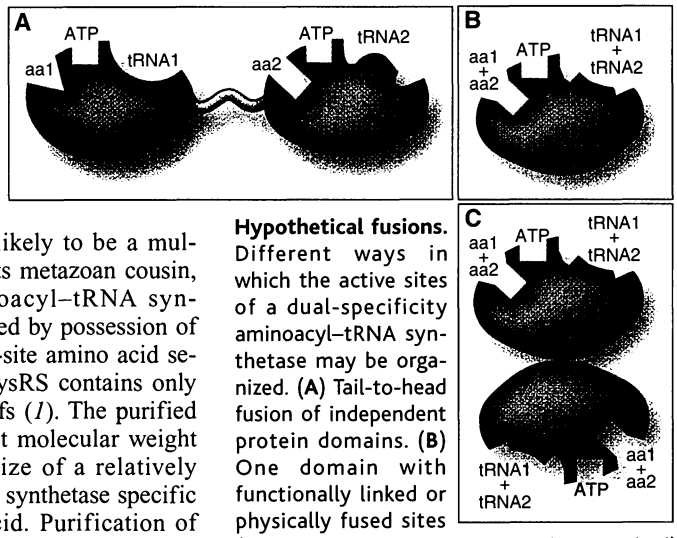
In contrast, early results on the archaeal ProCysRS suggest that it has a much more intricate fusion of enzymic activities (see the figure, panels B and C). The new archaeal molecule is unlikely to be a multidomain peptide like its metazoan cousin, GluProRS. All aminoacyl–tRNA synthetases are characterized by possession of type I or type II active-site amino acid sequences (*6*). But ProCysRS contains only one set of type II motifs (*1*). The purified protein has an apparent molecular weight of about 53 kD, the size of a relatively small aminoacyl–tRNA synthetase specific for only one amino acid. Purification of ProCysRS from an *Escherichia coli* clone makes it clear that a single subunit supports both activities. Thus, sites for two amino acids and two tRNAs must exist in close proximity or overlap within this unique small protein.



Hypothetical fusions. Different ways in which the active sites of a dual-specificity aminoacyl–tRNA synthetase may be organized. (**A**) Tail-to-head fusion of independent protein domains. (**B**) One domain with functionally linked or physically fused sites (these alternatives are not yet distinguished). (**C**) An oligomer of identical subunits with fused sites. The substrate (amino acid, aa; ATP; tRNA) for each site is shown. The notation a + b indicates that a site binds two substrates.

Cross inhibition between the specific proline and cysteine substrates also suggests linkage or overlap between their active sites. The relationship between the amino acid sites in the ProCysRS is presently not understood in detail. Envisioning an amino acid binding site with two specificities at first seems laborious. Perhaps cysteine could bind via catalyzed disulfide bond formation, thereby making minimal demands on remodeling the initial amino acid site. Any interactions between $tRNA^{Cys}$ and $tRNA^{Pro}$ sites within ProCysRS, if they exist, have not yet been studied. One intriguing observation (*1*), however, is that the presence of $tRNA^{Cys}$ is required for activation of cysteine. A few other synthetases, such as GlnRS (*7*), also have tRNA and amino acid activation sites that are strongly linked. Nonetheless, $tRNA^{Pro}$ is not required for proline activation and does not stimulate cysteine activation. So, experiments suggest that the sites in ProCysRS for activation of proline and cysteine are closely connected, and that the sites for $tRNA^{Cys}$ and cysteine are also linked. Site number, placement, and interaction are crucial because the ProCysRS must make both types of aminoacylated tRNAs in the presence of substrates that mutually inhibit each other's reactions. Perhaps a switch that minimizes interference is the binding of $tRNA^{Cys}$ to its site, with consequent conversion of the protein to a specific CysRS.

These collected properties are most easily understood if ProCysRS is a monomer (see the figure, panel B). Competition for one set of overlapping active sites then explains how the aminoacyl–tRNA synthetase activities mutually inhibit eachother. However, aminoacyl–tRNA synthetases are frequently oligomers. If ProCysRS is a homo-oligomer, then cross-inhibition implies communication between identical amino acid sites on different oligomers. In this case (see the figure, panel C), the extensive aminoacyl–tRNA synthetase literature offers many other possibilities for how the dual activities of ProCysRS could be coordinated. For example, one set of sites in an oligomeric enzyme can be active despite apparently identical inactive sites elsewhere in the oligomer (half-of-the-sites reactivity) (*8*). Either simple inhibition is transmitted between sites or, more surprisingly, a change in specificity to that matching the first-occupied site could be transmitted. The spreading effect of $tRNA^{Cys}$ on the activation of cysteine, and the dramatic effect of nucleotide modifications on the formation of Cys-tRNA (*4*) might conceivably be other symptoms of such allosteric variations.

The resolution of these mechanistic possibilities will be of interest to a group broader than aminoacyl–tRNA synthetase mavens. Consider that the molecule could arise by straightforward fusion between a ProRS and a CysRS (see the figure, panel A). Such a fusion might be selected when coordination of different synthetase activities becomes adaptive. On the other hand, if there are intimately fused bispecific sites (see the figure, panel B), or exquisitely communicating bispecific sites (see the figure, panel C), then there may exist molecular routes for fusion and evolutionary forces for the selection of specificity that have not yet been imagined.

References

1. C. Stathopoulos *et al.*, *Science* **287**, 479 (2000).
2. C. J. Bult *et al.*, *Science* **273**, 1058 (1996).
3. D. R. Smith *et al.*, *J. Bacteriol.* **179**, 7135 (1997).
4. C. S. Hamann *et al.*, *J. Bacteriol.* **181**, 5880 (1999).
5. C. Cerini *et al.*, *EMBO J.* **10**, 4267 (1991); *Eur. J. Biochem.* **244**, 176 (1997).
6. G. Eriani *et al.*, *Nature* **347**, 203 (1990).
7. V. L. Rath *et al.*, *Structure* **6**, 439 (1998).
8. A. R. Fersht *et al.*, *Biochemistry* **14**, 13 (1975).

PERSPECTIVES: QUANTUM COMPUTING

Quantum Information Processing Without Entanglement

Peter Knight

Suppose you had a really stupid friend, who can remember your phone number but not your name. How long would it take him to search the phone book database, comprised of n entries, before he found you? If he were really unlucky, it would take $n - 1$ queries. But on average, it would take him $n/2$ queries, which is clearly still a demanding task. Of course, I have assumed that the search was done with classical devices. But what would happen if you exploited a quantum device to make the search? Grover of AT&T showed in 1997 (*1*) that, if you took advantage of the massive parallelism within quantum mechanics, you could reduce the search to on the order of $n^{1/2}$ queries on average. The Grover algorithm, along with remarkably few other algorithms—including Shor's famous algorithm for factoring numbers (*2*)—is what has attracted so much attention to the newly emerging field of quantum computing.

Enhanced online at www.sciencemag.org/cgi/content/full/287/5452/441

The author is at the Blackett Laboratory, Physics Department, Imperial College London SW72BW, UK. E-mail: p.knight@ic.ac.uk

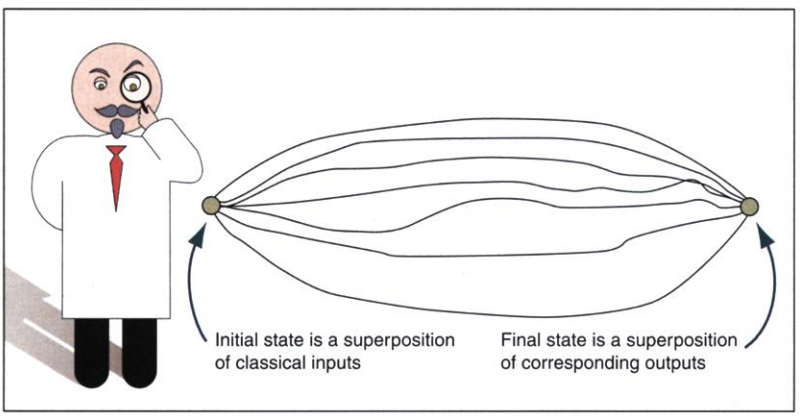
There have been a few pilot experimental studies of the Grover search algorithm, using nuclear magnetic resonance (NMR) (*3*) and interferometry (*4*). On page 463 of this issue, Ahn *et al.* report on a realization of Grover's search algorithm using a single atom as a quantum processor (*5*). In doing so,

they raise important questions about exactly what quantum physics allows over and above what the classical world permits in the processing of information.

The power of quantum computing derives from the interference inherent among the various quantum paths the system can take from a prepared input to an output that can be read out. Entanglement, that peculiar quantum property first identified by Erwin Schroedinger, is often invoked as the mechanism for the speedup of quantum processors over their classical counterparts. But it is not a necessary ingredient of a quantum information processor, although it does change the required physical resources and sometimes the speedup of the processor.

Ahn *et al.* address an important issue of current concern in the lively and newly emerging interdisciplinary field of quantum information processing: the degree to which atomic systems can implement algorithms using dramatically less physical resources, or fewer steps, than classical computing systems. What Ahn *et al.* do is important: They show that a single quantum system possessing no entanglement whatsoever can implement the search algorithm in a nonclassical and highly effective way. Their quantum system is a marked "unary" superposition of Rydberg states, which possess a high degree of parallelism. One then looks for this marked state within the atomic manifold, using short laser pulses to prepare the superposition, followed by ramped state dependent field ionization to read out the final outcome.

Most work in quantum information theory has concentrated on using two-state systems to represent information: the so-called qubit. If n such qubits are placed in a coherent superposition, then the number of potential states that can be exploited is 2^n, clearly a large number even for quite modest n: A register of say 1500 qubits, if it could be placed in superposition, could access more states than there are particles in the universe. The n qubits are the modest physical resource needed to access this astonishing potential. There is of course a catch: The $2n$ states need to evolve coherently with no disturbance from the outside world. Were an outside system in the larger world (including the inquisitive observer in the figure) to acquire information about which pathways were accessed in the quantum processor, then these lovely quantum superpositions would collapse, negating all the advantages of using a quantum processor. This is the "decoherence" problem that has proved to be such an obstacle in quantum computing.



Entanglement not required. Quantum information processing uses interference among quantum paths to access information in a potentially massively parallel fashion. Here we imagine that the processor sets up an initial superposition of states that evolve coherently according to a chosen algorithm. At the end of the evolution they come together to interfere, and if the algorithm has been set up intelligently, a readout of the final state enables information to be accessed that exploits this parallelism. Great care must be taken to isolate the quantum paths from the larger environment, because otherwise this outer world will become hopelessly entangled with the processor, resulting in a decoherent output lacking all the advantages of parallelism. Here, our inquisitive bystander must not access path information, or else the parallel development fails.

Most quantum processors one can envisage are not sufficiently isolated to permit much in the way of real parallelism. In contrast, the system used by Ahn *et al.* is pretty robust. *Ahn et al.* do not use n qubits as a resource, but rather a single atom prepared in a superposition of n Rydberg states, and it is this that is the "unary" representation. It lacks some of the resource advantages of an n-qubit system but retains others, particularly the ability to process in a parallel way using interference.

I heard a presentation of this work at the CLEO/QELS meeting in Baltimore in May 1999, where it appeared to be met with some suspicion from senior researchers. The origin of this misplaced suspicion seems to be a myth that quantum algorithms derive their power from entanglement. This myth is simply not true. It really needs to be stressed that quantum superpositions and parallelism are always needed for true quantum computation but that entanglement may not always be needed. Of course, it is needed for the Shor algorithm, where the change from exponential to polynomial times for execution that attracted so much attention requires both parallelism and entanglement (*6*). But neither the Grover algorithm nor the very first quantum algorithm due to Deutsch and Jozsa (*7*) need entanglement. This has already been discussed in an elegant paper by Lloyd (*8*). Lloyd discusses precisely the theory of the unary excitation of a superposition of atomic states experiment now performed by Ahn *et al.* He describes the various approaches to sorting, both classical and quantum and either using or not using entanglement. Finally, he discusses the implications for the resources needed if a unary approach rather than a qubit approach is adopted, as by Ahn *et al.* (*5*). That all you need is interference is in fact the reason why Cerf *et al.* could realize Grover's algorithm using classical interferometers (*4*) (although with a substantial resource overhead).

Is the approach developed by Ahn *et al.* going to be a practical device? Perhaps not, although Ahn *et al.* have hopes of extending it to much larger state spaces. Unlike the NMR route used up to now to implement Grover's algorithm, this one looks like it can be scaled up to databases of useful sizes. And it will tell us much about the nature of the necessary resources in quantum computing, itself an important development.

References

1. L. Grover, *Phys. Rev. Letters* **79**, 325 (1997).
2. P. Shor, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, S Goldwasser, Ed. (IEEE Computer Society, Los Alamitos, CA, 1994), p. 124.
3. I. Chuang, N. Gershenfeld, M. Kubinec, *Phys. Rev. Lett.* **80**, 3408 (1998); J. A. Jones *et al.*, *Nature* **393**, 344 (1998).
4. See, for example, N. J. Cerf *et al.*, *Phys. Rev. A* **57**, R1477 (1998).
5. J. Ahn, P. H. Bucksbaum, T. C. Weinacht, *Science* **287**, 463 (2000).
6. See, for example, A. K. Ekert and R. Jozsa, *Rev. Mod. Phys.* **68**, 733 (1996).
7. D. Deutsch and R. Jozsa, *Proc. Ry. Soc. London* A **439**, 553 (1992).
8. S. Lloyd, *Phys. Rev.* **A61**, R1 (1999); (available at xxx.lanl.gov/abs/quant-ph/9903057).