Solutions to homework 1

Warren D. Smith
wds@research.NJ.NEC.COM

date = October 13, 1998

1 $N \times N \times N$ Rubik's cube with all surface squares distinguished by numbered labels.

Let $R(N)$ be the number of configurations of the cube. Let $\phi(N)$ be the parity of N, i.e. 1 if N is odd, otherwise 0.

First, let's get some quick upper and lower bounds on $R(N)$. We observe the Rubik's cube has $6N^2$ surface squares. So $R(N) \le (6N^2)!$ by considering the number of ways to permute them. Next, observe the a square in the kth "layer" stays in that layer after any rotation of it (or some other layer parallel to it), and no matter what you do it will always be distance k to some edge of the face it is on. In fact, if the coordinates of a surface square on its face are (a, b), (where the face center is $(0, 0)$) then that surface square can be rotated to any of (a, b), $(-b, a)$, $(-a, -b)$, $(b, -a)$ on its face, and it can be moved also to any of the other 6 faces of the cube at these coordinates on the new face *only*, for a total of $24 = 4 \times 6$ possible locations *only*. Hence the 24 elements of each of $6N^2/24 = N^2/4$ equivalence classes can get permuted, and that is all. Hence $R(N) \le (24!)^{N^2/4}$.

We now show our upper bound is fairly tight. That may be seen by finding an explicit move sequence which will leave the cube unaltered except for performing a 3-cycle permutation $C = (1 \to 2 \to 3 \to 1)$ of any 3 desired images of any desired "typical" surface square. (A surface square is "typical" if it has 24 images and it is not on a cube edge. Only face centers and edge and corner squares can be atypical.) Such a sequence may be devised as follows if 1 and 2 are on the same face and 3 is on the opposite face. Let A be a permutation which swaps $1 \leftrightarrow 3$ while possibly fouling up the rest of the cube, but not fouling up the face 1 and 2 are on. Let T be the face twist which moves $2 \to 1$. Then $C = ATA^{-1}T^{-1}$ does the job (note how the A^{-1} unfouls the foulup coaused by A); this requires 8 moves. Any other choice of the surface squares 1, 2, 3 may be handled by doing $P^{-1}CP$ where P moves the 3 images to their appointed spots, possibly while fouling up the rest of the cube – that won't matter since the later P^{-1} unfouls it. This increases the "8" to a larger constant though. (I think 30 suffices easily). Since the 3-cycle $a \to b \to c \to a$ is writeable as a product of $a \leftrightarrow b$ and $b \leftrightarrow c$, this is an even permutation. Using 3-cycles we can achieve *any* even permutation (since also the product of two *disjoint* pair-swaps $a \leftrightarrow b$, $c \leftrightarrow d$ is achieveable by two 3-cycles a,b,c and d,c,a). Since there are $6(N-2)^2 - 6\phi(N)$ typical surface squares we see $R(N) \ge (24!/2)^{[(N-2)^2-\phi(N)]/4}$. So:

$$(\frac{24!}{2})^{[(N-2)^2-\phi(N)]/4} \le R(N) \le (24!)^{N^2/4}. \quad (1)$$

Actually the odd permutations are also reachable in typical typical cases since a single layer rotation is a 4-cycle (odd perm).

Getting the *exact* formula for $R(N)$ is considerably tougher, but doable; you need to consider exactly which permutations are reachable, there are a few funny parity restrictions. For example for the $N = 3$ case[1] it turns out that the group consists of: all even permutations which map edges to edges and corners to corners, which preserve the sum mod 2 of the edge-flippednesses (so of the 2^{12} thinkable edge subcube flips, only 2^{11} are actually achieveable), and which preserve the sum mod 3 of the corner-twists (so of the 3^8 thinkable corner subcube twists, only 3^7 are actually achieveable). If we regard the middle layers of the $3 \times 3 \times 3$ cube as fixed so the face centers never move (WLOG), the group has cardinality

$$\frac{1}{2}\, 8!\, 12!\, 3^7\, 2^{11} \;=\; 43252\,00327\,44898\,56000. \quad (2)$$

More generally, there are 8 types of cubies possible:

1. $6\phi(N)$ Face centers
2. 8 Corners. These have sum of twists mod 3 equals zero so only 3^7 of the 3^8 thinkable twists are achievable.
3. $12\phi(N)$ edge centers. These have sum of flips mod 2 equals zero so only 2^{11} of the 2^{12} thinkable flips are achievable.
4. $12[N-2-\phi(N)]$ non-central edges.
5. $12[N-2-\phi(N)]$ face diagonal squares.
6. $12[N-3]\phi(N)$ face horizontal or vertical midline squares.
7. $24[N-2-\phi(N)][N-4-\phi(N)]$ typical squares.

The total number of squares of the last 4 types is $12[N-2-\phi(N)][2N-6-\phi(N)]$. For each of these, all 24! permutations of each equivalence class are achieveable. The total

[1] In which *all* the surface squares are atypical, so our analysis above was rather weak...; incidentally when $N = 3$ it is not necessary to number all the surface squares since they are already all distinguishable!

number of configurations (now counting the 24 rigid rotations of the whole cube as *different* configurations and also realizing that each face center may be rotated in 4 ways – if you, like most people, want to get rid of these effects then divide by $24 \cdot 4^{6\phi(N)}$) is

$$24 \cdot 3^7 8! \, [2^{11} 12! \cdot 4^6]^{\phi(N)} \, 24!^{[N-2-\phi(N)][2N-6-\phi(N)]/2} \quad (3)$$

Now let $\Delta(N)$ be the maximum number of moves (layer rotations) required to get from any cube configuration to any other. In any configuration, you can make $9N$ possible non-null moves. Hence after M moves, we can reach $\leq (9N)^M$ configurations[2]. Hence $\Delta(N) \geq [\log R(N)]/\log(9N) \geq (N-2)^2 \log(24!)/[4\log(9N)]$. On the other hand, our magic 30-move sequence that performs a 3-cycle may be used to simply exchange all the pairs of typical surface squares that need exchanging, plus there are other constant-long move sequences to perform 2 edge cube flips and 2 corner twists (these tasks for large N add up to $O(N)$ moves, which is negligible by comparison) so that $\Delta(N) \leq 30 \cdot 2 \cdot 6N^2 + O(N)$. So

$$\left\lceil \frac{(N-2)^2 \log(24!)}{4\log(9N)} \right\rceil \leq \Delta(N) \leq 360N^2 + O(N). \quad (4)$$

There is still an order $\log N$ gap between these two bounds, unfortunately.

Finally, as far as algorithms to solve a cube scrambled with M moves are concerned: the naive method is to examine all possible M-long move sequences, which takes M^{9N} steps and consumes only $O(M)$ words of memory. But a faster method if you have lots of random access memory available is to make a big table of all configurations (as words in the permutation group of order $(6N^2)!$) reachable in $M/2$ moves from a pristine cube. There are at most $(9N)^{M/2}$ such configurations, each of which requires $\min(6N^2, M)$ words of memory to represent naively. Now, scan through the configurations, and for each, ask, what is the permutation required to get from here, to the scrambled configuration? That permutation will also be in our table if we happen to be at the halfway point, so we look it up and output the product of the two half-permutations required to do the full scrambling/unscrambling job. Note: the key idea here is that the state transitions of the cube form a *group*. They are not just any old transition diagram! The total number of steps in this algorithm is roughly $(9N)^{M/2}\min(M, N^2)$, i.e. about the *square root* of the naive algorithm's op count. The trouble with this algorithm is the huge requirement for $(9N)^{M/2}\min(6N^2, M)$ words of memory.

We can get other algorithm variants which use more space than the naive (but space-optimal) algorithm, but less space than our fast algorithm, and whose runtime is intermediate also. (There is in fact a whole space time tradeoff curve.)

[2] Actually moves after the first should not move the same layer as the preceding move, so $\leq (9N)(9N-3)^{M-1}$ is a better bound. Even better bounds could be got by explicitly counting the 2-move sequences, and demanding nonoverlap between successive 2-move sequences, and so on. In the $N = 3$ case we can refuse to allow the middle layers to move leading to $\leq 9^{M-1} \cdot 12 \leq 12^M$.

Example I: you can store all configurations at distance $M/3$ from the pristine and scrambled states in tables A and B. Now scan through table A and, by scanning through the table of $M/3$-length permutations again in an inner loop, compute (but don't store) the configurations at distance $2M/3$, looking for a match in table B. That takes $2(9N)^{M/3}\min(6N^2, M)$ words of memory and has runtime $(9N)^{2M/3}$ roughly.
Example II: Store all configurations at distance $M/4$ from the pristine and scrambled states in tables A and B. Now scan through table A and, by scanning through the table of $M/4$-length permutations again in an inner loop, compute, but do not store, the $M/2$-away configurations. For each one, ask if it is $M/4$ away from somebody in B (by scanning over B and seeing if the quotient of the two permutations, is in our table). The storage needs are now of order $2(9N)^{M/4}\min(6N^2, M)$, and the runtime is about $(9N)^{3M/4}$ steps.

Fiat, Moses, Shamir, et al [Symp. Foundations of CS 30 (1989) 274-279; see also Bright, Kasif, Stiller; AAAI-94 conf. 1341-1346] showed how to speed up the algorithm of example II. The idea is to use tables A and B to generate the $M/2$-away configurations from both sides, looking for a match. They have a clever algorithm which will generate all $2K$-away permutations *in lexical order* from a table of all K-away permutations. If you are a good programmer, you can probably see how to do this. You want to generate the permutations which map the first guy to the first guy first, then the rest; and these two generation subproblems are solved recursively. Thus, both camps can generate the $M/2$-away configurations in lexical order with only the camp which is currently behind in that order being active at any time. In this way, we can find a match in the midway set without consuming time quadratic in the cardinality of that set – we need only consume linear time. This idea gives us the same runtime (except for a constant factor) as our most space hungry algorithm, but roughly squareroots its space needs.

In these algorithms, the runtime may be reduced further by using any subroutine which computes a lower bound on the distance from an arbitrary configuration to the scrambled or pristine state. We could use these bounds to incorporate "pruning" into the algorithm generating the move sequences, allowing us never to generate a large fraction of the possible $M/4$-away or $M/2$-away configurations. It is hard to get a precise bound saying what the time and space savings will be from this idea, but I would expect them to be *at least* a factor of 10^4 in the $N = 3$ case.

In the $N = 3$ case, you can get effective distance lower bounds as follows. First, you could just add up L_1 distances of all the edge subcubes to their destinations, divide by 4 (since a move moves 4 edge subcubes at once), and round up to an integer. Second, the same trick could be applied with corner subcubes only. Third, you could explicitly store tables of the minimum number of moves you need to resolve all possible corner configurations (there are $8!3^7 = 88179840$ such configurations). Or you could have tables of the minimum number of moves you need to resolve all possible configurations of some other subset of subcubes. Take the best lower bound you get from any of these methods.

For $N=3$, our crude lower bound on $\Delta(3)$ is 5. More precise analyses[3] specialized to $N=3$ show $18 \le \Delta(3) \le 29$. David Seal has conjectured that the "superflip" configuration (everything in the pristine state except all edge subcubes are flipped) is the furthest from the pristine state and that this is exactly 20 moves away. The 20 move sequence in the notation[4] of Berlekamp, Conway, Guy: Winning ways volume 2 (Academic Press) is

$$RLD^2B^{-1}L^2F^2R^2U^{-1}DR^{-1} \quad (5)$$
$$D^2F^{-1}B^{-1}D^{-1}F^2D^{-1}R^2U^{-1}F^2D^{-1}.$$

This conjecture is partly motivated by the fact that the superflip is the unique "central" nonidentity element of the Rubik group[5]. Assume Seal's conjecture is correct. Then our naive algorithm with $N=3$ would take somewhere around $27 \cdot 24^{19} \approx 4.52 \times 10^{27}$ steps to solve the cube optimally. Meanwhile the age of the universe is about 10^{17} seconds, i.e. 10^{26} nanoseconds. Our better algorithm would take around $27^{10} \cdot 10 \approx 2.06 \times 10^{15}$ steps, which is only about a CPU month on a computer doing 1 step per nanosecond, but the storage requirements of around 10^{10} words of memory could be a burden. The further space and time improvements arising from distance lower bound based pruning ought to reduce this down to a feasible amount of computation, e.g. a few CPU hours on a contemporary desktop computer.

Getting a 10-times faster computer won't allow us to increase our maximum allowable N to $N+1$. This is typical of exponential time algorithms – you can hardly get anywhere, and with a 10 times faster computer, you can increase your maximum problem size by an additive constant.

2 Church's thesis

How to simulate a RAM with a Turing machine [solutions to this exercise are done in C. H. Papadimitriou: Computational complexity, Addison-Wesley, 1994 and other books] Well, the idea is that after T RAM steps, at most T RAM cells could be in use. So the simulating Turing machine makes a table of T entries to hold all the active RAM cell contents, and each cell also stores its address. Every time the RAM accesses a cell, the Turing machine does a brute force search through all T cells. Every time the RAM adds a new cell to the active set, the Turing machine has to shift its table by 1 over (at least, if it wants to preserve address ordering). If you don't want to preserve ordering, then just add it onto the end. Anyway, each RAM step is thus emulated by order $T^2(W+A)$ Turing steps, where an address is A bits long and a RAM word is W bits wide. This is polynomial slowdown. If each RAM cell is in fact an infinitely long tape, then we can emulate this with second-level addresses or by just shifting everything over by 1 whenever we add or delete a tape square. Of course, we need our Turing machine to have special characters in its alphabet denoting things like "end of subtape" and "current square marker" so the Turing machine knows when to stop walking.

History of "Church's" theses (as far as I know): Alonzo Church proposed [An unsolvable problem of elementary number theory, Amer. J. Math. 58 (1936) 345-363; A note on the Entscheidungsproblem, J. Symbolic Logic 1 (1936) 40-41 & Correction 101- 102] the idea that his "recursive" functions encapsulated the notion of "computability." Recursive functions were shortly thereafter shown by Turing to be what Turing machines compute. The notion of polynomial time equivalence, and the notion that that "polynomial time" meant approximately the same as "tractable," was not due to Church, and seems to have been promulgated by Jack Edmonds in the early 1960s [Paths, trees, and flowers, Canadian J. Math. 17 (1965) 449-467]. The fact that the strong and weak Church theses are really statements about *physics* seems to have been unnoticed, or anyhow not shouted on the streets, until the paper of Vergis,Stieglitz, and Dickinson [The complexity of analog computation, Math & Computers in Simulation 28 (1986) 91-113]. If this is really the first time this was said, it took amazingly long. I would hereby like to propose the "intermediate strength Church thesis" that a Turing machine may simulate any "computer" with at most polynomial increase in its *space* consumption.

3 Linear algebra

A matrix U is "unitary" if $U^{-1} = U^*$, where U^* means U transpose conjugate. A matrix H is "Hermitian" if it equals its complex conjugate transpose: $H = H^*$.

If $\exp(iH)$ is regarded as defined by the Taylor series for exp, then the fact that if $U = \exp(iH)$, then $U^{-1} = \exp(-iH) = U^*$ (thus verifying that U is unitary) may be verified by simply multiplying the two series to verify $UU^{-1} = I$ and considering them term by term to see $U^{-1} = U^*$ since every term is the complex conjugate transpose.

If A and B are Hermitian, then $A \pm B$, $AB + BA$, $iAB - iBA$, ABA, and A^n for all integer n, are Hermitian. But AB and $AB - BA$ ain't necessarily so as you can see by simply trying some random 2×2 numerical examples. The proof for $A \pm B$ is immediate from the definition of Hermitian. For $AB + BA$ and $iAB - iBA$, the second term cancels out the non-hermitianness in the first term, since the second term is the conjugate transpose of the first term (using $(AB)^T = B^T A^T$ for example) so that their sum

[3] The lower bound 18 is from the same analysis we gave before, only with the correct group cardinality (EQ 2) and using footnote 2. The upper bound 29 is from the latest worst case analysis (by Michael Reid: `reid@math.brown.edu`) of the 2-stage subgroup descent cube solving algorithm. Reid did an exhaustive search of the subgroup generated by $\{U, D, F^2, R^2, B^2, L^2\}$, to show it had maximal distance 18 from the identity. Plus it was already known you could always reach this subgroup in 12 moves, and sharing 1 move between the two solution stages is always possible. Then $29 = 18 + 12 - 1$. Other information is in Christoph Bandelow: Inside Rubik's Cube and Beyond, Birkhauser 1982.

[4] U, D, R, L, F, B are clockwise turns of the up, down, right, left, front and back faces of the cube (held in some fixed orientation).

[5] The "center" of a group G is the subgroup consisting of the $x \in G$ such that $gxg^{-1} = x$ for all $g \in G$.

has to be Hermitian. Similarly $(ABA)^* = A^*B^*A^*$ which if A and B are Hermitian is ABA. The proof for A^n is just a special case of the fact that $f(A)$ is Hermitian if f is any analytic function (below) but is anyhow obvious for $n \geq 0$ by considering multiplying the matrix A^n, i.e. $AAAA...A$, by a vector $\vec{x}$. Take the real symmetric subcase of complex Hermitian matrices for simplicity. Obviously if you view $\vec{x}$ as a column vector then $A\vec{x}$ has the same effect as $A^T\vec{x}$ since $A = A^T$. But then AA has the same effect as A^TA^T. And so on inductively, thus proving A^n is symmetric if A is.

But what the devil can it mean to take a general analytic function $f(A)$ of a *matrix*? Well, most reasonably advanced linear algebra books will tell you that, e.g. G.Golub & C.Van Loan: Matrix computations, Johns Hopkins Univ.Press; N.Dunford & J.Schwartz: linear operators volume 1 (Wiley), and also see R.F.Rinehart: Amer. Math. Monthly 62 (1955) 395-414 for an expository article. There are several equivalent definitions. The most powerful (since it works for operators as well as finite dimensional matrices) is

$$f(M) = \frac{1}{2\pi i}\oint f(z)(zI - M)^{-1}\mathrm{d}z \tag{6}$$

where the contour integral is for any sufficiently small contour going counterclockwise around the origin in the complex plane. You can use a circle of radius r. The idea is that if M is a scalar (i.e. 1×1) then this is just Cauchy's residue theorem applied to an integrand constructed to have a residue of value $f(M)$ at $z = M$. By doing the integral approximately using a Riemann sum, you get an algorithm for approximating $f(M)$. In cases like $f = \exp$ where the Taylor series converges everywhere (or continued fractions could be considered...) you could just use the Taylor series (in fact it suffices if all the eigenvalues of M lie in the convergence region).

A key observation is you can use the *Schur decomposition*

$$M = QTQ^* \tag{7}$$

where Q is unitary and T is upper triangular. Any matrix has such a factorization and it may be found by an efficient and stable numerical algorithm taking order N^3 flops for $N \times N$ matrices. This is very important in computational linear algebra. In the special case when M is Hermitian, then this is the *eigendecomposition* and T is the real diagonal matrix of M's eigenvalues, which are all real, and the columns of Q are the orthonormal eigenvectors of M.

Now note that

$$M^n = MM...M = QTQ^*QTQ^*...QTQ^* = QT^nQ^*. \tag{8}$$

Hence, for any analytic function (power series) in M, disregarding convergence issues, $f(M) = Qf(T)Q^*$ and so, it suffices to be able to evaluate matrix functions for upper triangular matrices only (assuming we can perform the Schur decomposition) to be able to do them for all matrices. And to be able to handle all Hermitian matrices,it suffices to be able to evaluate matrix functions on diagonal matrices (the eigenvalues) only – which is trivial: $f(D) = \mathrm{diag} f(D_{jj})$. So that explains, for example, how to evaluate $\arctan(H)$ if H is Hermitian:

1. Compute eigendecomposition $H = QTQ^*$, T diagonal and all real, Q unitary;
2. Compute the diagonal matrix F with $F_{jj} = \arctan T_{jj}$;
3. $\arctan H = QFQ^*$.

To evaluate $F = f(T)$ where T is upper triangular is also easy once you observe Lemma: $FT = TF$ (because T commutes with any T^n term in any Taylor series). Also lemma: F is also upper triangular (because T^n is). These lemmas allow us to write down linear equations for the entries of F, and assuming the diagonal entries are known (they are $F_{ii} = f(T_{ii})$) allows solving for the others. The resulting very slick algorithm due to Parlett will evaluate $f(T)$ in $O(N^3)$ flops (it is algorithm 11.1.1 in Golub & Van Loan).

Finally, we need to describe how to do things with unitary matrices if we know how to do things with Hermitian ones and vice versa. The trick is to observe that if H is hermitian then $U = (H + iI)(H - iI)^{-1}$ is[6] unitary. The reverse transform is $H = -i(I + U)(I - U)^{-1}$. This transformation was motivated by the fact that the conformal map $z \to (z + i)/(z - i)$ maps the real line (containing all eigenvalues of a Hermitian matrix) to the unit circle $|z| = 1$ (containing all the eigenvalues of a unitary matrix). Now if we, for example, find the eigenvalues λ of H using the Schur decomposition algorithm, then we can solve for the eigenvalues θ of U by $\theta = (\lambda + i)/(\lambda - i)$.

So if we want to show that $i \ln U$ is hermitian if U is unitary, we can apply the known Taylor series expressing $\ln z$ as a sum of powers of $x = (z - 1)/(1 + z)$

$$\ln z = 2[x + \frac{x^3}{3} + \frac{x^5}{5} + \cdots] \tag{9}$$

to see that, yes, indeed, we are going to get a Hermitian matrix, since every term in the Taylor series is Hermitian.

4 Electrons

Classically the energy density in an E field is $\epsilon_0 E^2/2$ and the E field at distance s from the center of a spherical charge e is $e/(4\pi\epsilon_0 s^2)$ (Coulumb's law). This is in my high school physics book. So the total field energy outside a hollow sphere electron of radius r is

$$\frac{\epsilon_0}{2}\int_r^\infty |\vec{E}(r)|^2 4\pi r^2 \mathrm{d}r = \frac{e^2}{8\pi\epsilon_0}\int_r^\infty s^{-2}\mathrm{d}s = \frac{e^2}{8\pi\epsilon_0 r}. \tag{10}$$

which, to be the same as $m_e c^2$, requires the radius of an electron to be

$$r = \frac{e^2}{8\pi\epsilon_0 m_e c^2} \approx 1.40897046 \times 10^{-15}\text{meters}. \tag{11}$$

Physicists usually also add to the field energy, a term (which happens to be exactly equal to it) which represents the energy needed to bring all that charge in from ∞ in infinitesimal increments. Thus the "classical electron radius" is

[6] Proof: $UU^* = (H + iI)(H - iI)^{-1}(H^* + iI)(H^* - iI)^{-1} = I$.

twice the value above, i.e. (CODATA value)

$$r_e \approx 2.81794092 \times 10^{-15}\text{meter}. \quad (12)$$

Rigid objects are impossible in special relativity (indeed, it is impossible for the speed of sound in an object to exceed c) so the sphere will be flexible. Assuming waves propagate at speed $v \leq c$, the frequency of the lowest tone will be $f \geq v/(2\pi r)$ roughly, which if $v = c$ is $\approx 1.69 \times 10^{22}$ Hertz, and hence its energy will be $E \geq hf \approx 1.12 \times 10^{-11}$ Joules= 7.00×10^7eV.

These energies are *easily* accessible to experimenters so we would have expected excited states of the electron (and consequent spectral lines) to have been observed if they were present[7] (unless the speed of sound in electron matter were 1000 times slower than the speed of light, which seems implausible... presumably any fundamental speed should actually *be* the speed of light). Hence, they are not present, and this picture of the electron is presumably wrong.

Now, Weisskopf's quantum field theoretic analysis of the field energy of an electron [Phys.Rev. 56 (1939) 72-85] says that the energy grows only logarithmically, not as a power law, to infinity as $r \to 0$. Unfortunately, Weisskopf is not able to estimate the dimensionless multiplicative constant(s) ("z_n" on his page 75) in front of his logarithm, very precisely. If he ignores that, he would conclude (also his page 75) the quantum electron radius would be about $10^{-58} r_e$. However, since his unknown constant ends up in an exponent, this estimate is highly unreliable. Anyway the quantum radius of the electron is expected to be perhaps *exponentially smaller* than r_e, for all we know somewhere randomly distributed on a log scale anywhere between maybe r_e and $10^{-200} r_e$. Thus perhaps the electron's spectral energies are exponentially larger, i.e. quite possibly *far* larger than any energies accessible experimentally, even in high energy cosmic rays. (For example they might be 10^{100} times larger.) Hence it does not seem a priori ridiculous for electrons to be tiny spheres, at least with the current precision of Weisskopf's radius versus energy analysis. Furthermore, spherical electrons would have smaller field energy than any other shape of electrons such as strings, doughnuts, cubes, etc, lying totally inside the sphere. Hence, on naive energetic grounds alone, there is no reason to prefer string theory, contrary to claims made in popular lectures by Ed Witten. (That is: the string or cube electron with size appropriate for its field energy to explain its mass, would have lower energy excitations than the sphere electron and hence would be *more* in conflict with the experimental lack of observation of such states, not less.)

However, there is another reason Witten and others think string theory (or nowadays, the fad is "M-brane theory") is a good thing. That has to do with quantum gravity. Electrons, if they are point particles, must be black holes. The Einstein-Maxwell equations (which have been totally solved in the case of "Kerr black holes" with given charge, mass, and angular momentum) disallow black holes with that high a charge and that little mass, so something is wrong. In our naive classical analysis, the electron of radius r_e would be far too large compared to its Schwarzchild radius $2Gm_e/c^2 \approx 1.35 \times 10^{-57}$ meters to be a black hole – so no problem. But, in our quantum analysis, sufficiently exponentially small electrons would certainly (according to classical general relativity, anyhow) be black holes, in which case something would have to be done about it. I do not know if the most natural such something is to assume electrons are really 10 dimensional "strings," though! Assuming an electron radius of 10^{-40} meters would seem to be essentially totally compatible with theory and observation so far (it would imply spectral energies far too high to be observed) while at the same time not being even close to black holeness and hence not encountering any fundamental worries there.

In conclusion: all these arguments constructed by Witten to convince the layperson of the need for string theory, presently are groundless and fall flat on their face. There probably are better arguments for string theory, such as the hope it can predict most of the fundamental constants of nature, but these arguments would currently seem to be, as we say in the computer biz, "vaporware!"

5 TUNNELING

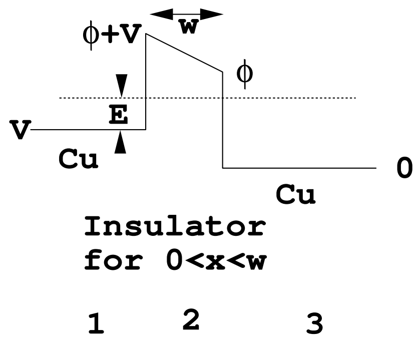

Figure 1: Tunneling problem potential function $U(x)$.

Schrödinger's time independent equation in our case is

$$[V + E - U(x)]\Psi = \frac{\hbar^2}{2m_e}\frac{\mathrm{d}^2}{\mathrm{d}x^2}\Psi?? \quad (13)$$

where $U(x)$ is drawn in the figure. Let $P = \hbar^2/(2m_e)$ for convenience.

The solution of

$$u'' + k^2 u = 0 \quad (14)$$

is

$$u(z) = c_1 e^{ikz} + c_2 e^{-ikz}. \quad (15)$$

Hence in the copper with $x < 0$, the solution is a plane wave of the form

$$\Psi(x) = a_1 e^{ixE/\sqrt{P}} + b_1 e^{-ixE/\sqrt{P}} \quad (16)$$

In the copper with $x > w$, the solution is a plane wave of the form

$$\Psi(x) = a_3 e^{ix(E+V)/\sqrt{P}} + b_3 e^{-ix(E+V)/\sqrt{P}} \quad (17)$$

[7] Although I admit, I have never heard of an explicit experimental search for this.

Information about the *Airy*[8] *functions* may be found in [Bender & Orszag: Advanced mathematical methods for scientists and engineers, McGraw; and Handbook of Mathematical Functions (ed. M.Abramowitz & I.Stegun) Dover, section 10.4]. Here it is pointed out that, for example, the solution of

$$u'' - k^2 z u = 0 \tag{18}$$

is

$$u(z) = c_1 f(kz) + c_2 g(kz) \tag{19}$$

where

$$f(z) = 1 + \frac{1}{3!}z^3 + \frac{1\cdot 4}{6!}z^6 + + \frac{1\cdot 4\cdot 7}{9!}z^9 + \cdots \tag{20}$$

$$g(z) = z + \frac{2}{4!}z^4 + \frac{2\cdot 5}{7!}z^7 + + \frac{2\cdot 5\cdot 8}{10!}z^{10} + \cdots \tag{21}$$

So in the insulator, the solution is

$$\Psi = a_2 f(\frac{x_0 - x}{\sqrt{P}}) + b_2 g(\frac{x_0 - x}{\sqrt{P}}) \tag{22}$$

where

$$x_0 = w + w(\phi - E - V)/V \tag{23}$$

is the location of the point where the diagonal line would cross the dashed one. The 6 undetermined coefficients $a_1, b_1, a_2, b_2, a_3, b_3$ have to be chosen so that Ψ and Ψ' are continuous at $x = 0$ and $x = w$ (4 conditions) and so that we get outgoing waves *only* when $x > w$ (1 condition $a_3 = 0$). That leaves us with one degree of freedom corresponding to the fact that our solution may be multiplied by any constant factor.

Then the transmission probability will be $|b_3/b_1|^2$ where b_3 is the complex amplitude of the outgoing plane wave in the $x > w$ copper and b_1 is the complex amplitude of the incoming plane wave in the $x < 0$ copper.

The known asymptotic formulae for Airy functions (10.4.82-85 in the Handbook of M.F.'s) or more directly the WKB approximation indicates that the transmission probability is asymptotic (in the limit where the tunneling probabilities are small because the barrier is high or thick) to

$$\exp\left(\frac{-2\sqrt{2m_e}}{\hbar}\int_0^w \sqrt{U(x) - E - V}\mathrm{d}x\right) = \exp\left(\frac{-4\sqrt{2m_e\phi}}{3\hbar V}\frac{-4w\sqrt{2m_e\phi}}{3\hbar V}\left[(\phi + V - E)^{3/2} - (\phi - E)^{3/2}\right]\right). \tag{24}$$

To try to compute the actual current-voltage curve for this 2-terminal electrical device is not this easy. First, one has to consider plane waves impinging at all angles of incidence, not just normal incidence. Suppose we knew the transmission probability formulae for all pairs of incoming and outgoing angles. Then, the Copper crystal lattice supports periodic solutions of Schrödinger's equation called "Bloch conduction band states." Suppose we knew all such states and the "density of states" in the copper as a function of both energy and angular orientation with respect to the Copper crystal lattice. Then, assume these states are filled by electrons (or not) with filling probabilities according to the Fermi-Dirac distribution as a temperature T. We could then integrate, over all states in the lefthand copper, the filling probability of that state, times the probability the corresponding states (also integrated over) in the righthand copper at energy V away were unfilled, times the transmission probability through the barrier for that angle-pair at that energy, to get an estimate of the actual tunneling rate. (We'd also have to consider backwards tunneling similarly.) Very messy; a computer would certainly be needed.

[8] Airy functions are very important because any solution of Schrödinger's equation near a classical "turning point" (where a classical particle would turn around) is approximately Airy.