monomodal Gaussian distribution peaked at a metallicity of −0.8) is consistent with a bimodal distribution in color.

The reason for the nonlinearity of the color-metallicity relation goes back to the evolution of stars of different metallicities. Old evolved stars pass through a helium-burning phase (the horizontal branch of the Hertzsprung-Russell diagram), which is predominantly blue at low metallicities and becomes rapidly redder as the metallicity increases from −1.0 to −0.5. The mean colors of the less-evolved giant and dwarf stars also become redder at higher metallicities, again in a nonlinear way.

Yoon *et al.* also sort out another aspect of cluster color. The fraction of clusters in each color mode, and the mean colors of the modes, are observed to vary with the brightness of the host galaxy. These variations are easily understood in the Yoon *et al.* picture. Brighter ellipticals have higher mean metallicities than fainter ellipticals; this has been known for decades. Yoon *et al*. show how the projection of different metallicity distributions affects the predicted color distribution. As the mean metallicity decreases, the fraction of clusters in the blue mode increases, and the colors of both modes become bluer, just as observed. Similar variations within individual ellipticals can also be understood simply as a consequence of the internal radial gradients of metallicity that have also been known for many years.

The conclusion from the argument of Yoon *et al*. is that two separate epochs of globular cluster formation in ellipticals may not be needed. A single broad distribution of cluster metallicity can produce a bimodal color distribution. This makes sense because broad distributions of metallicity arise naturally in galaxies, from their continuous chemical evolution. Although the results of Yoon *et al*. do not exclude the merger origin of ellipticals, color bimodality may no longer be strong evidence for the two epochs of cluster formation that were predicted in the merger picture.

Reference and Notes

1. K. Ashman, S. Zepf, *Astrophys. J.* **384** 50 (1992).
2. Metallicity is the ratio of "metals" to hydrogen, where metals include all elements heavier than helium. It is usually expressed logarithmically relative to the Sun, so metallicities of 0 and −2 represent (1.00 and 0.01) × the solar metallicity.
3. S.-J. Yoon, S. K. Yi, Y.-W. Lee, *Science* **311**, 1129 (2006).
4. S. Zepf, K. Ashman, D. Geisler, *Astrophys. J.* **443**, 570 (1995).
5. Alternatively, some authors have argued that the metal-rich red mode of clusters are the original clusters of the underlying parent elliptical, whereas the metal-poor blue-mode clusters have been accreted from smaller in-falling galaxies.

10.1126/science.1123992

PHYSICS

Implementing a Quantum Computation by Free Falling

Quantum computers may become more powerful than conventional computers, especially at solving hard problems. Finding efficient ways to tackle such problems turns out to resemble understanding the path taken by a falling object.

Jonathan Oppenheim

There are a number of arenas where quantum resources outperform their classical counterparts, but this improvement is particularly impressive in the theory of computation. Quantum computers can efficiently solve problems that are believed to be unfeasible on a classical computer, as they would need to run exponentially longer. What type of programs can be run on a quantum computer is a question that Nielsen *et al*. attack on page 1133 of this issue (*1*). Currently, we have only a handful of quantum algorithms, of which the most noteworthy are Shor's factoring algorithm (*2*) and Grover's search algorithm (*3*). To further our understanding, one of course wants to find more problems that can be solved faster on a quantum computer, and although progress has been made, this has proven to be a difficult task.

Although it is doubtful, it could even be that quantum computers can solve all problems in the class NP—those problems whose solutions can be efficiently checked on a classical computer (*4*). If such a thing were true, it would have radical implications not only for physics but for human thought in general. We believe that writing a great poem is more difficult than recognizing one, because many can do the latter but few the former. Likewise we believe that discovering a new theory of nature, which seems to require genius, is much harder than checking the correctness of the theory, a task that many are capable of. Yet at the moment we don't have a proof of the existence of problems whose solutions can be checked efficiently on a classical computer but not solved efficiently. Nor do we have a proof that quantum computers cannot solve such NP problems. Finding such an example is one of the great tasks of classical and quantum computer science.

What a computer does when it solves a problem is to implement a mapping between inputs to the computer and a set of outputs. Thinking of this in terms of a physical operation, one sees that the quantum computer is implementing a physical mapping from initial quantum states to final states. This physical mapping between states is what we call "unitary evolution" or sometimes

Arriving at a solution. A quantum computation could be viewed as a path along a landscape of hills and valleys. The desired unitary evolution of states in the computation is represented by U. For the quickest path to the target unitary there exists a computation that runs at approximately the travel time. One wants to learn whether there is an efficient computation (polynomial time) or whether the computation is inefficient (exponential time).

CREDIT: JOE SUTLIFF

The author is in the Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge CB3 0WA, UK. E-mail: j.oppenheim@damtp.cam.ac.uk

just a "unitary." We know that almost all unitaries cannot be efficiently implemented (*5*), but we don't have an example of one. Although understanding which unitaries can be efficiently implemented has proven difficult, one might be able to relate this problem to other problems that have been more thoroughly studied, and thus gain some greater insights. This is exactly what Nielsen *et al*. have done. They link a problem in Riemannian geometry—namely, finding the shortest path between two points—to the problem of deciding whether a unitary can be implemented efficiently. This allows ideas from each of these fields of research to inspire the other.

Given a family of unitaries U that act on registers of size *n* quantum bits (or qubits), we are interested in how long it takes a quantum computer to implement these unitaries. At each step of the computation, the computer performs one of some set of elementary interactions (called a gate). If the number of steps the computer uses scales polynomially in *n*, then we say that the computation is efficient. If the number of steps scales exponentially in *n*, then the computation is not efficient. Deciding whether the computation is efficient is a matter of decomposing the unitary into the smallest number of elementary gates. This is a daunting task, because there are all kinds of ways one can make this decomposition—how do we know that we have found an optimal one?

Nielsen *et al*., building on previous work (*6*, *7*), relate this question to geometry as follows. Imagine you are sitting at the center of a surface, and your goal is to reach some other point on it that represents your target U (see the figure). The authors show that if you take the shortest path to your target, then the time of your journey is close to the time it would take for a quantum computer to implement the unitary. If your journey takes a time that grows polynomially with *n*, then there exists an efficient implementation of the unitary (and vice versa). It works roughly like this: First put coordinates on the surface to guide you on your journey; the quantum computer will take the basis of your coordinate system to correspond to particular interactions it will apply during the computation. Next, the authors endow the surface with a metric, which tells us how to measure the time our journey to the target will take. The metric they choose causes clocks to run normally if we travel along directions that correspond to elementary interactions, but causes them to run very fast if we travel along directions that correspond to more complicated interactions involving more than two qubits. This forces us to avoid paths that travel in these directions if we wish to minimize our travel time. Now we want to take the shortest distance to our target—a geodesic. Geodesics are paths that a freely falling object would take, so to make our journey optimal, we should freefall. We thus begin our journey by picking a direction and speed—but we must pick carefully if we hope to reach our target. In general, most geodesics will not pass through our target, and it may also be that there are many geodesics that pass through the target, forcing us to find the shortest one. Once we have found the shortest geodesic, Nielsen *et al*. then show that it corresponds to an implementation that approximates the desired unitary and that is of a length polynomial in the time traveled along the geodesic. This, coupled with a lower bound proof (*7*) (with the caveat that it pertains to exact implementation of the unitary without additional work space), completes the correspondence.

Finding the shortest geodesic between two points is of course a difficult problem; however, Riemannian geometry is a much more mature field than quantum computing and has the luxury of dealing with continuous paths, bringing with it all the power of differential geometry. One thus hopes that insights from it may yield some results in computation. Likewise, insights from computation might yield some surprises in Riemannian geometry. Proving that a particular unitary is difficult to implement is of great interest, so one would like to remove the caveats contained in the proof of the lower bound. Many questions are raised here. Because quantum states are closely related to quantum operations, both from a mathematical and an operational perspective, one wonders whether analogous relationships could be found for quantum states. One might also be able to relate the workings of classical computers to questions of geometry. The relationship between geometry and the implementation of unitaries promises to be, at the very least, stimulating.

References and Notes

1. M. Nielsen *et al*., *Science* **311**, 1133 (2006).
2. P. Shor, *SIAM J. Sci. Statist. Comput.* **26**, 1484 (1997).
3. L. K. Grover, in the *Proceedings of the 28th Annual ACM Symposium on the Theory of Computing* (May 1996), p. 212.
4. The NP class of problems is so-named because they are problems that can be solved by a nondeterministic Turing machine in a time that is a polynomial function of the problem size.
5. E. Knill, http://arxiv.org/abs/quant-ph/9508006.
6. Related geometric methods have been used in control theory, for example, in (*8*).
7. M. Nielsen, http://arxiv.org/abs/quant-ph/0502070; also available in *Quantum Inform. Comput.*, in press.
8. N. Khaneja, S. J. Glaser, R. Brockett, *Phys. Rev. A* **65**, 032301 (2002).

10.1126/science.1124295

MOLECULAR BIOLOGY

"X"-Rated Chromosomal Rendezvous

Laura Carrel

Female mammals inactivate one of their two X chromosomes to ensure a dosage of genes equal to that of males who contain a single X. A brief union between the pair of X chromosomes may initiate this inactivation process.

The many ways in which men and women differ are attributed to the qualitative difference in the composition of their pair of sex chromosomes—XY chromosomes versus XX chromosomes, respectively. But difference in the number of X chromosomes also poses a potential problem. In mammals, most genes on one X chromosome are inactivated in females to equalize the "dose" of X-chromosome genes between XX females and XY males. Our understanding of this process remains incomplete, but two new reports, by Xu *et al.* on page 1149 of this issue (*1*) and by Bacher *et al.* (*2*), reveal an important facet of X chromosome behavior at the onset of X chromosome inactivation.

Enhanced online at www.sciencemag.org/cgi/content/full/311/5764/1107

The initial stages of this regulatory process are quite complex [reviewed in (*3*, *4*)]. Early in mammalian development, before X chromosome inactivation occurs, each cell must calculate the number of Xs and initiate inactivation only when more than one X is present. Furthermore, embryonic X chromosome inactivation is random—some cells initially decide to inactivate their maternally inherited X chromosome while others target the paternal X chromosome. Sequences regulating these counting and choice steps reside at the X inactivation center (*Xic*), a region on the X chromosome that includes three genes that encode noncoding RNA transcripts (*3*, *5*). The *Xist* gene, expressed only from the X chromosome that will be inactivated, encodes a structural RNA that coats the inactived X

The author is in the Department of Biochemistry and Molecular Biology, Pennsylvania State University College of Medicine, Hershey, PA 17033, USA. E-mail: lcarrel@psu.edu

Published by AAAS