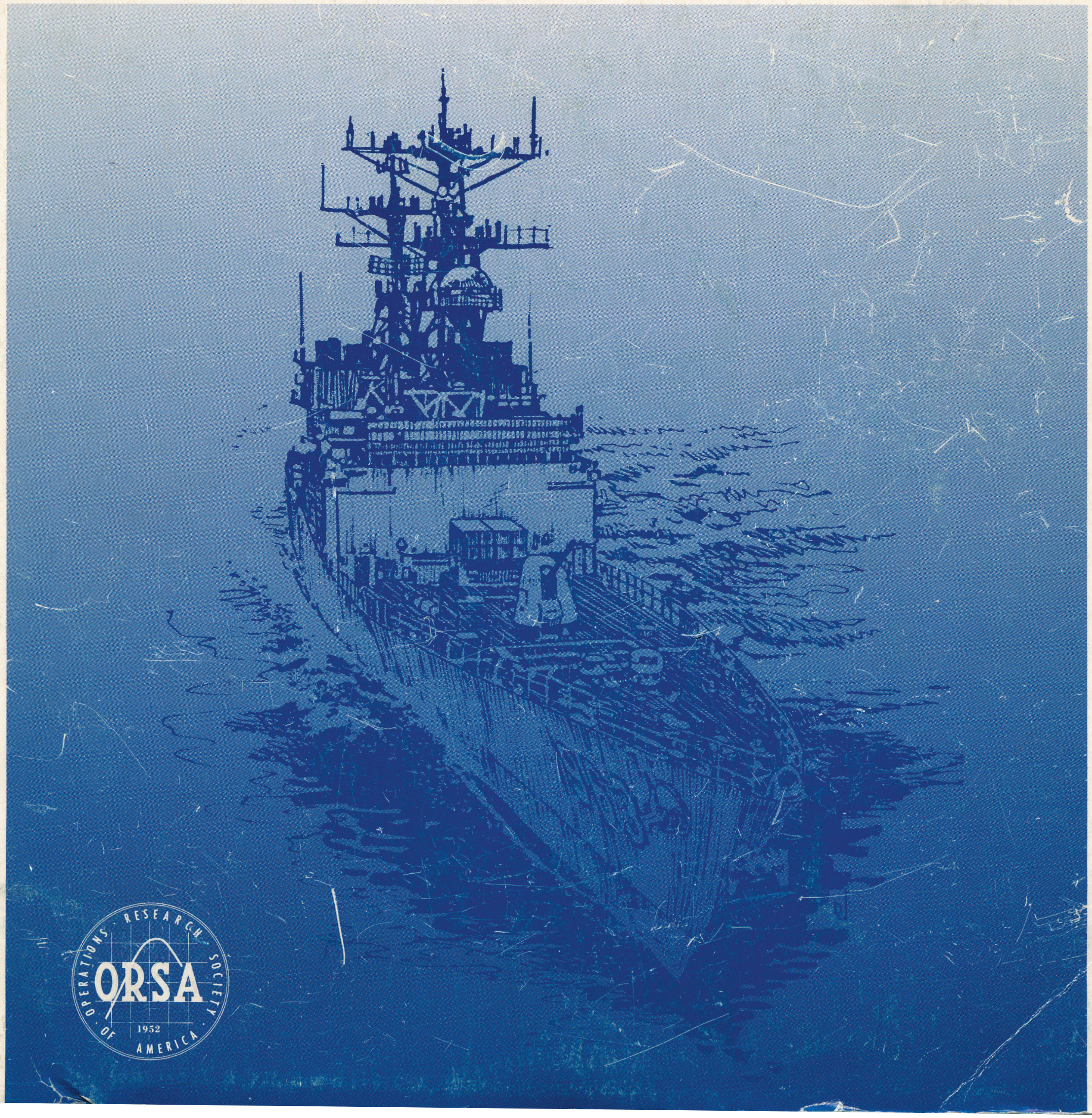
March–April 1990

OPERATIONS RESEARCH

Volume 38 Number 2



ARTICLES

GRAPHS, MANEUVERS AND TURNPIKES

ARTHUR T. BENJAMIN

Harvey Mudd College, Claremont, California

(Received May 1988; revision received December 1988; accepted January 1989)

We address the problem of moving a collection of objects from one subset of $\mathbf{Z}^m$ to another at minimum cost. We show that under fairly natural *rules for movement* assumptions, if the origin and destination are far enough apart, then a near optimal solution with special structure exists: Our trajectory from the origin to the destination accrues almost all of its cost repeating at most m different patterns of movement. Directions for related research are identified.

Consider the problem of maneuvering a collection of objects from one configuration to another at minimum cost, subject to various rules for movement. This *optimal movement of pieces* scenario is suggestive of applications in industrial robotics, military logistics, transportation science and (within a state-space setting) economic development planning. In these contexts, it is easy to see how, in some configurations, the pieces might "get in each other's way," thus blocking rapid progress toward the destination, while in other configurations, the pieces' relative positions might be mutually supportive in a way permitting exceptionally rapid further progress (*leapfrogging*). Although the environment through which the movement occurs is unlikely to be strictly homogeneous, some sort of local homogeneity may well be a good approximation, and the homogeneous case seems a suitably idealized starting point for research into such problems. We will deal with the simplest discrete homogeneous environment, namely the integer-point lattice $\mathbf{Z}^m$ in $\mathbf{R}^m$. This setting, it turns out, is already rich enough in structure to yield interesting questions, results and suggestive concepts.

In this section, we consider a series of attractive special cases arising from *jumping* problems and *sliding* problems. In every instance, the optimal trajectories have exhibited a special repetitive structure. The desire to explain and generalize this common feature provoked the investigations in the sections that follow.

The first example we consider is a game that resembles Chinese checkers. This solitaire puzzle is played with a finite set of indistinguishable pieces, using $\mathbf{Z}^2$ as our game board. At each move, exactly one piece is displaced. Suppose that a piece is situated at the point $x \in \mathbf{Z}^2$, and let e_i denote the ith unit vector of $\mathbf{Z}^2$. If $x + e_i$ is unoccupied, the piece can *shift* there; similarly for $x - e_i$. If $x + e_i$ is occupied, but $x + 2e_i$ is not, then the piece can *hop* over the occupant of $x + e_i$ to arrive at $x + 2e_i$, where it may either remain or hop over another adjacent piece, etc. (Similarly for a hop over $x - e_i$ to $x - 2e_i$.) A *move* consists either of a shift or a *jump* (a sequence of one or more hops by a single piece). Our objective is to transfer, in the minimum number of moves, the pieces from some configuration near the origin (0, 0) to a specified destination (d, d) where $d > 0$ is large.

While the above problem with more than four pieces is not fully resolved (see below), several related problems have known solutions which led to our more general results. For example, in Belur and Goldman (1985), the above problem with three pieces was solved. Here, the prescribed origin configuration was the "lower triangle" situated at the points (0, 1), (0, 0) and (1, 0). Our destination configuration is the "upper triangle" situated at the points $(d-1, d)$, (d, d), and $(d, d-1)$ for some prescribed positive d.

The solution is portrayed in Figure 1, in which the notation $X \xrightarrow{\mu} Y$ denotes using μ moves to reach configuration Y from configuration X. One point in the configuration is labeled with its position in $\mathbf{Z}^2$ and the positions of the remaining pieces, thereby, are automatically determined. The second and fifth configurations are merely translates of each other (in the direction (1, 1)), and the same sequence of three moves

Subject classifications: Military, logistics: optimal maneuvering. Networks/graphs: optimal maneuvering by graph theory approach. Transportation, mode/route choice: optimal maneuvering by turnpike theorems.

0030-364X/90/3802-0202 $01.25

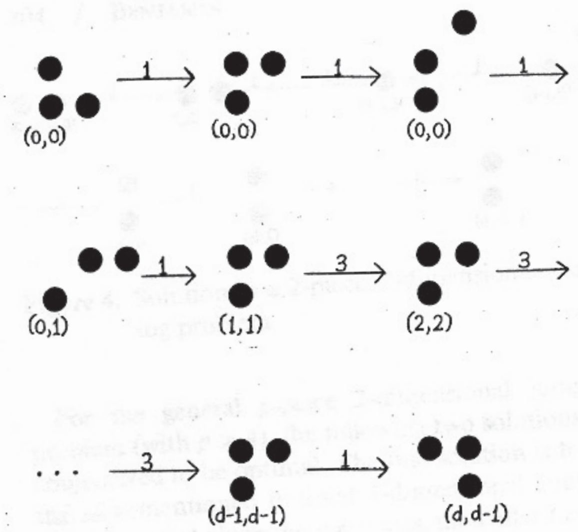


Figure 1. Solution to the 3-piece, 2-dimensional jumping problem.

is used to reach the subsequent configurations. The sequence requires $3d - 1$ moves and this was shown to be minimum.

The proof of the above minimality result (not presented here) involved *projecting* down to a simple 1-dimensional problem, similar to one analyzed by Castells and Goldman (1983). In the latter problem, we begin with $p \geq 3$ pieces on $\mathbf{Z}^1$ situated at points 0, 1, 2, ..., $p - 1$; using the aforementioned rules for jumping (restricted to one dimension), the objective is to move these pieces to the points $d, d + 1, d + 2, \ldots, d + p - 1$ in as few moves as possible. The solution is presented in Figure 2. (Here we are assuming $d \geq 2p - 1$ and that $d + p$ is odd. The solution when $d + p$ is even is similar.)

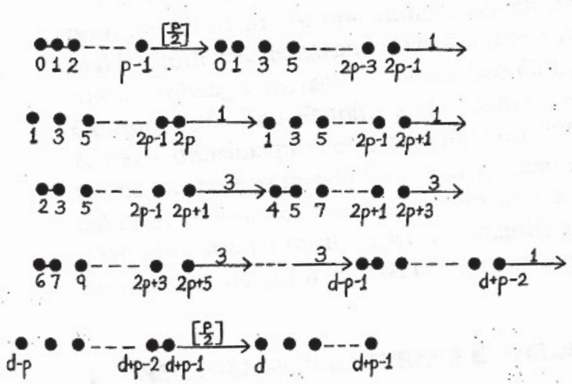


Figure 2. Solution to a 1-dimensional jumping problem when $d \geq 2p - 1$.

Again, the second and fifth configurations pictured in the figure are merely translates of each other (by two), and the same sequence of three moves is used to reach the subsequent positions. The *long jump* in each three-move sequence illustrates the leapfrogging concept mentioned earlier.

Similarly, consider a 1-dimensional *sliding problem* with p indistinguishable pieces on $\mathbf{Z}^1$. Here, a piece situated at point x can move to point y in one move, provided that y is unoccupied, and all points between x and y are occupied. (Note that if $y = x + 1$ or $y = x - 1$, then the move from x to y is simply a shift.) The pieces are originally situated at points $\{a_1 < a_2 < \ldots < a_p\}$ and we wish to maneuver them to occupy the points $\{b_1 < b_2 < \ldots < b_p\}$ in as few moves as possible.

In Benjamin (1987), it was shown that when $b_1 \geq a_p$, the unique solution to this problem is always to move forward the piece that is farthest back and not on a destination point. The optimal sequence of moves requires $b_p - a_1 - (p - 1)$ moves, as in Figure 3. Here leapfrogging is manifested within the *long slide*, a one-move sequence.

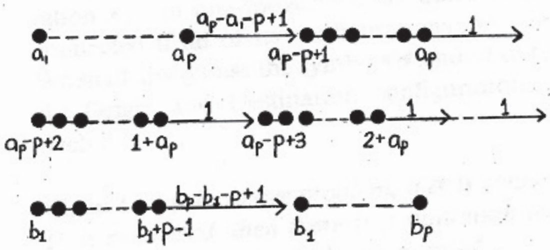


Figure 3. Solution to a 1-dimensional sliding problem.

Returning to the orignal 2-dimensional jumping problem, it is worth noting that with 2 pieces, the obvious trajectory from origin $\{(0, 0), (1, 0)\}$ to destination $\{(d, d - 1), (d, d)\}$, presented in Figure 4, is indeed optimal. Notice that almost all of the time is spent in configurations ·· and :.

The solution to the analogous four piece problem (analyzed by Auslander, Benjamin and Wilkerson 1988) maneuvers the pieces into a very efficient configuration, then repeatedly uses two moves to translate that configuration in the direction (1, 1) (see Figure 5) until we are close to (d, d); then it maneuvers the pieces to the destination. When d is large, the maneuvering time spent at the beginning and end is relatively negligible.

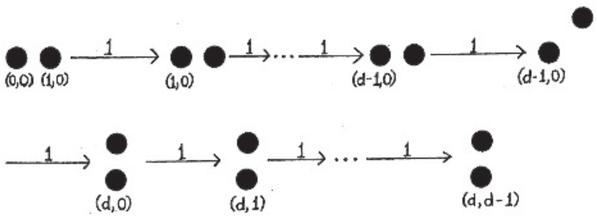


Figure 4. Solution to a 2-piece, 2-dimensional jumping problem.

For the general p-piece 2-dimensional jumping problem (with $p > 4$), the following two solutions are conjectured to be optimal. The first solution is to use the aforementioned optimal 1-dimensional configuration to crawl along the $y = 0$ axis, then after *turning the corner*, to crawl along the $x = d$ axis in a similar way. The other solution is to maneuver into a diagonal configuration, and repeatedly use a three-move procedure (see Figure 6) to translate it in the direction (1, 1) until we are near (d, d). (In Figure 6, the two alternatives for the third configuration correspond to the different possible parities of p.)

All the preceding solutions share a common feature. When the distance between the origin and destination (represented by the scalar d) is sufficiently large, most of the cost (i.e., the number of moves) is spent repeatedly translating one or two configurations (such as •• and : in the two-piece jumping problem solution). This resembles the phenomenon that if one had to efficiently drive a great distance (say from Baltimore to Los Angeles), one would spend most of the time on (perhaps only one or two) high speed interstate highways or *turnpikes*. (Strictly speaking, a turnpike is a high speed highway where some toll is charged, as opposed to a *freeway*. We shall not make use of this distinction.) Instances of this turnpike theme have been identified in the operations research literature, making both theoretical and algorithmic contributions toward solving knapsack problems (see Gilmore and Gomory 1966 and Shapiro and Wagner 1967) and Markov decision problems (see Shapiro 1968). The theme has been somewhat more prominent in mathematical economics (see Cass 1966 and McKenzie 1986). In a similar spirit, we wish to identify and prove turnpike theorems for general maneuvering problems.

1. ONE-DIMENSIONAL TURNPIKE THEORY

The general problem of finding a minimum cost sequence of moves from one subset of $\mathbf{Z}^m$ to another can be viewed as a minimum cost path problem on an infinite directed graph, where each node represents a possible placement of the pieces on $\mathbf{Z}^m$, and there exists an arc with weight c directed from node X to node Y if it is possible to reach Y from X in a single move with cost c. (In our earlier examples, each arc has a unit cost.) Of course, unless we make some additional assumptions about our rules for movement (and hence, the associated graph), we cannot hope to make any useful statements about the general problem. Toward that end, we first illustrate how the infinite configuration graph may be reducible to a *finite* graph.

1.1. Connectivity and the Finiteness of Configuration Space

Consider the 1-dimensional jumping problem with designated origin $\mathcal{O} = \{\sigma_1, \ldots, \sigma_p\}_<$ (notation: $\sigma_1 < \ldots < \sigma_p$) and destination $\mathcal{D} = \{\delta_1, \ldots, \delta_p\}_<$, with $\delta_1 \geqslant \sigma_p$. For simplicity of the following proof, let us further assume that our pieces are only allowed to move in the forward direction. For this problem, we define a configuration $\{x_1, \ldots, x_p\}_<$ to be *connected* if $x_i - x_{i-1} \leqslant 2$ for $i = 2, \ldots, p$. We define a *trajectory* to be a sequence $X_0, X_1, \ldots, X_n$ of configurations, where configuration X_i can be reached from configuration X_{i-1} in one move. We say that a trajectory is connected if all of its configurations are connected. We shall always use the symbols $\mathcal{O}$ and $\mathcal{D}$ to represent the Origin and Destination configurations, respectively.

Claim 1. *In the above problem, if $\mathcal{O}$ is connected and $\mathcal{D}$ is connected, then there is a minimum length trajectory from $\mathcal{O}$ to $\mathcal{D}$ which is connected.*

Proof. We can obviously find a (generally, disconnected) *brute force* trajectory with length $\sum_{i=1}^{p} (\delta_i - \sigma_i)$ by repeatedly shifting the front piece from σ_p to δ_p then shifting the next piece on σ_{p-1} to δ_{p-1}, and so on Since a feasible trajectory exists, a minimum lengtl trajectory must exist. Let S be the set of all minimun length trajectories from $\mathcal{O}$ to $\mathcal{D}$. To avoid trivial cases we shall assume that $p > 1$ and the length of eac minimum trajectory to be $n \geqslant 2$. We assert that contains a connected trajectory.

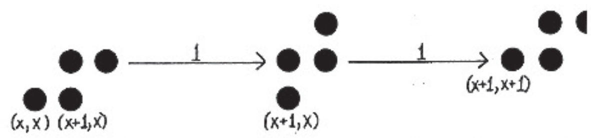


Figure 5. Solution to the 4-piece, 2-dimension jumping problem (intermediate phase).

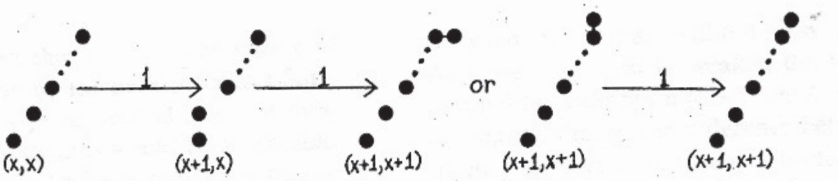


Figure 6. Conjectured solution to the p-piece, 2-dimensional jumping problem (intermediate phase).

Suppose, to the contrary, that no such connected trajectory exists. Thus, every minimum length trajectory contains a disconnected configuration. For each trajectory $T \in S$, $T = (\mathscr{O} = X_0, X_1, \ldots, X_{n-1}, X_n = \mathscr{D})$ (where X_k is the kth configuration, reachable in one move from X_{k-1}), let $i_T = \min_{i=0,\ldots,n} \{i : X_i$ is disconnected$\}$. Since $\mathscr{O}$ and $\mathscr{D}$ are connected, $1 \leqslant i_T \leqslant n - 1$, and the configuration $X_{i_T-1} \equiv \{a_1, a_2, \ldots, a_p\}_<$ must have been connected, and became disconnected after moving forward the piece located at, say, position a_j. Define $j_T = j$, and give T the label (i_T, j_T). Choose T^* to be any trajectory with label $(i_{T^*}, j_{T^*}) \equiv (i, j)$ where $i_{T^*} = \max_{T \in S} \{i_T\}$ and $j_{T^*} = \min_{T \in S} \{j_T: i_T = i\}$. In other words, T^* delays disconnecting until the last possible moment, and does so with the rearmost piece possible, without loss of optimality. Let $T^* = (\mathscr{O} = X_0^*, X_1^*, \ldots, X_{n-1}^*, X_n^* = \mathscr{D})$. Thus, $X_{i-1}^* = \{a_1, \ldots, a_p\}_<$ is connected, but after moving the piece on a_j forward, we reach $X_i^* = \{c_1, \ldots, c_p\}$, which is disconnected. Notice that since the piece on a_j either shifted forward to $1 + a_j$ or jumped over a piece on $1 + a_j$, we must have $c_k = a_k$ for $k = 1, \ldots, j - 1$ and $c_j = 1 + a_j$. It is clear that $j \neq 1$ and that the only disconnecting *gap* created by this move must exist between the pieces on $c_{j-1} = a_{j-1}$ and c_j (i.e., $c_j - c_{j-1} > 2$). Thus, since $\mathscr{D}$ is connected and backward movement is prohibited, we must eventually move forward one of the pieces located on a_k for some $k \in \{1, 2, \ldots, j-1\}$. Suppose that the next time we move one of these pieces is on the tth move where $t > i$. Let $X_{t-1} = \{b_1, \ldots, b_p\}$. (Now here is the key idea.) Since $b_1 = a_1, \ldots, b_{j-1} = a_{j-1}$ and $b_j - b_{j-1} \geqslant c_j - a_{j-1} > 2$, the piece situated at b_k may not move beyond $1 + b_{j-1}$ because no piece occupies $2 + b_{j-1}$. Therefore, all pieces situated beyond $1 + b_{j-1}$ are not relevant toward executing this move. Consequently, this same *move* (that is, physically moving the piece on $a_k = b_k$) could have been executed just before the move i actually made in T^*, rather than at move t. Since moves $i + 1, \ldots, t - 1$ do not concern the pieces on b_1 through b_{j-1}, we would still reach the same configuration X_t after the tth move. Hence, we have a new minimum trajectory that postpones the *offending* ith move another turn. If this new ith move preserves connectivity, then we have contradicted the definition of i_{T^*}. If this move disconnects the configuration, then we have contradicted the definition of j_{T^*} since $k < j$. Either way, we are provided with the desired contradiction.

What does such a claim do for us? It assures us that, for this particular problem, we can restrict our attention to connected configurations *without (asymptotic) loss of optimality*. We can, therefore, fit each configuration into a *box* of length $2p - 1$. If we consider two placements of our pieces to be *equivalent*, should they look the same when left justified in our box (that is, they are translates of each other), then we have reduced the number of possible different configurations down to 2^{p-1} (if the first piece is fixed at z_1, then $z_{i+1} = 1 + z_i$ or $2 + z_i$, $i = 1, \ldots, p - 1$), a quantity which not only is *finite*, but does not depend on the *distance* between $\mathscr{O}$ and $\mathscr{D}$. The usefulness of such a bound will soon become apparent.

Before presenting our 1-dimensional assumptions, we clarify the concepts of configuration and placement and develop a useful notation. At each moment in time (i.e., before each move) our pieces are arranged in some *configuration* X, whose back piece is situated at the position $a \in \mathbf{Z}^1$. We will refer to (X, a) as a *placement* of configuration X at the point a. For example, Figure 7 illustrates the situation where $p = 3$, $X =$ •• • and $a = 4$. Thus, in all that follows, *placement* corresponds to many of our earlier usages of *configuration*, and *configuration* to the preceding *distinct configuration*, that is, to equivalent classes of placements. In this more discriminating terminology, configuration matches the intuitive notion of *formation*, while a placement is a *placed configuration*. The notation $(X, a) \xrightarrow{c} (Y, b)$ denotes moving from (X, a) to (Y, b) with cost c (e.g., in c moves). If no c is

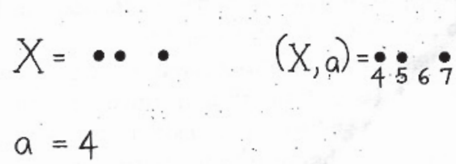


Figure 7. Configuration X and placement (X, a).

identified, it is assumed that $c = 1$. (The choice of representing the position by the back piece is a fairly arbitrary one. The front piece, second piece, or location of its center of gravity also would be acceptable, and in certain proofs, may be easier to work with. For instance, if one piece has special properties, its location may be a natural position parameter.) We are ready to state our 1-dimensional movement assumptions, which we abstract from the properties of the particular case just discussed.

1.2. Rules-for-Movement Assumptions Over $\mathbf{Z}^1$

We are interested in moving a collection of objects (called *pieces*) from one subset of $\mathbf{Z}^1$ to another at minimum cost, subject to restrictions on the allowable moves. We assume our rules for movement obey the following assumptions, to be discussed after their statements.

Finiteness. Without loss of optimality, we can prescribe a finite set $\mathscr{C}$ of allowable configurations for our pieces. From each configuration, there are a finite number of legal moves available.

Time Homogeneity. For all $(X, a) \in \mathscr{C} \times \mathbf{Z}^1$, the legal moves available from (X, a) do not depend on our particular moment in time.

Cost Homogeneity. For all $(X, a) \in \mathscr{C} \times \mathbf{Z}^1$, the legal moves available from (X, a) do not depend on the total cost accrued previously in reaching (X, a).

Space Homogeneity. For all $(X, a) \in \mathscr{C} \times \mathbf{Z}^1$, the legal moves available from (X, a), as well as their costs, do not depend on a. In terms of our notation, this says that for any $a, b, c, \delta \in \mathbf{Z}^1$, and $X, Y \in \mathscr{C}$, $(X, a) \xrightarrow{c} (Y, b)$ is legal if and only if $(X, a + \delta) \xrightarrow{c} (Y, b + \delta)$ is legal.

Brute Force Ability. There exist $r \geq 0$ and nonnegative integral brute force constants $\{c_\delta : \delta \geq r\}$ such that for all $X, Y \in \mathscr{C}$, $a \in \mathbf{Z}^1$ and $\delta \geq r$, $(X, a) \xrightarrow{c_\delta} (Y, a + \delta)$ is legal.

Positive Cycles. If $(X, a) \xrightarrow{c} (X, a + \delta)$ is legal, then $c \geq 0$. If $\delta \neq 0$, then $c > 0$.

Remarks

- For some rules for movement, the finiteness property is *explicit*—for instance, if the rules themselves actually list a finite number of legal configurations or require some sort of *connectivity* or *compactness of formation*. It is desirable, as in Claim 1, to derive useful sufficient conditions for our rules *implicitly* to yield finiteness, without loss of optimality. Also, we may wish to weaken the without-loss-of-optimality assumption to *without loss of asymptotic optimality*, that is, the difference between the minimum trajectory cost when restricted to our finite configuration set and the minimum (unrestricted) cost is bounded above by a constant, which does not depend on the distance between the origin and the destination.
- Even when time homogeneity is not strictly present, we can sometimes modify $\mathscr{C}$ so that time homogeneity is obeyed. For instance, suppose that our rules involve periodic *refueling* or *maintenance* restrictions like "you cannot go more than $t_i \geq 1$ time units without maneuvering into some configurations in the set $S_i \subseteq \mathscr{C}$, $i = 1 \ldots n$." Then one *simply* multiplies the number of configurations by $\prod_{i=1}^{n} (t_i)$ by associating, with each $X \in \mathscr{C}$, the new configurations $X^{(s_1, s_2, \ldots, s_n)}$, $0 \leq s_i < t_i$, $i = 1, \ldots, n$. The legal moves are precisely those of the following form: Supposing that $X \xrightarrow{c} Y$ when time is not a consideration (e.g., at time 0), that $J = \{j : Y \in S_j\}$, and that $s_j < t_j - 1$ for all $j \notin J$, then in our restricted problem

$$X^{(s_1, \ldots, s_n)} \xrightarrow{c} Y^{(s_1', \ldots, s_n')}$$

where

$$s_j' = \begin{cases} 0 & \text{if } j \in J \\ s_j + 1 & \text{if } j \notin J. \end{cases}$$

With $\mathscr{C}$ redefined to be $\{X^{(s_1, \ldots, s_n)} : X \in \mathscr{C}^{\text{old}}, 0 \leq s_i < t_i\}$ our rules now obey the time homogeneity assumption. In a similar way, one could accommodate restrictions of the form: you cannot make more than t_i consecutive moves of type i, $i = 1, \ldots, n$.
- Similarly, certain cost nonhomogeneities can be accommodated in the same way as time nonhomogeneities.
- It would be desirable to weaken the space-homogeneity assumption to allow for *boundaries* on (or *obstacles* in) an otherwise homogeneous environment.
- By space homogeneity, to verify Brute Force Ability it suffices to show that $(X, 0) \xrightarrow{c_\delta} (Y, \delta)$ is legal.
- We could allow the arc costs to be nonintegral, and all subsequent theorems would follow, provided that we reinterpret the notation $(X, a) \xrightarrow{c_\delta} (Y, a + \delta)$ in the brute force assumption to mean that we could maneuver from placement (X, a) to placement $(Y, a + \delta)$ with cost *not exceeding* c_δ.
- The nonnegativity of r (as in radius of maneuver) in the brute force assumption means that the existence

of any legal backward moves is not guaranteed, so that our destination had better be in the forward direction. A stronger assumption implying "there exists c such that $(X, a) \xrightarrow{c} (Y, a)$ is legal," would exclude situations like the one previously analyzed where backward movement was not allowed. (There would be no way to reach $(Y, 1)$ from $(X, 1)$ with $Y = \cdots$ and $X = \cdot\cdot\ \cdot$, while one could reach $(Y, 2)$ from $(X, 1)$.) We can often assume, without loss of optimality, that for any $X \in \mathscr{C}$, $a \in \mathbf{Z}^1$, $(X, a) \xrightarrow{1} (X, a)$ is legal.

- The positive cycle assumption is needed to ensure that we cannot make arbitrarily long progress without accumulating positive cost. The name, *positive cycles*, will be clear when we introduce the $\mathscr{C}$-Graph.

1.3. The $\mathscr{C}$-Graph (1-Dimensional)

If our rules for movement obey the aforementioned assumptions, we can conveniently represent our problem in terms of the following Configuration Graph (abbreviated $\mathscr{C}$-Graph). Our $\mathscr{C}$-Graph consists of a vertex-set (or node-set) $\mathscr{C}$ consisting of the (finite number of) allowable configurations, and a weighted arc (or directed edge) set E, where an arc exists from node X to node Y with cost c and *progress* δ if and only if $(X, a) \xrightarrow{c} (Y, a + \delta)$ is a legal move for some $a \in \mathbf{Z}^1$ (and hence, for all $a \in \mathbf{Z}^1$, by space homogeneity). In terms of our graph, the arc in Figure 8 represents the ability to move from placement (X, a) to $(Y, a + \delta)$ at cost c in a single move, for any $a \in \mathbf{Z}^1$. As before, if no c is present, then a cost of 1 is assumed. If no δ is present, then a progress of zero is assumed. Without loss of generality, we shall usually assume that a zero-progress, unit cost arc exists from every node to itself (to accommodate the brute force assumption).

Consider the 1-dimensional forward moving jumping problem analyzed at the beginning of this section, specialized to the situation where we have only $p = 3$ pieces. By the connectivity result, we need only consider four different configurations, namely

A $\cdots$

B $\cdot\cdot\ \cdot$

C $\cdot\ \cdot\cdot$

D $\cdot\ \cdot\ \cdot$

The corresponding $\mathscr{C}$-Graph appears in Figure 9.

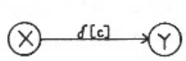


Figure 8. An arc in our $\mathscr{C}$-Graph.

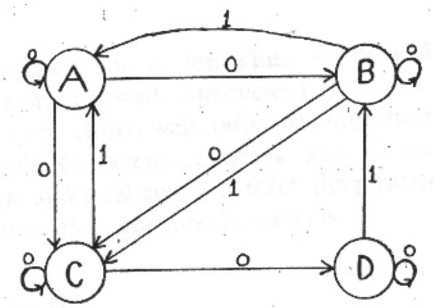


Figure 9. $\mathscr{C}$-Graph for the 3-piece, 1-dimensional jumping problem.

From the $\mathscr{C}$-Graph, we see that the brute force assumption is indeed valid with $r = 2$ and $c_\delta = 2\delta + 2$, as

$$(X, 0) \xrightarrow{4} (A, 2) \xrightarrow{1} (C, 2) \xrightarrow{1} (A, 3) \xrightarrow{2} (A, 4)$$
$$\xrightarrow{2} (A, 5) \xrightarrow{2} \ldots \xrightarrow{2} (A, \delta) \xrightarrow{2} (Y, \delta).$$

Notice that $(X, 0) \xrightarrow{4} (A, 2)$ and $(A, \delta) \xrightarrow{2} (Y, \delta)$ are possible for all X and Y (using the loops around X or Y, if necessary).

Define a *walk* on a graph to be an alternating sequence of vertices and arcs $(v_1, e_1, v_2, \ldots, e_{n-1}, v_n)$ where e_i is an arc connecting v_i to v_{i+1}. When the context is clear, we will omit mentioning the arcs. A walk is *closed* if $v_1 = v_n$. A closed walk is called a *cycle* (or circuit or dicycle or simple cycle) if $v_1, \ldots, v_{n-1}$ are all distinct. For a $\mathscr{C}$-Graph, we define the (*total*) *progress* of a walk to be the sum of the (progress) weights of the arcs of the walk, where each weight is counted as many times as the associated arc is used in the walk. Similarly, we can define the (*total*) *cost* of a walk. (When all arcs have unit cost, this is simply $n - 1$, the number of arcs in the walk counting repetitions.)

The original problem is to reach $(\mathscr{D}, d)$ from $(\mathscr{O}, 0)$ with minimum cost, subject to our rules for movement. This is equivalent to finding a minimum cost walk from node $\mathscr{O}$ to node $\mathscr{D}$ with total progress exactly d.

Notice that a closed walk (and, in particular, a cycle) beginning and ending at node X, with total progress δ and total cost c, represents the *translation* of pieces at (X, a) to $(X, a + \delta)$ with cost c for some arbitrary $a \in \mathbf{Z}^1$. Define the *speed* (or average speed or efficiency) of a cycle to be its total progress divided by its total cost. In our one-dimensional setting, a *turnpike configuration* X is one that lies on a maximum speed cycle of the $\mathscr{C}$-Graph. Recall the definition of r in the brute force assumption.

Theorem 1. (1-Dimensional Turnpike Theorem). *Consider the task of moving a collection of pieces over* $\mathbf{Z}^1$ *from* $(\mathcal{O}, 0)$ *to* $(\mathcal{D}, d)$ *at minimum cost. If the rules for movement obey the previously stated assumptions, and* $d \geqslant 2r$, *then there exists a turnpike trajectory of the following form: Maneuver the pieces from* $(\mathcal{O}, 0)$ *into some turnpike configuration, T. Repeatedly translate this configuration until you are close to* $(\mathcal{D}, d)$. *Then maneuver the pieces to* $(\mathcal{D}, d)$. *Furthermore, the difference between the cost of this trajectory and the cost of an optimal trajectory from* $\mathcal{O}$ *to* $\mathcal{D}$ *is bounded above by a constant that does not depend on d.*

Proof. In terms of our $\mathscr{C}$-Graph, the theorem (loosely) says that we can find a near minimum cost walk from $\mathcal{O}$ to $\mathcal{D}$ with total progress d, which spends most of its cost repeatedly traversing some one cycle of the $\mathscr{C}$-Graph.

Let C by a cycle of our $\mathscr{C}$-Graph with maximum average speed $s = p/q$ ($p > 0$ is the total progress of C; $q > 0$ is the total cost of C. Note that the finiteness, brute force and positive cycle assumptions imply the existence of such a cycle). Let T be an arbitrary node of C, and consider the following trajectory

$$(\mathcal{O}, 0) \xrightarrow{c_r} (T, r) \xrightarrow{q} (T, r + p) \xrightarrow{q} (T, r + 2p) \xrightarrow{q} \dots \xrightarrow{q} (T, r + xp) \xrightarrow{c_\delta} (\mathcal{D}, d)$$

where $x = \lfloor (d - 2r)/p \rfloor$ and $\delta = d - (r + xp)$. The cost of this trajectory is $c_r + xq + c_\delta \leqslant c_r + c_\delta + (d - 2r)/s$. This is our turnpike trajectory, which translates the configuration T (i.e., traverses cycle C) x times. Notice that $r \leqslant \delta \leqslant r + p - 1$, which does not depend on d.

Let $(\mathcal{O}, 0) \xrightarrow{c^*} (\mathcal{D}, d)$ be a minimum cost trajectory and consider the trajectory $(\mathcal{O}, 0) \xrightarrow{c^*} (\mathcal{D}, d) \xrightarrow{c_r} (\mathcal{O}, d + r)$. This represents a closed walk (from $\mathcal{O}$ to $\mathcal{O}$) along our $\mathscr{C}$-Graph with total progress $d + r$ and total cost $c^* + c_r$.

We can decompose the arcs of any closed walk into cycles. That is, if the cycles of our $\mathscr{C}$-Graph are $C_1, \dots, C_n$, we can find nonnegative integers $x_1, \dots, x_n$ such that if we traverse cycle C_i x_i times, $i = 1, \dots, n$, then every arc will be traversed exactly as many times as in the closed walk. (This can be proven by induction on the number of arcs (repetitions counted) of the walk as follows. If the closed walk is itself a cycle, we are done. Otherwise, it contains a node v that is visited twice (if the only such node is the first node, then it is re-visited before the end). Hence, our walk contains an internal closed walk which, inductively is decomposable into cycles. After removing this subwalk from our walk, the remaining walk remains closed, and this too, by induction, can be decomposed into cycles. Thus, we have decomposed our original walk into cycles.)

Thus, if our closed walk (after decomposition) traverses cycle C_i exactly x_i times, and C_i has total progress p_i and total cost $q_i > 0$ (cf. the positive cycle assumption), then our total progress is

$$d + r = \sum_{i=1}^{n} p_i x_i.$$

Our total cost is

$$c^* + c_r = \sum_{i=1}^{m} q_i x_i$$

and so our average speed is

$$\frac{d + r}{c^* + c_r} = \frac{\sum_{i=1}^{n} p_i x_i}{\sum_{i=1}^{n} q_i x_i} \leqslant s$$

where the inequality follows from $p_i \leqslant s q_i$. Therefore $c^* \geqslant (d + r)/s - c_r$. Since the cost of our turnpike trajectory is at most $c_r + c_\delta + (d - 2r)/s$ it follows that

$$\frac{d + r}{s} - c_r \leqslant c^* \leqslant \frac{d - 2r}{s} + c_r + c_\delta. \quad (1)$$

Thus, the difference between the cost of our turnpike trajectory and an optimal trajectory is at most

$$c_r + c_\delta + \frac{d - 2r}{s} - c^*$$
$$\leqslant \left(c_r + c_\delta + \frac{d - 2r}{s}\right) - \left(\frac{d + r}{s} - c_r\right)$$
$$= 2c_r + c_\delta - \frac{3r}{s}$$

with a bound (since $\delta \leqslant r + p - 1$) that does no depend on d.

The preceding result is analogous to theorems give by Chrétienne (1984), with nonconstructive proofs i the manner of Gilmore and Gomory, which impl that a "maximum valued" walk from $\mathcal{O}$ to $\mathcal{D}$ wit progress d necessarily spends most of its time travelin turnpike cycles as d gets large. Those theorems wei not extended to higher dimensions.

Note that we have shown the difference in cost (the optimal trajectory and our turnpike trajectoı to be bounded by a constant which becomes rel tively negligible as d gets large. That is, we have t Equation 1

$$\lim_{d \to \infty} \frac{c^*}{d/s} = 1.$$

Thus, $c^* \approx d/s$ for large d.

1.4. Examples

Returning to the $\mathscr{C}$-Graph for the three piece, 1-dimensional jumping problem (see Figure 9), we notice that it contains seven simple cycles, excluding the four zero-progress loops (see Table I). Cycles ABC′ and BCD′ denote cycles ABC and BCD where the zero-progress arc from B to C is used instead of the unit-progress arc.

The cycles ABC and BCD are turnpike cycles, with maximum speed $2/3$. Thus, if we let ABC play the role of our turnpike cycle with $p = 2$ and $q = 3$ and let B be our entering turnpike configuration within BCD, then our turnpike trajectory, from origin $(A, 0)$ to destination $(D, 99)$, is

$$(A, 0) \xrightarrow{6} (B, 2) \xrightarrow{3} (B, 4) \xrightarrow{3} (B, 6)$$
$$\xrightarrow{3} \dots \xrightarrow{3} (B, 94) \xrightarrow{3} (B, 96) \xrightarrow{8} (D, 99)$$

with a cost of $6 + 3(47) + 8 = 1.55$. To illustrate the merely asymptotic nature of the optimality provided by such a trajectory, we observe the lesser length, $150 = 1 + 49(3) + 1 + 1$, attained (via cycle BCD) by

$$(A, 0) \xrightarrow{1} (B, 0) \xrightarrow{3} (B, 2) \xrightarrow{3} (B, 4)$$
$$\xrightarrow{3} \dots \xrightarrow{3} (B, 98) \xrightarrow{1} (C, 99) \xrightarrow{1} (D, 99).$$

The trajectory is optimal because if we could maneuver from $(A, 0)$ to $(D, 99)$ at a cost $c \leqslant 149$, then the closed walk $(A, 0) \xrightarrow{c} (D, 99) \xrightarrow{1} (B, 100) \xrightarrow{1} (A, 101)$ would have a progress/cost ratio of $101/(c + 2) \geqslant 101/151 > 2/3$, which is impossible by Table I.

As another example, consider the previous problem with a distinguished piece. The same rules apply, but now only the distinguished piece is allowed to perform a double jump. Here we have $4 \times 3 = 12$ nodes $X1$, $X2$, $X3$ depicting whether the distinguished piece is in front, middle, or back, respectively, in the configuration $X \in \{A, B, C, D\}$. In the corresponding $\mathscr{C}$-Graph (see Figure 10), the dotted lines denote arcs with progress 0, solid lines denote arcs with progress 1, and all arcs have a cost of 1. We can prove that the

Table I
Speeds of Cycles of the Three-Piece Jumping Problem

Cycle	Speed
AB	1/2
ABC	2/3
ABC′	1/3
AC	1/2
ACDB	3/4
BCD	2/3
BCD′	1/3

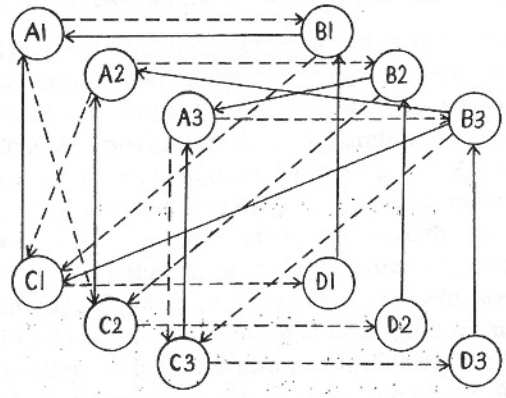


Figure 10. $\mathscr{C}$-Graph for a 3-piece, 1-dimensional jumping problem with a distinguished piece (loops omitted).

maximum cycle speed is $4/7$ as follows. First, we prove that all cycles that do not use the arc from $B3$ to $C1$ (corresponding to performing the double jump) have speed at most $1/2$. We see this by removing the arc from $B3$ to $C1$ and *projecting* to the $\mathscr{C}$-Graph in Figure 11. Here we have a solid (dotted) line from node X to node Y if there exists a solid (dotted) line from X_i to Y_j for some i, j. Notice that the only simple cycle utilizing two consecutive solid lines is cycle ACDB, with speed $1/2$. All other cycles must follow a solid arc with a dotted arc and therefore have speed at most $1/2$ in this graph, and consequently, in the original graph as well. Thus, any cycle with speed greater than $1/2$ must use the solid arc from $B3$ to $C1$ in the original graph. By branching from $C1$, we see that the minimum length path from $C1$ to $B3$ is of length 6, which by the preceding argument cannot have more dotted lines than solid. Hence, the speed of the cycle is at most $(1 + x/2)/(1 + x)$, $x \geqslant 6$, hence, at most $4/7$. This is attained by the cycle $C1 - A1 - C2 - A2 - B2 - A3 - B3$.

As a less obvious example, consider the knapsack-type problem

$$\text{minimize} \quad \sum_{j=1}^{n} f_j x_j$$

$$\text{subject to} \quad \sum_{j=1}^{n} h_j x_j = d$$

x_j nonnegative integer

where we assume $h_1 = 1$ to ensure feasibility, and that f_j, $h_j > 0$ for all j. Suppose further that for all j $(h_j/f_j \leqslant h_n/f_n)$. Then we can construct the single node $\mathscr{C}$-Graph with n (loop) arcs, where arc j has progress h_j and cost f_j. The problem then is to find a minimum

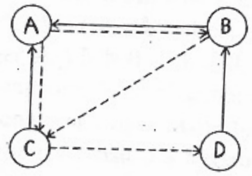


Figure 11. A projected problem.

cost walk from $(X, 0)$ to (X, d). (Note that $h_1 = 1$ easily gives us our brute force condition.) Our turnpike trajectory then spends most of its cost along the minimum cycle from X to X along the nth arc. This corresponds to the feasible solution

$$x_n = \lfloor d/h_n \rfloor, \quad x_1 = d - h_n \lfloor d/h_n \rfloor,$$
$$x_j = 0, \quad j \neq 1, n$$

which is nearly optimal for d sufficiently large. (This is essentially the result of Gilmore and Gomory.)

To make the correspondence explicit, observe that the above integer program can be represented by a simple one piece, one-dimensional problem in which our piece must move from 0 to d at minimum cost, and the jth of the n legal moves available from any configuration propels the piece forward h_j units at a cost of f_j.

As a last example, consider the NP-complete partition problem (Garey and Johnson 1978): Given a finite multiset $A = \{a_1, \ldots, a_N\}$ of positive integers, does there exist a subset $A' \subseteq A$ such that $\sum_{a \in A'} a = \sum_{a \in A - A'} a$? (We assume that $\sum_{i=1}^{N} a_i$ is even.) The problem can be transformed into a one-dimensional $\mathscr{C}$-Graph problem, namely, "Does there exist a walk from node 0 to node N in the $\mathscr{C}$-Graph in Figure 12 with total progress $\sum_{i=1}^{N} a_i/2$ and total cost $\leq N$?" This transformation is used in Benjamin (1989) to show that the decision problem associated with $\mathscr{C}$-Graphs is NP-complete.

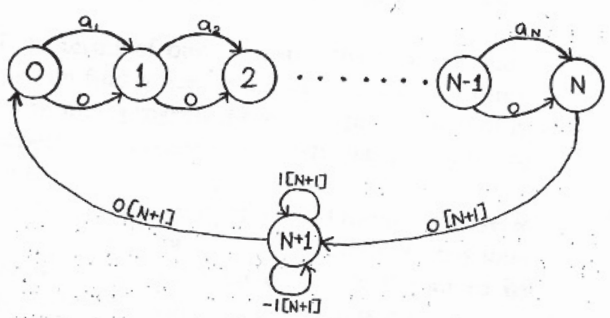


Figure 12. A $\mathscr{C}$-Graph for the partition problem.

2. HIGHER-DIMENSIONAL TURNPIKE THEORY

The preceding theory extends rather nicely to higher dimensions. When maneuvering our pieces of $\mathbf{Z}^m$, we make the following adjustments. $(X, \mathbf{a}) \xrightarrow{c} (Y, \bar{\mathbf{a}})$ denotes moving from configuration X placed at $\mathbf{a} \in \mathbf{Z}^m$ to configuration Y placed at $\bar{\mathbf{a}} \in \mathbf{Z}^m$ with cost $c \in \mathbf{Z}^1$. More specifically, we shall assume that $\mathbf{a} = (a_1, \ldots, a_m)$ and let $(X, \mathbf{a})$ denote that placement of X such that a_i is the minimum ith coordinate among all pieces in X. (For an example, see Figure 13.) As in the 1-dimensional case, other measures of location such as maximum coordinates, the location of some distinguishable piece, or the center of gravity, will also work, and may be more natural for certain problems. (The last quantity belongs to the set $(1/p)\mathbf{Z}^m$ where p is the number of pieces.)

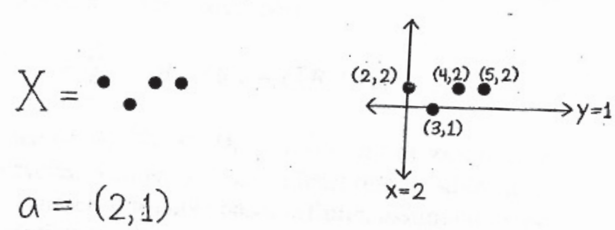


Figure 13. Configuration X and placement (X, a).

2.1. Rules-for-Movement Assumptions Over $\mathbf{Z}^m$

We are interested in moving a collection of objects from one subset of $\mathbf{Z}^m$ to another at minimum cost, subject to certain restrictions on the *elementary* movements. We assume our rules for movement obey the following assumptions.

Finiteness. Without loss of optimality, we can prescribe a finite set $\mathscr{C}$ of allowable configurations for our pieces. From each configuration, there are a finite number of legal moves available.

Time, Cost and Space Homogeneity. For all $(X, \mathbf{a}) \in \mathscr{C} \times \mathbf{Z}^m$ the costs and legal moves available from $(X, \mathbf{a})$ depend only on X. That is, for all $X, Y \in \mathscr{C}$, $c \in \mathbf{Z}$, and $\mathbf{a}, \bar{\mathbf{a}}, \delta \in \mathbf{Z}^m$, we have that $(X, \mathbf{a}) \xrightarrow{c} (Y, \bar{\mathbf{a}})$ is legal if and only if $(X, \mathbf{a} + \delta) \xrightarrow{c} (Y, \bar{\mathbf{a}} + \delta)$ is legal.

Brute Force Ability. There exist nonnegative integral brute force constants $\{c_\delta\}$ such that for all $X, Y \in \mathscr{C}$, and $\mathbf{a}, \delta \in \mathbf{Z}^m$, $(X, \mathbf{a}) \xrightarrow{c_\delta} (Y, \mathbf{a} + \delta)$ is legal. In particular, $(X, \mathbf{a}) \xrightarrow{c_0} (Y, \mathbf{a})$ is legal.

Positive Cycles. If $(X, \mathbf{a}) \xrightarrow{c} (X, \bar{\mathbf{a}})$ is legal, then $c \geq 0$. If $\mathbf{a} \neq \bar{\mathbf{a}}$, then $c > 0$.

The remarks following the 1-dimensional assumptions remain valid. We are assuming that our desired destination from $(\mathscr{O}, \mathbf{0})$ is $(\mathscr{D}, d\,\mathbf{b})$ where $\mathscr{D} \in \mathscr{C}$, d is a large positive integer and $\mathbf{b} \geqslant \mathbf{0}$. If $\mathbf{b} \not\geqslant \mathbf{0}$, then we can "re-coordinatize" without loss of generality. Notice that here we are using a stronger brute force assumption than in the one-dimensional version. We shall say more about this after the proof of the next theorem.

2.2. The $\mathscr{C}$-Graph (*m*-Dimensional Version)

The $\mathscr{C}$-Graph for the m-dimensional problem is similar to the 1-dimensional $\mathscr{C}$-Graph. Here, an arc is present from node X to node Y, with cost $c \in \mathbf{Z}^1$ and progress $\delta \in \mathbf{Z}^m$, if and only if in a single move, we can move from configuration $(X, \mathbf{a})$ to $(Y, \mathbf{a} + \delta)$ at cost c for any $\mathbf{a} \in \mathbf{Z}^1$. As before, if no c is present, then a cost of 1 is assumed. We shall usually assume that a zero-progress, unit cost arc exists from every node to itself, to accommodate the brute force assumption.

Also, as before, a closed walk from node X to X represents a translation of configuration X, with total progress and total cost defined, respectively, as the sum of the walk-arcs' cost weights and the (vector) sum of their progress weights. Determining a minimum cost trajectory from $(\mathscr{O}, \mathbf{0})$ to $(\mathscr{D}, d\,\mathbf{b})$, $d > 0$, $\mathbf{b} \geqslant \mathbf{0}$ is equivalent to finding a minimum cost walk in our $\mathscr{C}$-Graph from node $\mathscr{O}$ to node $\mathscr{D}$ with total progress $d\,\mathbf{b}$. If $\mathscr{O} = \mathscr{D}$, the walk is closed.

Theorem 2 (*m*-Dimensional Turnpike Theorem). *Consider the problem of moving a collection of pieces over $\mathbf{Z}^m$ from $(\mathscr{O}, \mathbf{0})$ to $(\mathscr{D}, d\,\mathbf{b})$ at minimum cost. If the rules for movement obey the previously stated assumptions, then there exists a turnpike trajectory of the following form. Letting $\mathscr{O} = T_0$, proceed as follows. For $i = 0, \ldots, m-1$, brute force maneuver from configuration T_i to an appropriate configuration T_{i+1}, then repeatedly translate T_{i+1} x_{i+1} times, x_{i+1} an appropriate nonnegative integer. Then, brute force maneuver from T_m to $\mathscr{D}$. Furthermore, the difference between the cost of this trajectory and that of an optimal trajectory is bounded above by a constant that does not depend on d or $\mathbf{b}$.*

Proof. In terms of our $\mathscr{C}$-Graph, the theorem states that we can find a near minimum cost walk from $\mathscr{O}$ to $\mathscr{D}$ with total progress $d\,\mathbf{b}$ which spends most of its cost repeatedly traversing m particular cycles of the $\mathscr{C}$-Graph.

Suppose that the cycles of our $\mathscr{C}$-Graph are n cycles $C^1, \ldots, C^n$, where for $i = 1, \ldots, n$, cycle C^i has total progress $\mathbf{a}_i \in \mathbf{Z}^m$ and total cost $c^i > 0$ (not to be confused with our brute force constants c_δ).

Let A denote the $m \times n$ matrix with ith column $\mathbf{a}_i$, $i = 1, \ldots, n$. We now use the brute force assumption to prove that A has full row rank, as follows. Consider any $X \in \mathscr{C}$, and any vector $\mathbf{v} \in \mathbf{Z}^m$. By the brute force assumption there is a closed walk from $(X, \mathbf{0})$ to $(X, \mathbf{v})$. As shown while proving Theorem 1, this closed walk can be decomposed into cycles. Therefore, $\mathbf{v}$ can be expressed as a (nonnegative integral) linear combination of $\mathbf{a}_1, \ldots, \mathbf{a}_n$, $i = 1, \ldots, n$. Hence, any integral vector $\mathbf{v}$ can be expressed as a nonnegative integral combination of some of the $\mathbf{a}_i$'s. Thus, A has full row rank.

Let $M = \max_{i,j} |a_{i,j}|$ and let $\mathbf{e}^T = (1, \ldots, 1) \in \mathbf{Z}^m$. For any linearly independent set of m column vectors $\{\mathbf{a}_{i_1}, \ldots, \mathbf{a}_{i_m}\}$, we can express $d\,\mathbf{b}$ as a linear combination of these vectors in precisely one way (namely as $d\,\mathbf{b} = \sum_{j=1}^{m} \mathbf{a}_{i_j} x_j$ where $x_j = (B^{-1} d\,\mathbf{b})_j \in \mathbf{Q}$ where $\mathbf{a}_{i_j}$ is the jth column of B). If $\mathbf{x} = (x_1, \ldots, x_m)$ is nonnegative, we say that the basis $B = \{\mathbf{a}_{i_1}, \ldots, \mathbf{a}_{i_m}\}$ is feasible, and has a total cost

$$\sum_{j=1}^{m} c^{i_j} x_j = \sum_{j=1}^{m} c^{i_j} (B^{-1} d\,\mathbf{b})_j \equiv \mathbf{c}_B^T B^{-1} d\,\mathbf{b}.$$

Since $(\mathscr{O}, \mathbf{0}) \xrightarrow{c_{d\mathbf{b}}} (\mathscr{O}, d\,\mathbf{b})$ is legal and decomposable into cycles, A must contain at least one feasible basis. The number of feasible bases is finite; assume for ease of notation that $\{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$ is a feasible basis with a minimum total cost $d\,\mathbf{c}_B^T B^{-1} \mathbf{b}$ where $B = [\mathbf{a}_1, \ldots, \mathbf{a}_m]$ and $\mathbf{c}_B^T = (c^1, \ldots, c^m)$. Let $x_i = (dB^{-1}\mathbf{b})_i \in \mathbf{Q}_+$. Let T_i be an arbitrary configuration node on cycle C^i, $i = 1, \ldots, m$. Then our turnpike trajectory can be constructed as

$$\begin{aligned}
(\mathscr{O}, \mathbf{0}) &\xrightarrow{c_\delta} (T_1, \mathbf{0}) \xrightarrow{c^1 \lfloor x_1 \rfloor} (T_1, \lfloor x_1 \rfloor \mathbf{a}_1) \\
&\xrightarrow{c_\delta} (T_2, \lfloor x_1 \rfloor \mathbf{a}_1) \xrightarrow{c^2 \lfloor x_2 \rfloor} (T_2, \lfloor x_1 \rfloor \mathbf{a}_1 + \lfloor x_2 \rfloor \mathbf{a}_2) \\
&\xrightarrow{c_\delta} \cdots \xrightarrow{c_\delta} \left(T_i, \sum_{j=1}^{i-1} \lfloor x_j \rfloor \mathbf{a}_j\right) \\
&\xrightarrow{c^i \lfloor x_i \rfloor} \left(T_i, \sum_{j=1}^{i} \lfloor x_j \rfloor \mathbf{a}_j\right) \\
&\xrightarrow{c_\delta} \cdots \xrightarrow{c_\delta} \left(T_m, \sum_{j=1}^{m-1} \lfloor x_j \rfloor \mathbf{a}_j\right) \\
&\xrightarrow{c^m \lfloor x_m \rfloor} \left(T_m, \sum_{j=1}^{m} \lfloor x_j \rfloor \mathbf{a}_j\right) \xrightarrow{c_\delta} (\mathscr{D}, d\,\mathbf{b})
\end{aligned}$$

where

$$\begin{aligned}
\delta &= d\,\mathbf{b} - \sum_{j=1}^{m} \lfloor x_j \rfloor \mathbf{a}_j \\
&= d\,\mathbf{b} - d\,\mathbf{b} + \sum_{j=1}^{m} a_j f_j = \sum_{j=1}^{m} (a_j f_j)
\end{aligned}$$

and $f_j \equiv x_j \bmod 1$, $0 \leqslant f_j < 1$. Consequently, $-mM\mathbf{e} \leqslant \delta \leqslant mM\mathbf{e}$, and since $\{\delta \in \mathbf{Z}^m : \|\delta\|_\infty \leqslant mM\mathbf{e}\}$ is a finite set, c_δ is bounded above by a constant $\bar{c}$ that does not depend on d or $\mathbf{b}$. The total cost of this trajectory is

$$\text{TPCOST} = mc_0 + \sum_{i=1}^{m} c^i \lfloor x_i \rfloor + c_\delta$$

$$\leqslant mc_0 + \sum_{i=1}^{m} c^i \lfloor x_i \rfloor + \bar{c}. \tag{2}$$

Notice that by construction, $\mathbf{x} = (x_1, \ldots, x_n)^T$ is a (basic feasible) optimal solution to the linear program

$$u^* = \min_{\mathbf{x}} \sum_{i=1}^{n} c^i x_i$$

subject to $A\mathbf{x} = d\mathbf{b}$, $\mathbf{x} \geqslant \mathbf{0}$.

Furthermore, if c^* denotes the minimum cost to reach $(\mathscr{D}, d\mathbf{b})$ from $(\mathscr{O}, \mathbf{0})$, then c^* cannot exceed the cost of the turnpike trajectory. Hence, by equation (2), we must have

$$c^* \leqslant \text{TPCOST} \leqslant mc_0 + \bar{c} + u^*. \tag{3}$$

On the other hand, consider some trajectory $(\mathscr{O}, \mathbf{0}) \xrightarrow{c^*} (\mathscr{D}, d\mathbf{b})$ with minimum cost c^*. Since the trajectory $(\mathscr{O}, \mathbf{0}) \xrightarrow{c^*} (\mathscr{D}, d\mathbf{b}) \xrightarrow{c_0} (\mathscr{O}, d\mathbf{b})$ is a closed walk on our $\mathscr{C}$-Graph, it can be decomposed into cycles. Thus, $d\mathbf{b} = \sum_{j=1}^{n} \mathbf{a}_j y_j$ for some nonnegative integers y_j, $j = 1, \ldots, n$. It follows that

$$c^* + c_0 \geqslant \min \sum_{j=1}^{n} c^j x_j$$

subject to $A\mathbf{x} = d\mathbf{b}$

$\mathbf{x} \geqslant \mathbf{0}$ integer

$$\geqslant \min \sum_{j=1}^{n} c^j x_j = u^*$$

subject to $A\mathbf{x} = d\mathbf{b}$

$\mathbf{x} \geqslant \mathbf{0}$.

That is

$$c^* \geqslant -c_0 + u^*. \tag{4}$$

Combining relations (3) and (4), we have

$$-c_0 + u^* \leqslant c^* \leqslant \text{TPCOST} \leqslant mc_0 + \bar{c} + u^*.$$

Consequently

$$\text{TPCOST} - c^* \leqslant (m + 1)c_0 + \bar{c} \tag{5}$$

which does not depend on d or $\mathbf{b}$.

By analogy to the 1-dimensional turnpike theorem, we can proceed to prove a similar theorem using a weaker brute force assumption. First, we point out that the apparent weakening, "There exist $r \geqslant 0$ and nonnegative brute force constants $\{c_i : i \geqslant r\}$ such that for all $X, Y \in \mathscr{C}$ and $\mathbf{a} \in \mathbf{Z}^m$ $(X, \mathbf{a}) \xrightarrow{c_{\|\delta\|}} (Y, \mathbf{a} + \delta)$ is legal whenever $\|\delta\| \geqslant r$," is actually equivalent to the current assumption, since for any $X, Y \in \mathscr{C}$, $\mathbf{a}, \delta \in \mathbf{Z}^m$ with $\|\delta\| < r$ $(X, \mathbf{a}) \xrightarrow{c_t} (Y, \mathbf{a} + (1 + r/\|\delta\|)\delta) \xrightarrow{c_r} (Y, \mathbf{a} + \delta)$ is legal, where $t = \|\delta\| + r$.

However, if we assume that $\mathbf{b} > 0$, then we can prove the previous theorem under a genuinely weaker brute force assumption, namely: There exist $\mathbf{r} \geqslant \mathbf{0}_m$ and nonnegative brute force constants $\{c_\delta : \delta \geqslant \mathbf{r}\}$, such that for all $X, Y \in \mathscr{C}$, $\mathbf{a} \in \mathbf{Z}^m$ and $\delta \geqslant \mathbf{r}$, $(X, \mathbf{a}) \xrightarrow{c_\delta} (Y, \mathbf{a} + \delta)$ is legal. This is analogous to the one-dimensional brute force assumption, and is motivated by the desire to include rules for movement where we are restricted to move only in forward directions.

Theorem 3. *When* $\mathbf{b} > 0$, *Theorem 2 is true under the weaker brute force assumption above, when* d *is sufficiently large.*

Proof. As before, let the cycles of our $\mathscr{C}$-Graph be $C^1, \ldots, C^n$, where for $i = 1, \ldots, n$, cycle C^i has total progress $\mathbf{a}_i \in \mathbf{Z}^m$ and total cost $c^i > 0$, and let A denote the $m \times n$ matrix with ith column $\mathbf{a}_i$, $i = 1, \ldots, n$. Since $\{\mathbf{v} \in \mathbf{Z}^m : \mathbf{v} \geqslant \mathbf{r}\}$ has dimension m, and (by the weak brute force assumption) lies in the column span of A, A has full row rank.

Let $M = \max_{i,j} |a_{i,j}|$, and let $\mathbf{e}^T = (1, \ldots, 1) \in \mathbf{Z}^m$. Let $\hat{\mathbf{b}} = d\mathbf{b} - mM\mathbf{e} - (m + 1)\mathbf{r}$. As in the previous proof, for any linearly independent set of m column vectors $\{\mathbf{a}_{i_1}, \ldots, \mathbf{a}_{i_m}\}$, we can express $\hat{\mathbf{b}}$ as a linear combination of these vectors in precisely one way.

Since $\mathbf{b} > 0$, we must have $\hat{\mathbf{b}} > \mathbf{r}$ for sufficiently large d. Thus $(\mathscr{O}, \mathbf{0}) \xrightarrow{c_{\hat{\mathbf{b}}}} (\mathscr{O}, \hat{\mathbf{b}})$ is legal and decomposable into cycles, so that A must contain at least one feasible basis for the system $A\mathbf{x} = \hat{\mathbf{b}}$. The number of feasible bases is finite; assume for ease of notation that $\{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$ is a feasible basis with minimum total cost $\mathbf{c}_B^T B^{-1}\hat{\mathbf{b}}$, where $B = [\mathbf{a}_1, \ldots, \mathbf{a}_m]$ and $\mathbf{c}_B^T = (c^1, \ldots, c^m)$. Let $x_i = (B^{-1}\hat{\mathbf{b}})_i \in \mathbf{Q}_+$. Let T_i be an arbitrary configuration node on cycle C^i, $i = 1, \ldots, m$. Then our turnpike trajectory can be constructed as

$$(\mathscr{O}, \mathbf{0}) \xrightarrow{c_r} (T_1, \mathbf{r}) \xrightarrow{c^1\lfloor x_1\rfloor} (T_1, \mathbf{r} + \lfloor x_1\rfloor \mathbf{a}_1)$$

$$\xrightarrow{c_r} (T_2, 2\mathbf{r} + \lfloor x_1\rfloor \mathbf{a}_1)$$

$$\xrightarrow{c^2\lfloor x_2\rfloor} (T_2, 2\mathbf{r} + \lfloor x_1\rfloor \mathbf{a}_1 + \lfloor x_2\rfloor \mathbf{a}_2)$$

$$\xrightarrow{c_r} \ldots \xrightarrow{c_r} \left(T_i,\, i\mathbf{r} + \sum_{j=1}^{i-1} \lfloor x_j\rfloor \mathbf{a}_j\right)$$

$$\xrightarrow{c^i\lfloor x_i\rfloor} \left(T_i,\, i\mathbf{r} + \sum_{j=1}^{i} \lfloor x_j\rfloor \mathbf{a}_j\right)$$

$$\xrightarrow{c_r} \ldots \xrightarrow{c_r} \left(T_m,\, m\mathbf{r} + \sum_{j=1}^{m-1} \lfloor x_j\rfloor \mathbf{a}_j\right)$$

$$\xrightarrow{c^m\lfloor x_m\rfloor} \left(T_m,\, m\mathbf{r} + \sum_{j=1}^{m} \lfloor x_j\rfloor \mathbf{a}_j\right)$$

$$\xrightarrow{c_\delta} (\mathscr{D}, d\mathbf{b})$$

where

$$\delta = d\mathbf{b} - \left(m\mathbf{r} + \sum_{j=1}^{m} \lfloor x_j\rfloor \mathbf{a}_j\right)$$

$$= d\mathbf{b} - m\mathbf{r} - \left(\hat{\mathbf{b}} - \sum_{j=1}^{m} \mathbf{a}_j f_j\right)$$

$$(\text{where } f_j \equiv x_j \bmod 1,\ 0 \leqslant f_j < 1)$$

$$= d\mathbf{b} - m\mathbf{r} + \sum_{j=1}^{m} \mathbf{a}_j f_j - d\mathbf{b} + (m+1)r + mM\mathbf{e}$$

$$= r + \sum_{j=1}^{m} (\mathbf{a}_j f_j) + mM\mathbf{e}.$$

Consequently, $\mathbf{r} \leqslant \delta \leqslant \mathbf{r} + 2mM\mathbf{e}$, and since $\{\delta \in \mathbf{Z}^m : \mathbf{r} \leqslant \delta \leqslant \mathbf{r} + 2mM\mathbf{e}\}$ is a finite set, c_δ is bounded above by a constant $\bar{c}$ that does not depend on d or $\mathbf{b}$. The total cost of this trajectory is

$$\text{TPCOST} = mc_r + \sum_{i=1}^{m} c^i \lfloor x_i \rfloor + c_\delta$$

$$\leqslant mc_r + \sum_{i=1}^{m} c^i \lfloor x_i \rfloor + \bar{c}. \tag{6}$$

Notice that by construction, $\mathbf{x} = (x_1, \ldots, x_n)^T$ is a (basic feasible) optimal solution to the linear program

$$u^* = \min_{\mathbf{x}} \sum_{i=1}^{n} c^i x_i$$

subject to $A\mathbf{x} = \hat{\mathbf{b}}, \quad \mathbf{x} \geqslant \mathbf{0}.$

Furthermore, if c^* denotes the minimum cost to reach $(\mathscr{D}, d\mathbf{b})$ from $(\mathscr{O}, \mathbf{0})$, then c^* cannot exceed the cost of the turnpike trajectory. Hence, by Equation 6, we must have

$$c^* \leqslant \text{TPCOST} \leqslant mc_r + \bar{c} + u^*. \tag{7}$$

On the other hand, consider some trajectory $(\mathscr{O}, \mathbf{0}) \xrightarrow{c^*} (\mathscr{D}, d\mathbf{b})$ with minimum cost c^*. Since the trajectory $(\mathscr{O}, \mathbf{0}) \xrightarrow{c^*} (\mathscr{D}, d\mathbf{b}) \xrightarrow{c_r} (\mathscr{O}, d\mathbf{b} + \mathbf{r})$ is a closed walk on our $\mathscr{C}$-Graph, it can be decomposed into cycles. Thus, setting $\overline{\mathbf{b}} = d\mathbf{b} + \mathbf{r}$, we have $\overline{\mathbf{b}} = \sum_{j=1}^{n} \mathbf{a}_j y_j$ for some nonnegative integers y_j, $j = 1, \ldots, n$. It follows that

$$c^* + c_r \geqslant \min \sum_{j=1}^{n} c^j x_j$$

subject to $A\mathbf{x} = \overline{\mathbf{b}}$

$\mathbf{x} \geqslant \mathbf{0}$ integer

$$\geqslant \min \sum_{j=1}^{n} c^j x_j \equiv z^*$$

subject to $A\mathbf{x} = \overline{\mathbf{b}}$

$\mathbf{x} \geqslant \mathbf{0}$

that is

$$c^* \geqslant -c_r + z^*. \tag{8}$$

Combining relations (7) and (8), we have

$$-c_r + z^* \leqslant c^* \leqslant \text{TPCOST} \leqslant mc_r + \bar{c} + u^*.$$

Consequently

$$\text{TPCOST} - c^* \leqslant (m+1)c_r + \bar{c} + u^* - z^*. \tag{9}$$

But u^* and z^* denote optimal objective function values to linear programs with parameters $(A, \hat{\mathbf{b}}, \mathbf{c})$ and $(A, \overline{\mathbf{b}}, \mathbf{c})$, respectively. By theorem (2.4) of Mangasarian and Shiau (1987), there exist optimal solutions $\hat{\mathbf{x}}$ and $\bar{\mathbf{x}}$ to the above linear programs satisfying

$$\|\hat{\mathbf{x}} - \bar{\mathbf{x}}\|_\infty \leqslant k_A \|\hat{\mathbf{b}} - \overline{\mathbf{b}}\|_\infty$$

where k_A is a constant depending only on A. Thus

$$u^* - z^*$$

$$= \mathbf{c}^T\hat{\mathbf{x}} - \mathbf{c}^T\bar{\mathbf{x}}$$

$$= \mathbf{c}^T(\hat{\mathbf{x}} - \bar{\mathbf{x}})$$

$$\leqslant \|\mathbf{c}\|_\infty \|\hat{\mathbf{x}} - \bar{\mathbf{x}}\|_\infty$$

$$\leqslant k_A \|\mathbf{c}\|_\infty \|\hat{\mathbf{b}} - \overline{\mathbf{b}}\|_\infty$$

$$= k_A \|\mathbf{c}\|_\infty \|d\mathbf{b} - mM\mathbf{e} - (m+1)\mathbf{r} - (d\mathbf{b} + \mathbf{r})\|_\infty$$

$$= k_A \|\mathbf{c}\|_\infty \|mM\mathbf{e} + (m+2)\mathbf{r}\|_\infty \tag{10}$$

which does not depend on d or $\mathbf{b}$.

Consequently, by relations (9) and (10), we see that the difference between the cost of our turnpike trajectory and the minimum cost trajectory is bounded by a constant which does not depend on d or $\mathbf{b}$.

Remark. The theorem of Mangasarian and Shiau is stronger than we need. The theorem says that there exist two solutions to the linear programs that are close together, when it suffices to show that their objective function values are close together. It would be interesting to see if this extra information could be exploited to yield stronger results.

3. IMPLEMENTATION IDEAS

The 1-dimensional turnpike trajectory problem is equivalent to finding a cycle in the given $\mathscr{C}$-Graph whose average speed (total progress/total cost) is maximized. Of course, that could be determined by enumerating all the cycles of the graph, but this would be inefficient. When every arc has unit cost, then the problem can be solved efficiently (time complexity: $O(\| V \| \| E \|)$ with vertex-set V and arc-set E) by an algorithm given in Karp (1978). When the arcs do not have unit costs, the problem can be solved in $O((\| V \| \| E \|)^2)$ time by an algorithm given in Megiddo (1979).

To approach the m-dimensional problem (with non-unit costs) directly through linear programming would require an enumeration of all cycles in the $\mathscr{C}$-Graph. This may be very difficult. For instance, if our $\mathscr{C}$-Graph contains a complete directed graph on n vertices, there are more than $(n - 1)!$ cycles (the number of traveling salesman tours). Furthermore, its number of vertices may be enormous (recall that the 1-dimensional jumping problem with p pieces has 2^{p-1} connected configurations). Does this mean that there is no hope of finding a reasonable solution to our problem? Not at all. Operations researchers face this sort of problem, for example, when formulating integer programs for crew scheduling of airlines. To consider all possible assignments of subsets of crew members to all possible flights would be overwhelming. Thus, one approach is to restrict attention to a manageable number of reasonable looking assignments. (In some versions, the *current optimum* can be tested for true optimality, in such a way that a negative outcome also generates a new member of the *manageable set*; cf. the use of *column generation*, below.) In a similar way, our rules for movement may suggest certain natural translations of configurations, and we might then restrict our attention to those translations (i.e., cycles in the $\mathscr{C}$-Graph). This idea can be extended to the case where we are unable to prove that the finiteness assumption is satisfied with our rules for movement. Here, we can guess at a finite number of natural-looking efficient configurations, and determine efficient translations of these, as before.

When the $\mathscr{C}$-Graph is known and of manageable size, we might attack the linear program

$$\min \sum_{j=1}^{n} c^j x_j$$

$$\text{subject to} \quad A\mathbf{x} = \mathbf{b}, \quad \mathbf{x} \geq \mathbf{0} \tag{11}$$

by reformulating it as a *minimum cost circulation problem* and solving

$$\min \sum_{j=1}^{\| E \|} \gamma^j y_j$$

$$\text{subject to} \quad P\mathbf{y} = \mathbf{b}, \quad B\mathbf{y} = \mathbf{0}, \quad \mathbf{y} \geq \mathbf{0} \tag{12}$$

where y_i is the amount of flow on the ith *arc*, from the arc-set E, γ_i and $\mathbf{p}_i$ are that arc's respective cost and progress, and B is the node-arc incidence matrix of the $\mathscr{C}$-Graph.

The correspondence between these two linear programs can be made explicit by noting that every circulation can be decomposed into cycles (with the same overall cost) and, likewise, every decomposition into cycles is a circulation. Note that a basic optimal solution to (12) uses at most $m + \| V \|$ arcs and thus can be decomposed into $m + \| V \|$ or fewer cycles in $O((m + \| V \|)^2)$ time. If this decomposition uses more than m cycles (and if we cannot reduce this number by inspection), we can then find a basic optimal solution of (11) directly, using only those columns (representing cycles) obtained in our decomposition. The minimum cost circulation problem should be most efficiently treatable by special "network with side constraints" algorithms (e.g., Chen and Saigal 1977).

Alternatively, we can employ a *column generation* scheme to solve (11) directly, as follows. If necessary, begin with an initial artificial basis consisting of m cycles, each making one unit of progress in a unit direction at enormous cost. Using the columns of our A matrix generated so far, solve the linear program to optimality. Let $\mathbf{x} \in \mathbf{R}^n$ and $\lambda \in \mathbf{R}^m$ be the primal and dual solutions to (11), restricted to the generated columns ($x_i = 0$ if the ith column has not been generated). From duality theory, $\mathbf{x}$ is an optimal solution to (11) if and only if $c^j - \lambda^T \mathbf{a}_j \geq 0, j = 1 \ldots n$. This can be determined directly on our $\mathscr{C}$-Graph by assigning the kth arc a weight of $\gamma_k - \lambda^T \mathbf{p}_k$, $k = 1 \ldots \| E \|$, and looking for a cycle of negative weight. If no such cycle exists, $\mathbf{x}$ solves (11), otherwise a negative cycle is generated, and its associated progress column and (unadjusted) cost is added to our set of generated columns. The new LP is solved, and the procedure is repeated. The negative cycle problem can be solved efficiently ($O(\| V \| \| E \|)$) by a modified shortest path algorithm (see Lawler 1976). Note that when $m \leq 3$, as will be the case with most maneuvering problems, $m \times m$ matrix inversions can be computed trivially, and a simplex method can be programmed easily without much worry about numerical issues.

We used the above procedure to solve the three-piece, two-dimensional jumping problem, restricted to the connected configurations, for all directions b. The $\mathscr{C}$-Graph has 46 nodes and 288 arcs. Starting from an artificial basis, the column generation scheme solved the problem very efficiently, generating only one superfluous column. Further algorithmic development and experimentation are in progress.

4. RESEARCH DIRECTIONS

We briefly mention some questions intended for continuation of this research.

- When is the finiteness assumption valid? Are there natural sufficient conditions that imply finiteness?
- How can we automate the construction of the $\mathscr{C}$-Graph from natural descriptions of its nodes (i.e., configurations) and arcs (i.e., legal moves)? Can this construction be usefully interwoven with the solution algorithms sketched in the preceding section?
- What happens when our configurations must stay within certain borders? Here, the space homogeneity assumption is violated, but only at the borders. It will be shown, in a subsequent paper based on Benjamin (1989), how a border-ignoring turnpike trajectory can be systematically modified to accommodate this situation.
- The main theorems indicate that the turnpike solution is almost as good as an optimal solution. When can we prove the stronger claim that there exists an optimal solution with the turnpike property?
- Clearly, it would be interesting to see how and how much these results can be generalized to $\mathbf{R}^m$ and other environments, both continuous and discrete (e.g., lattices other than $\mathbf{Z}^m$). Initial results of this type appear in Benjamin (1989).

We close by suggesting that the particular mathematical construct identified in this paper, that of "high-speed cycles in a $\mathscr{C}$-Graph," should prove generally valuable in the treatment of the optimal maneuvering problems described at the outset (at least for slowly varying environments of movement). The preceding results provide encouraging initial evidence, which we hope will be confirmed by the additional investigations outlined above.

ACKNOWLEDGMENT

This paper is based on dissertation research performed at The Johns Hopkins University, Department of Mathematical Sciences. The author is indebted to the faculty, staff, and especially to Professor Alan J. Goldman for inspiration, encouragement and support. The author also acknowledges the associate editor and the two anonymous reviewers for their valuable comments, and thanks Deena Dizengoff for her assistance with the figures. A preliminary version of this paper was awarded first prize in the George E. Nicholson student paper competition, 1988.

REFERENCES

Auslander, J., A. T. Benjamin and D. Wilkerson. 1988. Speed of Light Configurations in n-Dimensional Jumping Games. Unpublished manuscript.

Belur, A., and A. J. Goldman. 1985. Solitaire Chinese Checkers. Independent Research Report, Mathematical Sciences Department, The Johns Hopkins University, Baltimore.

Benjamin, A. T. 1987. Turnpike Theorems in Large Structured Networks. Dissertation Research Proposal, Mathematical Sciences Department, The Johns Hopkins University, Baltimore.

Benjamin, A. T. 1989. Turnpike Structures for Optimal Maneuvers. Ph.D. Dissertation, The Johns Hopkins University, Baltimore.

Cass, D. 1966. Optimum Growth in an Aggregative Model of Capital Accumulation: A Turnpike Theorem. *Econometrica* **34,** 833–850.

Castells, C., and A. J. Goldman. 1983. Analysis of Some Jumping Games. Independent Research Report, Mathematical Sciences Department, The Johns Hopkins University, Baltimore.

Chen, S., and R. Saigal. 1977. A Primal Algorithm for Solving a Capacitated Network Flow Problem With Additional Linear Constraints. *Networks* **7,** 59–79.

Chrétienne, P. 1984. Chemins Extrémaux d'un Graphe Doublement Valué. *R.A.I.R.O. Recherche Opérationelle/Operations Research* **18,** 221–245.

Garey, M. R., and D. S. Johnson. 1978. *Computers and Intractability.* W. H. Freeman & Co., New York.

Gilmore, P. C., and R. E. Gomory. 1966. The Theory and Computation of Knapsack Functions. *Opns. Res.* **14,** 1045–1074.

KARP, R. M. 1978. A Characterization of the Minimum Cycle Mean in a Digraph. *Discrete Math.* **23,** 309–311.

LAWLER, E. L. 1976. *Combinatorial Optimization: Networks and Matroids.* Holt, Rinehart & Winston, New York.

MANGASARIAN, O. L., AND T.-H. SHIAU. 1987. Lipschitz Continuity of Solutions of Linear Inequalities, Programs and Complementarity Problems. *SIAM J. Contr. Optim.* **25,** 583–595.

MCKENZIE, L. 1986. Optimal Economic Growth, Turnpike Theorems and Comparative Dynamics. In *Handbook of Mathematical Economics*, Vol. III, K. J. Arrow and M. D. Intrilligator (eds.). Elsevier Science Publishers, New York, 1281–1355.

MEGIDDO, N. 1979. Combinatorial Optimization With Rational Objective Functions. *Math. Opns. Res.* **4,** 414–424.

SHAPIRO, J. F. 1968. Turnpike Planning Horizons for a Markovian Decision Model. *Mgmt. Sci.* **14,** 319–341.

SHAPIRO, J. F., AND H. M. WAGNER. 1967. A Finite Renewal Algorithm for the Knapsack and Turnpike Models. *Opns. Res.* **15,** 319–341.