B) 4 3 6 5 2 1 C) 1 2 3 5 4 6

2. As you use the algorithm you were given, identify a couple of things you like best about it and identify a couple of things that you think might be improved.

*

*

*

*

3. Now discuss all the algorithms in your group. Identify the one that you think is the best. What makes it better than the others? What is your definition of best?

4. After comparing all the algorithms, select and revise the best algorithm to make it even better. Perhaps you can combine ideas from other algorithms to produce a new improved algorithm. Be prepared to present and test your group's algorithm.

Homework 1

An Improved Inversion Algorithm

Background terminology: This algorithm uses the concepts of a **breakpoint**, and **increasing & decreasing strips**.

A breakpoint in a sequence of *n* numbers occurs at the following places:

before the first element of a sequence, unless the first element is 1

after the last element of a sequence, unless the last element is *n*

between any two nonconsecutive numbers (examples: 4 7 or 5 1)

The sequence 3 4 8 7 6 1 2 5 has five breakpoints, the sequence 3 2 5 1 4 6 also has five breakpoints and the sequence 3 4 7 6 5 1 2 8 has four.

Draw in the breakpoints below:

| 3 4 \| 8 7 6 \| 1 2 \| 5 \| 3 2 5 1 4 6 3 4 7 6 5 1 2 8

Breakpoints separate a sequence into subsequences called **strips**. A strip is labeled **increasing** if it contains two or more elements that increase by 1 when read from left to right (examples: 3 4 5 or 1 2). A strip is labeled **decreasing** if it contains two or more elements that decrease by 1 (example: 7 6 5 4 or 4 3).

A strip with *exactly one element and appearing at the beginning or end* of the sequence is ill**decreasing** strips.

For example, reading left to right, the sequences below contain, 3 increasing & 1 decreasing; 1 increasing & 4 decreasing; and 3 increasing & 1 decreasing strips.

Label the decreasing and increasing strips in each sequence with a D or an I:

I D I I

| 3 4 \| 8 7 6 \| 1 2 \| 5 \| | 3 2 \| 5 \| 1 \| 4 \| 6 | 3 4 \| 7 6 5 \| 1 2 \| 8

The algorithm will specify which inversions to perform according to a series of rules. For example. Given the sequence at left above, the algorithm would instruct us to invert the sequence consisting of the strips 1 2 and 5 to yield the new sequence | 3 4 | 8 7 6 5 | 2 1 | which is closer to the identity sequence because it has one fewer breakpoint.

The subsequences you will be inverting will always consist of one or more complete strips as defined by breakpoints.

1) Mark all the breakpoints in the sequence and label the strips decreasing or increasing. Refer to page 1 of this handout for help in labeling the strips.

   In following the directions below, never create a new breakpoint by breaking a strip: do not perform an inversion of a subsequence unless the subsequence consists of one or more complete strips.

2) If there is at least one decreasing strip, find the decreasing strip with the smallest number. Call this number x and do ***one*** of the following:

   a. If x is 1 and is not in the first position, invert the subsequence beginning with the first position and ending with 1. (This inversion will put 1 in the first position.)

   or, if step (a) is not possible:

   b. Invert the strip (or group of adjacent strips) that results in x and being adjacent; note: sometimes you will invert a subsequence ending with x and other times you will invert a subsequence ending with x -1.

3) If there is no decreasing strip, create one by doing ***one*** of the following—consider the three options in order beginning with (a):

   a. If 1 is not the first element, invert the first increasing strip (of length 2 or more)

   or, if (a) is not possible:

   b. If n is not the last element, invert the last increasing strip (of length 2 or more)

   or, if (b) is not possible:

   c. Invert the strip between the first and second breakpoints located after 1.

4) Repeat steps 1 through 3 with each new sequence until you have produced the identity sequence (1 2 3 4 5 … n).

Homework 1

Use the Improved Inversion Algorithm to transform the sequences below into the identity sequence. Mark the breakpoints and label the increasing and decreasing strips. Also, next to each new sequence, write down which step of the algorithm is being applied during each inversion: 2a, 2b, 3a, 3b, or 3c.

1) 5 4 3 1 2

2) 5 6 1 2 3 4

3) 4 6 3 2 1 5

4) 1 3 4 5 2 6

5) 1 4 6 7 5 3 2

6) 1 4 5 2 3 6

7) Write down any questions you have about the algorithm. Also write down anything you notice about how the algorithm affects the number of breakpoints. Finally, take note of how often each of the different steps (2a, 2b, 3a, 3b, or 3c) was used.

Lesson 4

Activity

Using and Analyzing the Improved Inversion Algorithm

Work together to make sure that every member of your group can complete and explain all of the work below.

I Compare your homework solutions. Identify any errors and correct them by explaining any differences in how the algorithm was used.

II Complete the table below, by inserting breakpoints, labeling, and counting strips. Also, make up two sequences of your own that you think are interesting.

Name	Sequence Breakpoints & Labels	# of Inc. Strips	# of Dec. Strips
A	1 4 3 2		
B	1 2 3 4 5		
C	4 3 5 2 1 6		
D	3 4 6 5 12 8 9 11 7 1 2 10		
E			
F			

III Use the Improved Inversion Algorithm to transform the following sequences into the identity sequence. Note the total number of breakpoints at each step.

1) 5 1 2 3 4 6

2) 4 5 2 1 6 3 8 7 9

3) 1 3 2 5 7 4 6 8

4) 1 2 3 6 7 8 4 5

Group Activity

5. In step 2b of the algorithm, x and $x - 1$ become adjacent. What would happen if x and $x - 1$ were adjacent to begin with?

6. Explain why step 2 always decreases the number of break points by at least 1. In what case(s) does step 2 decrease the number of break points by more than 1? More than 2?

7. What is the largest number of break points that one step can eliminate?

8. The lower bound is the smallest number of inversions necessary to invert a sequence.

Note: The brackets represent the *ceiling function* in mathematics. The ceiling function means "the closest integer greater than or equal to c". For example, if c = 5, then = 5. And if $c = 3.5$, then = 4. It is a way of rounding up to the nearest integer, if a number is not already an integer.

If b is the number of breakpoints, explain why the lower bound is equal to .

9. Calculate the lower bound and upper bound for the inversion distance in each sequence below. The upper bound is the largest number of non-repetitive inversions needed.

a) 3 2 5 6 7 4 1 8

b) 3 2 5 7 4 6 8 1

c) 4 5 2 1 6 3 8 7 9

d) 3 2 1 5 4 8 9 7 10 12 6 11

10. Transform each of these sequences into the identity sequence and draw and label the most likely phylogenetic trees for the resulting sequences. Assume that the identity sequence is the most recently evolved living species.

a) 1 2 3 6 5 4

b) 5 4 3 1 2

Lesson 5

Assessment

1) Explain why the identity sequence has no breakpoints.

2) Give an example of a sequence that has exactly two breakpoints.

3) Can a sequence have exactly one breakpoint? Explain.

4) Write a sequence with four breakpoints that has no decreasing strips.

5) Assuming that the standard alphabet is the identity sequence, find the lower bound for the inversion distance of the following sequence.

a b c d e f j k l m p q r s u v z i h g n o t w y x

6) Is it possible that a mutation by inversion is a *good* thing for an organism? Explain.

7) Draw a flowchart for the Improved Inversion Algorithm.

8) Which of the two pairs of sequences below [A & B] or [C & D] are probably most closely related? Why?

Species A: 4 3 5 6 2 1 7 8

Species B: 1 2 3 4 5 6 7 8

Species C: 4 3 2 1 8 7 6 5

Species D: 1 2 3 4 5 6 7 8

9) Create the most likely phylogenetic tree for these two species:

5 4 3 2 1 7 6 and 1 2 3 4 5 6 7

10a) Explain how the study of evolution requires an understanding of both biology and of mathematics.

10b) Give examples of other situations (not the study of evolution) where a combination of biology and mathematics is needed.

Final Case Study

Phylogenetic Tree Creation for Newly Discovered Lizard

You are a Biomath graduate student conducting research in the Amazon Rainforest. One day you are lucky enough to stumble upon a previously undiscovered creature. To name this creature (perhaps after yourself!) it is important to determine its place on the evolutionary tree of life. After sequencing a section of chromosome 6 you use a computer to look for matches to any known species. The computer finds that the genes from chromosome 6 correspond to the genes in the frillneck lizard (a currently living species) but they are not in the same order. Your task now is to create a phylogenetic tree to show the relationship between the newly discovered creature and the frillneck lizard for your local natural history museum. The museum wants a display of your findings using model creatures complete with a drawing of the most likely phylogenetic tree. The order of genes is as follows:

New creature: 1 3 2 5 7 4 6

Frillneck lizard (*Chlamydosaurus kingii)*: 1 2 3 4 5 6 7

Public domain photograph by Miklos Schiberna.

Public domain photograph by Tim Vickers.

Overview for Teacher

Topics: genetic mutations, phylogenetic trees, mathematical algorithms and optimization.

Prerequisites: relationships among DNA, genes, and chromosomes

Target audience: Grades 9–12 mathematics or biology class or the two combined.

Length: Approximately one week.

Standards Addressed:

The five NCTM process standards all figure prominently in this module:

Problem Solving Reasoning and Proof Communication
Connections Representation

National Science Education Standards

Biological Evolution
Molecular Basis of Heredity
Investigating and Analyzing Science Questions

Goal: The student will understand the role of chromosome inversion mutations in evolution.

Objective 1: The student will be able to define a chromosome inversion mutation.
Objective 2: The student will be able to diagram an inversion event.
Objective 3: The student will be able to list the 3 possible outcomes of a genetic inversion.
Objective 4: The student will be able to explain why each outcome could arise.
Objective 5: The student will be able to calculate the number of inversions necessary to transform one sequence into another.
Objective 6: The student will be able to calculate the most likely phylogeny of a set of organisms using genome inversion data.

Goal: The student will understand algorithms and algorithmic thinking, particularly in the context of optimizing some value (in this case, minimizing inversions).

Objective 1: The student will be able to explain key aspects of algorithms and develop an algorithm to solve inversion problems.
Objective 2: The student will be able to apply a given algorithm to inversion problems.
Objective 3: The student will be able to compare algorithms' efficiencies.
Objective 4: The student will be able to describe the NP-Complete Problem as it relates to algorithm efficiency.

Goal: The student will understand the need for combining biological and mathematical approaches to answering some questions.

Objective 1: The student will be able to explain how the study of evolution requires both biological and mathematical understanding.
Objective 2: The student will be able to generate examples of other situations (not the study of evolution) where a combination of biology and mathematics is needed.

Synopsis of unit: Students will apply the basic concepts of DNA and evolution to a particular kind of genetic mutation. They play a game involving the rearranging of sequences by inverting subsequences. Next, they will be challenged to develop and write an algorithm for carrying out inversions. Finally, an improved algorithm will be introduced and analyzed. Students will connect the algorithm with the concept of gene mutation, and with the evolutionary distances that separate different species of animals.

Students will struggle with certain parts of this module and this struggle is intended. The purpose is to engage students in some open-ended problems that require logical analysis and some creativity or experimentation. In these parts, the pedagogical emphasis is on the process the students are engaged in and not as much on the correctness of the answers. Following these places of open-ended struggle, concrete answers or tools are provided enabling all students to continue with the module regardless of their level of success on the more challenging parts.

Timeline

Based on a typical 45-minute class. Adapt it for local circumstances.

Lesson	Topics	Materials	Homework	Key Terms
1	• Subsequence Inversions • Chromosomes and Genes • Genetic Mutations • Possible Mutation Outcomes	• Playing cards or numbered index cards, or Reverso computer game • Handouts – Activity 1 or Reverso Alternate Activity, Activity 2 (4 pp), Ancestor Homework, Inversions Puzzle • Computers with Reverso game and Internet connection for online material (optional)	• Handout – Ancestor Homework • Inversion Puzzle Homework	• cell • centromere • chromosome • DNA • evolution • gene • genome • mathematical model • mutation • protein • sex cell • species • subsequence • zygote
2	• Inversions and Evolution • Algorithms – understanding, developing and following algorithms	• Transparencies - (1) Joint genome Institute: Targets and Associated Diseases & (2) Inversions and Rotations • Algorithm Activity & Lesson 2 Homework Handout • paper cut into ~6" circles (2-3 per student) for Algorithm Activity	• Designing an inversion algorithm (could use extra time, perhaps over a weekend, or more class time)	• algorithm • breakpoints • identity sequence
3	• Algorithms – analyzing and optimization; • Inversion Algorithm • Upper and Lower Bounds	• Handout – Small Group Discussion Handout, Homework (3 pp)	• Improved Inversion Algorithm Homework (2pages)	• optimization
4	• Analyzing the Inversion Algorithm and NP-Completeness • Inversions and Evolution • Phylogenetic Trees	• Handout – Group Activity Handout (2 pp)	• None unless extension activities or assessments are started early.	• ceiling function • evolutionary distance • lower bound • NP-Completeness • phylogenetic tree • upper bound
5	• Comprehensive Assessment	• Handout – Assessment & Final Case Study	• None, or take-home assessment tasks.	

Glossary

algorithm – a step-by-step set of rules used to solve a given problem in a finite number of steps. Following an algorithm is often more successful and less frustrating than using trial and error.

breakpoint – a location in a sequence between two non-consecutive numbers or at the ends of a sequence if the first or last numbers are not in their correct places.

ceiling function – A mathematical function that rounds up to the nearest integer.

cell – smallest unit of life; controlled by the genetic information contained within DNA.

centromere – Region of a chromosome to which the spindle fibers become associated during cell division. Often constricted in mitotic chromosomes. The location of the centromere can be useful in determining if an inversion has occurred.

chromosome – a long segment of DNA that is coiled and packed around proteins
Complete Androgen Insensitivity Syndrome (CAIS) - a genetic disorder caused by an inversion on the X chromosome that makes a person with XY chromosomes insensitive to the effects of testosterone. The person has the external anatomy of a female, but the XY chromosomes of a male.

DNA – deoxyribonucleic acid; the code of life made up of a sequence of nucleotides referred to as A, T, C, G; it has a double helix shape.

DNA replication – the process whereby the genetic information is duplicated. Each strand in double-stranded DNA serves as a template for the construction of a complimentary strand.

evolution – a change over time; biologically speaking, it is the change in species over time.

evolutionary distance – the number of mutations that separate different species.

gene – sequences of DNA nucleotides that code for a protein; several genes are lined up in a particular order on one chromosome.

genome – the complete DNA information for a organism.

identity sequence – the sequential ordering of a group of numbers beginning with 1.

inversion – 1.(math) the reversing of a subsequence within a sequence, or the reversing of the entire sequence. 2. (biology) when a single gene or a group of genes detach from DNA, rotate 180°, and reattach.

lower bound (inversion number) – the lowest number that some unknown value might be able to have.

mathematical model – a mathematical representation of some non-mathematical situation.

mutation – a change in the DNA of an organism; the result may be beneficial, harmful, or have no effect on the organism.

NP-Complete Problem – a class of problems that are very hard to solve because the best method currently known is to exhaustively consider all possible solutions where the number of possible solutions grows very very quickly as the size of the problem increases.

optimization – finding the best solution, often the goal is to minimize or maximize some quantity subject to other constraints; for example, minimizing the time to complete a task or maximizing the amount of money earned.

phylogenetic trees – a branching diagram that represents possible evolutionary relationships among species.

protein – an organic compound made of amino acid subunits; the order of amino acids is coded for by the order of nucleotides in DNA. Proteins make up skin, hair, muscle, bones, cell structures, and enzymes.

sex cell – sperm or egg cell.

species – a group of organisms that can interbreed and produce fertile offspring.

subsequence – a sequence of two or more adjacent items that are usually a smaller portion of a larger sequence; however, sometimes an entire sequence might be considered a subsequence of itself.

upper bound – the largest number that some unknown value might be able to have.

zygote - the cell that results from fertilization of a sperm and egg; it will develop into an embryo.

Lesson 1

Introduction

Introduce this module by helping students understand the statement of the problem that motivates the work in the module. Encourage students to ask questions and make conjectures about how the problem might be solved and about the relevance of the problem for biology and for mathematics.

Problem Statement—Driving Question for the Module

In this module, we will work on the problem of determining how closely related two different species of animals are. Because the genetic codes of animals can be represented mathematically as sequences of numbers, we will focus on the related question of how closely related two numerical sequences are.

Ask students what they notice about the two pairs of sequences below. For example, most people would say that the first two sequences of numbers at left are much more similar than the two sequences at right. Next, pose the questions: How can we measure how closely related two given sequences are? What is the significance of this question for biology and for mathematics?

1 3 2 4 5 6	3 4 6 5 2 1
1 2 3 4 5 6	1 2 3 4 5 6

The biological significance is that these sequences model, or represent different species of animals whose evolutionary relationship we wish to study. The mathematical significance is that determining how similar the two sequences of numbers are is an interesting mathematical problem regardless of its application to evolution. At the center of mathematics is the study of patterns and the development of conceptual tools for working with patterns. In this module, a biological problem and a mathematical problem are seen to coincide.

The final task (a case study) on the Lesson 5 Assessment is stated below and should be introduced to students now as a motivating problem for the unit. It appears alone as a page in Lesson 1 without the numerical sequences and should be given to students now. Throughout the module, it should be used as the motivating context behind the lessons and referred to if students lose sight of the goal or the purpose of the work. At the end of the module hand out the second version of the case study that contains the sequences.

You are a BioMath graduate student conducting research in the Amazon Rain Forest. One day you are lucky enough to stumble upon a previously undiscovered creature. To name this creature (perhaps after yourself!) it is important to determine its place on the evolutionary tree of life. After sequencing a section of chromosome 6 you use a computer to look for matches to any known species. The computer finds that the genes from chromosome 6 correspond to the genes in the frillneck lizard (a currently living species) but they are not in the same order. Your task now is to create a phylogenetic tree to show the relationship between the newly discovered creature and the frillneck lizard for your local Natural History Museum. The museum wants a display of your findings using model creatures complete with a drawing of the most likely phylogenetic tree.

Note

There are two student pages to choose from for Lesson 1, (**Activity 1** or **Alternate Activity 1**). One makes use of an activity based on playing cards or numbered index cards, and the other makes use of a computer game (provided). Use one activity or the other during this lesson. The other activity could be an optional reinforcement or omitted.

Topical Focus of Activity 1

Using either the numbered-card activity or the *Reverso* alternative, the purpose is to develop an understanding of how subsequence inversions can be used to change one sequence into another.

A *subsequence* is a sequence of two or more adjacent items (numerals or colors) that are usually a smaller portion of a larger sequence; however, sometimes the entire sequence might be considered a subsequence of itself.

An *inversion* is the reversing of a subsequence within a sequence, or the reversing of the entire sequence.

Example: Suppose we begin with **A 4 5 2 6 3** and want to transform it into **A 2 3 4 5 6**.
Inverting the subsequence **4 5 2** within **A 4 5 2 6 3** yields **A 2 5 4 6 3**; inverting the subsequence **6 3** within **A 2 5 4 6 3** yields **A 2 5 4 3 6**; and finally, inverting the subsequence **5 4 3** within **A 2 5 4 3 6** yields the target identity sequence **A 2 3 4 5 6**.
So we used three inversions (of subsequences) to change **A 4 5 2 6 3** into **A 2 3 4 5 6.**

Different choices of inversions and different timings of inversions can result in different numbers of inversions used to change a sequence into another.

It is also possible to invert an entire sequence. For example, even though **6 5 4 3 A 2** seems to be far from the identity sequence, it can be transformed into the identity sequence with just two inversions. First, invert **A 2** to yield **6 5 4 3 2 A**. Finally, invert the entire sequence to yield: **A 2 3 4 5 6**.

As the students engage in either Activity 1 or the alternate activity, encourage them to think about what makes a transformation easy or hard, and ask them to think about what would be the easiest and hardest possible transformations.

Instructions for Orientation to Inversions (Activity 1)

Using playing cards (or numbered index cards) lay out the **Ace** through **6** of one suit in a horizontal row, in a random order. The goal is to use inversions of subsequences to restore the sequence to the identity order: **Ace 2 3 4 5 6**.

Have students pair up and give each other random sequences to transform into the identity sequence using only subsequence inversions. As the students play the game, encourage them think about what makes a transformation easy or hard, and ask them to think about what would be the easiest and hardest possible transformations.

Instructions for the *Reverso* Game: Orientation to Inversions (Alternate Activity 1)

Reverso is a computer game designed to prepare students for the concept of inversion in gene mutations by giving them some direct experience. The goal of the game is to transform one sequence of colors into another sequence using a series of subsequence inversions. The game is distributed free of charge. Install the game on the computers before the class meets.

Playing *Reverso*!
Playing a game to gain insights into the similarity of sequences.

Studying the similarity of two sequences isn't limited to numbers. *Reverso* is an interactive computer game based on sequences of colors. The goal is to change one sequence of colors into another sequence called the identity (or target) sequence.

Reverso is a Java applet that requires Java to be installed, usually as part of a web browser. The version of Reverso supplied on CD with this module is a beta version and some options might not work.

Instructions: After launching the applet, minimize other windows so that you can see the work space window and the initial window asking you how long of a sequence you want to work with (between 6 and 14). Begin with a small number, such as 6. You begin with two rows of colored tiles. The top row is the identity (ID, or target sequence). The second row is the sequence you need to transform into the ID sequence. You are to click on two tiles that mark the beginning and end of the subsequence you want to invert. X's appear on the two tiles, if you make a mistake, clicking again on a tile removes its X. Once you have two X's, click on "Go!". Another row of tiles will appear that results from making the subsequence inversion you specified. The goal is to make as few inversions as possible while transforming the sequence of colors into the ID sequence. When you succeed, a message appears telling you how many swaps (or inversions) you made. You can replay the game with the same sequences if you think you can complete the transformation in fewer inversions. To play again with a new sequence, select "New Game" from the "Game" menu. Note that you can also have students enter a seed value to have all students work with the same sequence.

If possible, it is beneficial to have students play *Reverso* several times prior to this module outside of class; additional work with the numbered cards activity reinforces in a concrete way what takes place when subsequences are inverted. At least one of these activities should be used during this lesson.

Sample Results from Activity 1

In Activity 1 students are instructed to collect some data on the numbers of inversions it takes to transform pairs of sequences. What follows are some sample results for each version of the activity.

Keep track of the lengths of your sequences and the numbers of inversions you used. Pay attention to the least and greatest numbers of inversions different pairs of sequences needed to transform one sequence of the pair into the other sequence of the pair.

Length of Sequences	**Pairs of Sequences**	**Numbers of Inversions Required**

Perhaps students find three different ways to transform the sequences each of which takes 3 inversions. They would record their data as:

4	2 4 3 1 & 1 2 3 4	3, 3, 3

Perhaps students only find two different ways to transform the sequences and maybe one of them takes many more inversions than are needed. They would record their data as:

5	1 3 5 4 2 & 1 2 3 4 5	3, 6

Or perhaps students can find only one way to transform a sequence, recording their data as:

3	Blue Red Green & Green Red Blue	1
6	B R G O Y T & G R B T O Y	3, 3

The students' work will vary. In general encourage students to try many different pairs of sequences and to try many different lengths of sequences (between 3 and 10 or so). This work forms a concrete foundation and generates familiarity with transforming sequences that is important for the remainder of the module. In Lessons 2 and 3 the data students collect now and their experiences in this activity will help them understand the idea of finding the smallest and largest numbers of inversions needed to transform one sequence into another.

Note: Some students might have unreasonably large numbers of inversions because some of the inversions they use will "undo" and "redo" some of the progress made with prior inversions. This is not too important at this time, but at some point help them see that the goal is to find the most efficient transformations.

Activity 2 Reviewing Some Biology

Activity 2 can be completed in class as a review discussion, in small groups, or as an additional homework. The five questions help students recall the key biological concepts used in this module. The Internet could provide reference material.

Homework 1

Assign the Ancestor Homework. Students may want to use the **Biology Background Review** handout, Internet resources, textbooks, or other books to answer the six questions about the human phylogenetic tree. They, or you, might wish to consult a biology teacher.

This homework, like **Activity 2**, is designed to bring to mind students' prior knowledge of biology. Discussion in Lesson 2 can help to clarify misinformation and fill in gaps in understanding the basics of genes, mutations, evolution, and phylogenetic trees.

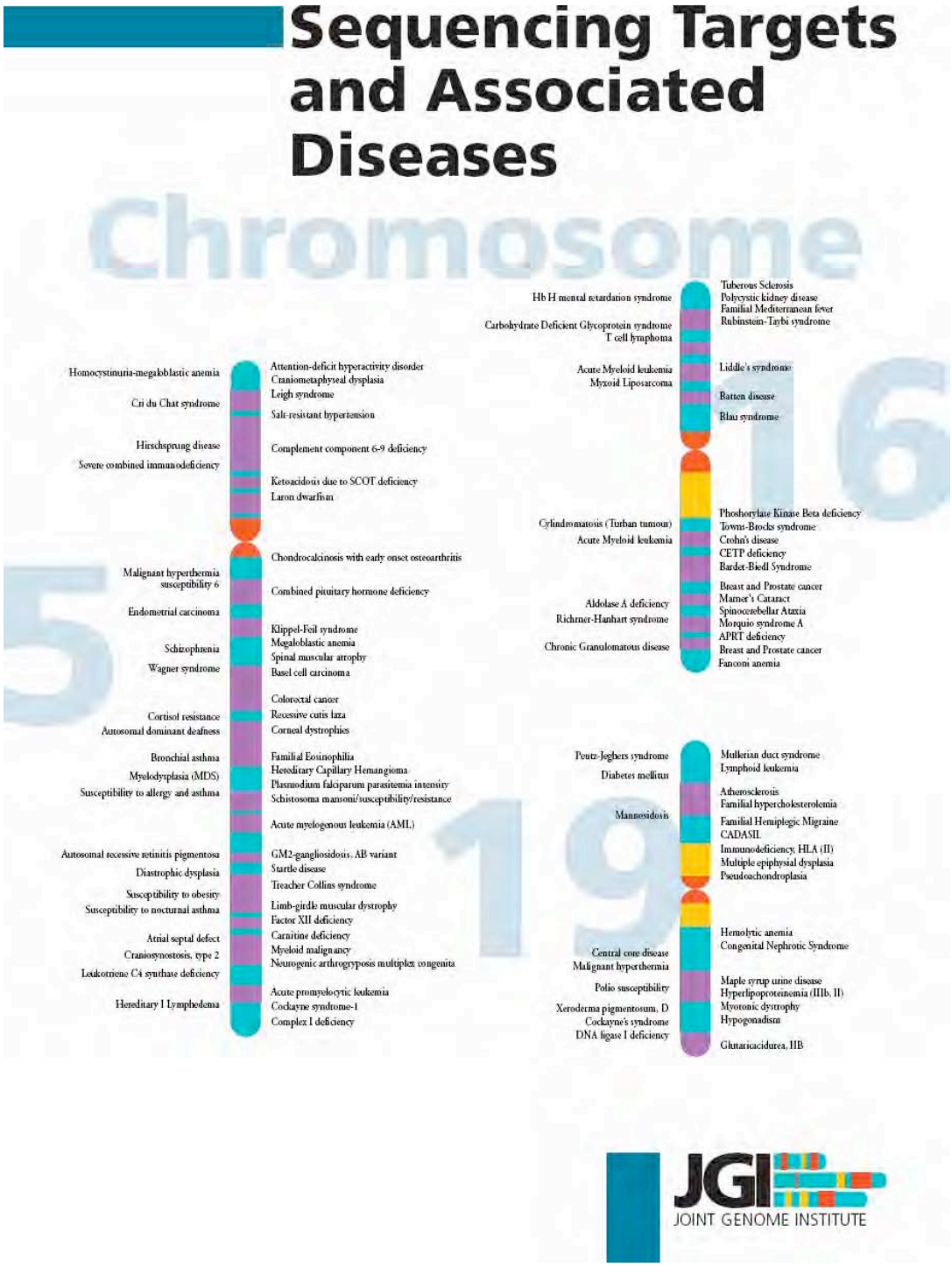
Also distribute the **Inversion Puzzle** and challenge students to solve the problem. They could begin this puzzle at the end of class if time permits. Completing Homework 1 is more important than completing Homework 2.

Note

The following page titled "Sequencing Targets and Associated Diseases" can be used in Lesson 1 or Lesson 2 as a transparency or handout in conjunction with the biology background reading or a class discussion to help students see the many diseases that have been mapped onto human chromosomes. This transparency helps connect the topic of this module with other aspects of genetics that students might have studied or will study in the future.

A larger version of this diagram is available at:

http://www.ornl.gov/sci/techresources/Human_Genome/graphics/chr51619.pdf


Sequencing Targets and Associated Diseases
Chromosome
5
Homocystinuria-megaloblastic anemia
Cri du Chat syndrome
Hirschsprung disease
Severe combined immunodeficiency
Malignant hyperthermia susceptibility 6
Endometrial carcinoma
Schizophrenia
Wagner syndrome
Cortisol resistance
Autosomal dominant deafness
Bronchial asthma
Myelodysplasia (MDS)
Susceptibility to allergy and asthma
Autosomal recessive retinitis pigmentosa
Diastrophic dysplasia
Susceptibility to obesity
Susceptibility to nocturnal asthma
Atrial septal defect
Craniosynostosis, type 2
Leukotriene C4 synthase deficiency
Hereditary I Lymphedema
Attention-deficit hyperactivity disorder
Craniometaphyseal dysplasia
Leigh syndrome
Salt-resistant hypertension
Complement component 6-9 deficiency
Ketoacidosis due to SCOT deficiency
Laron dwarfism
Chondrocalcinosis with early onset osteoarthritis
Combined pituitary hormone deficiency
Klippel-Feil syndrome
Megaloblastic anemia
Spinal muscular atrophy
Basel cell carcinoma
Colorectal cancer
Recessive cutis laxa
Corneal dystrophies
Familial Eosinophilia
Hereditary Capillary Hemangioma
Plasmodium falciparum parasitemia intensity
Schistosoma mansoni/susceptibility/resistance
Acute myelogenous leukemia (AML)
GM2-gangliosidosis, AB variant
Startle disease
Treacher Collins syndrome
Limb-girdle muscular dystrophy
Factor XII deficiency
Carnitine deficiency
Myeloid malignancy
Neurogenic arthrogryposis multiplex congenita
Acute promyelocytic leukemia
Cockayne syndrome-1
Complex I deficiency
16
Hb H mental retardation syndrome
Carbohydrate Deficient Glycoprotein syndrome
T cell lymphoma
Acute Myeloid leukemia
Myxoid Liposarcoma
Cylindromatosis (Turban tumour)
Acute Myeloid leukemia
Aldolase A deficiency
Richmer-Hanhart syndrome
Chronic Granulomatous disease
Tuberous Sclerosis
Polycystic kidney disease
Familial Mediterranean fever
Rubinstein-Taybi syndrome
Liddle's syndrome
Batten disease
Blau syndrome
Phoshorylase Kinase Beta deficiency
Towns-Brocks syndrome
Crohn's disease
CETP deficiency
Bardet-Biedl Syndrome
Breast and Prostate cancer
Marner's Cataract
Spinocerebellar Ataxia
Morquio syndrome A
APRT deficiency
Breast and Prostate cancer
Fanconi anemia
19
Peutz-Jeghers syndrome
Diabetes mellitus
Mannosidosis
Central core disease
Malignant hyperthermia
Polio susceptibility
Xeroderma pigmentosum, D
Cockayne's syndrome
DNA ligase I deficiency
Mullerian duct syndrome
Lymphoid leukemia
Atherosclerosis
Familial hypercholesterolemia
Familial Hemiplegic Migraine
CADASIL
Immunodeficiency, HLA (II)
Multiple epiphysial dysplasia
Pseudoachondroplasia
Hemolytic anemia
Congenital Nephrotic Syndrome
Maple syrup urine disease
Hyperlipoproteinemia (IIIb, II)
Myotonic dystrophy
Hypogonadism
Glutaricacidurea, IIB
JGI
JOINT GENOME INSTITUTE

Lesson 2

Class Discussion and Topical Focus

The purposes of this discussion are to answer student questions on the first Lesson's homework and to connect the concept of sequence inversions to chromosomal inversions and to evolution. Begin with an activity and then proceed to discuss questions students had on any of the work from Lesson 1 or from the homework.

Teacher-led Class Activity: Chromosomal Inversions—Details

When genes are actually inverted in an organism, they not only change position but also read 'backwards' within the chromosome. For example, in the sequence MOXIN, if the MOX is rotated 180° it would actually lead to XOWIN. Note the M is now 'upside down' to become the W. (Conveniently, the X and O look the same when rotated.)

Tape three strips of paper to the board (or use transparencies on an overhead) that contain the following capital letters:

I W O S S I NO

Challenge the students to create a word by rotating the strips of paper. If they are puzzled, show that by rotating the first and the third strips, they can form the word **OMISSION**.

This was not too difficult because the word was only broken into three pieces and the places where the breaks occurred, known as **break points**, were already given. But, what if the break points were not known? Or better yet, what if the location and the number of breakpoints were unknown? The topic of break points will return in Lessons 3 and 4.

Explain to students, using the transparency **Inversions and Rotations**, that in actual biological systems when chromosomes are "inverted" they are also rotated. Then explain that because this module is their first introduction to genetic inversions, that we will omit the rotation aspect of the problem. In this module, we will focus simply on inversions of the kind they used in class in Lesson 1. If some students are interested, urge them to pursue the more complete version of the problem as an extension to this module. A good resource is:
http://www.cse.ucsd.edu/groups/bioinformatics/GRIMM/

Making Connections: Inversions, Chromosomes, Mutation, and Evolution

Tracking chromosomal inversions is important biologically because it can help determine evolutionary phylogeny (the evolutionary development or history of related species). Mutations are the driving force of evolution and genetic inversions are one form of these mutations. By determining the most **parsimonious** (the shortest number of inversions is the most efficient) sequence of inversions to lead from the given sequence to the identity sequence, one can infer the minimum number of evolutionary steps between two organisms and the evolutionary order of a group of organisms connected by those inversions. For example, given the identity sequence 12345 for ones species and a second species' sequence of 32145, it can be seen that a single inversion might separate these two organisms. Therefore, it would be deduced that the two

species are closely related evolutionarily. On the other hand, 52314 would require three inversions to obtain the identity sequence and is therefore a more distant relative than the single inversion organism. One inversion sequence to obtain this is: 52314 – 13254 – 12354 – 12345.

Additionally, a chromosomal inversion can actually break in the middle of a gene, thus rendering that gene inoperable. For this module it is assumed that the chromosomal breaks do not break within a gene but rather shift complete genes in their entirety. And as was discussed above, in this module, we also do not pay attention to the changes in orientation (due to rotation) that accompany chromosomal inversions. In fact, some rotations do not affect gene function, so this simplification is warranted in many cases.

Discuss how the inversion homework activity, the game activity and the students' tables of data relate to genes, chromosomes, genetic inversion mutations, and evolution.

Some questions to pose to move the discussion along:
1. If a sequence of numbers represents the genes on a species' chromosome, then how is the number of inversions required to change a chromosome from one species into another species' chromosome related to the similarity of the two species?

Answer: It is likely that fewer inversions require less time for the needed mutations to take place, and that more time and more mutations are required for more inversions to take place. So we can infer (but not know for certain) that the more inversions that are present between two species' chromosomes, the more dissimilar and the more distantly related the two species are.

2. Can any sequence be transformed into the identity sequence using only inversions? Encourage students to develop a convincing argument for answering this question in the affirmative, or challenge them to provide a counter example. Encourage them to draw upon all the sequences that they have worked with in previous activities. A counter example would be a sequence that cannot be changed into the identity sequence by inversions.

Answer: Every sequence can be transformed into the identity sequence. Do not give too much of an explanation of this now as it will suggest an algorithm that students could use as an answer to tonight's homework. Students should not be steered too much toward this simple algorithm because they might be able to do better, or they might come up with the simple algorithm all on their own.

Detailed answer: After Activity 1 in Lesson 1, students should be able to convince themselves that every sequence can be transformed into the identity sequence. One easy way is to invert the sequence that places the 1 in the first place (if it is not already there), then invert the sequence that places the 2 in the second position (if it is not already there), and so on. This method is easy to understand and shows that the upper bound is n-1 inversions for a string of length n. The very last inversion will place the last *two* numbers in their correct places if they are not already there. This is the algorithm that students are likely to write for their homework. They will develop this upper bound in Lesson 3.
Example: 3 1 4 2 6 5 → 1 3 4 2 6 5 → 1 2 4 3 6 5 → 1 2 3 4 6 5 → 1 2 3 4 5 6

Inversions and Rotations

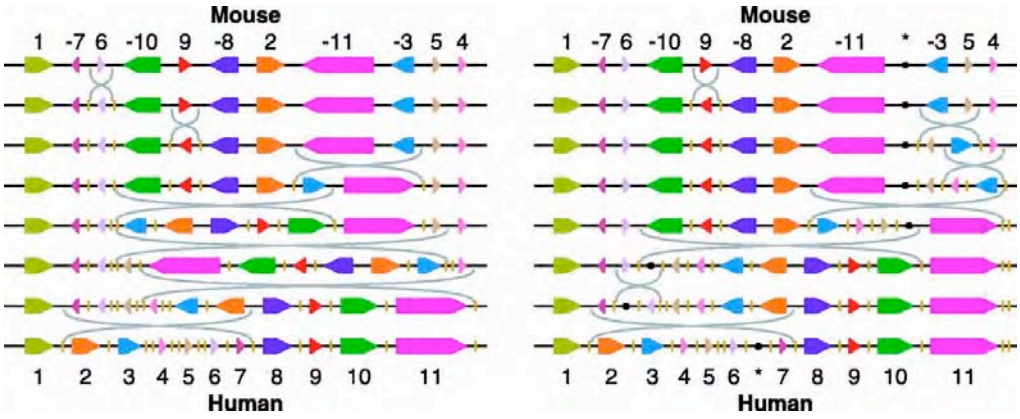
To take into account both inversions and rotations, we use negative and positive numbers.

I	W	O	S	S	I	N	O
-3	-2	-1	4	5	6	-8	-7

O	M	I	S	S	I	N	O
1	2	3	4	5	6	-8	-7

O	M	I	S	S	I	O	N
1	2	3	4	5	6	7	8

The negative signs denote a gene that is positioned 'backwards.' This notation is seen in the actual version of the mouse-to-human rearrangement shown below.



This diagram shows two ways a mouse x-chromosome can be converted to a human x-chromosome in 7 steps by simply using inversions, occurrences of which might be separated by thousands of years. Again, this module's version of the mouse to human conversion disregards these reversals of direction and simply concentrates on the sequence of numbers. In reality, a sequence of 1 2 3 4 -5 is not the same as 1 2 3 4 5. It will require one more inversion of the -5 to obtain the +5 direction. Taking both sequence and direction into account is an excellent extension of this module and involves the mathematics of graph theory. Recall however, that in this module we will not work with rotations and hence will not work with signed sequences. Work with signed sequences could make for a nice extension or independent student project.

Algorithm Activity: Introducing Algorithms

This activity is recommended unless your students are familiar with devising and writing algorithms. This activity while not directly related to genetic inversions, helps convey the nature of designing, writing, and following algorithms—issues of vital importance to all algorithms. Homework 2 is NOT optional!

Mathematics and science make use of a variety of **algorithms.** Suggest examples such as long division, solving equations, and following a science lab procedure. Ask for other examples of algorithms they have used in school. Define an algorithm as a finite list of precise steps that (if followed correctly!) results in the solution to some problem or task.

Although the final product (i.e., the final folded shape) is important, in this activity it is the processes of following and writing algorithms that are most important. Students might complain that an algorithm is hard to follow, and they should be reminded that all algorithms can be worded in different ways—and that different people might find different wordings clearer than others. Part of using or writing an algorithm is paying close attention to details.

Following a Simple Origami Algorithm

Students practice following an algorithm that transforms a circle to an equilateral triangle and then to a regular hexagon. The diagrams in Figure 1 provide you with a visual guide.

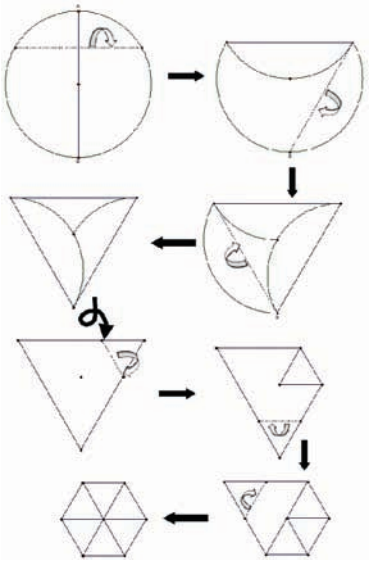


Figure 1.

Writing an Algorithm

Guide the students to create their own simple Origami algorithms and directions for making their shapes. When students are ready, pair them off to exchange algorithms. See directions on the student sheet. Remind students that they must be precise and complete, and caution them about

attempting too complicated a design! This task often reveals how much we take for granted about how another person will understand our directions.

Homework 2: Designing an Inversion Algorithm

Encourage students to be thoughtful and creative in approaching this homework. Some students might benefit from having more time or even a weekend over which to complete this work. Reassure them, that designing an algorithm is hard, but that their goal is to come up with any method of using inversions to put sequences in order. Advise them to pay attention to how they actually use inversions to solve specific problems and then to write directions that describe generally what they did in the specific instances.

Multiple answers are possible; in fact, you might emphasize that different people are likely to come up with different algorithms. In anticipation of Lesson 4, you should mention that there are some problems that despite decades of work, still have no good algorithms—either because they are not possible or because no one has been clever enough to find one yet.

Lesson 3

Topical Focus

Optimization is a recurring theme in mathematics. Often in applied problems, such as those students will encounter in algebra and calculus, the goal is to minimize or maximize some quantity subject to other constraints. For example, minimizing the time to complete a task or maximizing the amount of money earned. This lesson could be a good time to connect this module's mathematics to mathematical topics they will study later.

Small Group Discussion of Inversion Algorithms (Written in Homework 2)

Form groups of approximately 4 students. Help students follow the directions for their small group work as needed.

Dueling Algorithms

When several groups are ready, have a representative from one group use and describe his or her group's algorithm on each of the three sequences below (transforming each of them into the identity or normally increasing sequence). Repeat with any groups that have significantly different algorithms. Students could post their algorithms on chart paper around the room for comparison purposes and as displays of their work.

6 1 5 2 4 3 3 2 6 1 4 5 4 5 6 1 2 3

Closure of Lesson 3

As different algorithms are demonstrated, focus students' attention on the ideas of algorithm efficiency. Also remind students of the evolutionary context of this study of inversions. Ask what kind of algorithm would be the most useful for studying how difference species' chromosomes are related. The key insight is that we need an algorithm that will always use the *minimum* possible number of inversions to transform any given sequence into the identity sequence. An algorithm that efficiently transforms any sequence into the identity sequence using a *minimum* number of inversions is an *optimal* algorithm for the purpose of comparing chromosomes. Because evolution takes a long time, it is likely that the minimum number of inversions best measures how closely related two species are. Posing the following question and developing the chart below will help students understand the concept of an upper bound:

What is the maximum number of inversions that might be needed to transform a sequence of length n into the identity sequence? Using their table of data from Lesson 1, assist students if necessary in compiling the table below:

In the following table, n represents the length of the sequence being transformed. The next row represents the maximum number of inversions that some sequence of length n could require in being transformed into the identity sequence. It is important to note that some sequences of any given length might only require one inversion (if only two numbers in the sequence are transposed). What we are interested in here is a hunt for the worst-case sequence for a given value of n.

For example, if $n = 2$, there are only two possibilities: either the two numbers are in their correct order and no inversions are needed, or the two numbers are reversed requiring just one inversion. The entry for the table is 1 and not 0 because 1 represents the worst case for sequences of length n.

As another example, for sequences of length $n = 5$, some sequences might require zero, one, two, three, or four inversions, but no sequence of length $n = 5$ will require more than four inversions. So the worst-case scenario for sequences of length $n = 5$ is 4. Finally, with enough thought, students should convince themselves that among sequences of length n, the most inversions we would ever need is $n - 1$.

n:	2	3	4	5	6	n
Maximum # of inversions:	1	2	3	4	5	$n - 1$

Key Argument: The fact that the worst-case number of inversions is $n - 1$ is not based on the sequence of numbers in the second row of the table above. Such a pattern might not hold for larger values of n. The conclusion should be based upon the following insight. Any inversion affects at least two numbers in a sequence, and so the very last inversion performed will place at least two numbers in their correct position leaving no other numbers out of place. So if all the inversions preceding the last inversion place just one number each in place, in the worst case we will be left with only two numbers that are out of place and both of these numbers will be correctly ordered with just one inversion, meaning that we will never need an n^{th} inversion. Consider a sequence with eight elements: 1 2 3 5 4 6 7 8 that has only two numbers out of place, no matter how many inversions were required to arrive at this point, on the final inversion we get "two for the price of one". If needed, you can use the following example to help students understand where the $n - 1$ value comes from.

4 2 5 3 6 1 a sequence of length 6
1 6 3 5 2 4
1 2 5 3 6 4
1 2 3 5 6 4
1 2 3 4 6 5
1 2 3 4 5 6 Ah! We got two correct placements with just one inversion!

No matter where in the sequence the last inversion takes place, we get two correct placements. This is why the maximum number of inversions is one less than the length of the sequence we are transforming.

Concluding Observation
The maximum numbers are the *upper bounds* on the numbers of inversions. However, the goal in working with chromosome inversions is to find the *lower bound*—the minimum number of inversions needed to transform a given sequence into the identity sequence. Their homework is to study an improved algorithm for determining lower bounds for inversion problems.

See next page for Homework 3 description.

Homework 3

Assign **An Improved Inversion Algorithm** that introduces a new inversion algorithm. The challenge is for students to make sense of and apply an algorithm for solving sequence-inversion problems. If you think your students will not succeed with this task, you could work an example (provided below) in class if time permits. But for most students, the goal of this homework is for them to learn how to make sense of an algorithm by reading and interpreting carefully. In Lesson 4, in small groups they will be able to compare their experiences and help each other make sense of the algorithm. If a small group is unable to make sense, then you could use the examples below to help them understand what the steps in the algorithm mean.

Refer to Student Page **An Improved Inversion Algorithm** for Step #s.

		Inversion Step #	Breakpoints
Example A			
Given Sequence:	**4 3 2 1 7 8 5 6**		
	D I I | 4 3 2 1 | 7 8 | 5 6 | x	1	4
	1 2 3 4 | 7 8 | 5 6 |	**2a**	**3**
	I I I 1 2 3 4 | 7 8 | 5 6 |	1	
	1 2 3 4 | 7 8 | 6 5 |	**3b**	**3**
	I I D 1 2 3 4 | 7 8 | 6 5 | x	1	
	1 2 3 4 5 6 | 8 7 |	**2b**	**2**
	I D 1 2 3 4 5 6 | 8 7 | x	1	
Done!	**1 2 3 4 5 6 7 8**	**2b**	**0**

The sequence given above required 4 inversions that in turn reduced the number of breakpoints in the resulting sequences by 1, by 0, by 1, and by 2.

Worked Examples and Possible Errors

Some students might have trouble interpreting the algorithm. Below are some examples of the correct use of the algorithm and some examples of possible errors students might make. Note that the examples of what not to do are not intended steps of the algorithm; they are examples of how the algorithm might be mistakenly applied.

Example B 1 | 4 5 | 2 3 | 6

1 | 5 **4** | 2 3 | 6 — 3c
x

1 | 5 4 3 **2** | 6 — 2b — Not: 1 | 5 | 2 | 4 3 | 6 broke up a strip
x

1 2 3 4 5 6 — 2b — Not: | 3 4 5 | 1 2 | 6 broke up a strip

Example C 1 | 5 4 | 6 | 3 **2** |
x

1 2 3 | **6** | 4 5 | — 2b — Not: | 3 | 6 | 4 5 | 1 2 | broke up a strip
x

1 2 3 | 6 5 **4** | — 2b — Not: 1 2 3 4 | 6 5 | broke up a strip
x

1 2 3 4 5 6 — 2b — Whenever an inversion is the last one needed—perform that inversion!

Example D | 7 8 | 4 5 6 | 1 2 3 |

| 8 **7** | 4 5 6 | 1 2 3 | — 3a
x

| 8 7 6 5 **4** | 1 2 3 | — 2b — Not: | 8 | 5 4 | 7 6 | 1 2 3 | broke up a strip
x

| 8 7 6 5 4 3 2 1 | — 2b — Not: |8 7 6 5 | 2 1 | 4 3 | broke up a strip

1 2 3 4 5 6 7 8 — 2a

Lesson 4

Introduction

In small groups, or as a class, review the Improved Inversion Algorithm and answer any questions about the homework or on the **Analyzing the Improved Algorithm** worksheet. Tell the students that they are to become confident in using the algorithm, that they are to pay attention to what the algorithm reveals about sequence-inversion problems, and that they are to connect these ideas to the topics of genetic mutations and evolution.

Key Points

The lower bound (also called the *inversion number*) gives the smallest possible number of inversions needed to start with one sequence and end with the identity sequence. This number is called the *evolutionary distance* between the species, giving an approximation of how different the two species are.

The minimum number of inversions is the optimal number of inversions for our context of genetic evolution. In a different context, it might be the case that to optimize something, we would wish to maximize some number. What is considered optimal varies from one situation to another. In the context of genetic inversions, we want to optimize by minimizing the number of inversions because this corresponds to the fewest mutations required to transform one species into another.

The problem of determining the minimum number of inversions needed to transform one sequence into another is an incredibly challenging problem. As described in the Complexity Discussion, which you can lead at any time during Lesson 4 or 5, the general problem of creating an efficient algorithm that will always find the minimum number of inversions belongs to a notorious class of problems called NP-Complete problems. Students might be interested to know that currently there is a $1,000,000 prize for anyone who finds an efficient algorithm for these NP-Complete problems or proves that no such efficient algorithm is possible. See: http://www.claymath.org/millennium/P_vs_NP/

Complexity Discussion

Some versions of sequence-rearrangement problems are actually *very* hard optimization problems. They belong to the notorious group of problems referred to as NP-Complete. The problems in this large group are all very similar. To find optimal solutions for them requires time that grows exponentially as the sizes of the problems increase. To date, no one has found an algorithm that is faster than exhaustively checking all the possibilities—which is impossible even for relatively small realistic problems; nor has anyone found a proof that shows that a more efficient algorithm isn't possible. Perhaps the creativity of one of your students will lead to the solution of these perplexing problems.

In the case of our sequence-rearrangement problem, The Improved Inversion Algorithm presented in this module is in fact only an *approximate* algorithm. It does *not* guarantee that the number of inversions it uses to transform a sequence into the identity sequence is always the minimum possible number.

Point out to your students that much work and creative thinking needs to be done by their generation and that in addition to gaining a deeper understanding of genetic inversions, fame and possibly fortune awaits those who persist in the study of sequential rearrangement problems.

Wrapping up the Module: Making Connections and Drawing Trees

Lead the students in a class discussion of the ideas below.

Knowing the minimum number of inversions that separate two chromosomal sequences and inferring that this mathematical separation is also the biological evolutionary separation between the chromosomes allows us to construct the possible phylogenetic trees that connect the two species.

In constructing a phylogenetic tree, the nodes in the tree represent the organisms and two nodes are connected by a line only if the organisms are separated by one genetic inversion. The tree is oriented such that the most recently appearing organisms are located at the bottom of the tree; so as one moves up a tree one is moving into the evolutionary past. In this module, we do not pat attention to the lengths of the lines. The lengths of the lines represent time and would require additional information about the species to properly specify their exact positions on a timeline.

The tree with two species can only be drawn in one way with one node above the other as in Figure 2. Trees with three or more nodes could be represented by a variety of trees as shown in Figure 2.

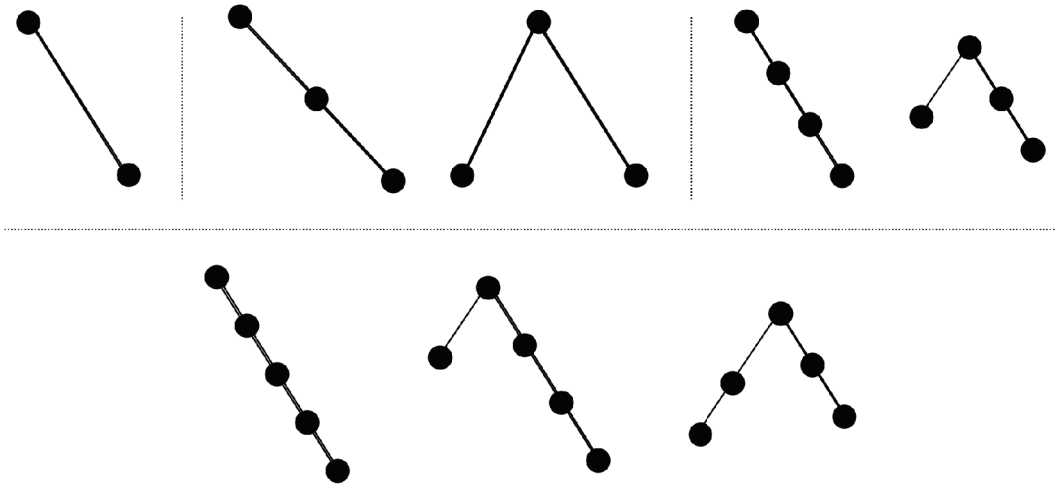


Figure 2. The possible relative arrangements of 2, 3, 4, and 5 nodes in a phylogenetic tree. The lengths of the lines are not to scale and are not important in this module.

The key to inferring which tree most likely represents the actual evolutionary relationship of the organisms involved depends upon what is known about the organisms and upon the relative chances of a series of mutations all leading to one single lineage or to a tree with multiple branching points.

Example

Consider the two sequences: 43125 and 12345.

Using the Improved Algorithm reveals that just two inversions connect these two sequences: 43125 → 43215 → 12345. So there are three possibilities for how the species represented by these sequences might be related. Two trees would have no branching points (left two trees in Figure 3) and would depend only on which species appeared first in evolutionary history. The other tree would have two twigs and one branching point (at right in Figure 3).

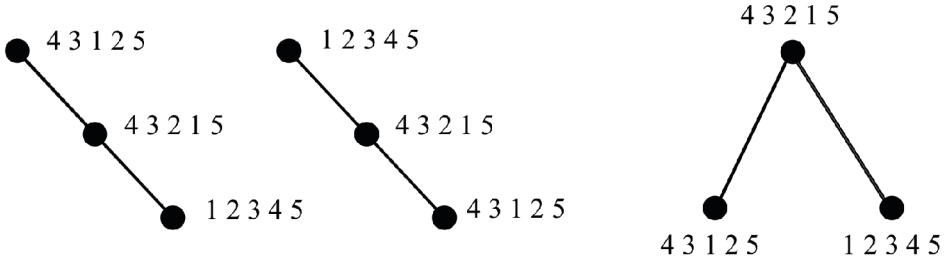


Figure 3.

Note that placing 4 3 2 1 5 at either end of a tree makes no sense because 4 3 2 1 5 is the node that connects each of the other two nodes by one inversion. Linking 1 2 3 4 5 and 4 3 1 2 5 violates our principle of assuming that only nodes separated by a single inversion should be joined by a line.

The tree at right In Figure 3 is the most likely relationship between the species. It is more likely that one species gave rise to two viable other species, each differing from the ancestor by one inversion (or mutation), than that one species gave rise to another which added an additional mutation and still yielded a viable species. In this simplistic model, it is important to note that *either* tree potentially could be the correct phylogeny; in a more complete and complex model, biologists would use additional data about the animals to gain confidence in which phylogeny they believe to be correct.

Note also that in addition to species 1 2 3 4 5, which is always assumed to be a living or extant, as we also assume about the species 4 3 1 2 5, the linking species 4 3 2 1 5 might be living or extinct. Whether this intermediate species is extant or extinct is independent of which of the phylogenetic trees we assume to be correct. Information on whether species are extinct or not comes from other sources.

Module Extension

If you have time and student interest in the topic is high, a nice extension of this module is to have the students make up 4 or more random sequences and then construct their phylogeny. To carry out this project, students will need to make some decisions about how to compare more than two sequences. For example, will all sequences be compared to one "standard" sequence, or will every pair of sequences be compared with each other and not to one standard reference, or is some hybrid comparison model justified. This open-ended problem simulates the open-ended nature of authentic science. In science, the best way to solve a problem is often unknown—scientific research does not involve "looking up the answer in the back of the book". Scientific research involves creativity and invention as well as logic.

If students want to pursue this extension, the example below might help those who are willing but stuck. The example shows how to compute the evolutionary distance between two random sequences, neither of which is the identity sequence. A way of "coding" the sequences is shown that will allow the students to use the Improved Inversion Algorithm.

Non-Identity Sequences Example

How many inversions are needed to transform 4 3 1 2 5 into 5 1 2 4 3?

Note that we can't use the Improved Inversion Algorithm because we are not working toward a sequence that is strictly increasing (1 2 3 4 5); and so the directions about increasing and decreasing strips won't make sense. For example, 5 1 2 4 3 is the target or ending sequence, and it has both increasing and decreasing strips as well as a jump from 5 to 1 and 2 to 4. This target sequence has breakpoints that should not be removed!

To get around this problem and still use the Improved Inversion Algorithm we can assign labels, or codes, to the target sequence using the identity sequence. To transform the initial sequence 4 3 1 2 5 into the target sequence 5 1 2 4 3 first code the target sequence:

5 1 2 4 3
1 2 3 4 5

Then use these codes (5—1, 1—2, 2—3, 4—4, 3—5) to code the initial sequence:

4 3 1 2 5
4 5 2 3 1

So our sequence inversion problem is now: transform 4 5 2 3 1 into 1 2 3 4 5. And because the target sequence is the identity sequence, we can now use the Improved Inversion Algorithm! Finally we decode the results to reveal the actual inversions:

Algorithm	Decoding	Actual Sequences	Actual Inversion
4 5 2 3 1	…	4 3 1 2 5	**4 3 1 2 5**
1 3 2 5 4	…	5 2 1 3 4	**2 1**
1 2 3 5 4	…	5 1 2 3 4	**3 4**
1 2 3 4 5	…	5 1 2 4 3	done.

So, 4 3 1 2 5 is separated from 5 1 2 4 3 by three inversions (indicated in bold above).

Optional Demonstration

Although the Improved Algorithm is good, professionals working on genetic inversion problems use a more sophisticated algorithm. The GRIMM program uses an algorithm that can also work with changes in direction (using signed numbers, including negative numbers, to represent directionality) as well as changes in order (the limited focus of this module). Below are two pages of directions and an example of how to use this tool to solve the unsigned inversion problems of the kind students have been working on. The following pages can also be used as handout for students wanting to extend their study of genetic inversions.

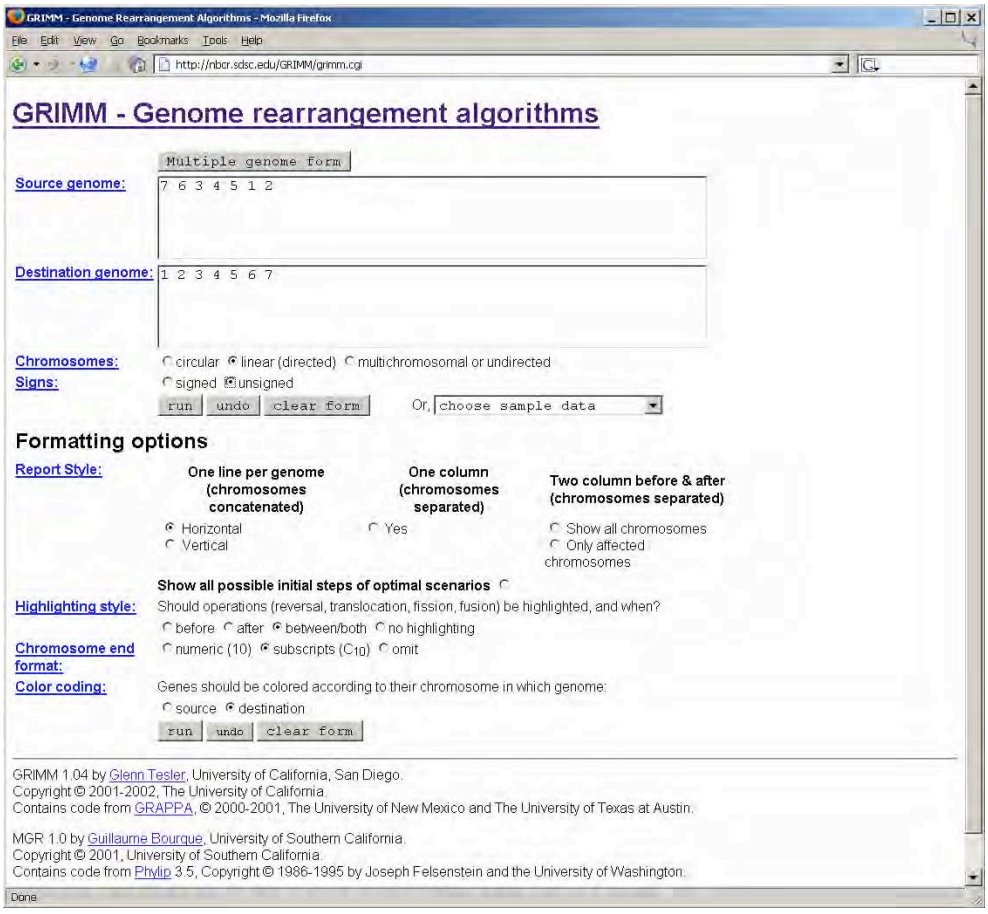
GRIMM Demonstration: Genome Rearrangements In Man and Mouse

http://nbcr.sdsc.edu/GRIMM/grimm.cgi

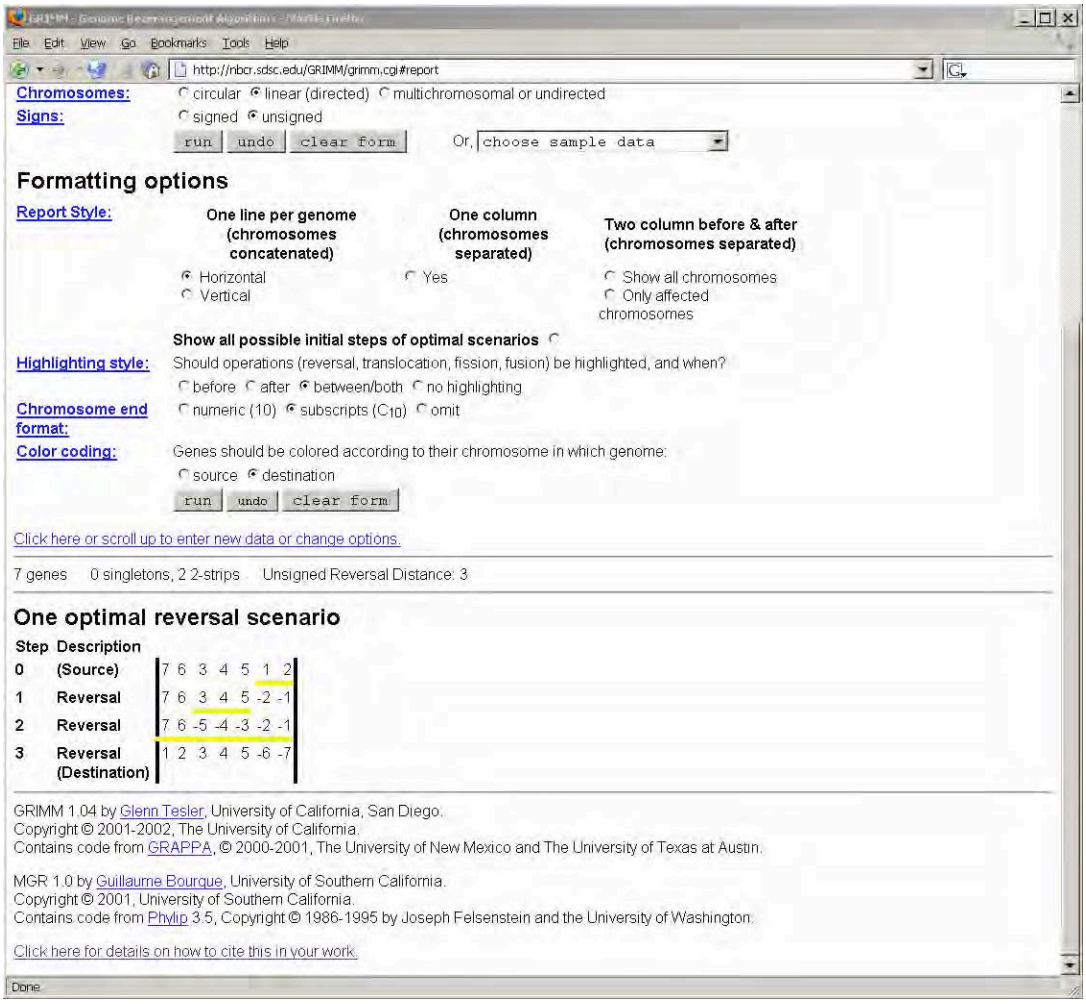
GRIMM is a powerful tool for analyzing rearrangements in pairs of genomes. It can be used to analyze reversal scenarios between pairs of genomes. To get started, open a web browser and navigate to the link above. Since the program itself is very powerful, the interface may seem fairly cluttered. However, this demonstration will only be using a few features for the purpose of this module.

Complete the following steps to determine a reversal scenario for the sequence 7 6 3 4 5 1 2.

1) Type the sequence 7 6 3 4 5 1 2 into the *Source Genome* box (be sure to separate the numbers with spaces)
2) Type the sequence 1 2 3 4 5 6 7 into the *Destination Genome* box.
3) Click the *Linear (directed)* radio button.
4) Click the *unsigned* radio button.
5) Click *Run*.



After running the program, the screen below should appear. The area of interest is the matrix at the bottom. The rows in the matrix represent intermediate steps in the evolution from the source sequence to the destination sequence. The underlined strip of numbers represents the portion of the sequence that was inverted in each step. For example, in the first step of the scenario below, the strip 1 2 was inverted. Notice that after the inversion these values became –2 –1. The negative notation indicates that the orientation of each gene has been reversed. For this module, any changes in orientation of individual genes are not considered. Therefore, disregard the sign of any number in the program output. Note that the given scenario uses three reversals. This is only one optimal scenario. There may be others.


File Edit View Go Bookmarks Tools Help
http://nbcr.sdsc.edu/GRIMM/grimm.cgi#report
Chromosomes: circular linear (directed) multichromosomal or undirected
Signs: signed unsigned
run undo clear form Or, choose sample data
Formatting options
Report Style:
One line per genome (chromosomes concatenated)
Horizontal
Vertical
One column (chromosomes separated)
Yes
Two column before & after (chromosomes separated)
Show all chromosomes
Only affected chromosomes
Show all possible initial steps of optimal scenarios
Highlighting style: Should operations (reversal, translocation, fission, fusion) be highlighted, and when?
before after between/both no highlighting
Chromosome end format: numeric (10) subscripts (C10) omit
Color coding: Genes should be colored according to their chromosome in which genome:
source destination
run undo clear form
Click here or scroll up to enter new data or change options.
7 genes 0 singletons, 2 2-strips Unsigned Reversal Distance: 3
One optimal reversal scenario
Step Description
0 (Source) 7 6 3 4 5 1 2
1 Reversal 7 6 3 4 5 -2 -1
2 Reversal 7 6 -5 -4 -3 -2 -1
3 Reversal (Destination) 1 2 3 4 5 -6 -7
GRIMM 1.04 by Glenn Tesler, University of California, San Diego.
Copyright © 2001-2002, The University of California.
Contains code from GRAPPA, © 2000-2001, The University of New Mexico and The University of Texas at Austin.
MGR 1.0 by Guillaume Bourque, University of Southern California.
Copyright © 2001, University of Southern California.
Contains code from Phylip 3.5, Copyright © 1986-1995 by Joseph Felsenstein and the University of Washington.
Click here for details on how to cite this in your work.
Done.

Lesson 5

Concluding Assessment

The assessment **What Have We Learned?** could be done individually or in small groups, or as a large group activity. It could also be done as a take-home assessment.

Assigning all the tasks might be too much to complete in one class period. Some could be used as a take-home component. Others could be used earlier in the module or skipped depending on what you want to emphasize.

If different students or groups work different sets of tasks, then they could present their work to each other so that all students benefit from all the tasks.

Note in particular that the final case study task is the culminating task related to the driving question that began this module. To vary this task students could generate their own two arbitrary non-identity sequences and create the possible phylogenetic trees for their sequences.

Concluding Discussion

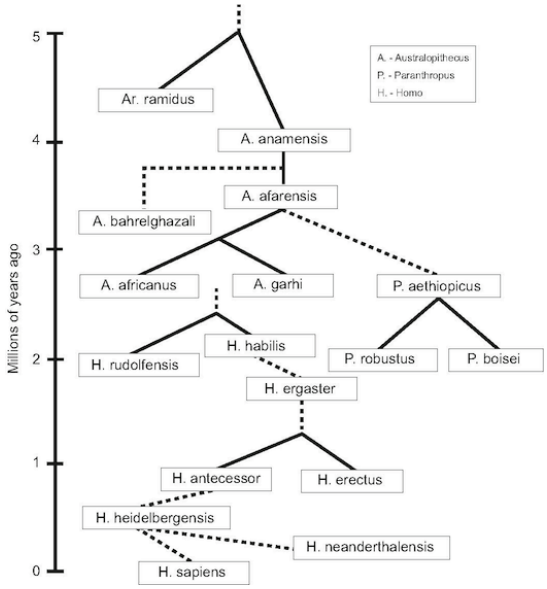
If students are having trouble seeing that some questions about genetic mutations and evolution require both biology and mathematics, help them understand the interdisciplinary nature of this module. Point out that at times it might not have been clear if they were doing biology or mathematics and that in fact, they were doing both simultaneously. In minimizing the number of inversions needed to transform a sequence, a person could simply talk about it from a mathematical perspective, but in this module, the sequence and the inversions had biological meanings. So the interest in minimizing the number of inversions was not an abstract mathematical exercise—it was driven by an understanding of how mutations help drive evolution. Encourage students to be alert for situations where a question or problem benefits from multiple perspectives or multiple disciplines. Increasingly, the most challenging problems we face in science and society at large require multiple approaches that draw from many previously separated disciplines.

Answer Key

Lesson 1 Activity 2

1. A gene is a part of chromosome that is a code for how to form a protein.
2. Inversion.
3. Advantageous, Disadvantageous, and Neutral or no effect.
4. Mutations in sex cells are passed along to the offspring of the parent.
5. Centromeres can be used to identify mutation events that might have an impact on evolution. If a centromere is found in an unexpected location within a chromosome, it is likely that a genetic inversion mutation has occurred.

Lesson 1 Ancestor Homework



1. Which organism is the oldest? *Ar(dipithecus). ramidus because it is at the top of the tree.*
2. Which two are more closely related, H. rudolfensis and H. erectus or P. robustus and P. boisei? *P. robustus and P. boisei because they are direct descendants of a common ancestor.*
3. Who is the direct ancestor of *P. boisei*? *(P. aethiopicus) Au. Garhi*? *The node is blank so we have not found that species yet*
4. What does it mean if a species has no descendants? *that species is still living – or it became extinct before it could evolve into something else*
5. What do we call the type of diagram above? A *Phylogenetic tree.*
6. How do you think they made decisions regarding where to place species on this tree? *answers may vary but will likely be based on the skull structure, or shape of some feature of the skull, or on other evidence gathered at the site*

7. What other ways could they have compared these species to place the on the tree? What way do you think would give the most accurate answer? *answers may vary but discussion should lead to the idea that we can now use genetic information to compare species*

Lesson 1 Inversion Puzzle Homework

Assign numbers 1-11 to the bottom row of shapes. Then based on this pairing, assign the corresponding number to each shape in the top row. The top row should read: 1 7 6 10 9 8 2 11 3 5 4 and the task is to turn it into the identity sequence 1 2 3 4 5 6 7 8 9 10 11.

The best answer we know of is 4 inversions: In order, invert 3 11; then invert 11 5 4; next invert 10 9 8 2 11 3 4 5; and finally invert 7 6 5 4 3 2.

Accept any plausible arguments. Students might happen upon a key insight: at most a subsequence inversion can at most correctly position two pairs of numbers that are not currently consecutive. Since there are 6 "breaks" or jumps in the original sequence, we would need at least 3 inversions. In fact the 4 at the end of the sequence is also not consecutive with what would come next were the sequence longer (the number 12), and so one more inversion is needed.

Lesson 2 Homework

Students will check each other's algorithms for completeness, precision, and effectiveness. The answers will vary, but are likely to include rather minimal statements of directions that will need to be refined. An algorithm might include directions along these lines:

1. Do the inversion that places 1 in the first position.
2. Do the inversion that places the largest number in the last position.
3. In order from smallest to largest, do the inversion that results in the largest number of numbers being correctly positioned.

Whatever the students come up with, validate their efforts but also challenge them to make their directions more precise and clear.

Lesson 3 Small Group Activity

Students will check each other's work and discuss it in their groups.

Lesson 3 Homework

The complete labelings of the three sequences on page 1 of the homework are:

```
  I     D   I   I          D  D  D  D  I          I     D     I   I
| 3 4 | 8 7 6 | 1 2 | 5 |    | 3 2 | 5 | 1 | 4 | 6    | 3 4 | 7 6 5 | 1 2 | 8
```

Solutions to problems 1 through 7 on student homework page 3 are:

1)
54312
54321 2b
12345 2a

2)
561234
651234 3a
654321 2b
123456 2a

3) 463215
123645 2a
123654 2b
123456 2b

4) 134526
125436 2b
123456 2b

5) 1467532
1235764 2b
1235467 2b
1234567 2b

6) 145236
154236 3c
154326 2b
123456 2b

7) Some of the things students might notice are that step 2b is a very common step, step 3c is a very rare, and that in general step 3 is not needed very often. This is because these steps involve only the smallest and largest numbers and their positions within the sequence. The other steps are more generic and apply to numbers regardless of their size.

Lesson 4 Group Activity: Using and Analyzing the Improved Inversion Algorithm

II

Name	Sequence Breakpoints & Labels	# of Inc. Strips	# of Dec. Strips
A	I D 1 | 4 3 2 |	1	1
B	1 2 3 4 5	0	0
C	D D D I | 4 3 | 5 | 2 1 | 6	1	3
D	I D D I D D I I | 3 4 | 6 5 | 12 | 8 9 | 11 | 7 | 1 2 | 10 |	4	4
E	student sequences will vary		
F	student sequences will vary		

III Use the Improved Inversion Algorithm to transform the following sequences into the identity sequence. Note the total number of breakpoints at each step.

1) 512346 3
543216 2
123456 0

2) 452163879 6
125463879 5
123645879 4
123654879 3
123456879 2
123456789 0

3)	13257468	6	4)	12367845	3
	12357468	5		12367854	3
	12347568	3		12345876	2
	12347658	2		12345678	0
	12345678	0			

Lesson 4 Group Activity: Using and Analyzing the Improved Inversion Algorithm

Student Page 15

Analysis of new algorithm

5. In step 2b of the algorithm, x and x - 1 become adjacent. What would happen if they were adjacent to begin with?

x - 1 would have been x or the strip with x and x - 1 would have been an increasing strip

6. Explain why step 2 always decreases the number of break points by at least 1. In what case(s) does step 2 decrease the number of break points by more than 1? More than 2?

When each of the numbers at the ends of the subsequence being inverted wind up next to consecutive numbers, then two breakpoints will be removed, and this is the only way two breakpoints can be removed. An inversion can never remove more than two breakpoints because an inversion only ever places two pairs of numbers adjacent to new numbers. All the numbers in the interior of the sequence do not change neighbors.

7. What is the largest number of break points that one step can eliminate? **2**

8. If b is the number of breakpoints, explain why the lower bound is equal to $\left\lceil \frac{b}{2} \right\rceil$.

The largest number of breakpoints that a single inversion can remove is two because only the two ends of the subsequence being inverted are affected by the inversion. So if in the best-case scenario we were able to always remove two breakpoints with an inversion then we would need b/2 inversions; but this b/2 only holds for sequences with an even number of breakpoints. For sequences with an odd number of breakpoints we would need to round b/2 up to the nexst nearest integer. For example a sequence with 9 breakpoints would require 4 inversions which at best can remove 8 breakpoints, so one more inversion would be required to remove the last breakpoint giving a total of 5 inversions or $\left\lceil \frac{9}{2} \right\rceil = 5$.

9. Calculate the lower bound for the inversion distance in each sequence below.

3 2 5 6 7 4 1 8
Lower Bound = 3

3 2 5 7 4 6 8 1
Lower Bound = 4

4 5 2 1 6 3 8 7 9
Lower Bound = 3

3 2 1 5 4 8 9 7 10 12 6 11
Lower Bound = 5

10a) 123654
\
123456

10b) 54321
/ \
54312 12345

Lesson 5 Assessment: Demonstrating What We've Learned

1) Explain why the identity sequence has no breakpoints.

All the elements are consecutive.

2) Give an example of a sequence that has exactly two breakpoints.

1 2 3 | 5 4 | 6 7 Many other examples exist.

3) Can a sequence have exactly one breakpoint? Explain.

No.

Case #1: If there is a breakpoint at an end of a sequence, then there must be an internal pair of values that differ by more than one creating another breakpoint.

Case #2: If there is a breakpoint in the interior of a sequence, then there are two values that differ by more than one. This creates two strips, one to the left of the breakpoint and one to the right. If this were the only breakpoint, then the strip to the left would have to be increasing (in order to start at one), and the strip to the right would also have to be increasing (in order to end with the largest number.) This however is impossible since this would only occur in a sequence that is already in order. Therefore there must be another breakpoint somewhere.

4) Write a sequence with four breakpoints that has no decreasing strips.

| 5 6 | 3 4 | 1 2 | Many other examples exist.

5) Assuming that the standard alphabet is the identity sequence, find the lower bound for the inversion distance of the following sequence.

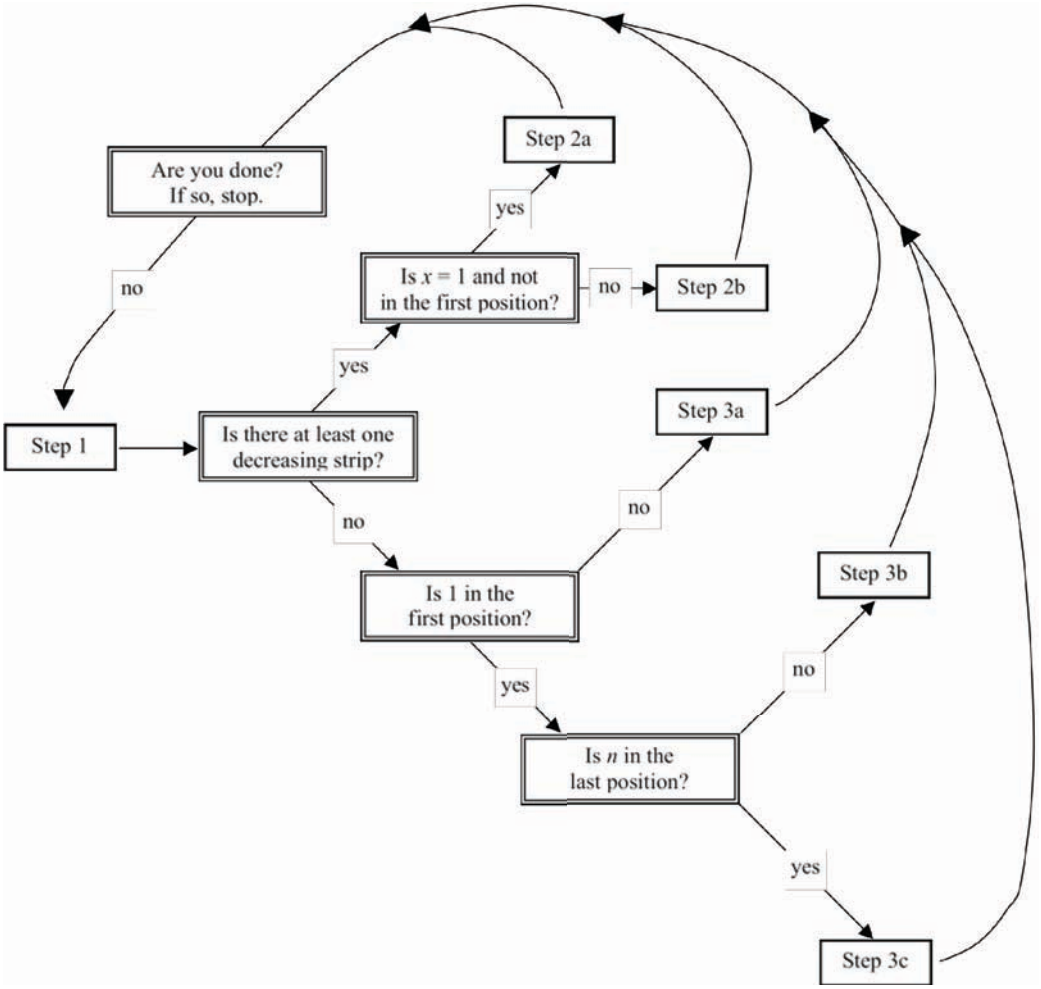
a b c d e f j k l m p q r s u v z i h g n o t w y x

a b c d e f | j k l m | p q r s | u v | z | i h g | n o | t | w | y x |
lower bound = 5

6) Is it possible that a mutation by inversion is a *good* thing for an organism? Explain.

Yes, in fact evolution relies on the fact that mutations result in organisms that are better able to survive in their environments and reproduce more effectively.

7) Draw a flowchart for the Improved Inversion Algorithm.
Students work will vary but should visually represent the written algorithm. One possible flowchart is show below.



8) Which of the two pairs of sequences below [A & B] or [C & D] are probably most closely related? Why?

Species A:	4 3 5 6 2 1 7 8	Species C:	4 3 2 1 8 7 6 5
Species B:	1 2 3 4 5 6 7 8	Species D:	1 2 3 4 5 6 7 8

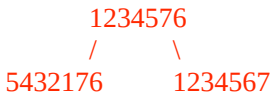
C&D require only two inversions and so are more similar than A&B which require three.

9) Create the most likely phylogenetic tree for these two species:

5 4 3 2 1 7 6 and 1 2 3 4 5 6 7

5 4 3 2 1 7 6
1 2 3 4 5 7 6
1 2 3 4 5 6 7

It is most likely that the two end sequences descended from the middle sequence:



10a) Answers will vary but should be along the lines of: To answer a question about how species are related via mutations an evolution requires biological understanding about genes, mutations, and evolution and to minimize the number of mutations that link species requires mathematical understanding of algorithms, upper and lower bounds, and optimization.

10b) Answers will vary but should make a case for how the mathematical work that is taking place has additional meaning because of what the mathematics represents. So for example, in the study of genetics, an understanding of biological reproduction is needed and an understanding of probability is needed.

Case Study: Phylogenetic Tree Creation

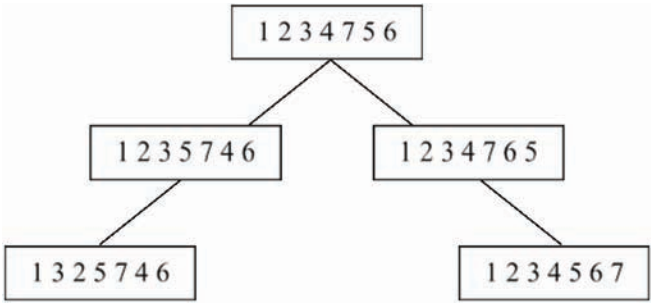
New creature: 1325746 Frillneck lizard (*Chlamydosaurus kingii)*: 1 2 3 4 5 6 7

1 3 2 5 7 4 6 = *start sequence*
1 2 3 5 7 4 6
1 2 3 4 7 5 6
1 2 3 4 7 6 5
1 2 3 4 5 6 7 = *identity sequence*

The start sequence and identity sequence represent two currently living organisms. The question is… **How related are they?** As noted from the inversion sequence above, they are only separated by four inversions. From an evolutionary perspective there are two basic interpretations of this data.

1. One sequence arose from the other through the series of inversions
2. Both sequences (start and identity) shared a common ancestor which can be located within the intermediate inversion steps. This is the more likely scenario.

If a tree is created to represent scenario 2 it might look like this:



About the Authors

Celeste Young is a life-science teacher at Monument Mountain Regional High School in Great Barrington, MA. She has been teaching biology, including anatomy and physiology, since 1998. When not in school, she enjoys taking her dog Mia for a hike, mountain bike ride, out to cross-country ski, or on a road trip.

Tom Fleetwood graduated from the University of Delaware in 1990 with a B.A.A.S. degree in biology. He then attended Life College in Marietta, Georgia, where he received a Doctor of Chiropractic degree in 1994. He began teaching at the Charter School of Wilmington in 1996 and has been teaching biology and human anatomy courses there ever since, where he is currently the science department chair. In his spare time, he likes to play chess and volleyball, and he enjoys playing the Wii with his 4-year-old daughter Kira.

Paul E. Kehle, a graduate of Beloit College and Indiana University, is Associate Professor of Mathematics Education and Associate Dean of Faculty at Hobart and William Smith Colleges in Geneva, NY. He is co-Principal Investigator on a five-year NSF grant to develop a curriculum that will engage high school students in computational thinking across all disciplines. When not teaching, designing curriculum, or photographing water, he conducts collaborative research with undergraduate students, introducing them to the frontiers of computational discrete mathematics; recent work has focused on hybrid circulant graphs and Ramsey theory.

UNITED STATES POSTAL SERVICE®

Statement of Ownership, Management, and Circulation (All Periodicals Publications Except Requester Publications)

1. Publication Title	2. Publication Number	3. Filing Date
The UMAP Journal	0197_3622	9/2/2010

4. Issue Frequency	5. Number of Issues Published Annually	6. Annual Subscription Price
Quarterly	Four (4)	$125

7. Complete Mailing Address of Known Office of Publication *(Not printer) (Street, city, county, state, and ZIP+4®)*

COMAP, Inc., 175 Middlesex Tpk., Suite 3B, Bedford, MA 01730

Contact Person: John Tomicek

Telephone *(Include area code)*: 781/862-7878 x130

8. Complete Mailing Address of Headquarters or General Business Office of Publisher *(Not printer)*

SAME

9. Full Names and Complete Mailing Addresses of Publisher, Editor, and Managing Editor *(Do not leave blank)*

Publisher *(Name and complete mailing address)*

Solomon Garfunkel, 175 Middlesex Tpk., Suite 3B, Bedford, MA 01730

Editor *(Name and complete mailing address)*

Paul Campbell, 700 College St., Beloit, WI 53511

Managing Editor *(Name and complete mailing address)*

Joyce Barnes, 175 Middlesex Tpk., Suite 3B, Bedford, MA 01730

10. Owner *(Do not leave blank. If the publication is owned by a corporation, give the name and address of the corporation immediately followed by the names and addresses of all stockholders owning or holding 1 percent or more of the total amount of stock. If not owned by a corporation, give the names and addresses of the individual owners. If owned by a partnership or other unincorporated firm, give its name and address as well as those of each individual owner. If the publication is published by a nonprofit organization, give its name and address.)*

Full Name	Complete Mailing Address
Consortium for Mathematics and	175 Middlesex Tpk., Suite 3B, Bedford, MA 01730
Its Applications, Inc. (COMAP, Inc.)	

11. Known Bondholders, Mortgagees, and Other Security Holders Owning or Holding 1 Percent or More of Total Amount of Bonds, Mortgages, or Other Securities. If none, check box → ☐ None

Full Name	Complete Mailing Address

12. Tax Status *(For completion by nonprofit organizations authorized to mail at nonprofit rates) (Check one)*

The purpose, function, and nonprofit status of this organization and the exempt status for federal income tax purposes:

☐ Has Not Changed During Preceding 12 Months

☐ Has Changed During Preceding 12 Months *(Publisher must submit explanation of change with this statement)*

13. Publication Title	14. Issue Date for Circulation Data Below
The UMAP Journal	Sep 2, 2010

15. **Extent and Nature of Circulation**			**Average No. Copies Each Issue During Preceding 12 Months**	**No. Copies of Single Issue Published Nearest to Filing Date**
a. Total Number of Copies *(Net press run)*			600	600
b. Paid Circulation (By Mail and Outside the Mail)	(1)	Mailed Outside-County Paid Subscriptions Stated on PS Form 3541(Include paid distribution above nominal rate, advertiser's proof copies, and exchange copies)	507	474
	(2)	Mailed In-County Paid Subscriptions Stated on PS Form 3541 *(Include paid distribution above nominal rate, advertiser's proof copies, and exchange copies)*	0	0
	(3)	Paid Distribution Outside the Mails Including Sales Through Dealers and Carriers, Street Vendors, Counter Sales, and Other Paid Distribution Outside USPS®	50	50
	(4)	Paid Distribution by Other Classes of Mail Through the USPS (e.g. First-Class Mail®)	0	0
c. Total Paid Distribution *(Sum of 15b (1), (2), (3), and (4))*			557	524
d. Free or Nominal Rate Distribution (By Mail and Outside the Mail)	(1)	Free or Nominal Rate Outside-County Copies included on PS Form 3541	0	0
	(2)	Free or Nominal Rate In-County Copies Included on PS Form 3541	20	38
	(3)	Free or Nominal Rate Copies Mailed at Other Classes Through the USPS (e.g. First-Class Mail)	0	0
	(4)	Free or Nominal Rate Distribution Outside the Mail *(Carriers or other means)*	0	0
e. Total Free or Nominal Rate Distribution *(Sum of 15d (1), (2), (3) and (4))*			20	38
f. Total Distribution *(Sum of 15c and 15e)*			577	562
g. Copies not Distributed *(See Instructions to Publishers #4 (page #3))*			23	38
h. Total *(Sum of 15f and g)*			600	600
i. Percent Paid *(15c divided by 15f times 100)*			97	93

16. Publication of Statement of Ownership

☒ If the publication is a general publication, publication of this statement is required. Will be printed in the Third issue of this publication.

☐ Publication not required.

17. Signature and Title of Editor, Publisher, Business Manager, or Owner	Date
	Sep 2, 2010

I certify that all information furnished on this form is true and complete. I understand that anyone who furnishes false or misleading information on this form or who omits material or information requested on the form may be subject to criminal sanctions (including fines and imprisonment) and/or civil sanctions (including civil penalties).