Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 14, no. 1, pp. 149–163 (2010)

Resource relocation on asymmetric networks

D. Jacob Wildstrom

University of Louisville

Abstract

The necessary information to optimally serve sequential requests at the vertices of an undirected, unweighted graph with a single mobile resource is a known result of Chung, Graham, and Saks; however, generalizations of this concept to directed and weighted graphs present unforeseen and surprising changes in the necessary lookahead for strategic optimization. A pair of edges of unequal weights and opposite orientation can serve to simulate a communication or transportation connection with asymmetric costs, as may arise in a transportation network from prevailing winds or elevation changes, or in a communication network from aDSL or a similar technology.

This research explores the complications introduced by asymmetric connections within even very small networks. We consider the dynamic relocation problem on a two-vertex system and find that, even in this simplest possible asymmetric graph, the necessary lookahead for optimal relocation may be arbitrarily large. This investigation also gives rise to a linear-time algorithm to determine the optimizing real-time response to any request sequence which uniquely determines an optimal response.

Submitted:	Reviewed:	Revised:	Accepted:
January 2009	November 2009	November 2009	November 2009
	Final:	Published:	
	November 2009	January 2010	

Article type: Regular paper

Communicated by: G. Woeginger

E-mail address: dwildstr@erdos.math.louisville.edu (D. Jacob Wildstrom)

1 Introduction

The dynamic location problem can be viewed as an extension of traditional problems in resource-location such as the k-median problem and the uncapacitated and capacitated resource location problem. These problems have dealt with selecting the optimal location for one or more servers on a network in a configuration which minimizes the cost of servicing a set of vertices designated as clients with the nearest server. Research into this problem in integer optimization has focused on the algorithmic feasibility of determining optimal configurations, and the effectiveness of approximation strategies.

By way of contrast, *dynamic* optimization problems deal not with algorithmic efficacy or approximation strategies, but with whether optimal strategies can be determined at all based on a limited worldview. A question relating to real-time relocation of a unique resource can be phrased as a choice of vertices $s_1, \ldots, s_n$ to minimize

$$\sum_{i=1}^{n} d(s_{i-1}, s_i) + d(s_i, r_i),$$

given chosen vertices $s_0, r_1, \ldots, r_n$ of a graph. This expression serves to calculate the costs associated with a resource moving at each timestep and then serving a request from the new position, assuming that both movement and service-provision costs are given by distance. Taken as a whole, the minimization of this cost is a simple combinatorial optimization problem. The dynamic aspect of this optimization is incorporated via a supplementary question: can an optimal sequence of location vertices be determined if each s_i is determined solely as a function of s_{i-1}, r_i, r_{i+1}, …,r_{i+k}?

This question, which will henceforth be called the *dynamic location problem*, has significance both to algorithmic simplification and to realistic service-scheduling considerations. The former application results from a simplification of the strategy-space which must be traversed in determining entries of the optimal relocation sequence. This algorithmic simplification is particularly exploitable in median graphs, as proven by Chung, Graham, and Saks [3], and as presented in an algorithmic context by Knuth [5]. The realistic considerations raised by this question follow directly from the system which it models; the question of how efficient finite-lookahead optimization can be and how much lookahead is needed to achieve specific goals, is of importance to service providers establishing scheduling policies.

The original statement of the dynamic location problem defined a graph parameter called the *window index* or *windex* to describe the minimum necessary lookahead for optimal choice of s_i. Thus, we say that the windex of a graph G, denoted $\mathrm{WX}(G)$, is the minimum natural number k yielding a strategy function $f : (V(G) \cup \{\emptyset\})^{k+1} \to V(G)$ such that, for any initial state s_0 and $r_1, \ldots, r_n$, an optimal choice of $s_1, \ldots, s_n$ is determined by $s_{i+1} = f(s_i, r_{i+1}, r_{i+2}, \ldots, r_{i+k})$ for $0 \leq i < n$. If no such k exists, G has *infinite windex*, denoted $\mathrm{WX}(G) = \infty$.

The windex of undirected, unweighted graphs has been completely characterized by Chung, Graham and Saks [2, 3], making use of a topological construct of Hell [4].

One notable case is that of the windex-2 graphs, which are retracts of Q_n: this class of graphs was shown by Bandelt [1] to be identical to the median graphs.

2 Fundamentals

The conventional approach to dynamic location, as practiced on undirected graphs, is explicitly constructive: to demonstrate an upper bound on a graph's windex, a particular finite-lookahead strategy is shown to yield optimal results in all cases, and to bound a graph's windex from below, a particular pair of request sequences differing only in their final terms is shown to require different optimizing strategies from its initial response. These tools, while developed for undirected, unweighted graphs, are suitable for simple investigations on weighted graphs.

While the original formulation of the dynamic location problem concerns relocation on graphs, the most general system to be considered here is that of an arbitrary distance metric. To maintain consistency of terminology with previous work, we shall retain the term "vertices" to describe nodes, and continue to call the metric a "graph", but rather than describing connectivity between vertices by means of weighted or directed edges, distances between vertices will be considered as an arbitrary function $d : V \times V \to \mathbb{R} \cup \{\infty\}$. For practical purposes, we shall limit our consideration to graphs in which each vertex is reachable from every other vertex, so $d(u, v) < \infty$ for all vertices u and v in our graph.

A *request sequence* will be described by a concatenation of vertices as such: $s_0 r_1 r_2 r_3 \cdots r_n$, with $r_1 \cdots r_n$ abbreviated ρ. A *response sequence* $s_1 s_2 s_3 \cdots s_n$, abbreviated σ, is also a sequence of vertices. The *cost* of a given response to a sequence is given by

$$\mathrm{cost}(s_0\rho, \sigma) = \sum_{k=1}^{n} d(s_{k-1}, s_k) + d(s_k, r_k)$$

Then the *offline optimal-response cost* of a given request is denoted by

$$\mathrm{OPT}(s_0\rho) = \min_{\sigma \in V(G)^n} \mathrm{cost}(s_0\rho, \sigma)$$

One critical consideration in exploring the windex is how a particular response sequence's optimal responses depend on the desired end-state. We thus explore the concept of an *end-constrained optimum*, which determines the most efficient way of responding to a certain request sequence and thereafter moving to a new state.

$$\mathrm{OPT}(s_0\rho; s) = \min_{\sigma \in V(G)^n} (\mathrm{cost}(s_0\rho, \sigma) + d(s_n, s))$$

Notably, the cost-minimizing choice of σ for $\mathrm{OPT}(s_0\rho; s)$ may be significantly unlike the cost-minimizing response for $\mathrm{OPT}(s_0\rho)$. This distinction explains the need for long lookahead: if the choice of response, even in the short term, is dependent on long-term needs, knowledge of the long-term future is necessary for optimization.

We denote the *optimal prefix-set* of a request as the set of possible choices of s_1 which are part of a cost-minimizing response to the request:

$$\mathrm{pref}(s_0\rho) = \{s_1 : \mathrm{cost}(s_0\rho, s_1 s_2 \dots s_n) = \mathrm{OPT}(s_0\rho)\}$$

and similarly define an end-constrained optimal prefix-set:

$$\mathrm{pref}(s_0\rho; s) = \{s_1 : \mathrm{cost}(s_0\rho, s_1 s_2 \dots s_n) + d(s_n, s) = \mathrm{OPT}(s_0\rho; s)\}$$

The relationship between chosen request suffixes and the associated prefixes of optimal-cost responses to those prefixes is made relevant to the lookahead problem through a concept of *suffix-dependency.* A request sequence is said to be *suffix-dependent* if there is no prefix which is an optimal response to the sequence with an arbitrary extension. Symbolically, $s_0\rho$ is suffix-dependent if

$$\bigcap_{r_{n+1} \in V(G)} \mathrm{pref}(s_0\rho r_{n+1}) = \emptyset$$

Here we have two different notions of "how a sequence ends": suffix-dependency utilizes choice of r_{n+1}, whereas end-constraint represents activity after the end of a sequence with s_{n+1}. Both considerations are useful, and they can be related simply:

Lemma 1 *If $s_0\rho$ is suffix-dependent, then $\bigcap_{s \in V(G)} \mathrm{pref}(s_0\rho; s) = \emptyset$.*

Proof: Optimal responses to $s_0\rho r_{n+1}$ must take on some value of s_{n+1}, and must be least-cost responses subject to such a choice of s_{n+1}. Thus, for each r_{n+1}, there is at least one s_{n+1} such that $\mathrm{OPT}(s_0\rho r_{n+1}) = \mathrm{OPT}(s_0\rho; s_{n+1}) + d(s_{n+1}, r_{n+1})$; since any choice of σ serving for the optimization on the right side of this equation also serves on the left side, it follows that $\mathrm{pref}(s_0\rho; s_{n+1}) = \mathrm{pref}(s_0\rho r_{n+1})$. Thus, considering such an s_{n+1} for each $r_{n+1} \in V(G)$, and assembling them into a set S, it is clear that

$$\bigcap_{s_{n+1} \in V(G)} \mathrm{pref}(s_0\rho; s_{n+1}) \subseteq \bigcap_{s_{n+1} \in S} \mathrm{pref}(s_0\rho; s_{n+1}) = \bigcap_{r_{n+1} \in V(G)} \mathrm{pref}(s_0\rho r_{n+1}) = \emptyset$$

□

The converse of this statement is not necessarily true. In the cases we are considering in this paper, however, the converse will hold.

Corollary 1 *In a two-vertex graph, $s_0\rho$ is suffix-dependent if and only if the set $\bigcap_{s \in V(G)} \mathrm{pref}(s_0\rho; s) = \emptyset$.*

Proof: For a given $s_0\rho$, if we assemble S as in the previous lemma, it must have at least one element by construction. If $|S| > 1$, then $S = V(G)$ and the inclusion in the statement of the lemma is thus an equality between the sets $\bigcap_{s_{n+1}\in V(G)} \mathrm{pref}(s_0\rho; s_{n+1})$ and $\bigcap_{r_{n+1}\in V(G)} \mathrm{pref}(s_0\rho r_{n+1})$.

If, on the other hand, $|S| = 1$, then if we denote the vertices $S = \{u\}$ and $V(G) \setminus S = \{v\}$, then since u is an optimal choice for s_{n+1} regardless of the value of r_{n+1}, it follows that $d(u, r_{n+1}) \leq d(v, r_{n+1})$ for all values of r_{n+1}. Since u is closer to every vertex than v, u is trivially a better state than v regardless of circumstance and is specifically always a better choice of s_1, so that if $|S| = 1$, no sequence $s_0\rho$ is suffix-dependent and every $\bigcap_{s\in V(G)} \mathrm{pref}(s_0\rho; s) = \{u\}$. □

If $s_0\rho$ is not suffix-dependent, it is called *suffix-independent*, and any element of $\bigcap_{r_{n+1}\in V(G)} \mathrm{pref}(s_0\rho r_{n+1})$ is called an *unambiguously optimal prefix* for $s_0\rho$. If every request sequence of a certain length has an unambiguously optimal prefix, then it is easy to show that the windex of the graph does not exceed that length.

Proposition 1 *If $s_0\rho$ is suffix-independent for every request sequence $|\rho| = k$, then the windex of the graph G is no more than k.*

Proof: Given $|\rho| = k + \ell$, we demonstrate optimizability via induction on ℓ. The case $\ell = 1$ follows naturally from the definition of suffix-independence: we choose s to be an unambiguously optimal prefix for $s_0r_1 \ldots r_k$. Then, since $s \in \mathrm{pref}(s_0\rho)$, there is an optimal response σ to ρ such that $s_1 = s$, and since $\mathrm{OPT}(s_0\rho) = d(s_0, s_1) + d(s_1, r_1) + \mathrm{OPT}(s_1r_2 \ldots r_{k+1})$, choices of $s_2, \ldots, s_{k+1}$ are yielded by choosing optimal responses to the sequence $s_1r_2 \ldots r_{k+1}$.

For larger values of ℓ, let us consider the request sequences $\rho_1 = r_1 \ldots r_k$ and $\rho_2 = r_{k+2} \ldots r_{k+\ell}$. Let s be an unambiguously optimal prefix for $s_0r_1 \ldots r_k$, as before. Note that

$$\mathrm{OPT}(s_0\rho) = \mathrm{OPT}(s_0\rho_1; s_{k+1}) + d(s_{k+1}, r_{k+1}) + \mathrm{OPT}(s_{k+1}\rho_2)$$

for some s_{k+1}. Since $s \in \mathrm{pref}(s_0\rho_1)$ for every choice of s_{k+1} and since both sides of this equation are the cost of the same response, it follows that $s \in \mathrm{OPT}(s_0\rho)$. Then, assuming a response to $s_0\rho$ with $s_1 = s$, we see that

$$\mathrm{OPT}(s_0\rho) = d(s_0, s) + d(s, r_1) + \mathrm{OPT}(s\rho_2)$$

and by the inductive hypothesis, since $|\rho_2| = k + \ell - 1$, an optimal response to $s\rho_2$ can be created with lookahead k. □

Conversely, a finite windex precludes long suffix-dependent sequences, and existence of an unambiguously optimal prefix is unaffected by extension of a request sequence.

Proposition 2 *If graph G has windex k, then any request sequence $s_0r_1 \ldots r_\ell$ is suffix-independent for $\ell \geq k$.*

Proof: Let $f(s_i, r_{i+1}, \ldots, r_{i+k})$ be an optimal k-lookahead strategy on G; in particular, for some $r_{\ell+1}$, and let $s_1 = f(s_0, r_1, \ldots, r_k)$. Since f is an optimal strategy, $s_1 \in \text{pref}(s_0 r_1 r_2 \ldots r_\ell r_{\ell+1})$. Since s_1 does not depend on $r_{\ell+1}$, $s_1 \in \bigcap_{r_{\ell+1}} \text{pref}(s_0 r_1 r_2 \ldots r_\ell r_{\ell+1})$. □

The following statement, while true in general, will be here proven only for the two-vertex systems discussed in the next section.

Proposition 3 *In a two-vertex graph, if $s_0\rho$ is suffix-independent, so is $s_0\rho r$ for any $r \in V(G)$; likewise, so is $s_0\rho\rho'$ for any sequence of vertices ρ'.*

Proof:
From Corollary 1, $s_0\rho$ is suffix-independent if and only if $\bigcap_{s\in V(G)} \text{pref}(s_0\rho; s) \neq \emptyset$; likewise, $s_0\rho r$ is suffix-independent if and only if $\bigcap_{s\in V(G)} \text{pref}(s_0\rho r; s) \neq \emptyset$. Since there is a choice of s_{n+1} such that $\text{OPT}(s_0\rho r; s) = \text{OPT}(s_0\rho; s_{n+1}) + d(s_{n+1}, r) + d(s_{n+1}, s)$, using the same response sequence for both optimizations, it follows that $\text{pref}(s_0\rho; s_{n+1}) = \text{pref}(s_0\rho r; s)$; collecting the s_{n+1} associated with each choice of S into a single set as in Lemma 1, we see that

$$\emptyset \neq \bigcap_{s_{n+1}\in V(G)} \text{pref}(s_0\rho; s_{n+1}) \subseteq \bigcap_{s_{n+1}\in S} \text{pref}(s_0\rho; s_{n+1}) = \bigcap_{r_{n+1}\in V(G)} \text{pref}(s_0\rho r; s)$$

Thus $\bigcap_{r_{n+1}\in V(G)} \text{pref}(s_0\rho r; s) \neq \emptyset$, so $s_0\rho r$ is suffix-independent. □

In light of the connection between windex and suffix-dependency, we may rephrase the definition of the windex of a graph in such terms:

$$\text{WX}(G) = \max\{|\sigma| : \sigma \text{ is suffix-dependent}\}$$

If G admits arbitrarily large suffix-dependent σ, then $\text{WX}(G) = \infty$. We may now use these tools to examine specific directed graphs.

3 Asymmetric paths

While even unweighted directed graphs such as cycles and connected tournaments indicate complexity beyond the traditional undirected case, they are highly inapplicable results, as unidirectional connections are rare in real-world networks. A common aspect of networks, however, is *asymmetric connection*; in a ground-transportation network, traffic and travel times may be unequal in different directions, while for air-transport, prevailing winds may suggest that one direction of travel has cost significantly lower than another, and in communication networks, transmission technologies such as aDSL lines have asymmetric data transfer rates. Thus, the advantage in allowing arbitrary distance metrics lies not in the ability to consider unidirectional connections, but in the use of nonsymmetric distance functions to model asymmetric connections.

We shall begin our investigation into asymmetric paths with the simplest possible model: a two-vertex graph, with possible fees for remaining in place

	11	22	12	21
111	$4a$	$b+2c+d$	$2a+b+c$	$a+b+2c$
112	$3a+b$	$b+c+2d$	$2a+b+d$	$2b+2c$
121	$3a+b$	$b+c+2d$	$a+2b+c$	$a+b+c+d$
122	$2a+2b$	$b+3d$	$a+2b+d$	$2b+c+d$

Table 1: Costs of responses to two-request strings on a two-vertex graph

or providing local service. In general, we shall consider a graph whose vertices are labeled 1 and 2, with distances $d(1,1) = a$, $d(1,2) = b$, $d(2,1) = c$, and $d(2,2) = d$. This may be denoted by the distance matrix $D = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$.

On a two-vertex graph, we shall find that suffix-dependent request sequences are in fact associated with pairs of complementary response sequences; that is to say, request sequences σ and σ' such that if $s_i = 1$ then $s'_i = 2$, and vice versa.

Proposition 4 *If $\sigma^r = s_1^r s_2^r \cdots s_n^r$ is a cost-minimizing response to $s_0\rho r$ for each $r \in V(G)$, and $s_0\rho$ is suffix-dependent, then for no i are all of the states s_i^r identical.*

Proof: Suppose there is such an i; then we may determine a single s_i such that $s_i = s_i^r$ for all r; let $\rho_1 = r_1 r_2 \cdots r_{i-1}$ and $\rho_2 = r_{i+1} r_{i+2} \cdots r_n$. Then, since a minimal-cost response to $s_0\rho r$ has ith state s_i, it is the case that $\mathrm{OPT}(s_0\rho r) = \mathrm{OPT}(s_0\rho_1; s_i) + d(s_i, r_i) + \mathrm{OPT}(s_i\rho_2)$, so since an optimal choice of s_1 is yielded by the first term in this sum, it follows that $\mathrm{pref}(s_0\rho r) \supseteq \mathrm{pref}(s_0\rho_1; s_i)$. Since r does not appear on the right side of this equality, $\bigcap_{r \in V(G)} \mathrm{pref}(s_0\rho r) \supseteq \mathrm{pref}(s_0\rho_1; s_i) \neq \emptyset$, violating the conditions of suffix-dependency. $\square$

Corollary 2 *If G has exactly two vertices 1 and 2, and $s_0\rho$ is suffix-dependent, then $s_0\rho 1$ and $s_0\rho 2$ have unique complementary response sequences.*

3.1 Two-request sequence responses

It is simple to exhaustively consider all possible responses to request sequences of length two on a two-vertex-graph. Without loss of generality we may consider the case $s_0 = 1$ to start with, leaving only four possible requests and four possible responses, whose associated costs are shown in Table 1. In addition, there are only two simple suffix-dependency possibilities: either 11 or 12 may be suffix-dependent; if neither of them are, then the windex of the graph in question is 1 or 0. By Corollary 2, we may establish that optimal response sequence pairs must be one of two conjugate pairs in one of two orders, so each of these cases comprises four possible subcases, making a total of eight potential results. By exhaustive consideration of each case, we can show that of the eight possible suffix-dependent sequences and their optimal responses, two are associated with a certain inequality, two are associated with an opposite inequality, and four can not occur. The four possible cases, labeled Case I, III, V, and VII, are shown here with the requisite inequalities; the other four appear in an appendix.

Case I: 11 is suffix-dependent: 111 and 112 respectively have unique optimal response sequences 11 and 22. Thus the unique minima of the first and second rows of Table 1 are in the first and second columns respectively. In particular, $4a < b + 2c + d$ and $b + c + 2d < 3a + b$, from which it follows that $a + d < b + c$. This case will be examined in more detail.

Case III: 11 is suffix-dependent: 111 and 112 respectively have unique optimal response sequences 12 and 21. The unique minima of the first and second rows of Table 1 must be in the third and fourth columns respectively. Since $2a + b + c < a + b + 2c$ and $2b + 2c < 2a + b + d$, it can be established that $a + d > b + c$.

Case V: 12 is suffix-dependent: 121 and 122 respectively have unique optimal response sequences 11 and 22. The unique minima of the third and fourth rows of Table 1 are in the first and second columns respectively. In particular, $3a + b < a + 2b + c$ and $b + 3d < 2b + c + d$, from which it follows that $a + d < b + c$. This case is analyzed alongside Case I.

Case VII: 12 is suffix-dependent: 121 and 122 respectively have unique optimal response sequences 12 and 21. The unique minima of the third and fourth rows of Table 1 must be in the third and fourth columns respectively, so $a+2b+c < a+b+c+d$ implies that $b < d$; $2b+c+d < a+2b+d$ implies that $c < a$, and thus $b + c < a + d$.

This casewise analysis allows for a natural classification of two-vertex systems: each possible case is associated with the condition $a + d < b + c$ or $a + d > b + c$. The condition $a + d = b + c$ is not associated with any of these cases and thus leads to a windex of 1 or 0; the other possibilities, each associated with two of the above cases, are amenable to individual consideration.

3.2 Two-vertex graphs with high parking fees

Since a and d represent the cost of remaining at a vertex, or providing service to the resource's current vertex, they may be referred to as *parking fees*, whereas b and c represent movement or communication among vertices and therefore might be called *transit fees*. The $a + d > b + c$ case is thus one in which parking fees are large in comparison to transit fees; we thus call this inequality the *high-parking-fee* condition. From the casewise analysis in Subsection 3.1, we find four potential suffix-dependent sequences of length 3: 111, 112, 121, and 122. As before, we are without loss of generality considering only cases where $s_0 = 1$. We shall see, via complete enumeration, that none of these possible sequences are in fact suffix-dependent. We are aided in our endeavor by the results in Subsection 3.1 indicating that, of the four possible 2-request response prefixes, only 12 and 21 are utilized. The limitation to these responses is sensible in consideration of the high-parking-fee criterion: when the choice $s_i = s_{i-1}$ incurs high costs, it appears useful to relocate.

	121	212	122	211
1111	$3a + b + 2c*$	$a + 2b + 3c$	$2a + b + 3c + d$	$3a + b + 2c*$
1112	$2a + 2b + 2c*$	$a + 2b + 2c + d$	$2a + b + c + 2d$	$2a + 2b + 2c*$
1121	$3a + b + c + d\dagger$	$3b + 3c$	$2a + b + c + 2d\dagger$	$2a + 2b + 2c$
1122	$2a + 2b + c + d\dagger$	$3b + 2c + d$	$2a + b + 3d\dagger$	$a + 3b + 2c$
1211	$2a + 2b + 2c$	$a + 2b + 2c + d*$	$a + 2b + 2c + d*$	$3a + b + c + d$
1212	$a + 3b + 2c$	$a + 2b + c + 2d*$	$a + 2b + c + 2d*$	$2a + 2b + c + d$
1221	$2a + 2b + c + d*$	$3b + 2c + d$	$a + 2b + c + 2d$	$2a + 2b + c + d*$
1222	$a + 3b + c + d*$	$3b + c + 2d$	$a + 2b + 3d$	$a + 3b + c + d*$

Table 2: Costs of responses to three-request strings on a two-vertex graph

Theorem 1 *If G is a two-vertex graph with $a + d > b + c$, then* $\mathrm{WX}(G) \leq 3$.

Proof: This argument proceeds by casewise enumeration as in the two-vertex case. There are 8 possible 3-request sequences subject to $s_0 = 1$ and 2 pairs of complementary responses. For a 2-request sequence to be suffix-dependent, its two extensions must be associated with complementary responses; given four request sequences of length 2, and four assignments of complementary responses to each, there are 16 possible suffix-dependencies that can occur. All possible costs considered in this argument are shown in Table 2. If identical expressions appear in separate cells of a single row, clearly neither of them is associated with a unique optimizing response, since two separate responses yield the same cost. Cells which are thereby forbidden from representing unique optimal responses are marked with an asterisk in Table 2. In addition, if we make use of the given condition that $a + d > b + c$, we furthermore find several cells guaranteed not to have the minimum value in their rows, denoted with a dagger. Since from each possible pair of extensions of a two-request sequence we have eliminated two non-complementary responses as possible unique optima, it is impossible to choose two request extensions with complementary optimal request sequences; thus, none of the two-request sequences are suffix-dependent. □

3.3 Two-vertex graphs with low parking fees

In contrast to the aforementioned high-parking-fee case is the case in which parking fees are lower than transit fees; that is, when $a + d < b + c$. Much as the optimal responses in the high-parking-fee case are marked by perpetual relocation, optimal responses in the low-parking-fee case are marked by staticity, and, in fact, the conjugate optimal responses associated with a suffix-dependent request are each, after the first step, completely motionless.

Lemma 2 *If G is the two-vertex graph with distances as given above with suffix-dependent request sequence $s_0\rho$ and $a + d < b + c$, then $s_0\rho 1$ has unique optimal response 1^{n+1} and $s_0\rho 2$ has unique optimal response 2^{n+1}.*

	1^n1	2^n2	1^n2	2^n1
$s_0\rho 1$	$C_1 + 2a$	$C_2 + c + d$	$C_1 + b + c$	$C_2 + a + c$
$s_0\rho 2$	$C_1 + a + b$	$C_2 + 2d$	$C_1 + b + d$	$C_2 + b + c$

Table 3: Costs of responses to arbitrary-length sequences

Proof: We proceed by induction on $|\rho| = n$. The base case $n = 1$ is derived from the casewise analysis in Subsection 3.1; since $a + d < b + c$, the only possible cases are I and V, and so any suffix-dependent sequence $s_0\rho$ of length 2 has unique optimal responses of 1^2 and 2^2 to $s_0\rho 1$ and $s_0\rho 2$ respectively.

For the inductive step, let ρ' be ρ with its last request truncated: since $s_0\rho'$ is suffix-dependent, we know by the inductive hypothesis that the optimal responses to $s_0\rho' 1$ and $s_0\rho' 2$, and thus the end-constrained optimal responses to $s_0\rho$ with end-constraints of 1 and 2, are 1^n and 2^n respectively. Let the costs of these responses to $s_0\rho$ be C_1 and C_2 respectively. Then there are 4 plausible responses to $s_0\rho 1$ and $s_0\rho 2$: these responses are 1^n1, 2^n2, 1^n2, and 2^n1. The costs of these responses are summarized in Table 3. Since $s_0\rho$ is suffix-dependent, we know that the optimal responses to $s_0\rho 1$ and $s_0\rho 2$ are unique and complementary. We may thus consider each of the four cases, seeing that only the case given in the statement of this lemma is possible:

Case I: $s_0\rho 1$ and $s_0\rho 2$ respectively have unique optimal responses 1^n1 and 2^n2. No contradiction arises from this supposition, and it matches our expectation from the inductive hypothesis.

Case II: $s_0\rho 1$ and $s_0\rho 2$ respectively have unique optimal responses 2^n2 and 1^n1. Since 2^n2 is a better response to $s_0\rho 1$ than 2^n1, it follows that $C_2+c+d < C_2+a+c$; likewise, since 1^n1 is a better response to $s_0\rho 2$ than 1^n2, it follows that $C_1+a+b < C_1+b+d$. These two inequalities are contradictory, as they are respectively equivalent to $d < a$ and $a < d$.

Case III: $s_0\rho 1$ and $s_0\rho 2$ respectively have unique optimal responses 1^n2 and 2^n1. Since 1^n2 is a better response to $s_0\rho 1$ that 1^n1, it follows that $C_2 + b + c < C_2 + 2a$; likewise, since 2^n1 is a better response to $s_0\rho 2$ than 2^n2, it follows that $C_2 + b + c < C_2 + 2d$. Adding and simplifying these inequalities yields $b + c < a + d$, which is contrary to the low-parking-fee condition.

Case IV: $s_0\rho 1$ and $s_0\rho 2$ respectively have unique optimal responses 2^n1 and 1^n2. We may derive from this possibility the opposite inequalities as in Case II; they are still contradictory.

Thus, the only possibility if $s_0\rho$ is suffix-dependent in a low-parking-fee graph is that $s_0\rho 1$ and $s_0\rho 2$ have respective optimal responses 1^{n+1} and 2^{n+1}. □

Theorem 2 *If G is the two-vertex graph with distances as given above with $a + d < b + c$ and the request sequence $s_0\rho$ consists of k occurrences of 1 and ℓ occurrences of 2, then $s_0\rho$ is suffix-dependent iff every prefix of $s_0\rho$ is suffix-dependent and $2d - a - b < (c + d - 2a)k + (2d - a - b)\ell < c + d - 2a$.*

Proof: Given $s_0\rho$ composed of the given individual requests, we may assign values to the costs C_1 and C_2 defined in Lemma 2. Providing service to k

requests at vertex 1 and ℓ at vertex 2 has associated costs of $(2k+\ell)a+\ell b$ and $kc+(k+2\ell)d$ when served from vertex 1 and 2 respectively. This simple calculation, however, neglects the necessity that, of the $k+\ell$ vertices in the sequence $s_0\rho$, one is not a request, but s_0, which has different associated costs than those for remote service. If $s_0=1$, then $C_1=(2k+\ell-2)a+\ell b$ and $C_2=b+(k-1)c+(k+2\ell-2)d$, while if $s_0=2$, then $C_1=(2k+\ell-2)a+(\ell-1)b+c$ and $C_2=kc+(k+2\ell-2)d$. Fortunately, the difference between these two cases is identical for both costs, so it has no effect on their relative value:

$$C_1 = 2ak+(a+b)\ell-2a-b+f(s_0)$$
$$C_2 = (c+d)k+2d\ell-2d-c+f(s_0)$$

where $f(1)=b$ and $f(2)=c$.

By Proposition 3, it is clearly necessary that $s_0\rho$'s prefixes be suffix-dependent in order for $s_0\rho$ to be suffix-dependent. By the inductive argument in Lemma 2, the necessary and sufficient conditions for such a sequence to be itself suffix-dependent is that $C_1+a<C_2+c$ and $C_2+d<C_1+b$. Thus, $s_0\rho$ is suffix-dependent iff its prefixes are suffix-dependent and

$$\begin{aligned}[2ak+(a+b)\ell-2a-b]+a+d &< [(c+d)k+2d\ell-2d-c]+c+d \\ &< [2ak+(a+b)\ell-2a-b]+b+c\end{aligned}$$

which simplifies to

$$2d-a-b<(c+d-2a)k+(2d-a-b)\ell<c+d-2a$$

□

This necessary and sufficient criterion for suffix-dependence in two-vertex systems leads to an easily-comprehended visual schematic for constructing suffix-dependent sequences in particular two-vertex systems. In this schematic, we consider k and ℓ as coordinates in the lattice $\mathbb{Z}^2$, and a sequence as a walk upwards and right on this lattice from the point $(0,0)$, with horizontal and vertical steps representing respective concatenations of 1 and 2 to the request sequence. Then suffix-dependence criteria can be presented entirely geometrically. Our base case guarantees that either walk to $(1,1)$ is suffix-dependent; thereafter, based on Theorem 2, our walk represents a suffix-dependent sequence if and only if it lies in the region between the lines $(c+d-2a)k+(2d-a-b)(\ell-1)=0$ and $(c+d-2a)(k-1)+(2d-a-b)\ell=0$.

As an example of this geometrical representation, let us consider the two-vertex system with distance matrix $D=\begin{bmatrix}1 & 13\\ 6 & 2\end{bmatrix}$. The boundaries of the suffix-dependent region are thus the lines $6k-10\ell=-10$ and $6k-10\ell=6$. These boundaries, and a lattice walk within it, are shown in Figure 1. For instance, in this case, we see that there is a pair of suffix-dependent request sequences with 7 requests:

$$\begin{cases}12121121\\ 21121121\end{cases}$$

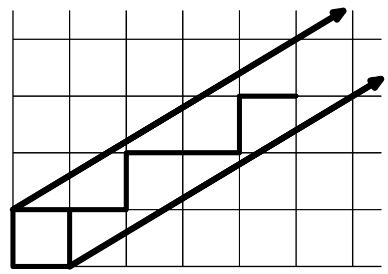

Figure 1: Coordinate-geometry representation of a suffix-dependent request on a two-vertex system

Furthermore, since no longer walk in the region shown is possible, these suffix-dependent request sequences are maximal, and the given two-vertex system thus has windex 8. We can generalize these specific results to describe the number and length of suffix-dependent requests for any two-vertex system.

Proposition 5 *For G a low-parking-fee two-vertex system, there are either exactly zero or two suffix-dependent request sequences of length n for every $n \geq 1$; furthermore, the two suffix-dependent request sequences differ only in the choice of s_0 and r_1.*

Proof: The base cases developed in Subsection 3.1 guarantee this is true for $n = 1$, which admits the request sequences 12 and 21. For $n > 1$, the geometric representation of a sequence of length n consists of a walk to $(1, 1)$ followed by an $(n-1)$-step walk within the region bounded by the lines $(c+d-2a)k+(2d-a-b)(\ell-1) = 0$ and $(c+d-2a)(k-1)+(2d-a-b)\ell = 0$. It is sufficient to show that there is no more than one such walk, which is identical to establishing that at no point during a walk within this region are both horizontal and vertical steps feasible. To illustrate this, suppose both possible steps from a point (k, ℓ) lie within our region: we may extend our path to $(k + 1, \ell)$ or $(k, \ell + 1)$. Thus, both

$$2d - a - b < (c + d - 2a)(k + 1) + (2d - a - b)\ell < c + d - 2a$$

and

$$2d - a - b < (c + d - 2a)k + (2d - a - b)(\ell + 1) < c + d - 2a$$

However, the first assertion is true only if $(c + d - 2a)k + (2d - a - b)\ell < 0$, while the latter is true only if latter is $(c + d - 2a)k + (2d - a - b)\ell > 0$; thus, since both cannot be satisfied for any (k, ℓ), each point within the region has at most one possible successor on a walk within the region; note that a point has *no* successors if $(c + d - 2a)k + (2d - a - b)\ell = 0$. □

Proposition 6 *A low-parking-fee two-vertex system G has a suffix-dependent sequence of length $\frac{(b+c-a-d)}{\gcd(c+d-2a,a+b-2d)} - 1$ and no longer suffix-dependent sequence; thus $\mathrm{WX}(G) = \frac{(b+c-a-d)}{\gcd(c+d-2a,a+b-2d)}$.*

Proof: Let $p = \frac{c+d-2a}{\gcd(c+d-2a,a+b-2d)}$ and $q = \frac{a+b-2d}{\gcd(c+d-2a,a+b-2d)}$. Then the boundaries of the region of suffix-dependency are $-p < pk - q\ell < q$. There are thus $p + q - 1$ possible distinct values for $pk - q\ell$ in the region, and since $\gcd(p, q) = 1$, all of these values are attainable; now it is sufficient to show that no path from $(1, 1)$ within the region can attain the same value of $pk - q\ell$ twice. Suppose (k_1, ℓ_1) and (k_2, ℓ_2) are two distinct points on a path in the region of suffix-independence such that $pk_1 - q\ell_1 = pk_2 - q\ell_2$, with (k_1, ℓ_1) lexicographically less than (k_2, ℓ_2) and (k_2, ℓ_2) as small as possible. Then $p(k_2 - k_1) = q(\ell_2 - \ell_1)$. Since p and q are relatively prime, $k_2 - k_1$ and $\ell_2 - \ell_1$ must be divisible by q and p respectively, and since $(k_1, \ell_1) < (k_2, \ell_2)$, it follows that $k_2 - k_1 \geq q$ and $\ell_2 - \ell_1 \geq q$. Thus, there are at least $p + q$ steps between (k_1, ℓ_1) and (k_2, ℓ_2). If any of these steps (k, ℓ) are such that $pk - q\ell = 0$, it is clear from the argument in the prior proof that (k, ℓ) has no successor in the region; likewise, if $pk - q\ell = pk_1 - q\ell_1$, it would contradict our selection of (k_2, ℓ_2) as the earliest residue repetition. Thus, the $p + q - 1$ intermediary steps between (k_1, ℓ_1) and (k_2, ℓ_2) can only take on $p + q - 3$ distinct values of $pk - q\ell$; since two of them will have the same value, this contradicts our selection of (k_2, ℓ_2) as minimal. Thus, the same residue cannot occur twice in a walk from $(1, 1)$ within this region, and since there are only $p + q - 1$ distinct residues in this region, a walk from $(1, 1)$ within the region can only take $p + q - 2$ steps, which together with the sequence prefix 12 or 21, creates a sequence of no more than $p + q - 1$ requests.

To show that a sequence of $p + q - 1$ requests can in fact be constructed, note from the previous proof that the only non-extensible suffix-dependent request sequences are those for which $pk - q\ell = 0$. The minimal nontrivial solution to this equation is $(k, \ell) = (q, p)$, so it is necessarily possible to take at least $p + q - 2$ steps from $(1, 1)$ within the region of suffix-dependence. □

This visual schematic also presents a simple algorithm for determining optimal relocation instructions based on limited information: the lattice walk associated with the sequence $s_i r_{i+1} r_{i+2} \ldots r_{i+k}$ either has all steps beyond the first within the suffix-dependent region, in which case optimal response to this request cannot be determined, or some step is outside the suffix-dependent region, in which case the correct choice is $s_{i+1} = 1$ if the first step outside the suffix-dependent region is to the left of the suffix-dependent region, and $s_{i+1} = 2$ if the first step outside is above the suffix-dependent region.

One peculiar consequence of this analysis of two-vertex systems is that all of these results developed in a model in which distances must be integral to will also hold in a model in which distances can be any real number. If the distances are rational, this extension is not notable inasmuch as rational distances could be scaled appropriately to become integers, but an unusual effect occurs if the ratio $\frac{c+d-2a}{a+b-2d}$ is irrational. In such a case, the boundaries of the suffix-dependency region have irrational slope, and the walk within the region never reaches a point where termination becomes necessary, since the only integral solution to $(c + d - 2a)k = (a + b - 2d)\ell$ is the point $(0, 0)$. For example, if $D = \begin{bmatrix} 0 & 1 \\ \pi & 0 \end{bmatrix}$, we can produce the arbitrarily long aperiodic sequence:

$$1222212221222122212221222122212221222212221222 1\ldots$$

By our known boundaries on the region of suffix-dependence, the number of occurrences of 2 in this sequence must be bounded between $\pi(k-1)$ and πk after the appearance of k occurrences of 1. Thus, the number of requests at 2 between the kth and $(k+1)$th requests at 1 is $\lceil \pi k \rceil - \lceil \pi(k-1) \rceil$, which is an aperiodic sequence of clusters of length 3 and 4. The patterns in this sequence are termwise differences in what is known as the $\lfloor n\alpha \rfloor$-sequence, described in significant detail by Niven [7, 6].

In summary, we find that when $\frac{c+d-2a}{a+b-2d} = \frac{p}{q} \in \mathbb{Q}$ for positive q and $(p,q) = 1$, the windex of the two-vertex system described by these distances is $p+q$. If $\frac{c+d-2a}{a+b-2d} \notin \mathbb{Q}$, then the windex of this two-vertex system is infinite.

Appendix: Impossible situations on length-2 requests

Case II: 11 is suffix-dependent: 111 and 112 respectively have unique optimal response sequences 22 and 11. The unique minima of the first and second rows of Table 1 must be in the second and first columns respectively. Thus $b+2c+d < 2a+b+c$ and $b+2c+d < a+b+2c$. Adding and simplifying these inequalities, $b+c+2d < 3a+b$, contradicting the presumption that $3a+b$ is the minimum value in the second row. This case will therefore never occur.

Case IV: 11 is suffix-dependent: 111 and 112 respectively have unique optimal response sequences 21 and 12. The unique minima of the first and second rows of Table 1 must be in the fourth and third columns respectively. If we add the resultant inequalities $2a+b+d < b+c+2d$ and $a+b+2c < 2a+b+c$, it follows that $a < d$. However, since $2a+b+d < 3a+b$ simplifies to $d < a$, this case is clearly impossible.

Case VI: 12 is suffix-dependent: 121 and 122 respectively have unique optimal response sequences 22 and 11. The unique minima of the third and fourth rows of Table 1 must be in the second and first columns respectively, so $b+c+2d < a+b+c+d$ and $2a+2b < a+2b+d$; adding these results in a contradiction, so this case cannot occur.

Case VIII: 12 is suffix-dependent: 121 and 122 respectively have unique optimal response sequences 12 and 21. The unique minima of the third and fourth rows of Table 1 must be in the fourth and third columns respectively, so both $a+b+c+d < b+c+2d$ and $a+2b+d < 2a+2b$, which add together to yield a contradiction, so this case too is impossible.

References

[1] H.-J. Bandelt. Retracts of hypercubes. *J. Graph Theory*, 8(4):501–510, 1984.

[2] F. R. K. Chung, R. L. Graham, and M. E. Saks. Dynamic search in graphs. In *Discrete algorithms and complexity (Kyoto, 1986)*, volume 15 of *Perspect. Comput.*, pages 351–387. Academic Press, Boston, MA, 1987.

[3] F. R. K. Chung, R. L. Graham, and M. E. Saks. A dynamic location problem for graphs. *Combinatorica*, 9(2):111–131, 1989.

[4] P. Hell. Graph retractions. In *Colloquio Internazionale sulle Teorie Combinatorie (Roma, 1973), Tomo II*, pages 263–268. Atti dei Convegni Lincei, No. 17. Accad. Naz. Lincei, Rome, 1976.

[5] D. Knuth. *Pre-Fascicle 0b*, volume 4 of *The Art of Computer Programming*. Addison-Wesley, in preparation.

[6] I. Niven. *Irrational numbers*. The Carus Mathematical Monographs, No. 11. The Mathematical Association of America. Distributed by John Wiley and Sons, Inc., New York, N.Y., 1956.

[7] I. Niven. *Diophantine approximations*. The Ninth Annual Series of Earle Raymond Hedrick Lectures of The Mathematical Association of America. Interscience Tracts in Pure and Applied Mathematics No. 14. Interscience Publishers, a division of John Wiley & Sons, New York- London, 1963.