Recollections on the discovery of the reverse search technique

David Avis
Kyoto University and McGill University
avis@cs.mcgill.ca

Komei Fukuda and I discovered the idea for reverse search during conversations in Tokyo in October 1990. We were working on the vertex enumeration problem for convex polyhedra. At the time I was visiting Masao Iri at the University of Tokyo and Masakazu Kojima at Tokyo Institute of Technology, supported by a JSPS/NSERC bilateral exchange. Komei was then working at the University of Tsukuba, Otsuka, which was a couple of subway stations from my office. So we met quite often.

One day Komei visited me at my office in Todai and explained to me the criss-cross method for solving linear programs, independently developed by S. Zionts [11], T. Terlaky [8, 9] and Z. Wang [10]. In this method one pivots in the hyperplane arrangement generated by the constraints of the linear program, without regard for feasibility - thus differentiating it from the simplex method. Komei and Tomomi Matsui, a Ph.D. student at Tokyo Institute of Technology, had developed an elegant new proof of the convergence of the criss-cross method [6], which Komei was explaining to me. Komei had drawn a line-arrangement on the blackboard, along with the path the criss-cross method would take from any given vertex to the optimum vertex of the LP. On the board all of these edges were shown in yellow with directions that eventually would lead to the root. In other words it was a directed spanning tree with single sink at the root, call it a criss-cross search tree. Suddenly Komei remarked that if we just followed the edges of the tree in the reverse direction starting at the root, we would be able to visit all the vertices of the arrangement. It was just a matter of deciding which edges of the arrangement were in the criss-cross search tree. This turns out to be easy: from a given vertex v of the arrangement one can easily generate all adjacent vertices in the arrangement by pivoting (assume for now the arrangement is simple). For an adjacent vertex w it suffices to ask if the criss-cross method would yield v in a single pivot. If so wv is an edge in the criss-cross search tree.

Using this edge finding oracle, the reverse search method can be viewed as just doing a depth first search in the criss-cross search tree, starting at its root. The beauty of this was immediately clear to us: if you do a depth first search in a tree you do not need to mark the visited vertices, as there cannot be two paths to the same vertex from the root. Furthermore the DFS can be implemented without a stack for backtracking, as the backtrack vertices can be simply recomputed rather than stored. Since hyperplane arrangements usually have exponentially many vertices, this savings in storage and book-keeping is very attractive.

Although the idea was pleasing, we did not immediately see how general it was. On the way home I remember thinking that no-one was really all that interested in the vertex enumeration problem for arrangements. The problem with lots of applications was the vertex enumeration problem for polyhedra. Then it occurred to me that if we just replaced the criss-cross rule with Bland's least subscript rule for the simplex method [5], the same technique would in fact only generate vertices of the feasible LP region. We were onto something.

We quickly saw that reverse search could be used for a number of related geometric enumeration problems, wrote it up, and published it as a T.I.T Technical Report [1]. During this period I explained the idea to Iri. He was enthusiastic, and invited us to submit a short note on the subject to the new journal, Applied Mathematics Letters, which had a very fast turnaround. Accordingly, our first journal paper on reverse search, without proofs, appeared in early 1991 [2].

We submitted an extended abstract to the 7th ACM Symposium on Computational Geometry and a conference version of the paper appeared in June 1991. It was invited for a special issue of Discrete and Computational Geometry based on the conference, and the full journal version appeared in 1992 [3]. Meanwhile we began developing a whole range of other applications, both geometric and combinatorial. The list kept growing, so this paper took a lot longer to finish. It appeared in 1996 [4].

The latter two papers have been widely cited: as of the time of writing 132 times and 141 times respectively, according to the Web of Science. As a summer project in 2008, John White prepared a database of papers that cite one or other of these papers. He divided these citations into applications and implementations of reverse search. He also prepared a demo to go along with a tutorial that I had written some time ago. This material is available from my McGill home page. I am extremely grateful for his excellent work, and will endeavor to keep the databases reasonably up to date.

As mentioned above, the "reverse search technique" simply amounts to a depth first search in a tree whose edges can be efficiently computed. It seems hard to believe that such an idea had not been used before, and in fact it had. Last year Ernst Mayr kindly informed Komei of Sipser's 1980 paper [7] where the idea, and in fact the name, was used in the completely different context of space bounded Turing machines. Perhaps it is older still.

Acknowledgments

I would like to thank my long time friend and colleague, Komei Fukuda, for sharing his recollections with me. Nevertheless, I take full responsibility for describing any imagined or otherwise fictional events that may or may not have happened some twenty years ago.

References

[1] D. Avis and K. Fukuda, "A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra," Tokyo Institute of Technology, Dept. of Information Science, Research Report B-237, November 1990, 14 pages. http://cgm.cs.mcgill.ca/ avis/doc/avis/AF90a.pdf

[2] D. Avis and K. Fukuda, "A basis enumeration algorithm for linear systems with geometric applications," Applied Mathematics Letters, Vol.4, Pages 39-42 (1991).

[3] D. Avis and K. Fukuda, "A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra," Discrete and Computational Geometry, Vol. 8, pp. 295-313 (1992).

[4] D. Avis and K. Fukuda, "Reverse search for enumeration," Discrete Applied Math, Vol. 65, pp. 21-46 (1996).

[5] R.G. Bland, "A combinatorial abstraction of linear programming," Journal of Combinatorial Theory Series B 23 (1977), pp. 33-57.

[6] K. Fukuda and T. Matsui, "On the finiteness of the criss-cross method," European Journal of Operations Research, Vol 52 pp. 119-124 (1991).

[7] M. Sipser, "Halting space-bounded computations," Theoretical Computer Science 10 (1980) 335-338.

[8] T. Terlaky, "A convergent criss-cross method," Mathematische Operationsforschung und Statistik Series Optimization, 16 (1985), pp. 683-690.

[9] T. Terlaky, "A finite criss-cross method for oriented matroids," Journal of Combinatorial Theory Series B 42 (1987), pp. 319-327.

[10] Z. Wang, "A conformal elimination free algorithm for oriented matroid programming," Chinese Annals of Mathematics 8 B 1 (1987).

[11] S. Zionts, "The criss-cross method for solving linear programming problems," Management Science 15 (1969), pp. 426-445.