Cutting a geometric figure into two Pieces

By Douglas A. Engel
9442 US 85 N
Littleton, CO 80125

With comments and proofs by Jaap Scherphuis
(See http://www.jaapsch.net)

Feb. 2, 2010
http://www.Puzzleatomic.com

Ecutmax

The problems here were suggested by trying to find new ways to make various kinds of puzzles by cutting certain kinds of figures. The big surprise is that the solutions result in Hamilton and Hamilton-like cycles on polyhedrons. I am very grateful that Jaap Sherphuis has added some of his analytical comments and proofsa and helped correct some mistakes in the figures. His website is a tour de force of sequential move puzzles. It has mathematical expose's and algorithmic solutions of a great many of the marketed puzzles out there. You can spend hours and days exploring this continuing work.

One statement of the problem is to take a tetrahedron that is formed only of its edges(a tetrahedron network) and determine the maximum number of edges you must cut through to form two separate pieces. Call this Ecutmax. Every cut must be necessary. If any one of the cuts is removed the network remains in one piece. If any new cut is added it forms three separate pieces. Figure 1 shows this process for a tetrahedron where Ecutmax = 4. Ecutmax for a convex polyhedron is either equal to or less than its number of faces, Ecutmax <= F.

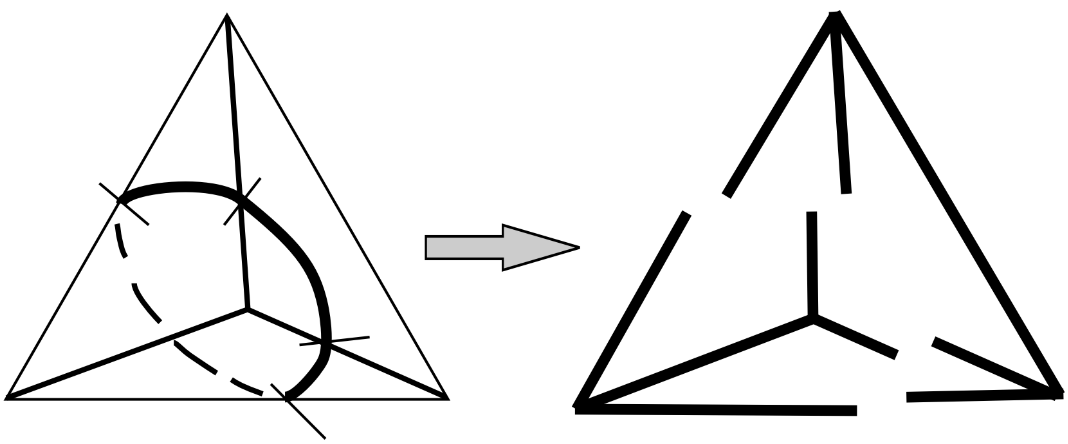


Fig. 1

Shown above on the left is a tetrahedron frame work that is made up only of edges. It can be cut in two by cutting through 4 of the edges as is shown in the right hand figure above.

In Figure 2 an octahedron is shown with eight cuts. The octahedron is drawn as a planar network where the area surrounding the figure is also considered to be a face.

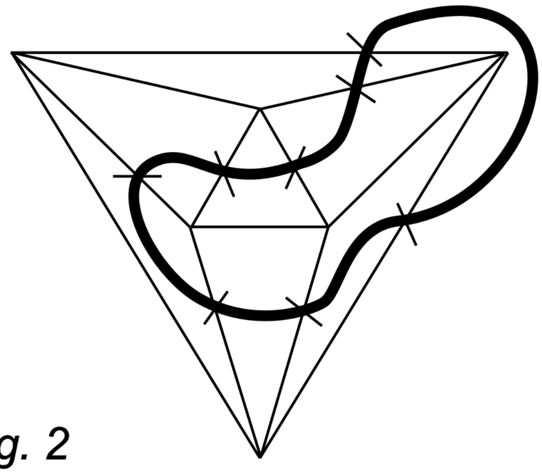


Fig. 2

`You can cut an octahedron network in two with 8 edge cuts maximum which is equal to its number of faces. If any one cut were removed a single connected figure would result.`

Figure 3 similarly shows a way of cutting a cube. The illustration shows the cube drawn in perspective as well as in a planar rendition.

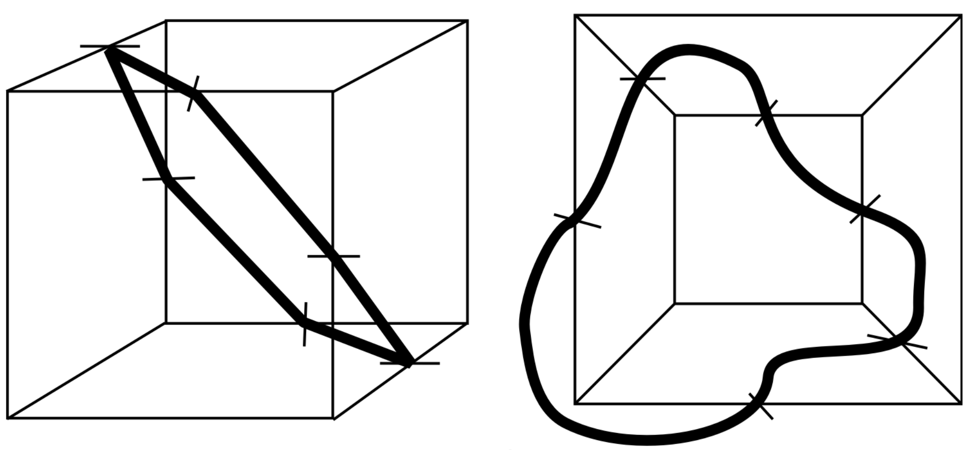


Fig. 3

`A cube requires a maximum of 6 cuts to sever the edge network into two pieces. The cuts can be seen as a recursive path going once through each face.`

Figures 4 and 5 show ways to cut the regular dodecahedron and icosahedron networks into two pieces. In each case the cut path is recursive and visits each face once and has Ecutmax equal to the number of faces. Finding such a cut path for the icosahedron is more difficult. It illustrates that finding such a recursive path can be more difficult as the network increases in complexity.

We would also like to know how many unique solutions exist for each of the regular polyhedrons. For instance the tetrahedron has only one unique solution. This is a problem already solved somewhere in the literature because Ecutmax is just a Hamilton cycle through the faces of a polyhedron. Note that if it is a figure composed of faces as well as edges then the Ecutmax path still works by thinking of it as cutting through the faces as well as the edges of the polyhedron.

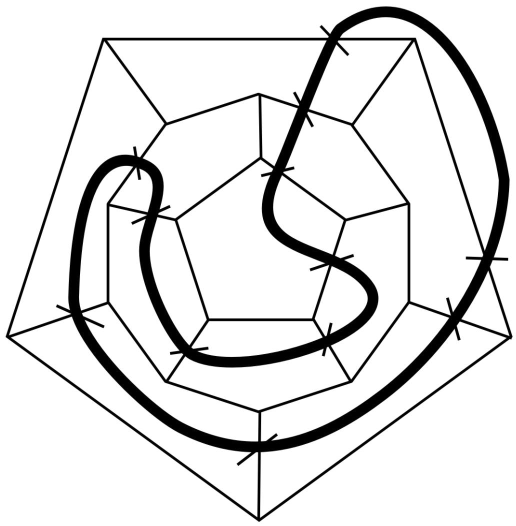


Fig. 4

```
This shows that a regular
dodecahedron can be cut in two
pieces with a maximum of 12 cuts.
```

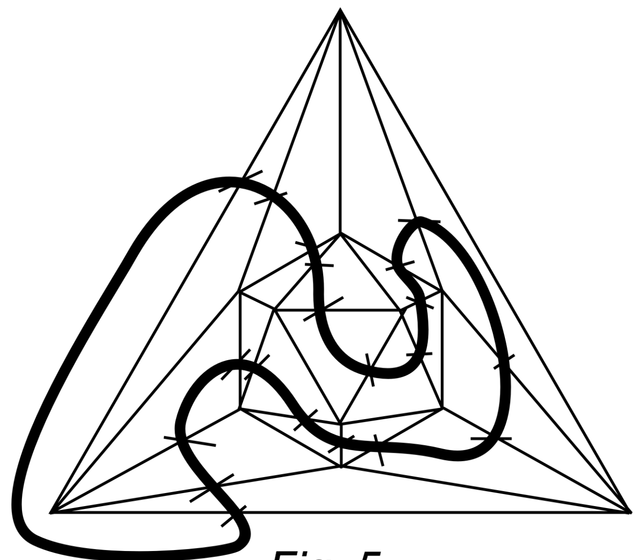


Fig. 5

```
A method of cutting a regular icosahedron in
two pieces with a maximum of 20 cuts is shown
above.  This recursive path was more difficult
to find. Is such a path always possible?
```

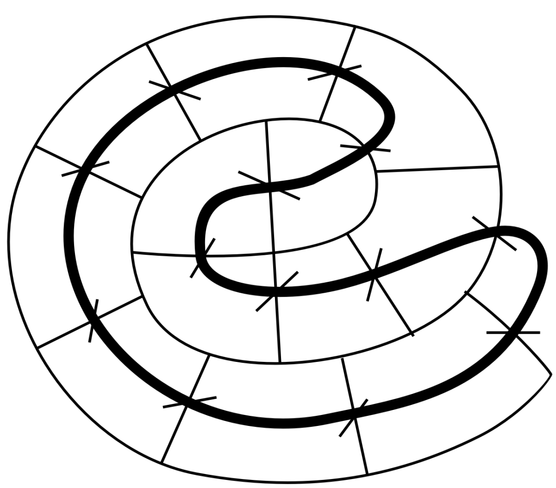


Fig. 6

```
This planar edge network has 13 faces
and requires 13 cuts maximum to cut in
two.  The area surrounding it is
considered to be a face.
```

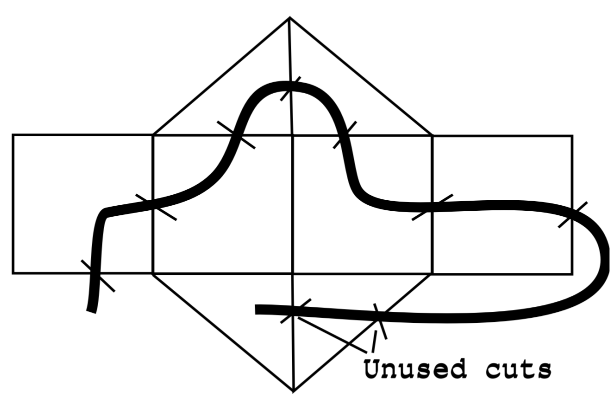


Fig. 7

```
Example of a network where a reccursive
cut path that visits every face exactly
once is not possible.  Its Ecutmax is less
than the number of faces and does not
divide every face of the network into two
connected parts, each with the same no. of
faces.
```

Figure 6 shows a more complicated figure but it also has an Ecutmax equal to its number of faces . Do examples exist where the Ecutmax path is non recursive? Figure 7 shows a network where the Ecutmax path does not go thru every face, so that Ecutmax < F.

The same question can be asked of any kind of edge connected network where the edge connection complexity cannot be seen as a simple set of faces. It then becomes a generalized network cutting problem. Ecutmax for a network like the internet would be a fair sized number that is constantly changing

Fcutmax

The above problem suggests going up a dimension. What is the maximum number of face cuts required to sever a tetrahedron into two pieces. Now we will see the faces as being part of the connectivity of the figure. We require that the cuts made in each face go from the midpoint of one edge to the midpoint of another edge. Without this requirement the maximum number of cuts could be made infinite by using a helical pattern of cuts. Once again each cut must be necessary to the dissection. Removing any one cut would leave a single connected figure. The problem of cutting faces is interesting to a puzzle maker since all sorts of different ways of cutting the faces of a cube exist. Figure 8 shows a tetrahedron cut in two pieces with 6 face cuts.

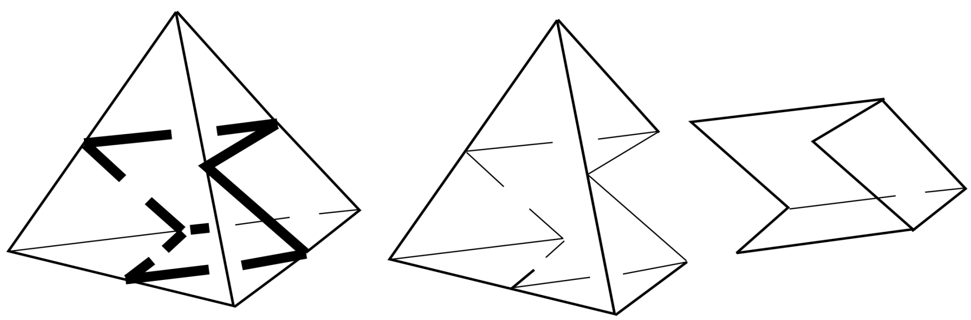


Fig. 8

A tetrahedron is cut into two pieces by a maximum of 6 face cuts. The face cuts are equal to the number of edges of the tetrahedron and form a recursive path. All 6 cuts are necessary.

Figure 9 shows an octahedron cut in two with 12 face cuts and Figure 10 shows this same process for a cube.

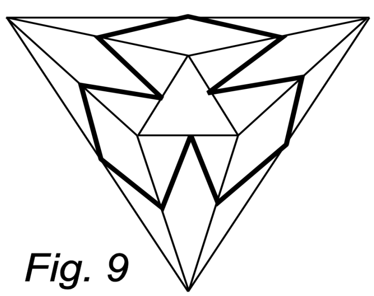


Fig. 9

You can cut a regular octahedron in two with 12 face cuts maximum, equal to its number of edges.

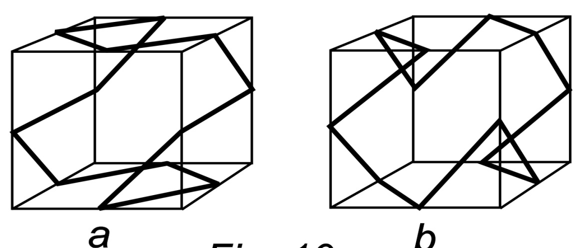


Fig. 10

Here a maximum 12 face cuts are made to cut a cube into two pieces. Figure 10a forms a box that can be opened. Figure 10b has a maximum 12 cuts and each face has two cuts. It was suggested by Jaap Scherphuis.

The best Fcutmax for a convex polyhedron is equal to the number of edges in the polyhedron, so Fcutmax = E. You would arrive at this conclusion by noting that a cut is allowed going to the midpoint of each edge. Thus each edge can add one cut to the cut path. Once again the best solution is to find a recursive path that visits each edge exactly once. If such a path exists then this is the best possible solution. Figures 11 and 12 show Fcutmax solutions for the dodecahedron and the icosahedron.

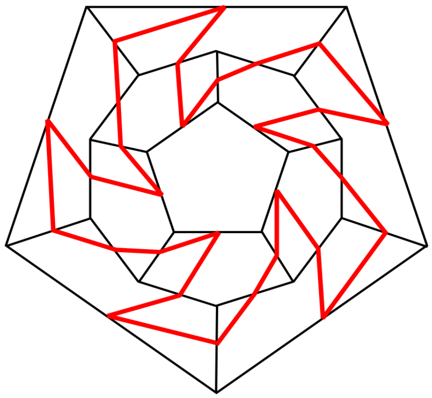


Fig. 11

This shows that a regular dodecahedron can be cut in two pieces with a maximum of 30 face cuts. Can the two pieces be separated?

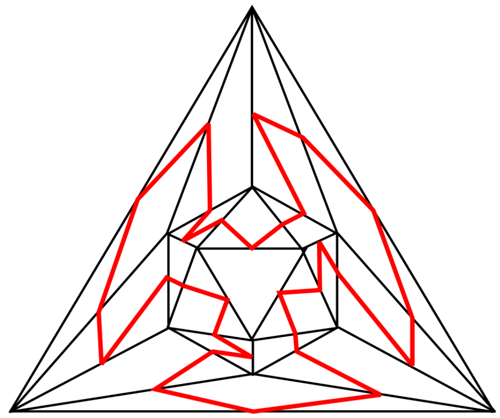


Fig. 12

Here is one way to cut a regular icosahedron in two pieces with a maximum of 30 face cuts. Can it be disassembled?

Nice box assembly puzzles can be made by allowing more cuts to the midpoints of the edges. The simplest rule is to first produce a Fcutmax dissection and then allow one extra cut arriving at an edge but not crossing the edge. In this case it can be seen that a 3 piece dissection of a cube will have one extra slice line on one face only or 13 cuts = 3 pieces.

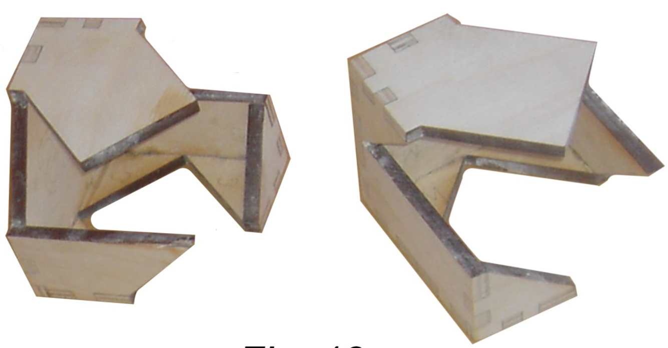


Fig. 13

A face cut cube with Fcutmax = 12. It can be disassembled as shown here.

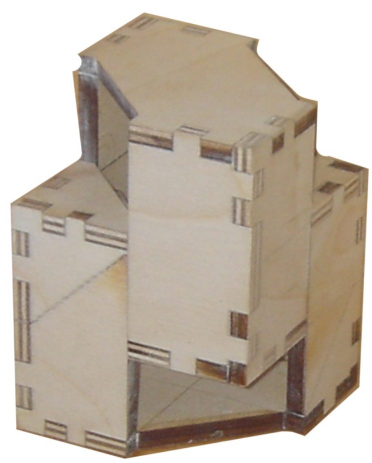


Fig. 14

A face cut cube with 8 face cuts. It does not satisfy Fcutmax. It can be slid open.

A four piece puzzle will have 14 cuts or Fcutmax(n) = 12 + (n-2) where n is the number of pieces. Thus for any standard polyhedron having Fcutmax = E we have Fcutmax(n) = E + (n-2) up to the maximum number of cuts allowed. This would not hold if cuts can cross each other. Similar results hold for dividing edge cut polyhedrons into more than 2 pieces. For instance if Ecutmax = F, then Ecutmax(n) = F +(n-2). Figure 13 shows an Fcutmax box that can be disassembled. On the other hand Figure 14 shows a non Fcutmax box that can also be disassembled.

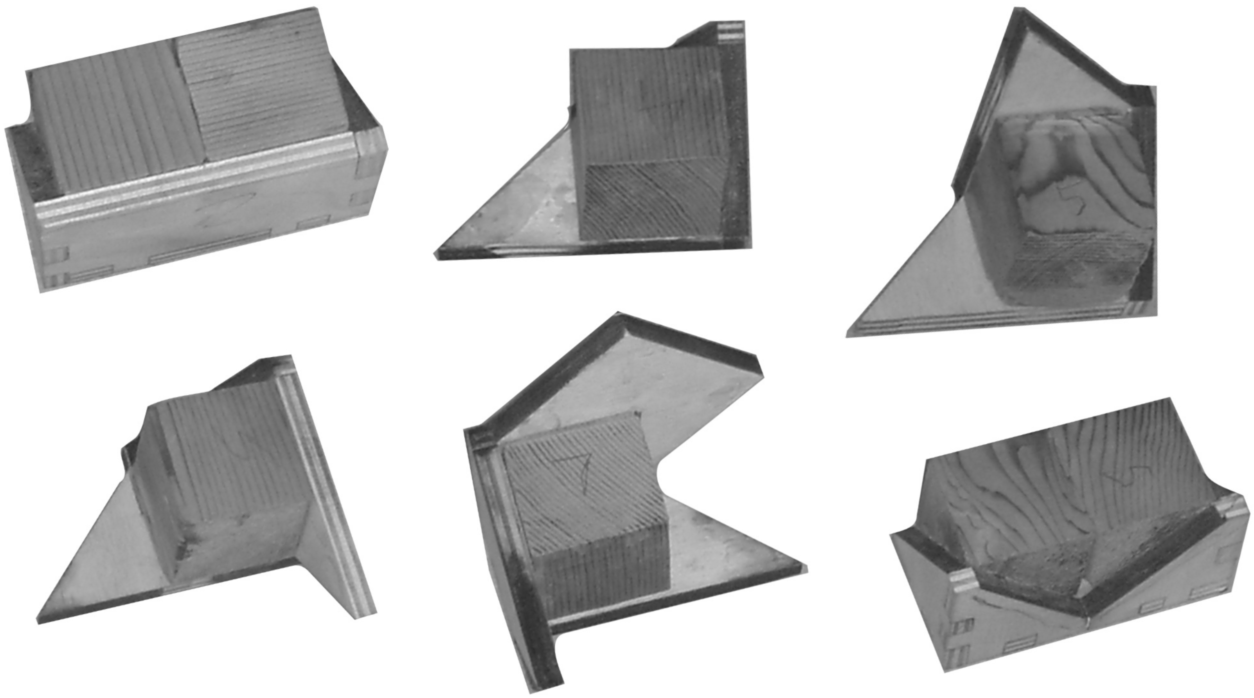


Fig. 15

A non recursive face cut cube puzzle with six pieces. Internal corner cubes added for stability and ease of assembly

The Puzzle dissecting design gets much more interesting if we do not restrict the number of cuts on each face. Figure 15 shows a wooden puzzle having a total of 14 face cuts resulting in 6 pieces. It does not have a recursive Fcutmax path and does not have a simple formula for number of pieces produced by x number of cuts, but it was inspired by Fcutmax. A corner cube has been glued into each cubic corner to make the puzzle more stable during assembly. These pieces can be assembled into a cube in more than one way. The pieces can be put together in many ways to simulate buildings and thus doubles as an architectural puzzle or toy. One or more of these sets is or will be sold on my website: http://www.Puzzleatomic.com

Vcutmax

The Hamilton cycle can be equated with a salesmans route. It visits every node of the network before returning to the starting node by traveling along the edges of the network which are the 'roads'. The salesmans problem is to find the shortest route that does this. Planar and standard polyhedral networks where the 'roads' or edges do not cross anywhere represent a simplified version of the problem. The salesmans problem now becomes the basis for a Vcutmax <= V cutting system. We cut along and down through the edges in a recursive path and therefore divide the network or polyhedron into two pieces. Every cut must be necessary so that more edge cuts create additional pieces. Box and pieces puzzles can easily be produced using a Vcutmax path as a starting point. The thick black line in Figure 16a shows this process for a dodecahedron. It also shows an Fcutmax path. The Vcut path alternately crosses over the Fcut path except where the Fcut path does not enter a face. In that case the Vcut path just moves along all nodes of the face before exiting it. Of course this is not the only way to make a Vcutmax path.

Four Coloring

In each one of the cut systems, Ecut, Fcut, and Vcut, two groups of faces are produced that have a branch like structure where all the branches are very thin due to the need for the paths to cover all elements of a given kind, faces, nodes, or edges. For instance a Vcut branch is never more than 1 face wide. Fcut and Ecut branches have similar thin widths. This gives us a possible system to try to find a simpler proof of the famous four color map coloring problem for a planar network. If four colors could be proven for the two thin branch like cutmax divisions perhaps it could be proven for the entire network. After making the above statements about 4 coloring Jaap informed me that a similar idea dominated the problem for many years.

Here are his comments;

"As you say a Vcut branch is 1 face wide. The two parts in fact have a tree-like shape. This means each can be two-colored, and so combine to give a four-coloring of the graph. This proves the four-color theorem for all planar graphs with a Hamilton cycle. The problem is that there isn't always a Hamilton cycle. The Rhombic Dodecahedron for example does not have one.

FYI, a bit of background on the four-coloring theorem:
They usually only consider cubic planar graphs, i.e. graphs with exactly 3 edges at each vertex. The reason is simple: any other planar graph can be turned into a cubic one that is at least as difficult to color. On the Rhombic Dodecahedron you can truncate all the corners where four faces meet so that you get a small square face there. Any vertex with n edges can be replaced by an extra face with n sides, with n new vertexes that each have 3 edges. Any coloring of the faces of the new graph is also a coloring of the original graph.

Had Tait's Conjecture (that all planar cubic graphs have a Hamilton cycle) been true then it would have led to an immediate proof of the four-color theorem. With cubic graphs there is a simple construction of the four-coloring from the Hamilton cycle, as can be seen here:
http://demonstrations.wolfram.com/FourColorMaps/

The method shown there does not work on non-cubic graphs, but you can still make a two-coloring of the inside and of the outside of the cycle, and combine them to make a four-coloring. It is just a little harder to make that into an airtight proof."

Figure 16b shows one portion of an Fcutmax path having 21 faces or facelets. Figure 16c shows the other portion of the same Fcutmax path and it also has 21 facelets. Figure 16d shows one portion of a Vcutmax path having 6 pentagonal faces and Figure 16e shows the other portion of the same path having 6 pentagonal faces.

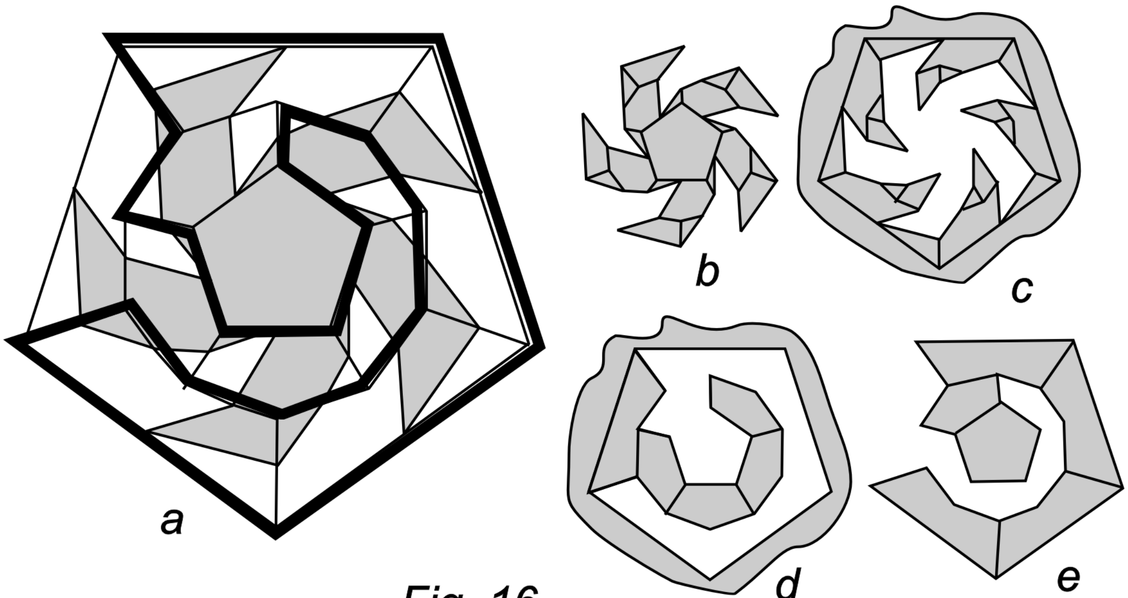


Fig. 16

The heavy black line represents a 'salesmans cut' or Hamilton cycle that visits each node once. It alternately croses the two Fcutmax areas except where a face has no cuts. In that case it traverses all the nodes of that face before leaving it. It cuts the faces into two separate pieces. Its formula is Vcutmax = V = 20. All cuts are necessary. Any additional V cuts will create additional pieces. Multipiece puzzles and boxes can be made using a polyhedral V cut system.

If we only allowed one straight cut to pass through each face of a face cutting, the ***maximim*** number of cuts would be equal to F instead of E. Every ***maximal*** face cutting of a planar network would produce the same number of facelets on each of the two cut halves. This is because the cut path divides each face into two parts. However as Jaap Scherpuis points out not every polyhedron has Hamilton cycle on its edges. He says;

"Just as the Rhombic Dodecahedron does not have a Hamilton cycle on its edges,its dual polyhedron, the cuboctahedron, does not have a closed path over the faces that visits each face exactly once. This is because the path must alternate over squares and triangle faces, and there are not the same number of each."

Solid cutmax in 3D

We can go up to three dimensions and ask what is the maximum number of edge cuts required to cut a 3D solid figure? A simple rule for a 3D solid such as a cube might be to follow the face cutting rule but cut to the center of the solid figure with each cut. Thus the resulting two pieces would simply be a solid version of the same two edge cut pieces. Figure 17a shows one such cutting for a cube. It is also possible to use the face cutting rule as shown in Figure 17b. Figure 17c is a vertex cutting for a cube. Finally Figure 17d shows a combination of face and vertex cuts used to produce two pieces that cannot be separated. It is non Hamiltonian in the sense that the cuts visit more than just the 8 vertexes of the cube. Some cuttings might leave the two pieces entangled or held in a knotted or linked system that can not be separated except by further dissection. For instance consider cutting a polyhedron with a hole in it. We could ask that the cutting allow the two pieces to be separated. This restriction might be interesting to a puzzle designer but might not be as interesting to a topologist. Perhaps more interesting to a puzzle designer would be to find a simple dissection of n pieces that can be disassembled.

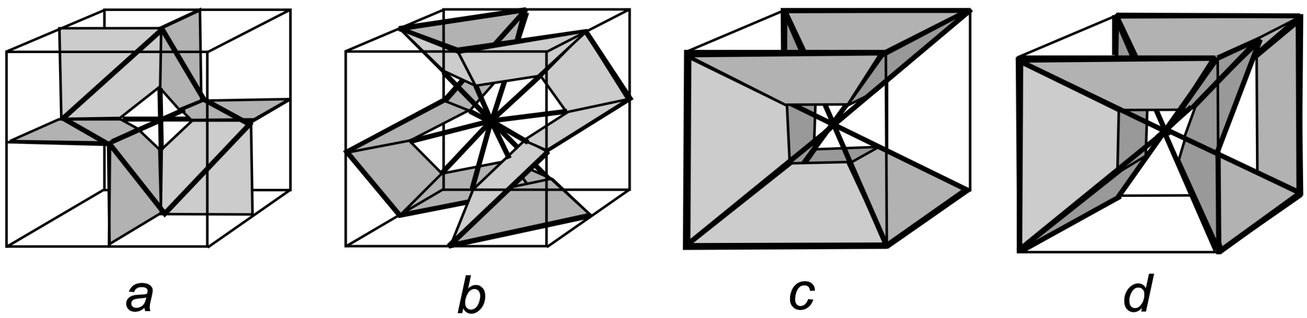


Fig. 17

This shows solid cutting. Figure a is an edge cut solid cube with 6 edge cuts. Figure b shows a face cut solid cube with 12 face cuts. Figure c shows a vertex cut solid cube with 8 cuts. The gray shaded cuts are only shown part way to center for clarity. All three examples can be separated after cutting. Figure d shows a vertex cut and face cut combined figure with 9 cuts. It is non Hamiltonian.

Planar Fcutmax in 4D

As a simple exercise Figure 18 shows a proposed face cutting for a hypercube. It is assumed that Figure 18 is an actual binary Fcutmax of a hypercube but this is certainly not proven as 1 or more flaws or mistakes could exist or the analogy with 3D figures does not work. Because three faces meet at each edge at least three cuts must meet at each edge whenever the main cut path passes from one face then over an edge and into another face to guarantee a maxcut. This extra cut can be called a ½ and must proceed through the third or common face to where another cut passes from one face over an edge to a different face. The main path should probably be drawn in such a way that it never makes two cuts simultaneously in one face. With this rule it is more difficult to find a Hamilton cycle. Since ½ cut extra exists at each edge, for a 4D hypercube the proposed formula is Fcutmax = E + E/2.

Figure 19a shows a perspective drawing of a 4D hypercube with a Hamilton cycle through the edges. This does not become Fcutmax until we add the twelve ½ cuts needed to create two separate pieces.

The extra cuts axiom or rule

At this point we might ask will the rule still hold that any more single face cuts will cut either of the two pieces in two after performing an Fcutmax on a hyper figure? As before all cuts of Fcutmax must be necessary. It is conjectured that rule **extra cuts yield extra pieces**, EC>>EP, will always hold for Fcutmax in any dimension since such a cut proceeds from one exposed cut edge to another exposed cut edge.

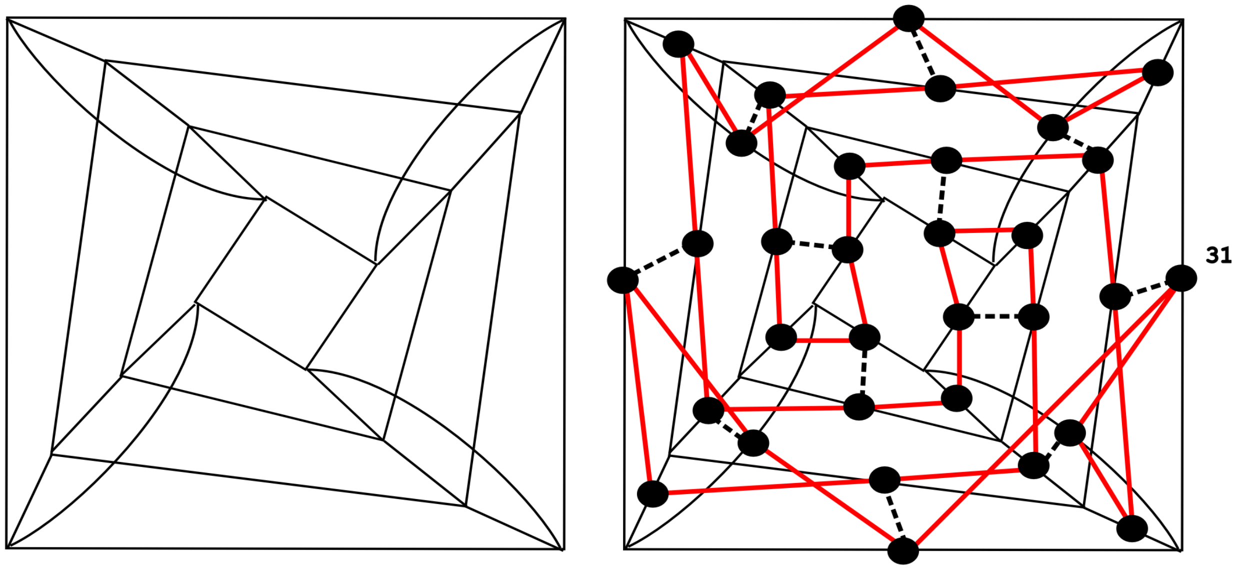
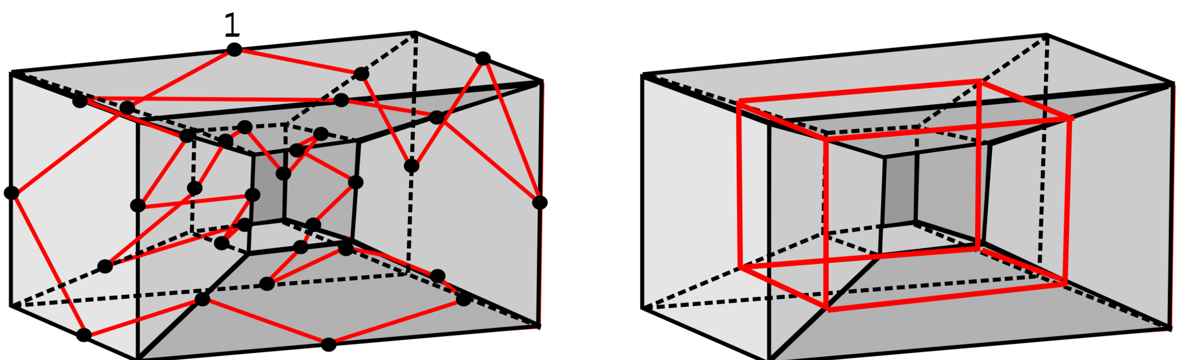


Fig. 18

A proposed Fcutmax = 48 for a hypercube. A hypercube has six 8 solid cubes, 24 square faces, 32 edges and 16 vertices. It is shown flattened as a planar network with crossing edges curved. It is twisted to clarify the cut path and the cubes with curved edges. The solid line is a Hamilton cycle through the edges. The dotted lines are the half cuts.



a *Fig. 19* *b*

Figure a is an attempt to show a Hamilton cycle for edges of a hypercube. Of course this is not Fcutmax without extra cuts so that every edge suffers 1.5 cuts. Figure b shows clearly that each cut node must have 3 cuts meeting at the node to be able to separate it into two pieces.

Planar Ecutmax in 4D

No formula is proposed for Ecutmax for a 4D figure. It seems a 4D tetrahedron has Ecutmax = 7, and a 4D cube has Ecutmax = 18. For a 4D tetrahedron we can make a cut in each of the edges of the outer tetrahedron for six cuts. Then if any of the remaining 4 inner edges are cut two pieces result. For a 4D cube we can cut all 12 edges of the outer cube then do a cutmax of 6 on the inner cube to create two pieces in 18 cuts.

Planar Vcutmax in 4D

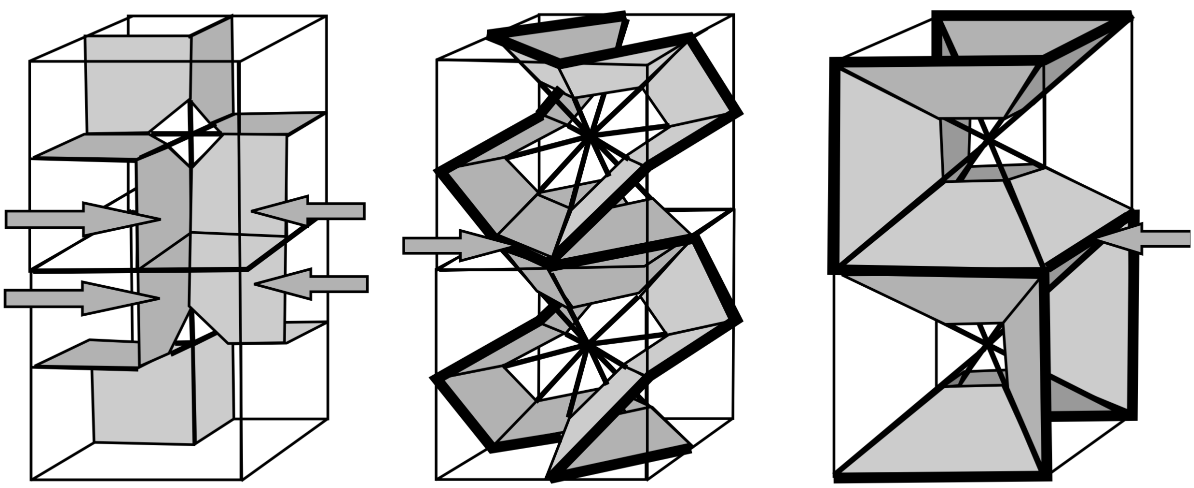
Since each edge has 3 faces emanating from it the vcut must decide which two faces to cut between on each cut so one could adopt the rule that all three faces are disconnected by the vcut.

Solids cut of Ecutmax in 4D

If the hyper figure has solid interiors included we need to make cuts to the center of the solids as discussed above for a 3D cube. For edge cuts call it Ecutmax. Each edge cut must be made to the center of all the solids common to the edge. As an illustration Figure 20 shows two cubes glued on one face. Where they are glued the edge cut must proceed to the center of both cubes thus being two edge cuts in each case as is shown by the pairs of arrows for each of these two cuts. To simplify our thinking these common cuts will always be considered to be one edge cut. No formula is known for solids Ecutmax for a hyper figure but it should be the same as for a planar 4D Ecutmax.

Solids cut of Fcutmax in 4D

For cutting faces the solid cuts will proceed to center as described in Figure 17b. In this case each cut must be executed to the center of both the solids common to a face. In addition another path line segment connecting two edge crossings must be added where two paths pass by each other on the same face. Where this extra path is made can often be chosen in two different ways. Figure 20b shows the extra double cut made by the extra path segment. We will consider cuts needing to go to two(or more) centers to be a single cut if emanating from a single edge. The formula should be the same as the planar 4D Fcutmax formula or the solids cut of 4D Fcutmax $<= E + E/2$.



a b c

Fig. 20

Figure 20 shows the basic concept of making solid cuts to centers of both cubes if they have a face, edge or vertex in common. The examples here show two cubes with one face in common.

Solids cut of Vcutmax in 4D

It is possible to cut a 4D figure using Vcutmax. Figure 20c shows this done for a pair of cubes with one common face. Where the cut line is on a common edge the cuts proceed to the center of both cubes and are considered to be one cut. For the two cube case an extra path line segment joins two vertexes as shown by the arrow. In a 4D cube three solids are common to each edge but generally only one or two cuts to centers may be needed per edge. No formula is known but solids 4D Vcutmax should be the same as the formula for planar 4D Vcutmax. As before we require all cuts to be necessary for Vcutmax to create two pieces.

Pcutmax in 3D and 4D

Another kind of cut can be made. Since the faces of 3D solids in a hyper figure or compound 3D figure are always common to two solids it is possible to simply cut through the plane of selected faces thereby severing the solid connection between cut faces. Call this Pcutmax for a polygon cut. Would the cut proceed along a polygon type Hamilton cycle? If we think of a 2x2x2 cube with Pcutmax cuts it would resemble the edge cut cube in Figure 17a where the edge cuts are now cutting between two common faces with each cut. No formula or guess is known for Pcutmax and it is unknown if this type of cutting always has a logical cutmax solution. We know that each cube of a 4D cube has six cubes common to it. A branch like piece must generally leave two faces connected to each so a cutting away of 4 cubes might cut through 3 x 4 = 12 faces or ½ the faces. One could work with a dual network got by connecting all adjacent centers of the solids. For the 4D cube in Figure 19 this looks like an octahedron with two sets of 3 central axes.

Conjectures and Observations

Ecutmax as discussed above generally required all cuts to be necessary. If we drop this requirement and allow cuts to be made at random then REcutmax, Random Ecutmax equals NEcutmax, Necessary Ecutmax. For instance we could cut a tetrahedron into two pieces by cutting the top off in three cuts. This leaves the bottom portion which is able to withstand one more cut and still remain in one piece. Thus Random Ecutmax for a tetrahedron is 4, the same as for Necessary Ecutmax for a tetrahedron. Proof: If a single cut is removed the two pieces are reunited. The removed cut is now moved by cutting another edge somewhere else. This can generally be done since there are generally more edges than faces. The new cut must then cut the figure into two pieces again since it must obey the rule before being cut that extra cuts mean extra pieces. This 'proof' may not be ironclad but it shows the general nature of the concept of cutmax. Ecutmax on a very complicated network of edges will not always look like a Hamilton cycle through the faces. It could look very branch like and the branches can be entertwined in various ways. Anyway we conclude that REcutmax = NEcutmax and the cutmax rule EC >> EP always holds for both cases up to the maximum number of pieces possible.

For Fcutmax we cannot compare necessary with unnecessary cuts since all cuts of a planar figure must always be a Hamilton cycle touching all the edges. Of course this assumes that such a Hamilton cycle for the edges is always possible. When we upgrade to 4D then a Hamilton cycle is less obvious since ½ cuts must exist and some rule of moving cuts and having unnecessary cuts equal to necessary cuts might be possible. Similar observations apply to Vcutmax.

General Observations

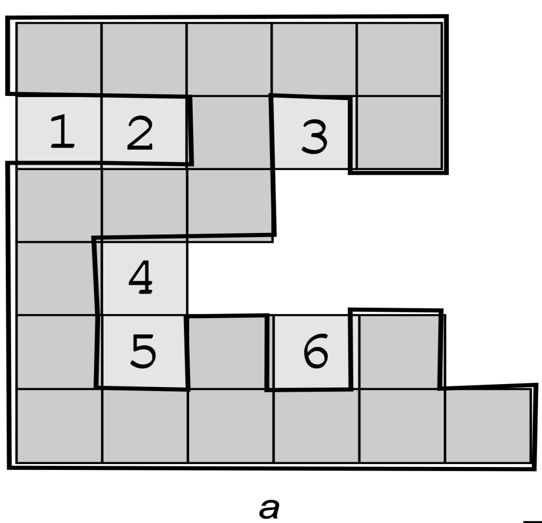
It can be seen from the above that when we get into higher dimensions the cutmax idea deviates significantly from the idea of a Hamilton path. It shows that the idea has a separate and distinct life all on its own and is an interesting subject by itself. A great deal more could be done with this basic idea as it pertains to art, sculpture, mathematics and above all puzzles.

Perhaps in four dimensions it would be possible to disassemble an Fcutmax hypercube in two pieces. But we live in 3D and since all the faces, vertexes and edges can actually exist in a 3D model you can build, you could make the two pieces separately. Then you could see if they can be put back together. If not then you could make additional cuts to create more pieces, one cut at a time, until assembly is possible. The project could be done more easily using a 3D printer. If this project is too daunting perhaps one could drop to a simpler 4D polyhedron such as a tetrahedron. One imagines that it could be very attractive and high tech looking if some or all of the pieces are transparent so you can see inside the '4D' assembly. Wonderful puzzles and objects will also be possible using the solid Fcutmax, Ecutmax, Vcutmax, Pcutmax systems.

The possibilities for 4D cuttings grow if we extend them to HSM Coxeter's and other mathematicians many wonderful Polytopes. This might make a neat project for artistic renditions using a graphic program or 3D printer.

Constant number of faces inside a Hamilton cycle.

After writing the above I noticed that the number of square faces in the interior of a Hamilton cycle on a square grid is always constant no matter how the path is drawn. This also implies that the number of squares on the exterior is always constant. This is not an obvious thing. Of course the path length is constant since it always equals the number of nodes, But one would think the number of faces could vary somewhat. If you remove some of the edges inside the grid the number of squares inside and outside the path can vary by more or less depending on size of grid and where and how many edges are removed. To maintain the constant property the grid can have any shape and size except it cannot have holes or surround an area as this would be equivalent to removing some edges. It could be L shaped, Z shaped or snake around in a large spiral with hundreds of branches. The constant area theorem still holds. Call it Fsum(a,b) where the a and b are the number of squares inside and outside the path.



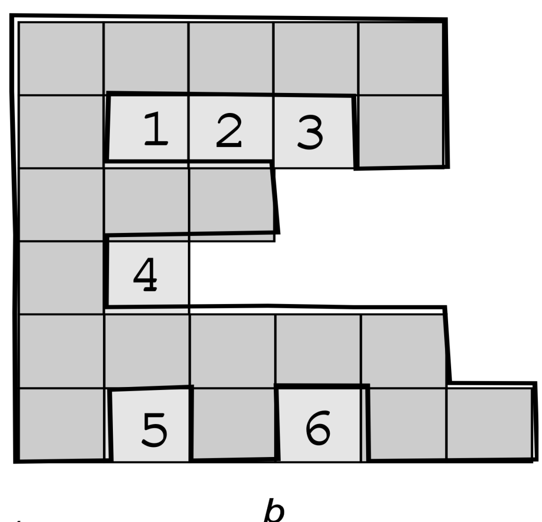


Fig. 21

Figure 20 shows how the number of faces on the inside and outside areas of a square grid are constant no matter how the Hamilton cycle is drawn.

Figure 21 shows an example of how a and b are constant for a small grid. Figure 22 shows how a and b can vary if the grid has some missing edges.

It would be easy to devise paper and pencil puzzles using this principle. For instance show a grid with some edges missing then challenge the solver to find a cycle with n squares showing up on the exterior.

Sums exist for edges, vertexes and faces in the two parts cut out by a Hamilton cycle or a near Hamilton cycle. The sums could have names as follows. EcutFsum(a,b), EcutEsum(a,b), EcutVsum(a,b), FcutFsum(a,b), FcutEsum(a,b), FcutVsum(a,b), VcutFsum(a,b), VcutEsum(a,b), VcutVsum(a,b), and cutmax is assumed and facelets for the E and Fcuts. The sums are either constant or vary by some upper and lower bound depending on the network and the particular chosen path. Obviously Euler's formula relates any group of three.

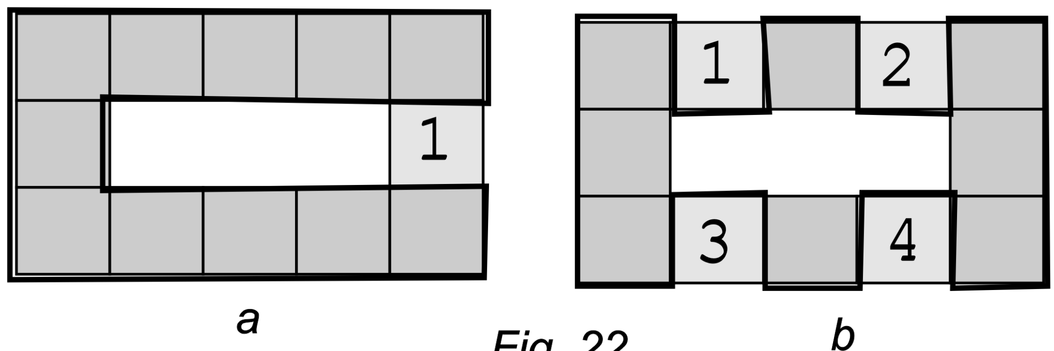


Fig. 22

Figure 21 shows how the number of faces on the inside and outside areas of a square grid are not constant if the Hamilton cycle is drawn differently when some of the edges are missing.

Comments and Proofs by Jaap Scherphuis

Proof of Ecutmax <= F: Proof by Jaap Scherphuis

Lemma: Let G denote a planar graph, F(G) its number of faces, and M(G) the maximum number of edges that can be cut without G becoming disconnected. Then M(G) = F(G)-1.

Any planar graph with 1 face is a tree graph, and removing any edge from it will disconnect it. Therefore M(G)=F(G)-1 for the base case F(G)=1.

Suppose M(G) = F(G)-1 this is true for any graph with f-1 faces.

If you have any planar graph with f faces, and cut an edge without disconnecting the graph, then that edge must lie in some loop, and so have different faces on either side of that edge. Cutting that edge is essentially the same as removing that edge, merging the two faces on either side, leaving you with a graph with one fewer edges and one fewer faces. Regardless of which edge you cut that doesn't disconnect the graph, this smaller graph has M(G)=f-2, so the original graph must have M(G)=f-1=F(G)-1.

By induction this is true for planar graphs with any number of faces.

Clearly Ecutmax <= M+1 by definition, so Ecutmax <= F

Ecutmax is defined in a way where every cut is necessary. There are graphs where any choice of F edges will have unnecessary cuts, and Figure 7 is indeed an example of that. You can cut 8 edges from that 9-faced graph such that any more cuts disconnect it, but in all cases some of the other cuts become unnecessary.

This necessity requirement is equivalent to requiring the cuts to lie on a single loop. If the cuts disconnect the graph then there must be at least some loop around one of the two parts. Any cuts not on that loop are extraneous. I think the smallest counterexample like Figure 7 is the graph consisting of a triangular face with three triangles adjacent to it.

You could also work with the dual graph. Put a node in the center of each face in your original graph, and connect two center nodes whenever their two faces are adjacent. A set of cuts in the original graph that attains Ecutmax then corresponds to a Hamilton cycle around the dual graph (a looped path that visits every node exactly once). Most planar graphs where all nodes having three or more edges have a Hamiltonian cycle. You will find it hard to find a counterexample that does not have this property. It is only relatively recently that such counterexamples were discovered:
http://en.wikipedia.org/wiki/Tait%27s_conjecture

When cutting faces instead of just edges, you are allowing for consecutive cuts to lie in the same face. On the cube you can find an example with 12 face cuts that does not do this. See Figure 10b(note 2). On a dodecahedron you can also visit each face only twice non-consecutively, but then you use up only four edges of each pentagon, cutting only 24 of the Fcutmax=30 edges. Take a look at the Moebius puzzle on my page for an example of that.
http://www.jaapsch.net/puzzles/moebius.htm

Of course any such path can trivially be extended to the full 30 edges when you allow consecutive cuts in the same face(note 3). I can't prove that Fcutmax = E can always be attained, but I'll have to think about that a bit more. It seems unlikely to be impossible, but you never know.

Proof of constant number of faces cut off by a Hamilton path through a square grid.
Proof by Jaap Scherphuis sent: Mon, Mar 1, 2010

The faces inside the hamilton cycle form a tree - if it were not a tree then there would be some loop of three or more faces, which would have internal vertices unvisited by the Hamilton cycle.

Suppose you have a tree of k squares. There will be k-1 internal edges used up in joining them together, each shared by two of the squares. The k squares have 4k edges all together, so there are 4k-2(k-1) = 2k+2 edges around the perimeter of the tree, which form the Hamilton cycle.

On an m by n grid of squares there are mn squares and (m+1)(n+1) vertexes. A hamilton cycle through all those vertexes therefore have (m+1)(n+1) edges.
So 2k+2 = (m+1)(n+1).

As an aside, this shows that no Hamilton circuit is possible if m and n are both even as k must be an integer.

We want the number of squares outside the Hamilton circuit, which is mn-k. A little algebra then shows this is (m-1)(n-1)/2.

The same reasoning works for any kind of planar graph where all the faces (excluding the exterior) have the same number of edges.
Jaap Scherphuis

Notes:

1. ***maximum, maximal;*** Jaap corrected me technically here by suggesting these words be added.
2. (Author's comment: I purposely wanted to be able to to freely make the cuts to make more puzzle designs possible but Jaap's comment here is well worth noting)
3. (Author's comment: By making the extra face cuts on a dodecahedron, etc., to realize Fcutmax = E the cutmax principle that any more cuts will result in more pieces is maintained.)
4. Please note the author(Doug Engel) takes full responsibility for any errors, omissions, etc.