XXX
Simpozijum o operacionim istraživanjima
SYM-OP-IS 2003
Hotel PLAŽA, Herceg-Novi, 30.09.-03.10.2003.

THE MINIMUM FUNDAMENTAL CYCLE BASIS PROBLEM: A NEW HEURISTIC BASED ON EDGE SWAPS

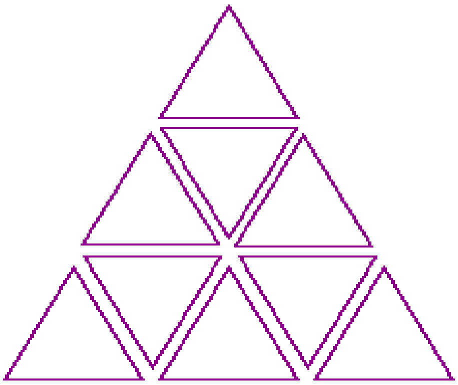
Edoardo Amaldi, Leo Liberti, Nelson Maculan, Francesco Maffioli
DEI, POLITECNICO DI MILANO, P.ZZA L. DA VINCI 32, I-20133 MILANO, EUROPE

Abstract: *This paper describes a new heuristic method for finding minimum fundamental cycle bases in biconnected, undirected graphs. At each iteration, a particular edge swap is applied to the spanning tree corresponding to the current fundamental cycle basis is computed. Our numerical experiments, obtained for square mesh graphs, show that this heuristic finds fundamental cycle bases with considerably lower costs compared to other available heuristics.*

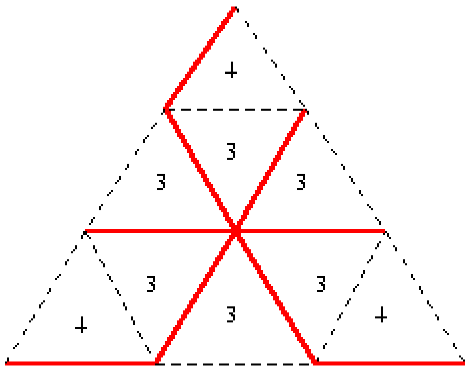
KEYWORDS: fundamental cycle basis, spanning tree, edge swap, local search.

1. INTRODUCTION

Let $G = (V,E)$ be a simple, undirected, biconnected graph with n nodes and m edges. A set of cycles in the graph is a *cycle basis* if it is a basis in the cycle vector space. The cost of a cycle is defined as the number of



Cycle basis which is not fundamental



Fundamental cycle basis with cost = 30

edges contained in the cycle. The cost of a set of cycles is the sum of the costs of each cycle in the set. Given any spanning tree T of G, the edges in $G \setminus T$ (the co-tree) are called *chords* of G w.r.t. T. Any chord uniquely identifies a cycle consisting of the chord itself and the unique path in T adjoining the two vertices incident on the chord. These $m - n + 1$ cycles are called *fundamental cycles*; they form a cycle basis which is called *fundamental cycle basis* (FCB) of G with respect to T. It turns out [6] that a cycle basis is fundamental if and only if each cycle in the basis contains exactly one edge which is not contained in any other cycle in the basis. A *minimum fundamental cycle basis* is quite naturally an FCB having minimum cost. Finding a minimum FCB is referred to as the **MinFCB** problem.

Minimum FCBs arise in a variety of application fields, such as VLSI design [1], periodic timetable planning [5], and generating minimal perfect hash functions [3], as well as being an interesting combinatorial optimization problem in itself.

Let T be a spanning tree of G. Any edge e of T naturally partitions T in two connected components T_1, T_2 (this becomes apparent when removing e from T). The cut t_e consisting of e and all the edges of G having one adjacent vertex in T_1 and the other in T_2 is called the *fundamental cut* of e with respect to T in G. Let e,f be edges of G such that e is in T and f is in $E \setminus T$, and let $\pi = (e,f)$ be an edge permutation (also called *edge swap*). We can define the action of π on the set $\mathcal{T}$ of all spanning trees of G by setting $\pi T = T'$ where T' is the spanning tree derived from T where e has been replaced by f. Note that π is well-defined as an action on $\mathcal{T}$ if and only if f is in the fundamental cut t_e.

Since any spanning tree of G gives rise to an FCB, we can define a mapping m between $\mathcal{T}$ and the set $\mathcal{F}$ of all FCBs of G. It turns out [6] that this mapping is surjective but not injective: if $t_e = \{e,f\}$ and $\pi = (e,f)$ then $m(T) = m(\pi T)$. In other words, edge swaps in fundamental cuts of cardinality 2 induce different spanning trees but the same FCB.

The **MinFCB** problem is NP-hard [2]. Moreover, it is proved in [4] that it cannot have a PTAS unless **P=NP**. In the same work, a $2^{O(\sqrt{(\log n \log\log n)})}$ approximation algorithm is presented. Thus, it makes sense to look for heuristic methods of solutions. Several heuristics for the **MinFCB** problems exist in the literature [2, 3], and all are based on a "spanning tree growth" strategy. This means that a spanning tree is constructed iteratively adding vertices and edges of G in the order that is likely to give rise to the FCB having least cost. Such approaches are usually very fast (and thus the only possible approaches for extremely large graphs) but do not find exceptionally good solutions. In this paper we shall present a different kind of heuristic for this problem, based on edge swaps. A good initial spanning tree is found and then some edge swaps are applied to it so that the corresponding FCB cost decreases.

The rest of this paper is organized as follows: in section 2 we shall describe the edge-swapping heuristic. In section 3 we shall present a structural result which allows an optimal implementation of the heuristic. Some computational results are presented in section 4.

2.THE HEURISTIC

The heuristic move described in this section works by applying a carefully chosen edge swap to an existing spanning tree. Let T be the initial spanning tree. Let $P = \{(e,f) \mid e \text{ in } T \text{ and } f \text{ in } t_e \text{ s.t. } |t_e| > 2, f \neq e\}$. For all π in P let d_π be the cost of $m(\pi T)$. Choose π in P such that d_π is minimum and replace T with πT. This heuristic move is inserted into a simple local search algorithm that terminates when π is the identity. Obviously, the choice of the initial spanning tree T is important. To this end any of the existing fast heuristics in [2, 3] can be used.

In order to calculate the cost of an FCB given the corresponding spanning tree we need to calculate the cost of each fundamental cycle. Thus, given the set of chords C of cardinality $m - n + 1$ (where m is the number of edges in the graph and n the number of vertices), for each chord c we need to find the unique path connecting the vertices adjacent to c. Using "least common ancestor" techniques, the computational complexity of such an operation is $O(n)$. Since we have to do this for every chord, the complexity of calculating the FCB cost is $O(mn)$.

The heuristic move requires an FCB cost calculation for each π in P. There are $n - 1$ edges in a spanning tree and in the worst case there are m edges in a cut, so there are at worst mn permutations in P (but this theoretical worst case is never attained in undirected graphs without parallel edges). Hence, the complexity of computing the set of FCB costs $\{d_\pi\}$ is $O(m^2 n^2)$. Finding the minimum d_π has complexity $O(mn)$ as $|P| < mn$.

Furthermore, for each edge e in T we need to compute the fundamental cut t_e. We do this in two steps: (a) find the connected components T_1, T_2 partitioning T which are obtained by removing e from T; and (b) identify the chords of G which have one adjacent vertex in T_1 and the other in T_2. Those chords form the fundamental cut t_e. Step (a) can be carried out in linear time proportional to n by a simple recursive depth-first search of $T \setminus \{e\}$ starting from either vertex of e. Step (b) can be carried out in linear time proportional to the number of chords of G, that is, $m - n + 1$. Since we have to find the fundamental cut for each of the n – 1 branches of T, we obtain a complexity of $O(mn)$.

Thus, the heuristic move consists of three steps: calculating FCB costs, choosing the minimum FCB cost over a finite set, and finding fundamental cuts. All in all, the complexity order of the whole heuristic move is $O(m^2 n^2)$.

3. EFFICIENT IMPLEMENTATION

In settings which require repeated applications of the heuristic move described above (like, for example, in a local search framework or in a tabu search scheme), it is easy to note that at each repeated application we end up with a tree πT which differs from the initial tree T only by an edge swap. Most of the times this means that recalculating the whole set of fundamental cuts and FCB costs in πT is overkill, since on average most of these entities will remain the same. This situation calls for a differential calculation of the set of fundamental cuts and FCB costs to be carried out with respect to the chosen permutation π.

In the rest of this section we shall state some structural results in graph theory (without proofs due to the size limitation of this paper) which will allow us to carry out fundamental cuts and FCB costs in function of π in an optimal way. To the best of our knowledge, these results have never been stated before.

Let $\pi = (e,f)$ with e in T and f in t_e. The following properties hold:

1. for any edge h in T, π changes t_h if and only if f is in t_h as well;
2. $\pi(t_e) = t_f$;
3. for any edge h in T, $\pi(t_h)$ is the symmetric difference of the edge sets t_h and t_e.

The above properties allow us to quickly compute the fundamental cuts at each repeated application of the heuristic move, by updating the existing fundamental cuts.

We now describe equivalent properties on the cycles. Let $\pi = (e,f)$ with e in T and f in t_e, and for all chords k of G with respect to T let c_k be the fundamental cycle in G defined by k. The following hold:

1.for any edge h which is not in t_e, π fixes c_h;
2.for any chord h in t_e , $\pi(c_h)$ is the symmetric difference of the edge sets c_h and c_f.

By using the above properties on fundamental cycles, we can compute the FCB cost by updating (only when necessary) the fundamental cycle corresponding to each chord in the graph.

4. COMPUTATIONAL RESULTS

We have carried out extensive numerical experiments on square mesh graphs with side N. Each such graph has N^2 vertices and $2N(N - 1)$ edges. We have found these graphs to be a hard challenge for the **MinFCB** problem. The following table lists N, the FCB cost found by the heuristic (inserted in a local search scheme) of the corresponding N-square mesh graph, and the computational time required to perform the computation. By comparison, we have also listed the computational result of our own implementation of the NT heuristic for the **MinFCB** problem described in [3]. All the tests have been carried out on a Pentium 4 2.66GHz machine with 1GB RAM and running linux. The source code, in C++, has been compiled by gcc v. 3.0.

N	**FCB Cost**	**CPU time**	**Deo: FCB cost**	**Deo: CPU time**
5	72	0.01s	78	0.01s
10	476	0.5s	518	0.05s
15	1320	00:00:06	1588	0.24s
20	2652	00:00:44	3636	00:00:01
25	4592	00:03:16	6452	00:00:02
30	6962	00:09:11	11638	00:00:04
35	10012	00:28:07	16776	00:00:08
40	13530	01:18:27	28100	00:00:13
45	18040	03:20:45	35744	00:00:24
50	23028	06:51:43	48254	00:00:37
55	28662	12:39:25	62026	00:00:48
60	34986	22:30:04	92978	00:01:06

The results table above shows that the edge-swapping heuristic obtains much better solutions than its tree-growth based counterpart, albeit at a considerable computational time cost. This indicates that edge-swapping is convenient either when the graph is not huge, or when there is a considerable need for a very good solution. Tree-growing heuristic should be used in very large graphs.

5. CONCLUSION

In this paper we gave a short description of the **MinFCB** problem, we proposed a new heuristic move based on swapping edges of the spanning tree corresponding to the current FCB, we established some properties of FCBs which help build an efficient implementation of the algorithm, and we gave some computational results concerning the application of the above-mentioned heuristic move (inserted within a local search scheme) to N-square mesh graphs. We compared these results with the best-known “tree-growth” based heuristic for the **MinFCB** problem. From the comparison it is apparent that “tree-growth” heuristics are much faster than edge-swapping heuristics, but they attain worse results in terms of FCB cost.

Obtaining lower bounds for the **MinFCB** problem is a hard task. The best lower bound found so far is the algebraic bound $4(n - 1)^2$ on n-square mesh graphs, derived from the number of “small squares” in the mesh.

A tabu search implementation built around the above-mentioned heuristic move is currently under way.

6. REFERENCES

[1] Brambilla, A. and Premoli, A., “Rigorous Event-Driven (RED) Analysis of Large-Scale Nonlinear *RC* Circuits”, IEEE Transactions on Circuits and Systems-I, 48(8):938-947, August 2001.
[2] Deo, N., Prabhu, G.M., and Krishnamoorthy, M.S. “Algorithms for generating Fundamental Cycles in a Graph”, *ACM Transactions on Mathematical Software*, 8(1):26-42, March 1982.
[3] Deo, N., Kumar, N., and Parsons, J., “Minimum-length Fundamental-cycle Set Problem: New Heuristics and an Empirical Investigations”, *Congressus Numerantium*, 107:141-154, December 1995.
[4] Galbiati, G., and Amaldi, E., “On the Approximability of the Minimum Fundamental Cycle Basis Problem”, *submitted to Workshop on Approximation and Online Algorithms (WAOA03), Budapest, Sept. 2003*.
[5] Liebchen, C., “Finding Short Integral Cycle Bases for Cyclic Timetabling”, TU Berlin, Institut für Mathematik, *Internal report 2003/12*, June 2003.
[6] Sysło, M., “On some Problems related to Fundamental Cycle Sets of a Graph”, in Chartrand, R. (ed.), *Theory of Applications of Graphs*, 577-588, Wiley, New York, 1981.