A summary of

Dynamic location problems with limited look-ahead

by Fan Chung and Ronald Graham

The following problem arose in connection with studies of Internet web page caching. The general setting is as follows:

In some fixed metric space M, k "servers" $S_1, \ldots, S_k$ are given with some arbitrary initial locations in M. Requests for service at certain points $\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_N$, in M arrive over time. Immediately after request σ_t is received, exactly one of several mutually exclusive actions must be taken:

(i) Some server is moved to σ_t, with a resulting cost of $c(\sigma_t)$, the "cost" of the point σ_t.

(ii) No server moves. In this case, the cost for "no service" is defined to be $\min_k \ d(S_k, \sigma_t)$, where $d(x, y)$ denotes the distance between x and y in M.

A further feature of our model is that two parameters $u, w \geq 0$ are specified, which are used as follows. Before having to decide how to service request σ_t, the servers have at their disposal the knowledge of the $u + w$ requests σ_i with $t - u \leq i \leq t + w - 1$. Thus, the servers can only "remember" or store the past u requests $\sigma_{t-u}, \sigma_{t-u+1}, \ldots, \sigma_{t-1}$ but are allowed to know the w future requests $\sigma_t, \sigma_{t+1}, \ldots, \sigma_{t+w-1}$ before having to service σ_t.

The rules which govern the choices made for servicing all the σ_t define some algorithm A. In this model, A is deterministic and can only depend on the values of the σ_i which it currently knows, and nothing else. In particular, A is not allowed to make probabilistic choices based on some source of randomness. We denote by $A(\sigma)$, the cost of servicing the request sequence $\sigma = (\sigma_1, \ldots, \sigma_N)$.

Of course, if we are allowed to know *all* the σ_t before having to act, it is very likely the cost of servicing σ can be decreased. Let us denote by $\mathrm{OFF}(\sigma)$ the minimum possible cost of servicing σ in this case. Thus, $\mathrm{OFF}(\sigma)$ is the cost for an optimal *off-line* algorithm for σ.

The problem of interest here is to investigate the effect of being able to look ahead (as well as utilizing the past history) on the performance of such algorithms.

For a given request sequence, let $\mathrm{OPT}^{(u,w)}(\sigma)$ denote the minimum cost of any (u, w)-window algorithm servicing σ, and we define

$$\rho(u, w) : \;\; = \;\; \inf_{\sigma} \; \frac{\mathrm{OPT}^{(u,w)}(\sigma)}{\mathrm{OFF}(\sigma)}$$

over all σ with $\mathrm{OFF}(\sigma) \to \infty$ to avoid certain inessential degeneracies. (In our model, we are assuming our "adversary", i.e., the source generating the request sequence σ, is all-powerful. That is, the adversary knows everything about the servers' strategy and can predict everything they will do.) The basic problem then is to understand how $\rho(u, w)$ is affected by the various parameters in our model, i.e., the metric space (M, d), and the choice of the cost function c.

This model can be further extended by allowing "excursions", i.e., where servers are allowed to go to arbitrary points in M in response to a request σ_t. In other words, we can replace (i) by :

(i') Some server, say S_i, is moved to some vertex v, with a resulting cost of $c(v) + d(S_k, v) + d(v, \sigma_t)$.

We note that a fair amount is known when option (ii) is not allowed, $w = 1$, and the cost function C depends on $d(S_i, \sigma_t)$ and not just σ_t. In other words, some server must always be moved to the current request, but the cost only depends on the distance moved (see [2] or [7] for a survey). It is shown in [5] that for this model, finite look-ahead does not help asymptotically. However, for the model of allowing excursions, finite values of w do make a difference.

By examining several special cases of the general problem, there is an example of the "windex phenomenon". That is, we have $\rho(0,4) = 3 = \rho(t,0)$ for all $t \geq 4$. In other words, if you can't see into the future at all, then it doesn't help to see more than four steps into the past. Does this phenomenon occur for all $w < \infty$? Is there an efficient algorithm for determining $\rho(u,w)$ for given large values of u and w? What happens for varying costs $c(o)$ and $c(1)$? On more general graphs? With $k > 1$ servers? These questions among many others (unfortunately) illustrate how incomplete our knowledge is here at this point in time.

References

[1] S. Ben–David, A. Borodin, R. Karp, G. Tardos and A. Wigderson, On the power of randomization in on-line algorithms, Proc. 22nd Symp. on the Theory of Algorithms, ACM, 1990, 379-386.

[2] Marek Chrobak and Lawrence L. Larmore, The server problem and on-line games, *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, **7**, pp. 74-94, 1992.

[3] F. R. K. Chung, R. L. Graham and M. E. Saks, Dynamic search in graphs, in Discrete Algorithms and Complexity, Academic Press, New York, 1987, 351–388.

[4] ____________, A dynamic location problem for graphs, Combinatorica 9 (1989), 111–131.

[5] Fan Chung and Ronald Graham, Dynamic location problems with limited look-ahead, *Theoretical Computer Science*, to appear.

[6] A. Fiat, Y. Rabini and Y. Ravid, Competitive k-server algorithms, *31st Annual Symposium on Foundation of computer Science*, *J. Comput. System Sci.* **48** (1994), 410-428.

[7] Amos Fiat and Gerhard J. Woeginger (Eds.), *Online Algorithms: State of the Art*, Springer-Verlag, 1998.

[8] E. Grove, The harmonic on-line k-server algorithm is competitive, *On-line Algorithms*, DIMACS SER. Discrete Math. Theoret. Comput. Sci., **7**, 1992, 65-75. Amer. Math. Soc., Providence, RI.

[9] M. S. Manasse, L. A. McGeoch and D. D. Sleator, Competitive algorithms for server problems, J. Algorithms 11 (1990), 208–230.

[10] M. E. Saks, A limit theorem for (min,+) matrix multiplication, *Math of OR*, **13** (1988), 606-618.

[11] D. Sleator and R. E. Tarjan, Amortized efficiency for list update and paging rules, Comm. ACM 28 (1985), 202–208.