Chapter 3

Adapting Search Theory to Networks

Brian Alspach[1], Kristen Jaskie[2], Theodore Kolokolnikov[3], Denis Kosygin[4], Nathan Krislock[5], Aihna Li[5], Jacky Li[6], Eva-Marie Nosal[3], Moshe Rosenfeld[2], Sarah Sumner[7], Ling Zhao[8]
Report prepared by Brian Alspach

3.1 Introduction

The problem under study here was brought by Dr. Allan Douglas of the Communication Security Establishment. The Communication Security Establishment (hereafter referred to as the CSE) includes a team composed of Defence Scientists assigned from the Operational Research Division, National Defence Headquarters, Department of National Defence, Ottawa.

The CSE is interested in the general problem of locating objects in networks. Their interest in this type of problem arises because of the emerging concern regarding security issues for information operations.

The concept of transmitting code from one computer to another has been around for more than forty years. As a programming practice, it has evolved from an occasional concern of systems programmers working at the deepest levels of operating systems to a common and widespread practice. It is now utterly unremarkable for a web browser to bring in a web page containing a JAVA applet which then executes.

The next stage, in which objects move under conditions of their own determination, already is upon us. Development environments are being created to facilitate this. One of these, for example, is for the creation and use of “Aglets” which are examples of mobile network agents. Mobile network agents are programs that can be dispatched from one computer and transported to a remote computer for execution. At the remote computer they present their credentials in

[1]University of Regina
[2]University of Washington
[3]University of British Columbia
[4]Princeton University
[5]Loyola University
[6]Dalhousie University
[7]University of Ottawa
[8]University of Alberta

order to gain access to local services and data. It is apparent that mobile network agents are going to undergo considerable development and become extensively used. The networked world is going to see many of these objects.

Prudence dictates the expectation that some of the mobile objects will not be benign. Defensive information operations will have to deal with mobile attackers. The problems of determining whether or not attackers are present or likely to be present, and determining their present locations become of considerable interest.

The CSE people were familiar with the literature involving search theory, where the searches are carried out over two- and three-dimensional regions. The techniques for these kinds of searches typically involve partitioning the region into cells and considering the problem of getting the searcher (or one of several searchers) and the target in the same cell.

They also were familiar with an old paper by G. Pólya [34] in which he considered random searches on the n-dimensional grid with a single searcher and a single target. They were not aware of any other work done on searching in graphs.

Because of their exposure to search theory, the problem they brought to the workshop was phrased in terms of adapting search theory to networks. Thus, the first step was the introduction of an already existing healthy literature on searching graphs.

T. D. Parsons, who was then at Pennsylvania State University, was approached in 1977 by some local spelunkers who asked his aid in optimizing a search for someone lost in a cave in Pennsylvania. Parsons quickly formulated the problem as a search problem in a graph. After his paper [31] appeared, many subsequent papers appeared. Subsequent papers led to two divergent problems. One problem dealt with searching under assumptions of fairly extensive information, while the other problem dealt with searching under assumptions of essentially zero information. These two topics are developed in the next two sections.

3.2 Complete Information

There is a variety of models we may use for attempting to find an evader or evaders in a graph. The notion of *searching* a graph involves the evaders and searchers being located at vertices. Evaders and searchers may stay at their current locations, or move along an edge of the graph to a neighboring vertex. The simplest clock is one which ticks at regular intervals and all moves take place when the clock ticks. In one version, any subset of searchers and evaders may move at each tick. In a second version, subsets of searchers move on odd numbered ticks and subsets of evaders move on even numbered ticks — the point being that evaders and searchers alternate moves. In the most general version, the clock is continuous and participants may move at any time. In all of these models, capture takes place whenever an evader and a searcher occupy the same vertex at the same time.

Another decision which must be made for searching is whether or not capture takes place if an evader and a searcher "pass" each other along the same edge. For example, if at a tick of the clock, an evader moves along an edge from vertex u to v and a searcher moves along the same edge from vertex v to vertex u, does capture take place?

The notion of *sweeping* a graph involves evaders and searchers being able to be located at vertices or along an edge. In this case, we view the graph as being embedded in euclidean space.

The movement of an evader or searcher corresponds to a continuous function from the interval $[0, \infty)$ to the embedded graph. If there are k searchers, let $f_1, f_2, \ldots, f_k$ denote the k functions describing the searchers' movements in the graph. If there are ℓ evaders, let $e_1, e_2, \ldots, e_\ell$ denote the ℓ functions describing the evaders' movements in the graph. Capture takes place whenever there exist t, i, j such that $f_i(t) = e_j(t)$. In other words, a searcher and an evader occupy the same place at the same time.

Parsons' first paper on this topic [31] considered sweeping a graph. He made the following definition.

Definition 3.2.1 Let G be a finite graph. The *sweep number* of G, denoted $\mathrm{SW}(G)$, is the smallest integer k such that k searchers can sweep G and capture a single evader.

He observed that $\mathrm{SW}(G)$ always exists because $|V(G)| + 1$ searchers can always capture an evader. This is done by placing a searcher at each vertex and then using an additional searcher to move along every edge of the graph. If the graph is not connected, the additional searcher may sweep each component separately.

He proved the following theorem as well.

Theorem 3.2.2 *If T is a tree and $k \geqq 1$, then $\mathrm{SW}(T) \geqq k + 1$ if and only if T has a vertex v at which there are at least three branches T_1, T_2, T_3 satisfying $\mathrm{SW}(T_j) \geqq k$ for $j = 1, 2, 3$.*

He was able to use the theorem to recursively characterize all trees with a given sweep number. At the end of [31], Parsons suggested many other variations of the problem for investigation.

A. LaPaugh [22] first proved that a graph may be optimally swept without going over any edge twice. D. Bienstock and P. Seymour gave a new proof in [5].

Other people took up the problem obtaining many results. There also was a shift towards considering searching rather than sweeping, and considering a single evader. The usual model was alternating moves and complete information, that is, everybody knows everybody's location.

Definition 3.2.3 Let G be a finite graph. The *search number* of G, denoted $\mathrm{SE}(G)$, is the smallest integer k such that k searchers can search G and capture a single evader using alternate moves and with complete information.

One direction taken by various researchers was to consider classes of graphs. The class of Cayley graphs is an interesting class because of their use as models for network architectures. P. Frankl [17] was the first person to consider Cayley graphs. He proved the following result.

Theorem 3.2.4 *If $X = X(G; S)$ is a Cayley graph on the finite abelian group G, then $\mathrm{SE}(X) \leqq (|S| + 1)/2$.*

Since the n-dimensional cube is a Cayley graph on a finite abelian group, the preceding result immediately yields the following corollary.

Corollary 3.2.5 *If Q_n is the n-dimensional cube, then $\mathrm{SE}(Q_n) \leqq (n + 1)/2$.*

M. Aigner and M. Fromme [1] characterized the class $\mathcal{C}$ of graphs for which $\mathrm{SE}(X) = 1$ whenever $X \in \mathcal{C}$. In particular, they showed that $\mathcal{C}$ contains all finite trees.

M. Maamoun and H. Meyniel [26] generalized the Aigner-Fromme result about trees in the following way.

Theorem 3.2.6 *If X is a cartesian product of n finite trees, each of which has at least two vertices, then* $\mathrm{SE}(X) = \lceil (n+1)/2 \rceil$.

Aigner and Fromme [1] also proved the following interesting result for another important family of graphs.

Theorem 3.2.7 *If X is a planar graph, then* $\mathrm{SE}(X) \leqq 3$.

T. Andreae [3] generalized the preceding result in the direction of classes of graphs not containing a fixed graph as a minor. A. Quilliot [35] extended the result in the following direction.

Theorem 3.2.8 *If X is a graph of genus k, then* $\mathrm{SE}(X) \leqq 3 + 2k$.

The preceding results determine either upper bounds or exact values for the search number of graphs. P. Frankl [18] determined a lower bound in terms of girth and minimum degree.

Theorem 3.2.9 *If X is a graph with girth at least $8t - 3$ and minimum degree larger than d, then* $\mathrm{SE}(X) > d^t$.

A negative result about lower bounds was established by T. Andreae [2]. He proved that for any positive integers m and d, there exists a regular graph of degree d for which the search number is bigger than m.

In [4] it is shown that for each fixed m, there is a polynomial time algorithm determining whether or not a fixed graph X satisfies $\mathrm{SE}(X) \leqq m$. Complexity also was studied by A. Goldstein and E. Reingold in [19].

R. Nowakowski and P. Winkler [30] established a structural result by characterizing the graphs for which one searcher can always capture one intruder under the complete information model.

3.3 Zero Information

Zero information means that nobody knows anything about the location of anyone else. In fact, the searchers do not know if there even is an intruder in the graph they are searching. The problem is to devise a search mechanism whereby m searchers are guaranteed of finding an intruder or establishing that the graph is free of any intruders.

Almost all the work which has been done on this version is the study of collision properties of two random walks in graphs. The random search assumptions are that if someone is located at a vertex of valency d, then at the next tick of the clock, he moves to a neighboring vertex with probability $1/d$ for each of the d neighbors. Under this assumption, the following theorem holds.

Theorem 3.3.1 *Let u and v be any two distinct vertices of a graph X, let* $\mathrm{dist}(u, v)$ *denote the distance between u and v in X, let $\Delta(X)$ denote the maximum valency of X, let $T(u, v)$ denote the first time two random walks beginning at u and v occupy the same vertex, and let* $\mathrm{ET}(u, v)$ *denote the expected value of $T(u, v)$. For any two distinct vertices u and v of X,*

- $\mathrm{Prob}(T(u,v) < \infty) = 1$,
- $\mathrm{ET}(u,v) < 5\Delta(X)\mathrm{dist}(u,v)|V(X)|$, *and*
- *for every* $\epsilon > 0$, *there exists* $C(\epsilon) > 0$ *such that* $\mathrm{Prob}(T(u,v) > C(\epsilon)|V(X)|) \geqq 1 - \epsilon$.

Increasing the number of searchers does materially affect the preceding result. For a good discussion of this material see [23].

3.4 Workshop

On the afternoon of the first day, the group had its first meeting. Several participants were aware of some of the existing literature on searching and sweeping graphs. Our first course of action was to launch an extensive literature search, visit the excellent mathematics library on campus, and make copies of those papers which seemed most relevant.

By the end of the afternoon, most of the references given in the bibliography had been discovered and copies of approximately ten of the papers were distributed to the group. The object was to be able to begin to discuss their contents Tuesday morning.

Tuesday morning was spent discussing the contents of the papers, and exploring the potential usefulness with regard to the problem Dr. Douglas had in mind. Dr. Douglas himself found some of the results of considerable interest, but was mostly impressed by the fact these kinds of problems had been examined by a variety of researchers.

It became clear that different subgroups were becoming interested in pursuing different aspects of the problem. The rest of Tuesday was spent trying to work out clear objectives for different subgroups. Simultaneously, we were being presented with a nice description of random walks in graphs by Dr. Kosygin. By the time Tuesday evening rolled around, it was clear we still were not well organized.

Wednesday morning was spent completing the development of objectives for several subgroups. The subgroups then went their separate ways agreeing to meet late in the afternoon to sum up what they had accomplished during the day.

One subgroup was working on various problems dealing with trees. Their first foray into this was an attempt to independently work out proofs of several of the results on trees that we found in the literature. After coming to grips with the proofs, they then began to think about possible algorithms for computing search numbers for trees.

One approach is based on reducing a tree by removing all the leaves in a single stage. After this stage, the smaller tree then has its leaves removed. This is a layered approach and is recursive. It is shown that one less searcher is needed at each stage.

A divide and conquer approach for trees is based on deleting a central vertex, leaving a searcher at the deleted vertex to prevent the target from moving from one subtree into another subtree, and then searching each subtree separately. It is clear that the minimum number of searchers needed to search the entire tree is one more than the maximum number needed to search any of the subtrees created by the deletion of the vertex.

The last activity undertaken was an attempt to modify some of the known results on trees to other search models. They made a presentation to the group on Thursday.

A second subgroup wanted to look at the behavior of random searches. They decided to concentrate on the n-dimensional cube, with some consideration of complete graphs.

Let's look at what was done for the 10-dimensional cube first. The following table contains information on how long it took six searchers, using random search, to capture one target in the 10-dimensional cube, and the following table one searcher.

The column headed *Frequency* gives the number of random searches using a number of steps in the range shown in the column headed *Time*.

Frequency	Time
1	1 – 71
71	72 – 142
57	143 – 213
38	214 – 285
29	286 – 356
18	357 – 427
16	428 – 499
9	500 – 570
7	571 – 641
2	642 — 713
3	714 – 784
3	785 –

Six searchers, one target

Frequency	Time
1	11 – 366
72	367 – 722
50	723 – 1,080
38	1,081 – 1,435
24	1,436 – 1,792
24	1,793 – 2,148
18	2,149 – 2,505
5	2,506 – 2,861
8	2,862 – 3,217
5	3,218 — 3,574
3	3,575 – 3,930
1	3,931 – 4,287
3	4,288 – 4,643
5	4,644 –

One Searcher, one target

The next table provides some data correlating the number of searchers and the average number of steps required to capture a single target, using random search, in the 10-dimensional cube. The data indicates a definite trend, but many more trials are necessary to increase the accuracy. The column headed *Time* is the average number of steps required using the number of searchers under the column headed *Number*.

Number	Time
1	1,060
2	567
3	412
4	220
5	20
6	195
7	170
8	140
9	130
10	120
11	95
12	103
13	82
14	94
15	72
16	70
17	62
18	60
19	55
20	51

Number of searchers and time

For one searcher and one target, the next table relates steps to capture and the dimension of the cube.

Dimension	Time
1	0
2	2
3	10
4	25
5	40
6	85
7	130
8	320
9	590
10	1100

Dimension and time

We ran random searches, using one searcher and one target, on complete graphs whose orders were multiples of 100. We started with 100 vertices and went through 2,000 vertices. The number of trials again needs to be increased considerably in order to introduce more accuracy. Still, the average time behaved reasonably except for 1,300 and 1,400 vertices. From 100 through 1,800 vertices, the average number of steps required was approximately equal to the number of vertices. For 1,900 and 2,000 there were steep increases. There is a lot of room for studying this question more.

The subgroup working on generating random searches also made a presentation to the group on Thursday.

The rest of Thursday was used for amalgamating our efforts into a report to be presented Friday.

3.5 Summary

The following summarizes what we discovered during the week.

- Search results are greatly affected by the paradigm used.
- Searches on well defined families of graphs already present challenging problems.
- Simulations on random graphs and the n-dimensional cube agreed with computed expectations.
- Random searches usually perform as well as structured searches.
- The surface of the problem has been only scratched.

3.6 Acknowledgements

The team wishes to thank the University of Washington and the organizers for providing us with wonderful facilities, well organized structure, and excellent responses to our requests, even those that might have been somewhat unreasonable.

Most of all we wish to thank Dr. Allan Douglas of CSE for his patience, humor, detailed explanations in response to our questions, willingness to spend the entire week at the session, and the stimulating question he brought to the Workshop.

I, personally, want to thank Allan for his encouragement to go ahead and develop a MITACS proposal based on the Workshop experience. A proposal has been submitted and at the time of this report, we are awaiting a final decision.

Bibliography

[1] Aigner, M. and Fromme, M., *A game of cops and robbers*, Discrete Appl. Math., **8** (1984), 1–12.

[2] Andreae, T., *Note on a pursuit game played on graphs*, Discrete Appl. Math., **9** (1984), 111–115.

[3] Andreae, T., *On a pursuit game played on graphs for which a minor is excluded*, J. Combin. Theory Ser. B, **41** (1986), 37–47.

[4] Berarducci, A. and Intrigila, B., *On the cop number of a graph*, Adv. in Appl. Math., **14** (1993), 389–403.

[5] Bienstock, D. and Seymour, P., *Monotonicity in graph searching*, J. Algorithms, **12** (1991), 239–245.

[6] Brightwell, G. and Winkler, P., *Maximum hitting time for random walks on graphs*, J. Random Structures and Algorithms, **1** (1990), 263–276.

[7] Clarke, N.E., The Effects of Replacing Cops and Searchers with Technology, M.Sc. Thesis, Dalhousie University, 1999.

[8] Clarke, N.E. and Nowakowski, R.J., *Cops, robber and photoradar*, Ars Combinatoria, **LVI**, 97–103.

[9] Clarke, N.E. and Nowakowski, R.J., *Cops, robber and traps*, Utilitas Math., (to appear).

[10] Coppersmith, D., Tetali, P. and Winkler, P., *Collisions among random walks on a graph*, SIAM J. Discrete Math., **6**, (1993), 363–374.

[11] Ellis, J.A., Sudborough, I.H. and Turner, J.S., *The vertex separation and search number of a graph*, Inform. and Comput. **113** (1994), 50–79.

[12] Fitzpatrick, S.L., *The isometric path number of the Cartesian product of paths*, Proc. Thirtieth Southeastern Conf. Combinatorics, Graph Theory, and Computing, Congr. Numer. **137** (1999), 109–119.

[13] Fitzpatrick, S.L. and Nowakowski, R.J., *Copnumber of graphs with strong isometric dimension two*, Ars Combin. **59** (2001), 65–73.

[14] Fitzpatrick, S.L. and Nowakowski, R.J., *The strong isometric dimension of finite reflexive graphs*, Discuss. Math. Graph Theory, **20**, 23–38.

[15] Fomin, F.V., *Note on a helicopter search problem on graphs*, Proc. Conference Optimal Discrete Structures and Algorithms—ODSA '97, Discrete Appl. Math., **95** (1999), 241–249.

[16] Fomin, F.V., *Helicopter search problems, bandwidth and pathwidth*, Discrete Appl. Math., **85** (1998), 59–70.

[17] Frankl, P., *On a pursuit game on Cayley graphs*, Combinatorica, **7** (1987), 67–70.

[18] Frankl, P., *Cops and robbers in graphs with large girth and Cayley graphs*, Discrete Appl. Math., **17** (1987), 301–305.

[19] Goldstein, A.S. and Reingold, E.M., *The complexity of pursuit on a graph*, Theoret. Comput. Sci., **143** (1995), 93–112.

[20] Hahn, G. and Tardif, C., *Graph homomorphisms: structure and symmetry*, Graph Symmetry, Kluwer Acad. Publ., Dordrecht, 1997, 107–166.

[21] Hamidoune, Y.O., *On a pursuit game on Cayley digraphs*, European J. Combin., **8** (1987), 289–295.

[22] LaPaugh, A., *Recontamination does not help to search a graph*, Technical Report No. 335, Department of Electrical Engineering and Computer Science, Princeton University, 1982.

[23] Lovász, L., *Random Walks on graphs: a survey*, Combinatorics, Paul Erdös is Eighty, Vol. 2, Keszthely, 1993, 353–397.

[24] Lovász, L. and Winkler, P., *A note on the last vertex visited by a random walk*, J. Graph Theory, **17** (1993), 593–596.

[25] Lovász, L. and Winkler, P., *Mixing of random walks and other diffusions on a graph*, Surveys in Combinatorics, London Math. Soc. Lecture Note Ser., **218**, Cambridge University Press, 1995, 119–154.

[26] Maamoun, M. and Meyniel, H., *On a game of policemen and robbers*, Discrete Appl. Math., **17** (1987), 307–309.

[27] Neufeld, S., The Game of Cops and Robber, M.Sc Thesis, Dalhousie University, 1990.

[28] Neufeld, S. and Nowakowski, R.J., *A game of cops and robber played on products of graphs*, Discrete Math., **186**, 253–268.

[29] Nowakowski, R.J., *Search and sweep numbers of finite directed acyclic graphs*, Discrete Applied Math., **41** (1993) 1–11.

[30] Nowakowski, R. and Winkler, P., *Vertex-to-vertex pursuit in a graph*, Discrete Math., **43** (1983), 235–240.

[31] Parsons, T.D., *Pursuit evasion in a graph*, Lecture Notes in Math., **642**, Springer-Verlag, Berlin, 1978, 426–441.

[32] Parsons, T.D., *The search number of a connected graph*, Proc. Ninth Southeastern Conf. Combinatorics, Graph Theory, and Computing, Cong. Num., **21** (1978), 549–554.

[33] Polat, N., *Retract-collapsible graphs and invariant subgraph properties*, J. Graph Theory, **19** (1995).

[34] Pólya, G. *Über eine Aufgabe der Wahrscheinlichkeitsrechnung betreffend die Irrfahrt in Strassennetz*, Math. Ann., **84** (1921), 149–160.

[35] Quilliot, A., Thèse d'Etat, Université de Paris VI, 1983.

[36] Quilliot, A., *A short note about pursuit games played on a graph with a given genus*, J. Combin. Theory Ser. B, **38** (1985), 89–92.

[37] Quilliot, A., *About the complexity of some homomorphism problems on graphs*, Topics in Combinatorics and Graph Theory, Physica, Heidelberg, 1990, 561–567.

[38] Schroeder, B.S.W., *The copnumber of a graph is bounded by* $\lfloor \frac{3}{2}\ \textit{genus}\ (G) \rfloor + 3$, Trends Math., Birkhäuser Boston, 2001, 243–263.

[39] Tošić, R., *On cops and robber game*, Studia Sci. Math. Hungar., **23** 1988.

[40] Winkler, P., *Dependent percolation and colliding random walks*, Random Structures Algorithms, **16** (2000), 58–84.