FEATURE COLUMN *Monthly Essays on Mathematical Topics*

Urban Geometry

The tradition of mathematical playfulness set in motion by Euler is still alive well today...

Joseph Mall
York College
malkevitch at york.cuny.

The geometry of urban services

Although we have moved from 2007 to 2008, the celebration of the 300th anniversary of the birth of Leonard Euler (born A 1707) is still in full swing.

Euler, in 1736, created the field of graph theory with his famous paper in which he solved the Königsberg Bridge Problem. In a r this problem posed the question of whether or not it was possible to design a tour of a certain collection of Königsberg's bridges each bridge was traversed once and only once. The first diagram below shows an historic map of Königsberg and the second some of the bridges highlighted.

Figure 1

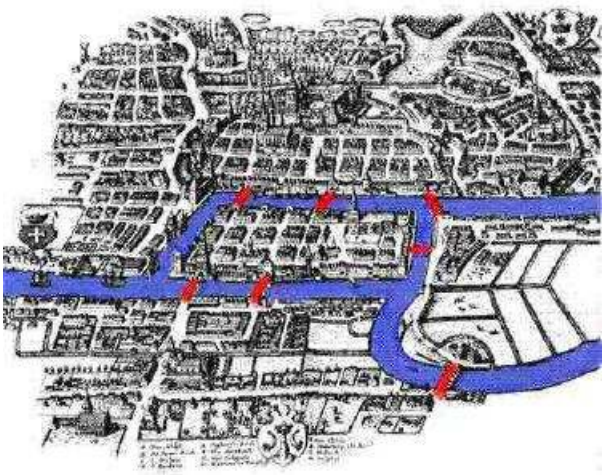


Figure 2

While at first glance what Euler did seems to be an elegant solution to a problem in recreational mathematics, it eventually ope door to a wide variety of improved ways to provide a host of urban services such as: street sweeping, garbage collection, ar removal. Euler's work is a gift of geometric insight into mathematics which eventually, in the hands of other clever people, evolve tool to make all of our lives better.

A graph theory primer

To solve the bridge problem we can, following in Euler's footsteps, use a geometrical tool called a *graph*. A graph is a diagram co of a finite number of dots called *vertices*, which are joined by a finite number of line segments (which can be straight or curved *edges*. (Here we will not allow an edge to join a vertex to itself, just as a matter of convenience, but it is not difficult to mo discussion to take into account the allowing of "loops.") Typically the dots represent a collection of objects, perhaps ATM machir two dots are joined by an edge if objects they represent obey some condition or property. For example, in the case of ATM mach might join two vertices representing ATM machines by an edge if it were possible to drive under 1/2 mile to get between them. T in the spirit of what Euler did, we can use a dot to represent a "land mass" and join two dots by an edge if the land masses they re have bridges between them. In the famous problem Euler attacked there were cases where two bridges connected the same land (banks of the river or an island), so we use two edges to connect the dots in such a case.

The diagram below shows a graph theory model (representation) that can be used to analyze the problem Euler faced.

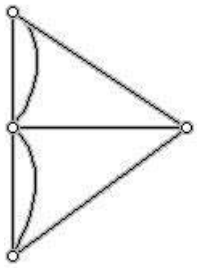


Figure 3

It is often convenient to gives names to the vertices (and/or edges) to help with describing what is going on. The diagram below s labeling of the graph above.

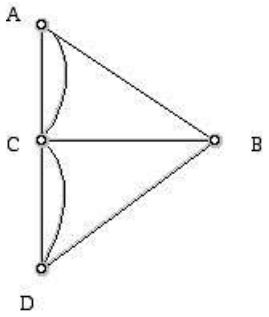


Figure 4

The question that was posed to Euler, in more modern terms, was: Is it possible to find a route which crosses each of the bri Königsberg once and only once? We will also require that the tour begin and end in the same place. A version of this problem formulated in a much more tantalizing way as a problem about urban services. Suppose we interpret the diagram in Figure 4 as network in an urban neighborhood, with a snow plow located at B. Is it possible for the plow to traverse each street once and on returning to B?

If such a tour for the snow plow exists, then we will call it an *Eulerian circuit*, in honor of Euler. It is relatively easy by trial ar (brute force) to verify that this is not possible for the graph in Figure 4. However, Euler being the extraordinary mathematician was, did not only care about the particular problem for this graph. He also wanted to know: Given a graph, when is it possible t route (e.g. an Eulerian circuit) which traverses each edge of the graph once and only once and starts and ends at the same vertex?

To answer this question it is first helpful to have some additional terminology and insight into properties that different kinds of gra have or not have. First, the graph in Figure 4 has 4 vertices and 7 edges. However, unlike the graph in Figure 4 which has o "piece," the graph in Figure 5 has two pieces. If a graph M has more than one piece, that means that there are two vertices in M s it is not possible to move along edges to get between the two vertices. We will say that a graph is *connected* if, given any two ve the graph, it is possible to move along edges of the graph to get between these vertices.

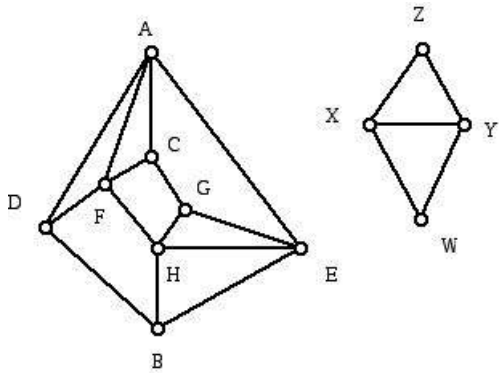


Figure 5

Another useful piece of information about a graph is to know for a particular vertex Q of the graph what is the number of edges th at vertex Q. This number is called the *valence* or *degree* of the vertex. The degree of vertex X is 3 and the degree of D is 3, w degree of vertex E is 4.

As an example of a "puzzle" that these new ideas allow one to quickly formulate, the sequence of numbers below is a list of al valences of the vertices in Figure 5.

4, 4, 4, 4, 3, 3, 3, 3, 3, 3, 2, 2

Can you find a connected graph with these valences?

Here are three more puzzles in the same spirit.

Can you find a connected graph which has the valences shown for each sequence?

6, 4, 4, 4, 3, 3, 3, 2, 2, 2, 2

6, 4, 4, 4, 3, 3, 2, 2, 2, 2

7, 4, 4, 4, 4, 1

(Can you find connected graphs which work for the sequences above if given any two vertices, there is at most one edge joining th

vertices?)

One of the exciting aspects of mathematics, especially important mathematics, is how quickly one can get to "puzzles" based on ε understand ideas and concepts. The crux of mathematics is not its signature strange symbols but its ability to engage the mi problems involving the search for patterns. Sometimes it may not be apparent how to apply solutions of such puzzles to the world of mathematical investigation. However, over and over again the "pure mathematics" of the past has become the "applicable mathe of today.

One fact about graphs is related to the puzzles above and was among the first facts about graphs that were discovered, in this Euler himself. Since every edge has exactly two end points (remember we are not allowing a vertex to be joined to itself by an e we add up the degrees (valences) of all the vertices in a graph we must get an even number, and this number is twice the nu edges of the graph. When we sum the valences of all of the vertices of a graph we count each edge twice, once at each of its endpc

We can state this as a theorem, and deduce a surprisingly useful corollary of this theorem:

Theorem: The sum of the valences of the vertices of a graph is even and equals twice the number of edges.

Corollary: The number of vertices of odd valence in a graph must be even.

Proof: If there were an odd number of vertices of odd valence, then summing the valences of the vertices of the graph would give number which contradicts the theorem.

You can verify that the graph in Figure 3 has four odd-valent vertices while the graph in Figure 5 has six odd-valent vertices.

Euler's spectacular insight

Euler's insight into the Königsberg Bridge Problem was to notice that not only is it not possible to find what has come to be know Eulerian circuit in the graph in Figure 3, but he found an extremely simple condition that can be applied to any connected graph. is *connected* (as mentioned above) if it is possible to get between any two vertices of the graph by moving along the edges of the g

Suppose a graph has an odd-valent vertex, that is, a vertex at which an odd number of edges meet. If this odd-valent vertex ι initial vertex of an Eulerian circuit T which visits each edge once and only once, we are in trouble. Every time we leave and reen the tour T we use up a pair of edges. Thus, if the valence of *w* is odd, we will leave *w* one more time than we are able to reenter i an odd-valent vertex can not be the initial vertex of an Eulerian circuit. A very similar argument shows that if a vertex *w* is not th vertex of a tour visiting each edge of a graph once and only once, and returning to its starting point, then *w* can not be odd. Ag reason is that one must have entered such a vertex one more time than one left the vertex. Hence, such a vertex would hav odd-valent.

We have shown that in order for an Eulerian circuit to exist, a connected graph must have every vertex be of even valence.

If a graph is connected and even-valent, does it have an Eulerian circuit?

There are numerous ways one can show that a graph will have an Eulerian circuit when the graph is connected and even-vale process can be illustrated using the even-valent (and connected) graph in Figure 6.

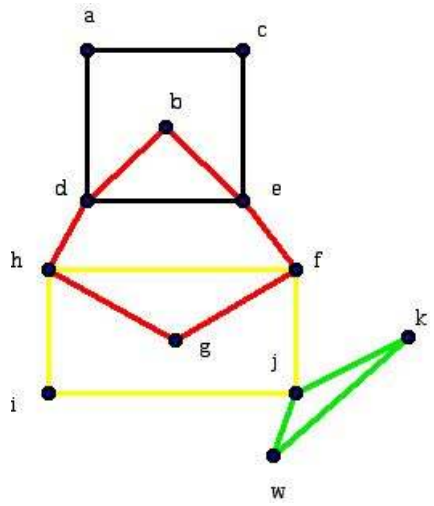


Figure 6

If a graph G has every vertex being even-valent, then G can be written as the disjoint union of (simple) circuits. In the graph ab example, the color- coded edges that form (some of the) circuits of the graph. For example, *d, e, f, g, h, d* is a simple circuit

while *w, j, k, w* is a simple circuit (in green). We can form an Eulerian circuit by piecing together these simple circuits: *w, j, f, e, e, b, d, h, ,g, f, h, i, j, k, w*. The circuits of different colors have been pieced together to form longer tours of edges at vertices circuits have in common. (It is worth noting that usually a graph can be written as the union of circuits whose edge sets are dis more than one way. For example, in the graph above we could have used *dbed* and *adhgfeca* instead of *adeca* and *bdhgfeb*.) Ty there are many Eulerian circuits in a connected even-valent graph, which means in applied problems one can choose an Eulerian that allows one to achieve some additional special goal.

An alternative approach to finding an Eulerian circuit is insightful in a different way. Suppose one uses the *wander at random ru* traversing a graph which is connected and even-valent:

If one is at a vertex u*, one can move on to another vertex along any edge that is present at* u *that has not already been traversed e the tour.*

If one applies this rule starting at *w*, and one can no longer apply the rule, where must one be? It is easy to see that one must b Suppose you are not at *w* but are at the vertex *v*. It must be the case that you have entered *v* one more time than you were able *v*. Thus, *v* has odd valence, contrary to the assumption that the graph G has every vertex having even valence. Hence, apply wander at random rule returns one to *w*. If the edges traversed include all the edges of G, we have an Eulerian circuit already. If n there are even-valent sub-graphs of G that have not yet been traversed. One of these pieces, because the original graph is cor must have a vertex *t* in common with the tour which is being built up to an Euler circuit that has been constructed up to this point. apply the wander at random rule starting at *t* to get edges of G that are used once and only once that return to *t*. We can piece this tour with the previous tour to get a longer partial tour, or an Eulerian circuit. Continuing in this way an Eulerian circuit is finally p because a graph must have a finite number of vertices and edges.

The final result, inspired by Euler's amazing start is:

A connected graph G has an Eulerian circuit if and only if G is even-valent.

As a theorem in mathematics, of interest for its own sake, this result has been extended in many directions. One such relatively direction is to consider tours that traverse each edge of a connected graph once and only once but begin and end at different vertic

Here is the result for this situation:

A connected graph G has a tour that starts at vertex v *and ends at vertex* w *where* v *and* w *are different vertices of G if and on valence of* v *and* w *are odd and the valence of all the other vertices of G is even.*

This result can be proven by using the previous result for Eulerian circuits. Thus, for example, if G has a tour from *u* to *w* (vertices) such that each edge is traversed once and only once, then if we add an additional edge *e* to G which joins *u* to *w*, we c find an Eulerian circuit which starts and ends at *u* using the additional new edge. Thus, this graph must be connected and even which means that the original graph must have had exactly two odd-valent vertices (and been connected) before the edge *e* was a G!

Our motivation in looking at Euler circuits was to be able to provide urban services such as street sweeping, snow removal, mail and garbage collection in an efficient manner. However, we now see that many situations, when the graph to be traversed has od vertices, can not be handled as ideally as we would like. However, people will still want their snow removed, so we have to find way of thinking of the applied situation.

When one uses mathematics to solve an applied problem, one has to check that the proposed insight that mathematics provides is enough for general circumstances. Here, when a graph G has an Eulerian circuit, that is, G is connected and even-valent, then one a wonderfully efficient solution. One never has to do any unnecessary work. However, when the graph is not even-valent, what d do? We have to modify our goal to be more realistic. Obviously, if one is collecting garbage, removing snow, or sweeping the stre can not achieve "efficiency" at the expense of not providing the underlying urban service for some customers. Thus, one will hav some "needless" work in order to guarantee that all customers get served. If the graph model for providing the urban service even-valent, we will have to repeat some of the edges of the graph even when no work is being done when we retraverse an edge represents some section of street or highway. It should be clear that when edges have different lengths (traversal times), there situations where retraversing more edges with lower cost will be better than retraversing fewer edges with high costs of retraver the moment, let us assume that the time (or cost) to traverse each edge of a graph is the same. (There are some situations wher actually realistic.) In this case, what we want to find is a route that starts and ends at the same place and traverses each edge of th at least once and which repeats a minimum number of edges.

For example, in the graph below (Figure 7) we see that the graph has no Eulerian circuit because it is not even-valent.

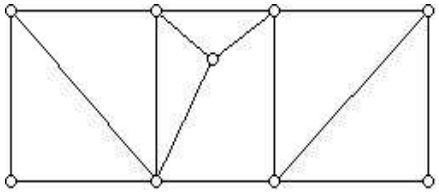


Figure 7

Suppose we duplicate existing edges in this graph in such a manner as to make the resulting graph even-valent. The graph v duplicated edges will have an Eulerian circuit and we can interpret moving along a duplicated edge as retraversing an edge in the graph. These retraversals are sometimes referred to as deadheads since no work is being done during the second traversal. Mig necessary to retraverse some edge more than twice in achieving the minimum repetition route? A moment's thought will show that 3 repeats were necessary to make the ends of edge *e* even-valent when they started as odd-valent, we could leave off two repeats, down to 1 repeat, and still have the ends of *e* even-valent. In the graph below the blue edges show those that correspond to du edges.

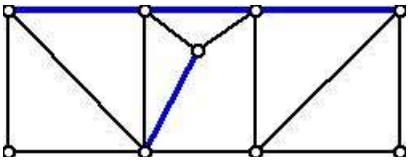


Figure 8

Finding an Eulerian circuit in the graph where the blue edges are duplicated results in a route which retraverses 4 edges. However, a route that retraverses only three edges.

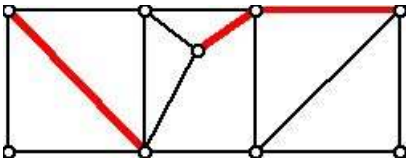


Figure 9

Is it possible to get an efficient algorithm for finding which edges to duplicate in order to get a route which repeats as few e possible? We can start to see what is involved by considering the graph below:

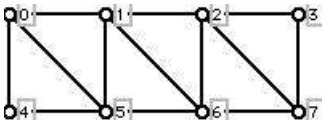


Figure 10

Since this graph has exactly two odd-valent vertices, those with the labels 0 and 7, we need to duplicate the edges on some pa vertex 0 to vertex 7. However, there are many such paths. Which one should be chosen? If all of the edges have the same weight, find a minimum cost tour we need to find a shortest path from vertex 0 to vertex 7. In this graph, the shortest path from 0 t length 3 and there are several such paths. If there were weights on the edges, we would need an algorithm for finding the shortes path between 0 and 7.

Chinese postman problem

Our improved model, which goes beyond the analysis of Euler and yet involves ideas concerning Eulerian circuits, is often referr the *Chinese postman problem*. The reason for this is that at the surprisingly recent date of 1962, the Chinese mathematician Meiç (also often known as Mei-Ko Kwan) raised issues related to efficient urban service in a paper published in Chinese.

The general solution of this problem for undirected graphs where weights were assigned to the edges of the graphs required sophi work. Here, I will try only to suggest the flavor of what is involved and give the idea of what is going on for a simple example.

Given an (undirected) connected graph G with an even number of vertices, a *perfect matching* for the graph is a collection of edges which includes all of the vertices of the graph. For example, the graph below has several perfect matchings, though all must contain the edge joining vertices 6 and 7.

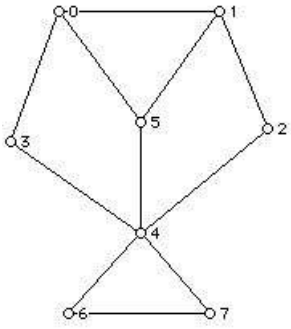


Figure 11

Not all graphs which have an even number of vertices have perfect matchings, even when there are other conditions such as havi vertex have valence 3; however, we are not particularly interested in the existence questions for matchings. In the particular gra concern us such perfect matchings will always exist. More specifically, we are concerned with having a complete graph with number of vertices, a graph where every pair of vertices are joined by exactly one edge, and with weights (usually positive) on the We want to find a perfect matching such that the sum of the weights of the edges in the matching is a minimum over all matchings.

To give you the flavor of what is involved, suppose we have a graph with exactly four odd-valent vertices. Unlike the analysis v able to make when we had exactly 2 odd-valent vertices, the complexity of matters grows rapidly here. If we have general weights edges, and even if we do not, we need to know how to duplicate existing edges to minimize the cost of the edges that we duplicate this we need to know the length in terms of weight (not number of edges, in the case where the edges can have different v between pairs of odd-valent vertices. Mathematicians and computer scientists have developed efficient algorithms for various l shortest path problems. For example, one might want to find the shortest path between one vertex and many other vertices in a w graph. A generalization is to find an efficient algorithm for the shortest paths between every pair of vertices in a weighted graph. (the vertices of a graph are odd-valent, then to solve the Chinese postman problem this would be required.) One breakthrough res was the work of Edsger Dijkstra , a physicist who became a pioneer of computer science. One of Dijkstra's important contributi to developing shortest path algorithms for graphs. (A recent application of shortest path algorithms is in the software that is many automobiles to provide directions for drivers to get from one location to another as quickly as possible.)

We can record the information about the shortest paths between the four vertices involved (the four odd-valent vertices) either i matrix (table) or along the edges of a complete graph on 4 vertices, as shown below.

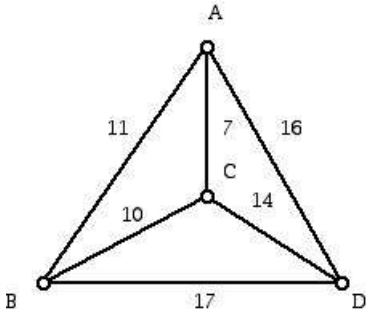
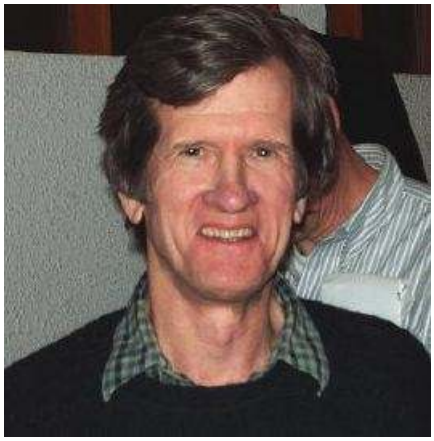


Figure 12

By trial and error it is easy to see that there are three different perfect matchings in this graph: 7 + 17; 10 + 16; 11 + 14. The m sum corresponds to repeating the edges on a shortest path from A to C and from B to D.

Putting the germs of the ideas laid down here to find practical, rigorous and computationally efficient algorithms for the Chinese p problem is no easy task. The task was achieved by work due to Jack Edmonds and to the joint work (1973) of Jack Edmond Ellis Johnson .

(Photo of Jack Edmonds, courtesy of Jack Edmonds and Michel Las Vergnas)



(Photo of Ellis Johnson, courtesy of Ellis Johnson and Jack Edmonds)

Although the work of Edmonds and Johnson gives rise to a polynomial time algorithm for solving the general Chinese postman p many new developments were set in motion by their work. In particular, practical versions of Chinese postman problems are mod only by undirected graphs with weights but also call for "directed" graph models where there are directed edges (each edge has a on it, corresponding to, say, a certain road that only allows one-way traffic). Sometimes it is natural to use a network (graph) whe of the edges are directed and some are undirected. A somewhat unexpected development was that the undirected and directed cases of the Chinese postman problem have polynomial time algorithms, while the mixed Chinese postman problem (direc undirected edges are allowed) has been shown to be NP-complete. (This means that there is no known polynomial time algorithm problem, nor a proof that the only way to solve the problem will require an exponential time algorithm.) The work of Edmo Johnson still inspires workers to find variant models for the Chinese postman problem with additional conditions, specialized alg which can be shown to be better for special classes of weighted graphs, and new applications of the older ideas, all having been by a problem in recreational mathematics of Leonhard Euler!

The tradition of mathematical playfulness set in motion by Euler is still alive and well today. Mathematical puzzles beget new mathe applications of mathematics, and in turn new mathematical puzzles to stimulate and amaze us, in a never ending progression.

References:

Beltrami, E. and L. Bodin, Network and vehicle routing for municipal waste collection, Networks 4 (1974) 65-94.

Brucker, P., The Chinese postman problem for mixed networks, in Proceedings of the International Workshop on Graph theoretic C in Computer Science, Lecture Notes in Computer Science 100, Springer-Verlag, New York, 1980, pp. 354-366.

Chachra, V. and P. Ghare, J. Moore, Applications of Graph Theory Algorithms, North-Holland, New York, 1979.

Christofides, et al., An Optimal Method for the Mixed Chinese Postman Problem, in System Modeling and Optimization, Lecture I Control and Information Science 59, Springer-Verlag, New York, 1984.

Edmonds, J., The Chinese Postman Problem, Operations Research, 13 (Supplement 1) (1965) 373.

Edmonds, J., Maximum matching and a polyhedron with 0,1 vertices, J. Research Nat. Bur. Stand., 69B (1965) 125-130.

Edmonds, Paths, Trees, and Flowers, Canadian J. Math., 17 (1965) 449-467.

Edmonds, J., and E. Johnson, S. Lockhart, Blossom I: a computer code for the matching problem. Unpublished report, IBM T.J. Research Center, Yorktown Heights, New York, 1969.

Edmonds, J. and E. Johnson, Matching, Euler Tours, and the Chinese Postman Problem, Mathematical Programming, 5 (1973) 88-1

Fleischner, H., Eulerian Graphs and Related Topics, Part 1, Volume 2, Annals Discrete Mathematics 50, North Holland, Amsterdam,

Fredrickson, G., Approximation Algorithms for some postman problems, J. ACM, 26 (1979) 538-554.

Fredrickson, G. and M. Heckt, C. Kim, Approximation algorithms for some routing problems, SIAM J. Computing, 7 (1978) 178-193

Guan, M., Graph programming using even and odd points, Chinese Mathematics 1 (1962) 273-277. (Note: There are many version spelling of Guan's name using latin characters. This is the most common recent version but one also sees Mei-ko Kwan for Meigu older references.)

Guan, M., On the windy postman problem, Discrete Appl. Math., 9 (1984) 41-46.

Kappauf, C. and G. Koehler, The mixed postman problem, Discrete Appl. Math., 1 (1979) 89-103.

Lin, Y. and Y. Zhao, A new algorithm for the directed chinese postman problem, Computers and Operations Research 15 (1988) 57

Minieka, E., Optimization Algorithms for Networks and Graphs, Marcel Dekker, New York, 1978.

Minieka, E., The chinese postman problem for mixed networks, Management Science 25 (1979) 643-648.

Norbert, Y. and J. Picard, An optimal algorithm for the mixed chinese postman problem, Networks 27 (1996) 95-108.

Orloff, C., A fundamental problem in vehicle routing, Networks 4 (1974) 35-64.

Papadimitriou, C., On the complexity of edge traversing, J. ACM, 23 (1976) 544-554.

Pearn, W., Solvable cases of the k-person chinese postman problem on mixed networks, Operations Research Letters, 16 (1994) 24

Raghavachari B. and J. Veerasamyh, A 3/2-approximation algorithm for the mixed postman problem, SIAM J. Discrete Math., 12 425-433.

Ralphs, T., On the mixed chinese postman problem, Operations Research Letters, 14 (1993) 123-127.

Toth, P. and D. Vigo (Eds.), The Vehicle Routing Problem, SIAM, Philadelphia, 2002.

Zin, A., on the windy postman problem on Eulerian graphs, Mathematical Programming, 44 (1989) 97-112.

Joseph Mall
York College
malkevitch at york.cuny.

Those who can access JSTOR can find some of the papers mentioned above there. For those with access, the American Mathe Society's MathSciNet can be used to get additional bibliographic information and reviews of some these materials. Some of th above can be accessed via the ACM Portal , which also provides bibliographic services.