6 Computational Results

6.1 Experimental Setup

We compiled a test suite of various instances on which we evaluated our linear/integer program (1)–(5) and the iterated rounding technique for *St-Mat*$_2(P)$. The suite includes ten instances with up to 442 points from the TSPLIB [21] (last point removed from odd cardinality instances; therefore the results reported here are more meaningful than those in [10]); the C-class ("clustered") of Solomon's instances of the vehicle routing problem [24] with 100 points each; 25 regular grids with 20 to 360 points, based on grids of size 5×5 up to 20×20 in which 20% of the points are removed (chosen uniformly at random); and a set of instances with up to 100 random points in the plane. Even though we experimented with available separation routines for blossom inequalities these are not included in the LPs on which we report below, as solution quality is already excellent.

Tables 1 and 2 display our results on a 2.8GHz Pentium 4 Linux PC with 1GB main memory, using the commercial LP/IP solver CPLEX 9.1. For each instance we list its name, which indicates the number of points; this number is reduced by one for odd names to allow a perfect matching. Also listed are the optimal objective function values for the linear (LPopt) and the integer (IPopt) program, together with the respective CPU time in seconds. The last column displays the approximate stabbing number obtained from iterated rounding. For solving the integer programs we set a time limit of four CPU hours which was exceeded for some large instances. This is indicated by brackets around the corresponding value. In that case we report the CPU seconds it takes to obtain the listed best known solution. To provide some intuition what fractional matchings of minimum (fractional) stabbing number look like, we show several of them in Figures 12–15. The edge weight is proportional to the thickness of edges in the drawing.

6.2 Brief (Additional) Observations

In fractional solutions variables may assume rather arbitrary fractional and small values; this is also true when blossom inequalities are added. The colinearity of points in the grid instances enables us to reduce the number of stabbing constraints, resulting in significantly reduced computation times. The clustering of points in the vehicle routing instances obviously facilitate the LP/IP solution process, as was to be expected. However, this observation is interesting in practice where the data is usually well structured, as opposed to randomly distributed.

In our experiments, the stabbing number obtained from iterated rounding is extremely close to the optimum: it is never off by more than by an *absolute value* of one or two, i.e., much better than predicted by our analysis (Lemma 13). Computation times are comparable to solving the linear program because an LP solver will exploit the fact that linear programs only differ very slightly from iteration to iteration, and will perform a "warm start." We also experimented with a "one good shot at once" approach that is based on the fact that each fractional matching is the convex combination of perfect matchings, by finding a maximum weight perfect matching in the support graph of the LP solution. This tends to give very good feasible solutions and certainly deserves further evaluation, both from a computational and from a theoretical point of view.

We also made an experiment (reported in [10]) to show that the stabbing constraints seem to completely destroy the polyhedral structure of the matching polytope. Half of the original TSPLIB instances we used are infeasible (because they originally had an odd number of points), and this is not

Instance	LP opt	LP CPU	IP opt	IP CPU	iterated
ulysses22	1.992	0.00	2	0.01	2
berlin52	2.815	0.02	4	0.90	5
lin105	5.500	0.15	6	80.57	8
bier127	4.297	0.34	(6)	(3.90)	7
u159	15.000	0.15	15	2.37	15
ts225	13.700	0.35	(15)	(122.28)	16
tsp225	11.500	0.32	12	7.66	12
a280	10.500	4.26	(12)	(284.80)	12
lin318	8.113	12.65	(10)	(6825.48)	11
pcb442	16.500	20.71	17	3289.41	18
c101	7.000	0.03	7	0.54	8
c102	7.000	0.03	7	0.54	8
c103	7.000	0.03	7	0.53	8
c104	7.000	0.03	7	0.53	8
c105	7.000	0.03	7	0.53	8
c106	7.000	0.04	7	0.54	8
c107	7.000	0.03	7	0.54	8
c108	7.000	0.03	7	0.53	8
c201	6.000	0.03	6	2.37	7
c202	6.000	0.03	6	2.37	7
c203	6.000	0.03	6	2.36	7
c204	6.000	0.03	6	2.37	7
c205	6.000	0.04	6	2.35	7
c206	6.000	0.04	6	2.37	7
c207	6.000	0.04	6	2.46	7
c208	6.000	0.40	6	4.18	7

Table 1: TSPLIB and clustered instances: Comparison of fractional and integer optimal stabbing number $\mathit{St\text{-}Mat}_2(P)$, and the one obtained from iterated rounding. Brackets around values indicate an exceeded time limit of four CPU hours, and we report the best known solution obtained after the time given.

Instance	LP opt	LP CPU	IP opt	IP CPU	iterated
grid5a	2.500	0.00	3	0.01	3
grid5b	2.750	0.00	3	0.00	3
grid5c	2.750	0.01	3	0.01	4
grid5d	2.000	0.00	3	0.01	3
grid5e	2.500	0.00	3	0.01	3
grid8a	5.003	0.01	6	0.03	6
grid8b	5.125	0.01	6	0.05	6
grid8c	5.000	0.00	5	0.05	6
grid8d	5.429	0.01	6	0.04	7
grid8e	5.403	0.00	6	0.21	6
grid10a	4.250	0.01	5	0.17	6
grid10b	4.250	0.01	5	0.13	6
grid10c	5.250	0.01	6	0.19	6
grid10d	4.500	0.02	5	1.17	6
grid10e	5.000	0.01	5	0.32	6
grid15a	6.000	0.03	6	15.92	7
grid15b	7.500	0.03	8	1.11	8
grid15c	6.000	0.03	(7)	(7.01)	7
grid15d	6.500	0.03	7	54.73	7
grid15e	6.750	0.02	7	761.33	7
grid20a	9.167	0.98	(11)	(43.17)	12
grid20b	9.250	0.27	(11)	(84.97)	11
grid20c	9.500	1.54	(11)	(33.05)	12
grid20d	9.500	2.24	(11)	(448.00)	12
grid20e	10.000	2.89	11	1169.66	12
random10a	1.750	0.00	2	0.01	2
random10b	1.834	0.00	2	0.00	2
random10c	1.750	0.00	2	0.01	2
random10d	1.700	0.00	2	0.01	2
random10e	1.813	0.00	2	0.01	2
random50a	2.595	0.25	3	19.83	4
random50b	2.628	0.23	3	1.91	4
random50c	2.669	0.23	4	30.77	4
random50d	2.662	0.22	4	15.99	4
random50e	2.790	0.33	4	25.54	4
random100a	3.376	5.57	(5)	(14.00)	6
random100b	3.406	1.04	(5)	(13.81)	5
random100c	3.247	0.99	(5)	(16.31)	6
random100d	3.211	0.89	(5)	(7.35)	6
random100e	3.233	0.90	(5)	(5.84)	5

Table 2: Results for grids and random instances

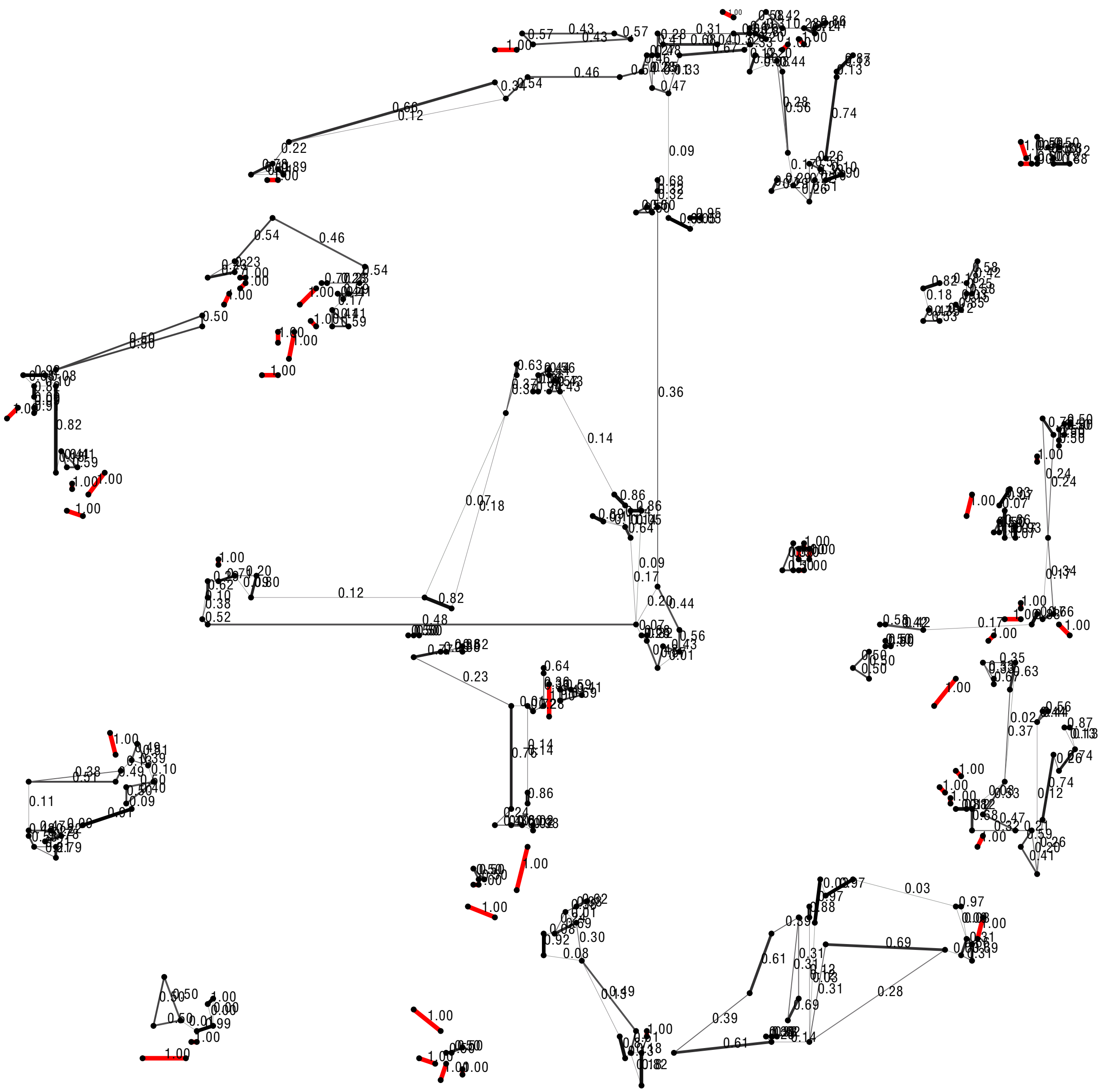

Figure 12: LP optimal solution for a “clustered” instance similar to c101, but with 400 points

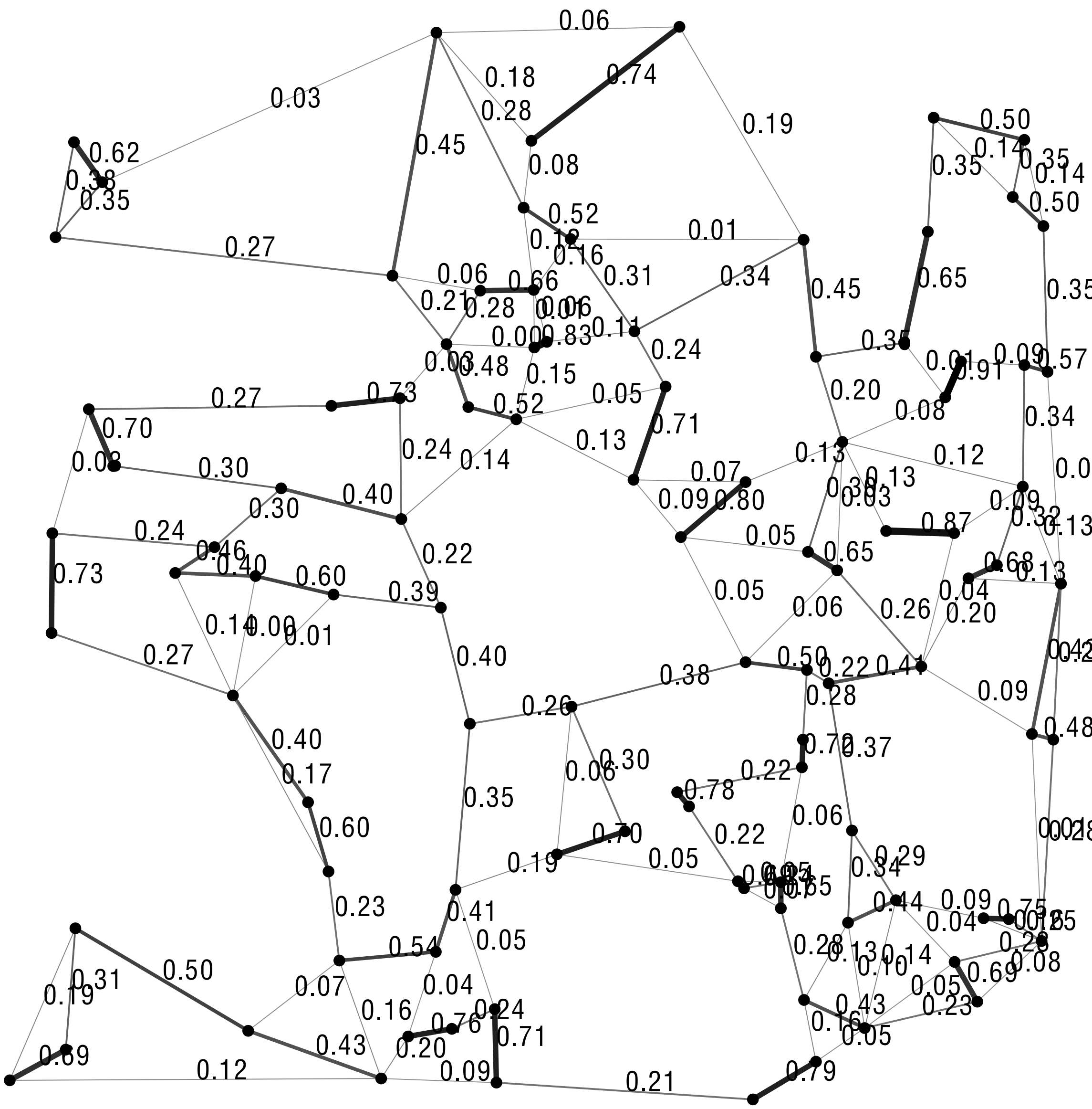

Figure 13: LP optimal solution for random100a

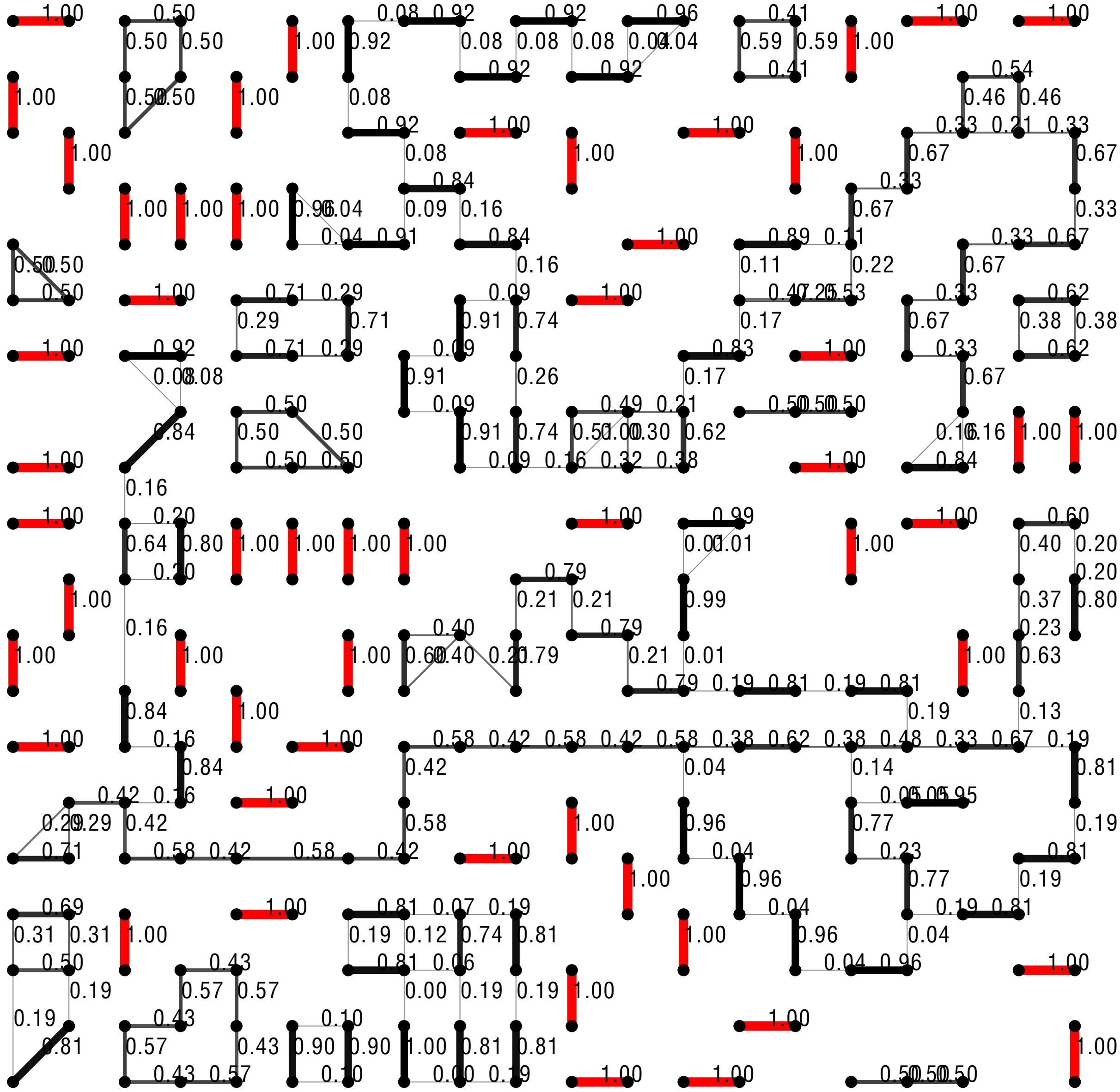

Figure 14: LP optimal solution for grid20b

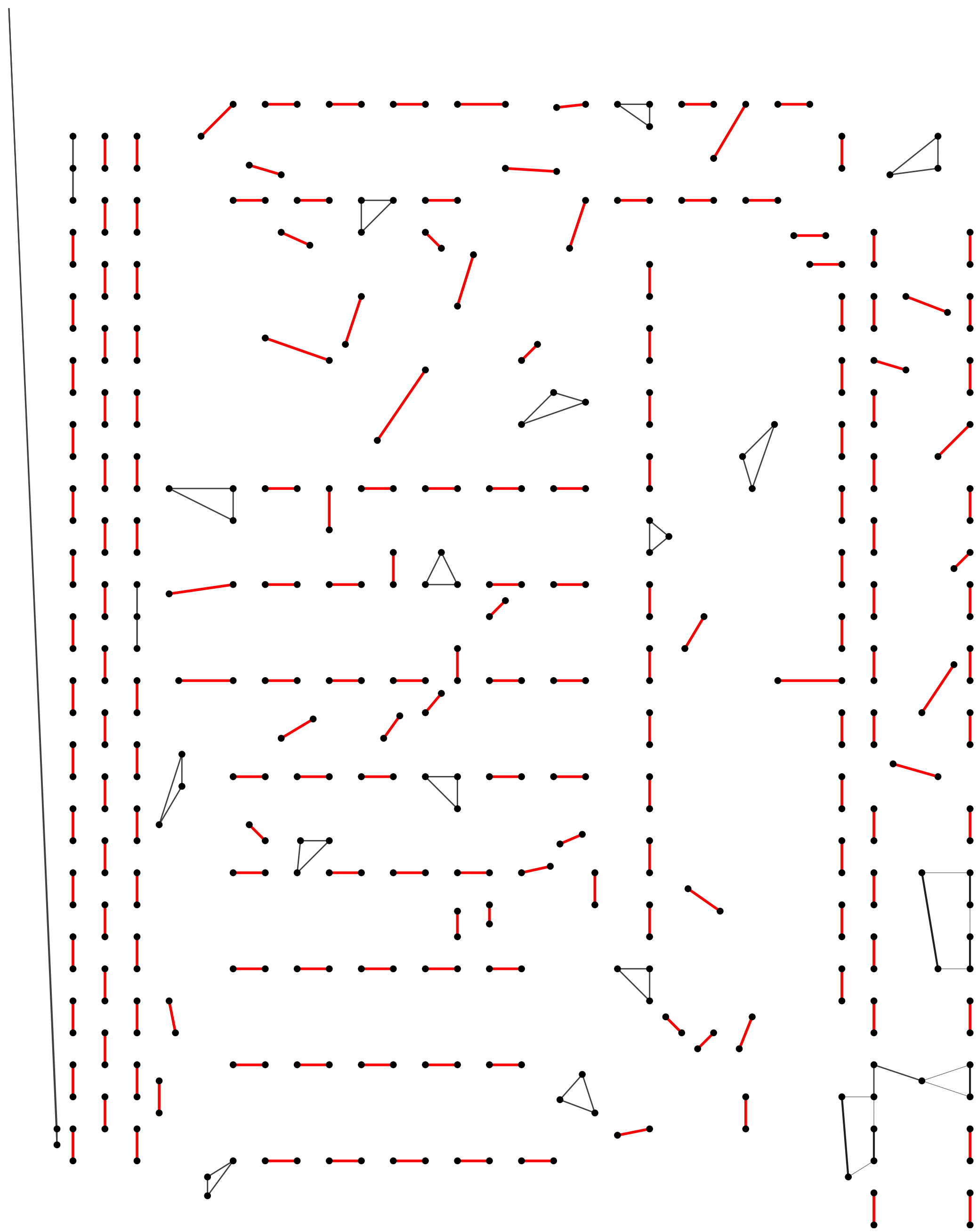

Figure 15: LP optimal solution for pcb442; ‘pcb’ stands for ‘printed circuit board’

detected by the state-of-the-art CPLEX IP solver within four CPU hours. Iterated rounding terminates (quickly) in this case with a non-perfect matching with one point unmatched.

Feasible integer solutions of good quality are usually obtained rather quickly by our integer programs. The time-consuming part appears to be a proof of optimality, but the lower bound increases only slowly in the branch-and-bound tree. It would be worthwhile to investigate strengthening the lower bound obtained from the LP relaxation by means of valid inequalities ("cutting planes").

7 Notes and Conclusion

We have presented the first algorithmic paper on stabbing numbers, resolving the long-standing open question of complexity, and providing an approach that appears to be useful in theory and in practice. There are a number of interesting open questions.

We were not able to extend our $\mathcal{NP}$-hardness proof to the case of finding a triangulation of minimum (general) stabbing number. Our proofs rely on a strong degeneracy of the point set, and it would be interesting to see a proof for points in general position.

Probably the most intriguing open question spawned by our work is whether the iterated rounding scheme suggested by the existence of a heavy edge in an optimal fractional solution to our linear programs (Lemmas 13 and 14) does indeed lead to a constant-factor approximation algorithm. Also, the use of the Ellipsoid method (at least as a theoretical argument) is not "combinatorial", which always has to be considered a drawback.

Another interesting question is to decide the existence of structures of small constant stabbing number. As the hardness proof for deciding the existence of a matching of stabbing number 5 illustrates, this is still not an easy task. From some solvable special cases, we only note one:

Theorem 16 *St-Tre*$_2(P)$*=2 and St-Mat*(P)*=2 can be decided in polynomial time.*

One may also ask for minimizing the *average* instead of the maximum stabbing number, and refer to the average over the whole continuum of lines intersecting a set of line segments, instead of just a combinatorial set of representatives. This, however, amounts to solving problems of minimum length, with all implications to hardness and approximation.

Theorem 17 *A set of line segments has minimum average (axis-parallel, resp.) stabbing number, iff the overall Euclidean (Manhattan, resp.) length of all line segments is minimum.*

We remark that a linear program for minimizing the average stabbing number can be written with a sum in the objective function (instead of a maximum as we had to model it), allowing to directly apply our iterated rounding technique and obtaining the desired approximation factors of 3 and 5, respectively.

Acknowledgments

We thank Joe Mitchell for pushing us to work on this problem, and repeated discussions that assured further progress. We also thank Kamal Jain for some discussions on iterated rounding.