Memo on urban renewal (James Propp)

Part 1 (of 6): Urban renewal lets you count (perfect) matchings of arbitrary bipartite planar graphs.

Two simple graphical moves, called urban renewal and vertex-splitting/merging, gives us easy proofs of the (EKLP) powers-of-2 theorem, the (Stanley) multi-q version of that result, and the (Yang) powers-of-5 theorem. They do so by allowing one to reduce the weighted enumeration of matchings of an Aztec diamond graph of order n to a (differently) weighted enumeration of matchings of an Aztec diamond graph of order n-1 (where the weight of a matching is the product of the weights of its edges, and the weighted enumeration of matchings of a graph is just the sum of the weights of its matchings). I

But there's more. Many bipartite planar graphs G can be associated with a subgraph G' of the square grid such that the matchings of G and G' are equinumerous (often you don't even need to split vertices of G to get G'), and such a graph G' can often in turn be modified (by adding isolated edges) to get a graph isomorphic to a spanning subgraph of an Aztec diamond graph of order n, provided that n is sufficiently large. A spanning subgraph can be thought of as just a 0,1-weighting of the full graph, for purposes of weighted enumeration of matchings --- so urban renewal gives us a way to count matchings of many bipartite planar graphs. In fact, it can be made to work for all bipartite planar graphs (if one does enough vertex-splitting). Indeed, it works for all weighted enumerations of matchings of bipartite planar graphs.

Here's the rule (to be immediately followed by an example): Given an Aztec diamond graph of order n whose edges are marked with weights, divide it into n^2 cells (in Mihai Ciucu's sense), replace each marked cell

```
      / \
 w  /     \  x
  /         \
/             \
\             /
  \         /
 y  \     /  z
      \ /
```

by the marked cell

```
                          / \
            z/(wz+xy) /     \ y/(wz+xy)
                    /         \
                  /             \
                  \             /
                    \         /
            x/(wz+xy) \     / w/(wz+xy)
                          \ /               ,
```

and strip away the outer flank of edges, so that an edge-marked Aztec diamond graph of order n-1 remains. Then (as I'll show in a later posting) the sum of the weights of the matchings of the graph of order n equals the sum of the weights of the matchings of the graph of order n-1 times the product of the n^2 (different) factors of the form wz+xy. So, if you perform the reduction process n times, obtaining in the end an Aztec diamond of order 0 (for which the sum of the weights of the matchings is 1), you'll find that the weighted sum of the matchings of the original graph of order n is just a product of n^2 + (n-1)^2 + ... + 2^1 + 1^2 terms, read off from the cells.

To show you how easy this is, I'll show you how to calculate the
number of perfect matchings of the 4-by-4 grid (16 vertices).
I imbed the the 4-by-4 grid in the Aztec diamond of order 3 in
such a way that the vertices in the complement can be matched.
I then enforce this matching on the complement by giving all
other edges in the complement weight 0:

```
      / \             / \             / \
    /     \         /     \         /     \
  / 1     0 \     / 1     1 \     / 0     1 \
/             \ /             \ /             \
\             / \             / \             /
  \ 0     1 /     \ 1     1 /     \ 1     0 /
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  / 1     1 \     / 1     1 \     / 1     1 \
/             \ /             \ /             \
\             / \             / \             /
  \ 1     1 /     \ 1     1 /     \ 1     1 /
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  / 0     1 \     / 1     1 \     / 1     0 \
/             \ /             \ /             \
\             / \             / \             /
  \ 1     0 /     \ 1     1 /     \ 0     1 /
    \     /         \     /         \     /
      \ /             \ /             \ /
```

Now I apply the replacement rule described above:

```
      / \             / \             / \
    /     \         /     \         /     \
  / 1     0 \     /1/2   1/2\     / 0     1 \
/             \ /             \ /             \
\             / \             / \             /
  \ 0     1 /     \1/2   1/2/     \ 1     0 /
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  /1/2   1/2\     /1/2   1/2\     /1/2   1/2\
/             \ /             \ /             \
\             / \             / \             /
  \1/2   1/2/     \1/2   1/2/     \1/2   1/2/
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  / 0     1 \     /1/2   1/2\     / 1     0 \
/             \ /             \ /             \
\             / \             / \             /
  \ 1     0 /     \1/2   1/2/     \ 0     1 /
    \     /         \     /         \     /
      \ /             \ /             \ /
```

Next I strip away the outer layer (come to think of it, I didn't really need to compute all those values, since I'm now throwing them away):

```
      / \             / \
    /     \         /     \
  / 1    1/2\     /1/2    1 \
/             \ /             \
\             / \             /
  \1/2   1/2/     \1/2   1/2/
    \     /         \     /
      \ /             \ /
      / \             / \
    /     \         /     \
  /1/2   1/2\     /1/2   1/2\
/             \ /             \
\             / \             /
  \ 1    1/2/     \1/2    1 /
    \     /         \     /
      \ /             \ /
```

Now I do it again (only this time I won't bother to show you the weights that end up getting thrown away):

```
      / \
 4/3/     \4/3
  /         \
/             \
\             /
  \         /
 4/3\     /4/3
      \ /
```

I do it one more time, and I've got the empty Aztec diamond.

Now we multiply together all the cell-factors:

$$2^5 \text{ times } (3/4)^4 \text{ times } (32/9) = 36.$$

Greg would correctly note that this is "just" row-reduction of Kasteleyn matrices, but observe that there are no signs to get wrong, and no cleverness required in doing the reductions.

What's more, if your interest is in proving *theorems* about weighted matchings of an infinite sequence of graphs rather than just counting weighted matchings of some particular graph), then this method is also good, because if your original weighting scheme was highly symmetrical, then after a few turns of the graphical-reduction crank you will be led to an Aztec diamond graph with the same pattern of weights you started with, which means that you now have a multiplicative recurrence relation for the matchings of the graphs in your sequence (where the multipliers are just the cell-factors you've seen along the way).

One slight subtlety concerning the "wz+xy method" concerns situations in which a product wz+xy vanishes because certain edge weights were 0 in your original weighted graph (or because you've decided to do some signed enumeration in search of interesting q=-1 phenomena). In such a case, you should replace 0's with epsilons as needed; the result will be a rational function of epsilon whose limiting value as epsilon goes to zero is desired. For this purpose, every rational function of epsilon can be written as a polynomial or power series in epsilon and replaced by its leading term, so calculations aren't as bad as one might think.

Part 2: Urban renewal lets you determine edge-probabilities for random weighted matching of arbitrary bipartite planar graphs.

Before I give the formal statement and proof for urban renewal, I want to point out that it does even MORE than I advertised in Part 1. Namely, it gives us an efficient iterative scheme for computing edge probabilities for uniform random matching of an arbitrary bipartite planar graph (or more generally, for random matching in which the probability of a particular matching is proportional to its weight).

What we do is thread our way back through the reduction process, computing the edge probability in successively larger and larger weighted Aztec diamonds, making use of the weights that were computed during the reduction process. Suppose we have computed edge probabilities for the weighted graph of order n-1. To derive the edge probabilities for the weighted graph of order n, we imbed the smaller graph in the larger (concentrically), divide the larger graph into n^2 cells, and SWAP the numbers belonging to edges on opposite sides of a cell (the example below will make this clearer). When we have done this, the numbers on the edges are not yet the true probabilities, but they are decent approximations to them, and we can add a correction term that makes them exact. Let's zoom in on a particular cell to see how it works: Consider our friend the cell

```
        / \
   w  /     \  x
    /         \
  /             \
  \             /
    \         /
   y  \     /  z
        \ /
```

and suppose that the edges with weights w, x, y, and z (before the swapping has taken place) have been given approximate probabilities p, q, r, and s, respectively. Then the *exact* probabilities for these respective edges are

s + (1-p-q-r-s) wz/(wz+xy),
r + (1-p-q-r-s) xy/(wz+xy),
q + (1-p-q-r-s) xy/(wz+xy), and
p + (1-p-q-r-s) wz/(wz+xy).
The number 1-p-q-r-s is called the deficit, and the numbers wz/(wz+xy) and xy/(wz+xy) are called (net) creation biases.

Applying this to the 4-by-4 grid graph, we find that for the weighted Aztec graphs of orders 1, 2, and 3 that we found (in reverse order) when we *counted* the matchings, the edge-probabilities are as follows: For the order 1 graph, we have

```
      / \
 1/2/     \1/2
  /         \
/             \
\             /
  \         /
 1/2\     /1/2
      \ /
```

We imbed this in the Aztec graph of order 2:

```
      / \             / \
    /     \         /     \
  / 0     0 \     / 0     0 \
/      .      \ /      .      \
\             / \             /
  \ 0    1/2/     \1/2    0 /
    \     /         \     /
      \ /             \ /
      / \             / \
    /     \         /     \
  / 0    1/2\     /1/2    0 \
/      .      \ /      .      \
\             / \             /
  \ 0     0 /     \ 0     0 /
    \     /         \     /
      \ /             \ /
```

Note that I've put a dot in the middle of each of the four cells.
We swap each number with the number opposite it in its cell:

```
      / \             / \
    /     \         /     \
  /1/2    0 \     / 0    1/2\
/      .      \ /      .      \
\             / \             /
  \ 0     0 /     \ 0     0 /
    \     /         \     /
      \ /             \ /
      / \             / \
    /     \         /     \
  / 0     0 \     / 0     0 \
/      .      \ /      .      \
\             / \             /
  \1/2    0 /     \ 0    1/2/
    \     /         \     /
      \ /             \ /
```

These are the approximate, inexact edge-probabilities. To find the exact values, we compute that in each of the four cells the deficit 1-p-q-r-s is 1/2. The weights on the edges (1's and 1/2's) give us creation biases (1/2)/(3/4) = 2/3 and (1/4)/(3/4) = 1/3. So we increment the 1/2's (and the 0's that are opposite them) by (1/2)(2/3) = 1/3, and we increment the other 0's by (1/2)(1/3) = 1/6, obtaining

```
      / \             / \
    /     \         /     \
  /5/6   1/6\     /1/6   5/6\
/             \ /             \
\             / \             /
  \1/6   1/3/     \1/3   1/6/
    \     /         \     /
      \ /             \ /
      / \             / \
    /     \         /     \
  /1/6   1/3\     /1/3   1/6\
/             \ /             \
\             / \             /
  \5/6   1/6/     \1/6   5/6/
    \     /         \     /
      \ /             \ /
```

Imbed this in the graph of order 3, and swap the numbers in each cell:

```
      / \             / \             / \
    /     \         /     \         /     \
  /5/6    0 \     /1/6   1/6\     / 0    5/6\
/      .      \ /      .      \ /      .      \
\             / \             / \             /
  \ 0     0 /     \ 0     0 /     \ 0     0 /
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  /1/6    0 \     /1/3   1/3\     / 0    1/6\
/      .      \ /      .      \ /      .      \
\             / \             / \             /
  \1/6    0 /     \1/3   1/3/     \ 0    1/6/
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  / 0     0 \     / 0     0 \     / 0     0 \
/      .      \ /      .      \ /      .      \
\             / \             / \             /
  \5/6    0 /     \1/6   1/6/     \ 0    5/6/
    \     /         \     /         \     /
      \ /             \ /             \ /
```

The four corner-cells have deficit 1/6, the other four outer cells have deficit 2/3 and the inner cell has deficit -1/3 (or surplus +1/3). So we do our adjustments, this time with all creation biases equal to 1/2 (except in the corners, where the creation biases are 0 and 1):

```
      / \             / \             / \
    /     \         /     \         /     \
  / 1     0 \     /1/2   1/2\     / 0     1 \
/      .      \ /      .      \ /      .      \
\             / \             / \             /
  \ 0    1/6/     \1/3   1/3/     \1/6    0 /
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  /1/2   1/3\     /1/6   1/6\     /1/3   1/2\
/      .      \ /      .      \ /      .      \
\             / \             / \             /
  \1/2   1/3/     \1/6   1/6/     \1/3   1/2/
    \     /         \     /         \     /
      \ /             \ /             \ /
      / \             / \             / \
    /     \         /     \         /     \
  / 0    1/6\     /1/3   1/3\     /1/6    0 \
/      .      \ /      .      \ /      .      \
\             / \             / \             /
  \ 1     0 /     \1/2   1/2/     \ 0     1 /
    \     /         \     /         \     /
      \ /             \ /             \ /
```

Lo and behold, these are the true probabilities associated with the original graph. That is, if I take a uniform random matching of the 4-by-4 grid, these are the probabilities with which I'll see the respective edges occur in the matching.

This method can be applied to hexagon graphs, fortress graphs --- you name it! In fact, one of the higher-priority applications of the method is to find a recurrence governing edge-probabilities for matchings of the fortress (by first converting the unweighted fortress graphs into weighted Aztec graphs); this ought to give us a conjectural formula for the curve bounding the temperate zone.

Incidentally, this method of computing probabilities is just a generalization of the scheme that Dan Ionescu hatched a few years ago for Aztec diamonds.

Part 3: Urban renewal lets you generate random perfect matchings of arbitrary bipartite planar graphs.

The ideas mentioned in Parts 1 and 2 also yield a way of generating a random perfect matching of an arbitrary bipartite planar graph, with arbitrary weights on the edges. This amounts to nothing more than a weighted version of the shuffling algorithm introduced in Elkies, Kuperberg, Larsen, and Propp.

[Note: Where I wrote "matchings" in the first two Parts, I meant "perfect matchings". Here I'll be more careful in my use of the term "matching", since I'll need to talk about partial matchings as well as perfect ones.]

As in Parts 1 and 2, it suffices to consider the case of a weighted Aztec diamond graph. Before one can begin to generate random perfect matchings of a weighted Aztec diamond graph of order n, one must apply the reduction algorithm from my first message, obtaining weighted Aztec diamond graphs of orders n-1, n-2, etc. But once this has been done, generating random perfect matchings of the graph is quite easy. The algorithm is an iterative one: starting from a Aztec diamond graph of order 0, one successively generates random perfect matchings of the graphs of orders 1, 2, 3, etc., using the weights that one computed during the reduction-process. At each stage, the probability of seeing any particular perfect matching can be shown to be proportional to the weight of that matching (where the constant of proportionality naturally changes as one progresses to larger and larger Aztec diamond graphs).

Here's how one iteration of the procedure works. Given a perfect matching of the Aztec diamond graph of order k-1, imbed the matching inside the Aztec diamond graph of order k, so that you have a partial matching of an graph of order k. (Note that the 4-cycles that were cells in the small graph become the holes between the cells in the large graph). In the new enlarged graph, find all (new) cells that contain two matched edges and delete both edges. Then replace each edge in the resulting matching with the edge opposite it in its cell. It can be shown that the complement of the resulting matching (that is, the graph that remains when the matched vertices and all their incident edges are removed) can be covered by 4-cycles in a unique way (and it's easy to find this covering just by making a one-time scan through the graph). Each such 4-cycle has weights attached to each of its 4 edges, so we can choose a random perfect matching of the 4-cycle using the weights to determine the respective probabilities of the two different ways to match the cycle. Taking the union of these new edges with the edges that are already in place, we get a perfect matching of the Aztec diamond graph of order k.

Let's see how this works with the 4-by-4 grid (for which we've already worked out the reduction process). To generate a random perfect matching, we start with the empty matching of the graph of order 0, imbed it in the graph of order 1, and find that the complementary graph of the empty matching is a single 4-cycle, which we match in one of two possible ways. In this case, each of the four edges has weight 4/3, so both matchings have equal likelihood; say we choose

```
                                    o
                                   /
                                 /
                               /
                             /
                            o       .       o
                                           /
                                         /
                                       /
                                     /
                                    o
```

Now we imbed this matching in the Aztec diamond graph of order 2:

```
        o       o



o       .       o       .       o
               /
             /
           /
         /
        o       o
               /
             /
           /
         /
o       .       o       .       o




        o       o
```

Neither of the two edges needs to be destroyed, because they belong
to different cells. Replacing each by the opposite edge in its cell,
we get

```
                            o               o
                           /
                         /
                       /
                     /
                    o       .       o       .       o




                            o               o




                    o       .       o       .       o
                                                   /
                                                 /
                                               /
                                             /
                            o               o
```

The complementary graph has a unique cover by 4-cycles. In each 4-cycle, three edges have weight 1/2 and one has weight 1, so we're twice as likely to see the one of the matchings as the other. Let's suppose that when we choose matchings with suitable bias, we get the more likely of the two matchings in both cells (as happens 4/9 of the time). Thus we have

```
        o               o
       /                 \
     /                     \
   /                         \
 /                             \
o       .       o       .       o
                 \
                   \
                     \
                       \
        o               o
         \
           \
             \
               \
o       .       o       .       o
 \                             /
   \                         /
     \                     /
       \                 /
        o               o
```

Now we imbed the matching in the graph of order 3:

```
        o               o               o




o       .       o       .       o       .       o
               /                 \
             /                     \
           /                         \
         /                             \
        o               o               o
                         \
                           \
                             \
                               \
o       .       o       .       o       .       o
                 \
                   \
                     \
                       \
        o               o               o
         \                             /
           \                         /
             \                     /
               \                 /
o       .       o       .       o       .       o




        o               o               o
```

This time we must delete the two central edges, because they belong to the same cell. After we have done this, we replace each of the four remaining edges with the opposite edge of its 4-cycle, obtaining

```
        o               o               o
       /                                 \
     /                                     \
   /                                         \
 /                                             \
o       .       o       .       o       .       o




        o               o               o




o       .       o       .       o       .       o




        o               o               o




o       .       o       .       o       .       o
 \                                             /
   \                                         /
     \                                     /
       \                                 /
        o               o               o
```

There is a unique cover of the complement by 4-cycles. As it happens, each edge in the complement has weight 1 in the (original) weighted Aztec diamond of order 3, so we can use a fair coin to decide how to match each of the 4-cycles.

Part 4. Some proofs

There are three claims to be proved. The first concerns weighted enumeration of matchings; the second concerns edge-probabilities; the third concerns random generation. In this posting, I'll only treat the first of the three.

To prove the claim, we begin with a lemma (the "urban renewal lemma"): If you have a weighted graph G that contains the local pattern

```
        \|/
       A o
         |
         |
         |
         o
        / \
      w/   \x
\B    /     \    C/
-o---o       o---o-
/     \     /     \
      y\   /z
        \ /
         o
         |
         |
         |
       D o
        /|\
```

(here A,B,C,D are vertex labels and w,x,y,z are edge weights, with unmarked edges having weight 1) and you define G' as the graph you get when you replace this local pattern by

```
        \|/
       A o
        / \
       /   \
      /     \
   w'/       \x'
    /         \
   /           \
\B/             \C/
-o               o-
/ \             / \
   \           /
    \         /
   y'\       /z'
      \     /
       \   /
        \ /
       D o
        /|\
```

with
w' = z/(wz+xy),
x' = y/(wz+xy),
y' = x/(wz+xy), and
z' = w/(wz+xy),
then the sum of the weights of the matchings of G equals wz+xy times
the sum of the weights of the matchings of G'. The proof is verification:

```
For each possible subset S of {A,B,C,D},
    check that
the total weight of the matchings of G in which
    the vertices in S are matched with vertices inside the patch
    and the vertices in {A,B,C,D}\S are matched outside the patch
equals wz+xy times the total weight of the matchings of G' in which
    the vertices in S are matched with vertices inside the patch
    and the vertices in {A,B,C,D}\S are matched outside the patch.
```

(I hope my poetic page-layout makes the preceding awful sentence a bit
easier to parse, if not digest.) As it turns out, 10 of the 2^4=16 cases
are trivial, and of the remaining 6, four are related by symmetry, so it
comes down to three cases.

S = {A,B}: If a matching of G matches A,B inward and C,D outward, we can make a matching of G' by getting rid of the edge of weight z that the matching must contain and replacing it by an edge of weight w' = z/(wz+xy). When we do this, the weight goes down by a factor of wz+xy.

S = {A,B,C,D}: This time, a single matching of G (of weight Q, say) corresponds to two matchings of G', one of weight w'z'Q and the other of weight x'y'Q. The combined weight of the two matchings is (w'z'+x'y')Q = Q/(wz+xy), so the weight has again gone down by a factor of wz+xy.

S = {} (empty set): Here two matchings of G, of weight wzQ and xyQ, are replaced by a single matching of weight Q. Q may vary, but the ratio of the weights remains 1 to wz+xy.

This completes the proof of the lemma.

Now, to prove the Aztec reduction theorem, suppose we have a weighted Aztec graph of order n.

```
   o   o   o
  / \ / \ / \
 o   o   o   o
  \ / \ / \ /
   o   o   o
  / \ / \ / \
 o   o   o   o    (n=3; weights not shown)
  \ / \ / \ /
   o   o   o
  / \ / \ / \
 o   o   o   o
  \ / \ / \ /
   o   o   o
```

We perversely begin our reduction by splitting each vertex into three vertices:

```
      o       o       o
      |       |       |
      o       o       o
      |       |       |
      o       o       o
     / \     / \     / \
o-o-o   o-o-o   o-o-o   o-o-o
     \ /     \ /     \ /
      o       o       o
      |       |       |
      o       o       o
      |       |       |
      o       o       o
     / \     / \     / \
o-o-o   o-o-o   o-o-o   o-o-o
     \ /     \ /     \ /
      o       o       o
      |       |       |
      o       o       o
      |       |       |
      o       o       o
     / \     / \     / \
o-o-o   o-o-o   o-o-o   o-o-o
     \ /     \ /     \ /
      o       o       o
      |       |       |
      o       o       o
      |       |       |
      o       o       o
```

It's easy to convince oneself that this vertex-splitting doesn't change the sums of the weights of the matchings, as long as the new edges we add are given weight 1.

Now we apply urban renewal n^2 times to get a graph of the form

```
      o       o       o
      |       |       |
      o       o       o
     / \     / \     / \
    /   \   /   \   /   \
   /     \ /     \ /     \
o-o       o       o       o-o
   \     / \     / \     /
    \   /   \   /   \   /
     \ /     \ /     \ /
      o       o       o
     / \     / \     / \
    /   \   /   \   /   \
   /     \ /     \ /     \
o-o       o       o       o-o
   \     / \     / \     /
    \   /   \   /   \   /
     \ /     \ /     \ /
      o       o       o
     / \     / \     / \
    /   \   /   \   /   \
   /     \ /     \ /     \
o-o       o       o       o-o
   \     / \     / \     /
    \   /   \   /   \   /
     \ /     \ /     \ /
      o       o       o
      |       |       |
      o       o       o
```

with new weights w',x',y',z' replacing the old weights w,x,y,z (which I should really write with subscripts i,j ranging from 1 to n, since they're permitted to be distinct from one another). Here's the coup de grace: The pendant edges that we see must belong to EVERY perfect matching of the graph, so we can delete them from the graph, obtaining an Aztec graph of order n-1:

```
    o       o
   / \     / \
  /   \   /   \
 /     \ /     \
o       o       o
 \     / \     /
  \   /   \   /
   \ /     \ /
    o       o
   / \     / \
  /   \   /   \
 /     \ /     \
o       o       o
 \     / \     /
  \   /   \   /
   \ /     \ /
    o       o
```

The theorem now follows.

One can prove the "fortress" theorem by similar means. Here, the initial graph is

```
                          o           o
                          |           |
                          o     o     o
                         / \   / \   / \
                      o-o   o-o   o-o   o-o
                         \ /   \ /   \ /
                          o     o     o
                          |     |     |
                          o     o     o
                         / \   / \   / \
                        o   o-o   o-o   o
                         \ /   \ /   \ /
                          o     o     o
                          |     |     |
                          o     o     o
                         / \   / \   / \
                      o-o   o-o   o-o   o-o
                         \ /   \ /   \ /
                          o     o     o
                          |           |
                          o           o
```

One's first impulse is to remove the pendant edges, but it's better still to apply urban renewal to every OTHER "city" in this graph. Then you get an Aztec diamond graph in which roughly half of the edges have weight 1 and the rest have weight 1/2. The same techniques that were used above can also be applied here to allow one to conclude that the number of matchings of such a graph is a certain power of 5 (or twice a certain power of 5). The "5" comes from the cell-factor (1)(1)+(1/2)(1/2) (when multiplied by 4), just as (in the unweighted Aztec diamond theorem) the "2" comes from the cell-factor (1)(1)+(1)(1).

Part 5. More proofs

Now I’ll prove the claim about edge probabilities in weighted Aztec diamond graphs (that is, I’ll prove that the iterative scheme for computing these probabilities that I described last week does in fact work). Here, too, we need a lemma, but it’s just a mild generalization of a result whose proof I’ve already posted to domino (back when I called it the "4-gon conclusion"): Say we are given a weighted bipartite planar graph G containing a 4-cycle of the form

```
        \|/
       A o
        / \
       /   \
      /     \
w [p]/       \x [q]
    /         \
   /           \
\B/             \C/
-o               o-
/ \             / \
   \           /
    \         /
y [r]\       /z [s]
      \     /
       \   /
        \ /
       D o
        /|\
```

Here w,x,y,z are weights and p,q,r,s are the probabilities of the respective edges belonging to a random weighted matching of G if each matching is chosen with probability proportional to its weight (which is just the product of the weights of its edges). Then I claim that the probability of a random matching of G containing both edge AB and edge CD is ps + qr(wz/xy), while the corresponding probability for edges BC and DA is qr + ps(xy/wz).

Note, as a plausibility check, that the ratio of the two probabilities is wz/xy, which is the ratio of the weights of any two matchings that differ only in that one contains AB and CD while the other contains BC and DA.

I won't write up the proof of the lemma here. You can prove it using the same reasoning that I used two-three years ago, or you can derive it from the known, unweighted result by permitting multiple edges of weight 1 to connect a pair of vertices, deducing the result for integer weights, deducing the result for rational weights (by scaling), and then deducing the result for real weights (by continuity).

Note that the sum of the two probabilities is (wz+xy)(ps/wz + qr/xy).

Now let's go back and give urban renewal a fresh look. Here are the graphs G and G', with edges marked with both their weight and their probability (in the format "weight [probability]").

G:

```
        \|/
       A o
         |
         |
         |
         o
        / \
  w [p]/   \x [q]
\B    /     \    C/
-o---o       o---o-
/     \     /     \
  y [r]\   /z [s]
        \ /
         o
         |
         |
         |
       D o
        /|\
```

(The unmarked edges have weight 1, and their probabilities are 1-p-q, 1-q-s, et cetera.)

G':

```
          \|/
         A o
          / \
         /   \
        /     \
w' [p']/       \x' [q']
      /         \
     /           \
  \B/             \C/
  -o               o-
  / \             / \
     \           /
      \         /
y' [r']\       /z' [s']
        \     /
         \   /
          \ /
         D o
          /|\
```

Recall that we're in the situation where we know p', q', r', and s', and are trying to compute p, q, r, and s. Recall also that we have w' = z/(wz+xy), x' = y/(wz+xy), y' = x/(wz+xy), and z' = w/(wz+xy).

Here's our plan: First, we'll work out the probabilities of all the local patterns in G'. Then we'll use the urban renewal correspondence to deduce the probabilities of the local patterns in G. Finally, we'll deduce the probabilities of the individual edges in G.

I'm going to write p',q',r',s',w',x',y',z' as P,Q,R,S,W,X,Y,Z, to save typing and eye-strain. Keep in mind the motivation that the primed quantities come from a LARGER Aztec graph than the unprimed.

Let S denote the set of vertices in {A,B,C,D} that are matched to a vertex inside the patch. Then here are the respective probabilites of the local patterns in G':

```
S={A,B,C,D}: (WZ+XY)(PS/WZ+QR/XY)
S={A,B}:     P-PS-QR(WZ/XY)
S={B,C}:     similar
S={C,D}:     similar
S={D,A}:     similar
S={}:        1-P-Q-R-S+(WZ+XY)(PS/WZ+QR/XY)
```

(The second formula is obtained by recalling that the probability of edge AB being present in a random matching, which can happen in two ways according to whether or not edge CD is present, must be P; the last formula is obtained by recalling that the probabilities of the six local configurations must sum to 1.)

It follows from the last of these, and the urban renewal correspondence, that the probability that a random matching of G contains two edges from the 4-cycle ABCD is 1-P-Q-R-S+(WZ+XY)(PS/WZ+QR/XY). The probability that a random matching of G contains edges AB and CD must be equal to this quantity times wz/(wz+xy) = WZ/(WZ+XY), which gives (WZ/(WZ+XY))(1-P-Q-R-S) + PS + QR(WZ/XY). The probability that a random matching of G contains edge AB but *not* edge CD (by another application of the urban renewal correspondence) equals the probability that a random matching of G' contains edge CD but not edge AB, which is S - PS - QR(WZ/XY).

Adding, we find that the probability that a random matching of G contains edge AB is S + (WZ/(WZ+XY))(1-P-Q-R-S). Thus we conclude that

```
           wz
p = S + (-----) (1 - P - Q - R - S) ,
         wz+xy

           ^             ^
           |             |
           |             |
    creation bias     deficit
```

and similar formulas hold for the other edges.

Part 6. Final proofs

Now I'll show that (generalized) domino shuffling works.

First, let's note that the shuffling algorithm I mentioned last week really *is* a form of domino shuffling, as described in "Alternating-sign matrices and domino tilings" by Elkies, Kuperberg, Larsen, and myself. [If you're unfamiliar with that article's treatment of the shuffling algorithm, you can skip this paragraph, since my whole point is that we DIDN'T understand what shuffling really was about back then.] The operation of removing from a matching those matched edges that belong to the same cell as another matched edge corresponds to the process of removing odd blocks to obtain an odd-deficient tiling; the operation of swapping the remaining edges to the opposite side of the cell corresponds to the process of shuffling dominoes (resulting in the creation of an odd-deficient tiling of a larger Aztec diamond); and the operation of introducing new edges corresponds to the creation of new two-by-two blocks.

Next, let's observe that a domino tiling of the Aztec diamond of order n determines an n-by-n alternating-sign matrix (as described in Elkies et al.), which we can understand in terms of matchings of the dual graph. Specifically, for each of the n^2 cells of the dual graph, write down the number of edges from that cell that participate in the matching and subtract 1. For instance:

```
    o       o       o       o
   /       /         \       \
  /       /           \       \
 /       /             \       \
o   0   o  +1   o   0   o   0   o
               /
              /
             /
    o       o       o       o
     \               \       \
      \               \       \
       \               \       \
o  +1   o  -1   o  +1   o   0   o
 \               \
  \               \
   \               \
    o       o       o       o
           /               /
          /               /
         /               /
o   0   o   0   o   0   o  +1   o
 \               \             /
  \               \           /
   \               \         /
    o       o       o       o
           /
          /
         /
o   0   o  +1   o   0   o   0   o
 \             /       /       /
  \           /       /       /
   \         /       /       /
    o       o       o       o
```

Label the weights of the edges of the cell in row i and column j with w_{ij}, x_{ij}, y_{ij}, and z_{ij}, thus:

```
      o
     / \
   w/   \x
   /     \
  o       o
   \     /
   y\   /z
     \ /
      o
```

Define also a "cell weight" D_{ij} = w_{ij} z_{ij} + x_{ij} y_{ij}. Call two edges "parallel" if they belong to the same cell.

The bias in the creation rates for generalized shuffling guarantee that if two perfect matchings of the Aztec graph of order n are associated with the same n-by-n alternating-sign matrix, then their RELATIVE probabilities are correct. So all we need to do is verify that the aggregate probability of these matchings (for fixed alternating-sign matrix A) is what it should be. Equivalently, one can simply consider partial matchings of the Aztec graph of order n, in which all parallel edges have been removed. The weight of all such partial matchings should be defined as the sum of the weights of all perfect matchings that extend it; it is equal to a product of edge-weights of the form w_{i,j}, x_{i,j}, y_{i,j}, or z_{i,j}, together with some cell-weights D_{i,j} corresponding to those cells in which two parallel edges canceled each other. Note that the locations (i,j) that contributes a cell-weight are exactly those locations in which the alternating-sign matrix A has a +1; the locations (i,j) that contribute an edge-weight are exactly those locations in which A has a 0; and the remaining locations (i,j), which make no contribution to the weight of the partial matching, are those in which A has a -1.

Now let's run shuffling backwards, to see where all the partial matchings associated with the alternating-sign matrix A "came from". Each edge in the partial matching of the Aztec graph of order n came from an edge in a partial matching of the Aztec graph of order n, whose weight is the same except for a factor of 1/D_{i,j}. These factors of 1/D_{i,j} --- associated with those locations in A in which a 0 occurs --- cancel some of the n^2 factors in the product of the D_{i,j}'s that "go in the front" for the process of urban renewal. Since the matching is only partial, we need to multiply its weight by further factors that take into account the missing edges in the matching of the (smaller) graph, which no longer are associated with cells but rather with "co-cells" (the spaces between the cells). These weight factors are easily checked to be of the form 1/D_{i,j}, where (i,j) is a location in which A had a -1. The only uncancelled factors are those cell-factors D_{i,j} for which (i,j) is the location of a 1 in the alternating-sign matrix A.

The upshot is, if you define the weight of a partial matching as the sum of the weights of the matchings that extend it, then the assignment of weights in urban renewal has the property that the weight of a partial matching of the Aztec diamond graph of order n associated with the alternating-sign matrix A equals
D_{1,1} D_{1,2} ... D_{n,n}
times the weight of the partial matching of the graph of order n-1 that is paired with it via shuffling. This factor is independent of A, so the relative weights of all the partial matchings is correct.

It follows that if you take a random matching of the smaller graph (in accordance with the edge weights given by urban renewal) and apply destruction, shuffling, and creation, then you'll get a random matching of the larger graph.

Part 7. Last thoughts

Two last thoughts concerning urban renewal, domino-shuffling and whatnot.

1) I think there might be a continuum analogue for the weighted version of shuffling, generalizing William Jockusch's PDE approach. (It's a shame that none of that work got written down, since it might be the easiest way to understand certain things.) This might even give us a spatial PDE for placement-probabilities, which is one of the Big Things that we're after. Does anyone have any thoughts about this? An interesting feature of the problem is that the evolution of the system (from its "big bang" at time zero to its steady-state at time infinity) depends on the edge-weights, which themselves are determined by running a different but related process *backwards* in time.

2) Can anyone think of a way to extend urban renewal to matchings of NON-bipartite planar graphs, or equivalently, can anyone come up with a form of routinized row- and column-reduction for Kasteleyn's Pfaffians?