

A decorative geometric shape in the top-left corner, composed of several interconnected polygons in shades of teal and orange.

2016/20

Optimum Turn-Restricted Paths, Nested Compatibility, and Optimum Convex Polygons

MAURICE QUEYRANNE AND LAURENCE A. WOLSEY

A large, stylized number '50' in a golden-yellow color, rendered with thick, rounded lines. It is positioned on the right side of the page, partially overlapping the '50 YEARS OF' text and the 'CORE' text.

50 YEARS OF
CORE
DISCUSSION PAPERS

CORE

Voie du Roman Pays 34, L1.03.01

Tel (32 10) 47 43 04

Fax (32 10) 47 43 01

Email: immaq-library@uclouvain.be

<http://www.uclouvain.be/en-44508.html>

CORE DISCUSSION PAPER
2016/20

Optimum Turn-Restricted Paths, Nested Compatibility, and Optimum Convex Polygons

Maurice Queyranne^{1,2} and Laurence A. Wolsey²

¹Sauder School of Business, University of British Columbia, Vancouver, Canada

²Center for Operations Research and Econometrics (CORE), Université
catholique de Louvain, Louvain-la-Neuve, Belgium

June 2, 2016

Abstract

We consider two apparently unrelated classes of combinatorial and geometric optimization problems. First, we give compact extended formulations, i.e., polynomial-size linear programming formulations with integer optima, for optimum path problems with turn restrictions satisfying a *nested compatibility* condition in acyclic digraphs. We then apply these results to *optimum convex polygon* problems in the plane, by interpreting certain Dynamic Programming algorithms as sequences of optimum turn-restricted path problems with nested compatibility in acyclic digraphs. As a result, we derive compact extended formulations for these geometric problems as well.

Keywords: shortest paths, acyclic networks, turn restrictions, convex polygons, convex subsets, extended formulation, dynamic programming

AMS 2010 Mathematics Subject Classification: 90C27, 90C35, 90C39, 90C57

1 Introduction

We consider two apparently unrelated classes of combinatorial optimization problems.

(i) *Optimum turn-restricted paths in networks*: In these problems we seek optimum paths in a network with given edge costs or revenues, subject to turn penalties or restrictions. *Turn penalties* $t(i, j, k)$ are incurred if the path reaches node j from node i and then immediately goes (“turns”) to node k ; in other words, if edges (i, j) and (j, k) immediately follow one another in the path. Such turn penalties occur frequently in the modeling of urban and freeway traffic, (e.g., [4]) and for route planning in transportation networks, (e.g., [2]). *Turn restrictions* are hard constraints which prohibit a path from containing such a subpath (i, j, k) of successive nodes. Such turn restrictions (which may also be enforced using very large or infinite turn penalties $t(i, j, k)$) may arise from traffic regulations, (e.g., [10]) or from physical constraints in motion planning, (e.g., [13]). An *optimum* path is a feasible path (i.e., satisfying all turn restrictions) with either minimum total edge costs and turn penalties, or (depending on the context) maximum total revenue net of turn penalties.

(ii) *Optimum convex polygons*¹: In these problems we seek a convex polygon with vertices from a given set of points in the plane, which satisfies certain restrictions and is optimum in a suitable sense. For example, in the *Optimum Convex Subset Problem* each given point has a weight (of arbitrary sign) and we seek a convex polygon which maximizes the total weight of all the given points it contains. This and related problems arise in applications such as statistical clustering and pattern recognition (see [8]), data mining and data compression (see [3]); see Section 4 below for details and further references.

We connect these two problem classes by interpreting the Dynamic Programming algorithm of Epstein et al. [8] and its modified version in [3] for optimum convex polygon problems as sequences of optimum turn-restricted (s, t) -path problems with nested compatibility in certain acyclic digraphs. We use these connections and derive *compact extended formulations*, i.e., tight linear programming formulations using polynomially bounded num-

¹The first author would like to thank the following colleagues for useful discussions and suggestions on optimum convex polygon and convex subset problems – in approximate chronological order: Marcos Goycoolea (Universidad Adolfo Ibáñez), Jeremy Barbay (Universidad de Chile), Miguel Constantino (Universidade de Lisboa), Marek Chrobak (University of California, Riverside), Keno Merckx and Jean-Paul Doignon (Université Libre de Bruxelles).

bers of additional variables and linear inequality constraints (see, e.g., [5]).

In Section 2 we describe an optimum (s, t) -path problem with turn penalties and restrictions in an acyclic digraph. We recall Caldwell's [4] reduction to an unconstrained optimum path problem in an extended network, and a resulting, well-known compact extended formulation for this problem. In Section 3 we introduce a "nested compatibility" condition on the turn restrictions, which allows a reduction in the size of the extended formulation. In Section 4 we interpret the DP algorithm of [8], as adapted by [3] for optimum convex polygon problems, as a sequence of optimum (s, t) -path problems with turn restrictions and nested compatibility in certain acyclic digraphs. Combining with the results in Section 3 and Balas's Disjunctive Programming principle [1] we derive a compact extended formulation for these optimum convex polygon problems. We conclude with a few remarks.

2 Optimum Paths with Turn Penalties and Restrictions in an Acyclic Digraph

In this preliminary section, and in view of the geometric applications in Section 4 below, we consider the optimum (s, t) -path problem with turn penalties and restrictions in an acyclic digraph. We recall how Caldwell's arc-based representation [4], which reduces the problem to an ordinary optimum path problem in a larger acyclic network, naturally leads to a compact extended formulation.

We are given an acyclic digraph $D = (V, A)$ with node set $\{0, 1, \dots, n\}$, arc set $A \subseteq \{(p, q) : 0 \leq p < q \leq n\}$ (i.e., we assume the given node numbering is consistent with the arc directions), and arc revenues $c(p, q)$ of arbitrary sign. The turn penalties and restrictions are given as a list L of *compatible triples*,

$$L = \{(p, q, r) : \text{arc } (q, r) \in A \text{ may follow arc } (p, q) \in A \text{ in the path}\},$$

together with turn penalties $t(p, q, r)$ for all $(p, q, r) \in L$. This representation is similar to the *turn tables* representation used in route planning [6, 9]. A path $P = (p_0, p_1, \dots, p_k)$ in D is *feasible* if every three successive nodes p_{j-1}, p_j, p_{j+1} form a compatible triple ($j = 1, \dots, k - 1$), and its *net value*

$$v(P) = \sum_{j=1}^k c(p_{j-1}, p_j) - \sum_{j=1}^{k-1} t(p_{j-1}, p_j, p_{j+1})$$

is the total revenue of the arcs it uses minus the total cost penalty it incurs.

As compatibility constraints only restrict pairs of successive arcs, a (forward) Dynamic Programming approach to finding a feasible $(0, n)$ -path with maximum net value uses as DP states the last edge (p, q) of the feasible $(0, q)$ -paths. Thus, letting the DP functional $G(p, q)$ denote the maximum net value of a feasible $(0, q)$ -path ending with edge (p, q) , the resulting DP recursion is:

$$\begin{aligned} G(0, p) &= c(0, p) && \text{for } (0, p) \in A; \\ G(q, r) &= \max_{p:(p,q,r) \in L} (G(p, q) + c(q, r) - t(p, q, r)) && \text{for } q \geq 1 \text{ and } (q, r) \in A; \\ z &= \max_{q:(q,n) \in A} G(q, n) \end{aligned}$$

giving the maximum net value z of a feasible $(0, n)$ -path. The running time is $O(n^3)$.

This can be interpreted as a standard recursion for a “longest”, i.e., maximum value, path problem in a larger acyclic digraph $\overline{D} = (\overline{V}, \overline{E})$ with nodes $\overline{V} = \{ij \text{ and } \overline{ij} : (i, j) \in A\}$, and two types of arcs $\overline{E} = E_1 \cup E_2$ where the arcs $E_1 = \{(pq, \overline{pq}) \text{ for all } (p, q) \in A\}$ handle the revenues $c(p, q)$ and the compatibility arcs $E_2 = \{(\overline{pq}, qr) : (p, q, r) \in L\}$ model the turn restrictions and penalties [4]. There are revenues $c(p, q)$ on all arcs $pq \in E_1$ and $-t(p, q, r)$ on all arcs $(\overline{pq}, qr) \in E_2$. We seek a path Q of maximum total value among all paths in \overline{D} from any node $0p$ to any node \overline{qn} .²

It is well known that such a problem can be formulated and solved as a linear program. Specifically we introduce indicator variables:

$$\begin{aligned} y_{p,q} &= 1 && \text{if path } Q \text{ contains the } E_1\text{-arc } (pq, \overline{pq}), \text{ and } 0 \text{ otherwise;} \\ z_{p,q,r} &= 1 && \text{if path } Q \text{ contains the } E_2\text{-arc } (\overline{pq}, qr), \text{ and } 0 \text{ otherwise.} \end{aligned}$$

The resulting formulation is:

$$\max_{y,z} \sum_{(p,q) \in A} c(p, q) y_{p,q} - \sum_{(p,q,r) \in L} t(p, q, r) z_{p,q,r}$$

²If needed, we could add an origin 0 and a destination n connected to all $0p$ nodes, and from all \overline{qn} nodes, respectively, with zero-revenue arcs, so we seek a maximum value $(0, n)$ -path in the resulting expanded network. This is, however, unnecessary for our purposes.

subject to

$$\sum_{p:(p,q,r) \in L} z_{p,q,r} = y_{q,r} \quad \text{for } 1 \leq q \text{ and } (q,r) \in A; \quad (1)$$

$$\sum_{s:(q,r,s) \in L} z_{q,r,s} = y_{q,r} \quad \text{for } (q,r) \in A \text{ and } r \leq n-1; \quad (2)$$

$$\sum_{p=1}^n y_{0,p} = 1; \quad (3)$$

$$y, z \geq 0. \quad (4)$$

Constraints (1) and (2) are flow conservation constraints at nodes qr and \overline{qr} respectively, and equation (3) ensures that the path starts in one of the nodes $0p$. This formulation has $|A| + |L| = O(n^3)$ nonnegative variables and about $2|A| = O(n^2)$ linear constraints (1)–(3).

Let $P(1)$ denote the polytope defined by (1)–(4).

3 Nested Compatibility

From now on we will assume that the set L of compatible triples satisfies the *Nested Compatibility* condition (i)–(iii) below. Recall that in an acyclic digraph $D = (V, A)$, node p is a *predecessor* of node q if $(p, q) \in A$, and node r is a *successor* of q if $(q, r) \in A$; let $\text{Pred}(q)$ and $\text{Succ}(q)$ denote the set of predecessors and successors of q , respectively. Thus, we now assume that, for each node $q \in \{1, \dots, n-1\}$,

- (i) $(0, q, r) \in L$ for every successor $r \in \text{Succ}(q)$;
- (ii) $(p, q, n) \in L$ for every predecessor $p \in \text{Pred}(q)$; and
- (iii) there exists an ordering $(p_q^0, \dots, p_q^{\pi(q)})$ of $\text{Pred}(q)$ and an ordering $(r_q^{\pi(q)+1}, \dots, r_q^{\sigma(q)})$ of $\text{Succ}(q)$ such that if $(p_q^i, q, r_q^j) \in L$ then $(p_q^h, q, r_q^k) \in L$ for all $h \leq i$ and $k \geq j$.

Remark: In the language of [7], condition (iii) means that the compatibility relation is “realizable” with respect to the specified orderings of the predecessors and successors of q ; see [7] for further information on connections with Guttman scales, interval orders, and related notions.

Lemma 1. *Under the Nested Compatibility condition, for each node q there exists a partition $A_q^1, \dots, A_q^{K_q}$ of its predecessors and a partition $B_q^1, \dots, B_q^{K_q}$*

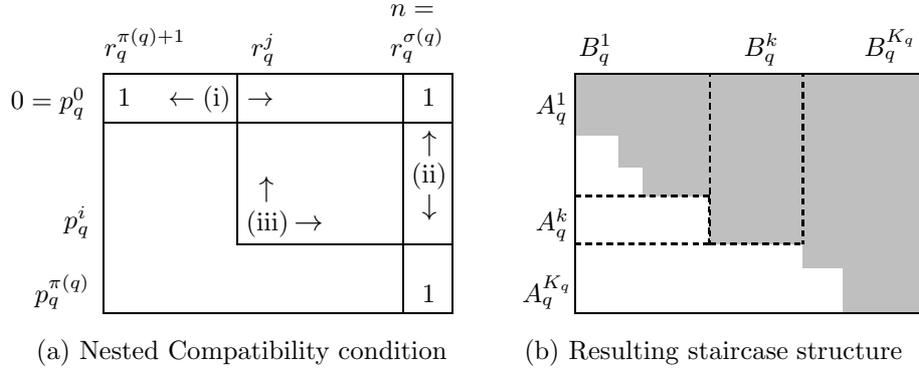


Figure 1: Local compatibility graph $G_{L,q}$ (adjacency matrix)

of its successors such that $(p, q, r) \in L$ if and only if $(p, r) \in \bigcup_{k=1}^{K_q} (A_q^k \times \bigcup_{\ell \geq k} B_q^\ell)$.

Sketch of proof. The local compatibility graph $G_{L,q}$ at node q is the bipartite graph $G_{L,q} = (\text{Pred}(q) \cup \text{Succ}(q), A_{L,q})$ where $(p, r) \in A_{L,q}$ iff $(p, q, r) \in L$. Figure 1(a) pictures the effect of the Nested Compatibility conditions (i)–(iii) on the adjacency matrix of the local compatibility graph $G_{L,q}$. It follows that this adjacency matrix has a staircase structure, the sets A_q^k represent the vertical steps and B_q^k the horizontal steps, see Figure 1(b). \square

A formal proof of Lemma 1 is included, for completeness, in the Appendix.

As we are aiming for a formulation with $O(n^2)$ variables, we assume that all turn penalties are zero, so we may ignore them in the rest of this paper. The resulting problem is called the Optimum Path Problem with Nested Compatibilities (OPP-NC).

We will now show that under the Nested Compatibility condition,

- (i) by modifying the arcs in the digraph \bar{D} and contracting some nodes, it is possible to construct an equivalent problem on an acyclic digraph \tilde{D} having only $O(n^2)$ arcs (Section 3.1); and
- (ii) by further eliminating the variables on certain arcs, we obtain a linear program that solves the original problem and defines the convex hull of L -compatible solutions in the space of the arc variables y_{pq} only (Section 3.2).

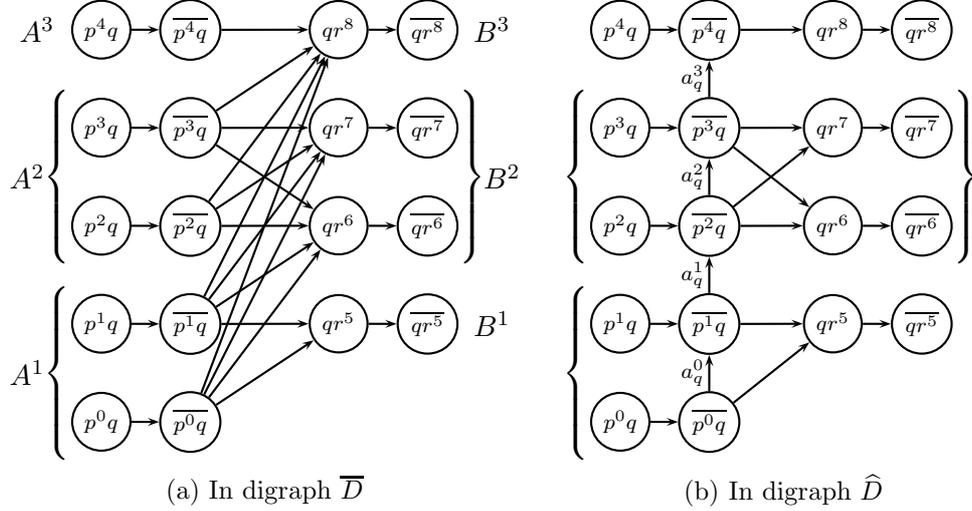


Figure 2: Arcs associated with node q

3.1 Reducing the number of arcs from $O(n^3)$ to $O(n^2)$

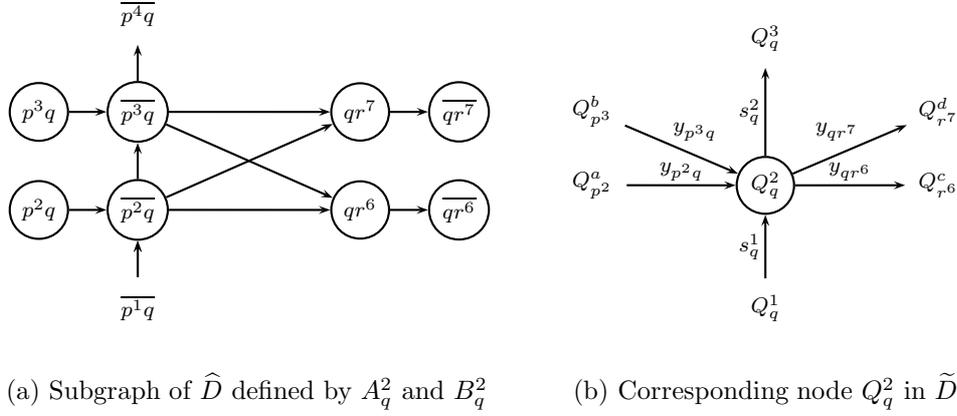
In Figure 2(a), where all subscripts q have been omitted for clarity, we show the structure of \bar{D} in the neighbourhood of a node q (note that the inner subgraph is isomorphic to the local compatibility graph $G_{L,q}$ defined in the proof of Lemma 1). There are arcs $(pq, \overline{p\bar{q}})$ entering nodes $\overline{p\bar{q}}$, arcs $(qr, \overline{q\bar{r}})$ leaving nodes qr and arcs $(\overline{p^i q}, qr^j)$ if $(p^i, r^j) \in \bigcup_k (A_q^k \times \bigcup_{\ell \geq k} B_q^\ell)$ (i.e., if $(p^i, r^j) \in A_{L,q}$).

We now use the Nested Compatibility condition and modify the digraph in such a way that the set of feasible paths and their values in the new acyclic digraph \hat{D} are unchanged:

- i) Add new arcs $a_q^i = (\overline{p^i q}, \overline{p^{i+1} q})$ for $i = 1, \dots, q-1$.
- ii) Remove all arcs $(\overline{p\bar{q}}, qr)$ with $(p, r) \in \bigcup_k (A_q^k \times \bigcup_{\ell > k} B_q^\ell)$.

The resulting modified subgraph associated with node q is shown in Figure 2(b). The next lemma justifies this modification. Its proof is straightforward and thus omitted.

Lemma 2. *There is a path from $\overline{p^i q}$ to qr^j in the modified digraph \hat{D} if and only if there is an arc $(\overline{p^i q}, qr^j) \in \bar{E}$ (i.e. $p^i qr^j \in L$). In particular for every $p \in A_q^k$ and $r \in B_q^k$ for some k , then there is an arc $(\overline{p\bar{q}}, qr)$ in \hat{D} .*



(a) Subgraph of \widehat{D} defined by A_q^2 and B_q^2 (b) Corresponding node Q_q^2 in \widetilde{D}

Figure 3: Reducing the modified digraph \widehat{D}

Consider now the subgraph of \widehat{D} consisting of nodes \overline{pq} for $p \in A_q^k$ and nodes qr for $r \in B_q^k$ for some k , see Figure 3(a). Any flow entering by an arc (pq, \overline{pq}) or the new arc $a_q^i = (\overline{p_q^i q}, \overline{p_q^{i+1} q})$ (where $p_q^i \in A_q^{k-1}$ and $p_q^{i+1} \in A_q^k$) can reach any leaving arc (qr, \overline{qr}) or the new leaving arc $a_q^j = (\overline{p_q^j q}, \overline{p_q^{j+1} q})$ (where $p_q^j \in A_q^j$ and $p_q^{j+1} \in A_q^{j+1}$). Therefore we may:

iii) Contract this subgraph into a single node Q_q^k .

See Figure 3(b). Note that now the arcs entering Q_q^k correspond to arcs pq for $p \in A_q^k$, plus a new *slack arc* from Q_q^{k-1} to Q_q^k if $k > 1$; and the arcs leaving Q_q^k to arcs qr for $r \in B_q^k$, plus the slack arc from Q_q^k to Q_q^{k+1} if $k < K_q$.

Let \widetilde{D} denote the resulting acyclic digraph. Note that \widetilde{D} has $O(n^2)$ nodes and $O(n^2)$ arcs. An optimum $(0, n)$ -path can thus be found in $O(n^2)$ time, using the following recursion, where $G^k(q)$ denotes the maximum net value of a feasible $(0, q)$ -path ending with arc (p_q^k, q) in D :

$$G^1(0) = 0; \tag{5}$$

$$G^k(q) = \max \left\{ \max_{p \in A_q^k} \left[G^{\ell(p,q)}(p) + c(p, q) \right], G^{k-1}(q) \right\} \quad \text{for } 1 \leq k \leq K_q \text{ and} \\ 1 \leq q \leq n-1; \tag{6}$$

$$z = \max_{q:(q,n) \in A} \max_{k=1, \dots, K_q} G^k(q) + c(q, n) \tag{7}$$

where $\ell(p, q)$ is the unique integer ℓ for which $q \in B_p^\ell$. (For example in Figure 3(b), a stands for $\ell(p^2, q)$ and b for $\ell(p^3, q)$.) Recursion (5)–(7) generalizes the DP recursions given in [3] for geometric problems to optimum path problems with nested compatibilities; see Section 4 below.

To every slack arc from Q_q^k to Q_q^{k+1} ($k = 1, \dots, K_q - 1$), associate the variable s_q^k . The resulting formulation is

$$\max_{y, s} \sum_{(p, q) \in A} c(p, q) y_{p, q} \quad (8)$$

subject to

$$\sum_{p \in A_q^k} y_{p, q} + s_q^{k-1} = \sum_{r \in B_q^k} y_{q, r} + s_q^k \quad \text{for } 1 \leq k \leq K_q \text{ and } 1 \leq q \leq n-1; \quad (9)$$

$$\sum_{p=1}^n y_{0, p} = 1; \quad (10)$$

$$y, s \geq 0; \quad (11)$$

$$s_q^0 = s_q^{K_q} = 0 \quad \text{for } 1 \leq q \leq n-1. \quad (12)$$

Let $P(2)$ be the polytope defined by (9)–(12).

3.2 Formulation in the original space with additional constraints

We now eliminate the s_q^k variables from the formulation (9)–(12) using projection. We first sum the constraints (9) over $\{k, k+1, \dots, K_q\}$ and, using $s_q^{K_q} = 0$, we obtain the following set of inequalities, equivalent to (9):

$$\sum_{\ell \geq k} \sum_{p \in A_q^\ell} y_{p, q} + s_q^{k-1} = \sum_{\ell \geq k} \sum_{r \in B_q^\ell} y_{q, r} \quad \text{for } 1 \leq k \leq K_q \text{ and } 1 \leq q \leq n-1. \quad (13)$$

Each nonnegative variable s_q^k now appears just once in (13), and may thus be eliminated. This leads to the final formulation:

$$\sum_{p:(p,q) \in A} y_{p,q} = \sum_{r:(q,r) \in A} y_{q,r} \quad \text{for } 1 \leq q \leq n-1; \quad (14)$$

$$\sum_{\ell \geq k} \sum_{p \in A_q^\ell} y_{p,q} \leq \sum_{\ell \geq k} \sum_{r \in B_q^\ell} y_{q,r} \quad \text{for } 2 \leq k \leq K_q \text{ and } 1 \leq q \leq n-1; \quad (15)$$

$$\sum_{p=1}^n y_{0,p} = 1; \quad (16)$$

$$y \geq 0 \quad (17)$$

where the “balance equations” (14) follow from (13) with $k = 1$ and $s_q^0 = 0$, since $\bigcup_{k \geq 1} A_q^k = \text{Pred}(q)$ and $\bigcup_{\ell \geq 1} B_q^\ell = \text{Succ}(q)$. Objective (8) and constraints (14)–(17) define a linear program just involving the arc variables of the original digraph D . It can also be seen as a (fractional) shortest path problem with additional linear constraints that model the nested incompatibilities. Let $P(3)$ be the polytope defined by (14)–(17).

As the path formulations lead to integral polyhedra and the projections of integral polyhedra are integral, we obtain our main combinatorial result, where “ $y \in P(j)$ ” stands for “ $y \in \text{proj}_y P(j)$ ” (i.e., $(y, z) \in P(1)$ for some z when $j = 1$; $(y, s) \in P(2)$ for some s when $j = 2$; and of course $y \in P(3)$ when $j = 3$):

Theorem 3. *For $j = 1, 2, 3$, the linear program*

$$\max_{y \in P(j)} \sum_{(p,q) \in A} c(p, q) y_{pq}$$

solves the optimum path problem under the Nested Compatibility condition. In particular, $P(3)$ is a tight linear programming formulation with $O(n^2)$ constraints in the original arc variables $y \in \mathbb{R}^A$, and its constraints (14)–(17) define the convex hull of the characteristic vectors $y^P \in \{0, 1\}^A$ of the arc sets of the feasible $(0, n)$ -paths P .

4 Geometric Applications

Here we consider certain classes of optimum convex polygon problems, starting in Subsection 4.1 with the minimum weight convex subset problem. In all these problems we are given a set X of m distinct points $x^i = (x_1^i, x_2^i)$

($i = 1, \dots, m$) in the plane, with certain attributes (weight, color, etc.). In contrast with much work in computational geometry, we do not assume the given points to be in “general position”. In particular, we allow three (or more) given points to be aligned, i.e., we allow some given points to be in the convex hull of *two* other given points; such a possibility is also discussed in the last paragraph of Section 4 in [8]. Indeed, interesting (and nontrivial) instances of these problems, such as those involving grid points, do not seem to be easily reducible to instances with points in general position.

Korte and Lovász [11] observe that certain families of complements of convex sets form *antimatroids*, a subclass of *greedoids*, structures on which certain optimization problems may be solved by greedy algorithms [12]. They present some algorithmic and polyhedral results for antimatroids.

We shall derive compact extended formulations for these geometric problems after observing that they may be solved as sequences of optimum (s, t) -path problems with nested compatibility, treated in Section 3.

4.1 The Minimum Weight Convex Subset Problem

Recall from the Introduction that in the optimum convex subset problem (OCSPP) each point $x^i \in X$ has a given weight f_i of arbitrary sign, and we seek a convex polygon C with vertices in X such that the total weight of all the points of X that are in C is maximized.

A standard approach is to divide the problem into m subproblems: the t^{th} subproblem involves the subset of points

$$X^t = \{x^i \in X : x_2^i > x_2^t \text{ or } (x_2^i = x_2^t \text{ and } x_1^i > x_1^t)\},$$

i.e., all points in X that lie above x^t or level and to the right of x^t ; in other words, x^t is the leftmost of the bottommost vertices of the convex polygon. In such case, we say that the polygon is *based at* x^t , and x^t is its *base*. Let $n_t = |X^t| + 1$.

We shall construct an acyclic directed network (V^t, A^t) in which $(0, n_t)$ -paths are related to edge sequences that define the boundary of convex polygons based at x^t . For this, we define the node set V^t of our network as follows. Take the point x^t to be node 0 and arrange the points in X^t in clockwise order as seen from x^t , where points aligned with x^t are arranged in arbitrary order, giving the nodes $1, \dots, n_t - 1$. Finally take node n_t to also correspond to x^t ; see Figure 4. Hereafter we shall identify points in $X^t \cup \{x^t\}$ with the corresponding nodes in V^t .

We associate an arc $(p, q) \in A^t$ with every ordered pair $p < q$ of points which may define an edge of a convex polygon C based at x^t ; in other words,

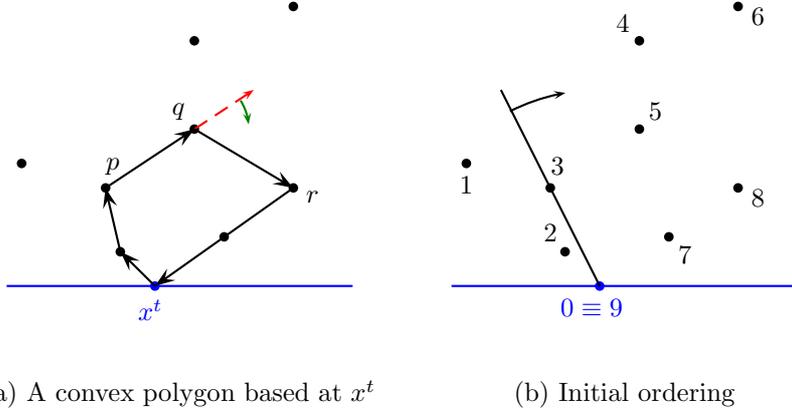


Figure 4: A 9-point instance.

such that vertex p appears immediately before vertex q as the boundary of C is traversed clockwise starting from node 0 and ending in node n_t . For the present problem we have

$$A^t = \{(p, q) \in V^t \times V^t : p < q \text{ and } 0, p, q, \text{ if all distinct, are not aligned}\}.$$

Note that A^t includes all arcs $(0, p)$ and (p, n_t) for $1 \leq p \leq n_t - 1$, as well the arc $(0, n_t)$ for the degenerate case where the polygon $C = \{0\}$. As a result, every convex polygon C based at x^t defines a $(0, n_t)$ -path in the network, satisfying additional restrictions due to the convexity requirement on C .

To enforce the convexity of polygon C , note that as we travel from vertex p to vertex q to vertex r in clockwise order around the boundary of C , we may only make a “right turn” at q . Thus define the compatibility list

$$L^t = \{pqr : (p, q) \text{ and } (q, r) \in A^t; p, q, r \text{ are not aligned; and } 0 \in \langle p, q, r \rangle\}$$

where $0 \in \langle p, q, r \rangle$ means that node 0 is in the convex cone with apex q induced by angle $\angle pqr$; see Figure 4(a). It follows that a $(0, n_t)$ -path is L^t -compatible if and only if it defines the (clockwise) boundary of a convex polygon based at x^t .

Now observe that L^t satisfies the inclusion condition. To see this, for each node $q \in \{1, \dots, n_t - 1\}$ construct the sets $A_q^{t,k}$ and $B_q^{t,k}$ as follows. Take a line through q , rotate it clockwise and list the neighbours of q in order they are hit, starting from $p_q^0 = 0$ and terminating with $r_q^{\sigma(q)} = n_t$. Let the resulting sequence of node sets be $A_q^{t,1} B_q^{t,1} \dots A_q^{t,k} B_q^{t,k} \dots A_q^{t,K_q^t} B_q^{t,K_q^t}$

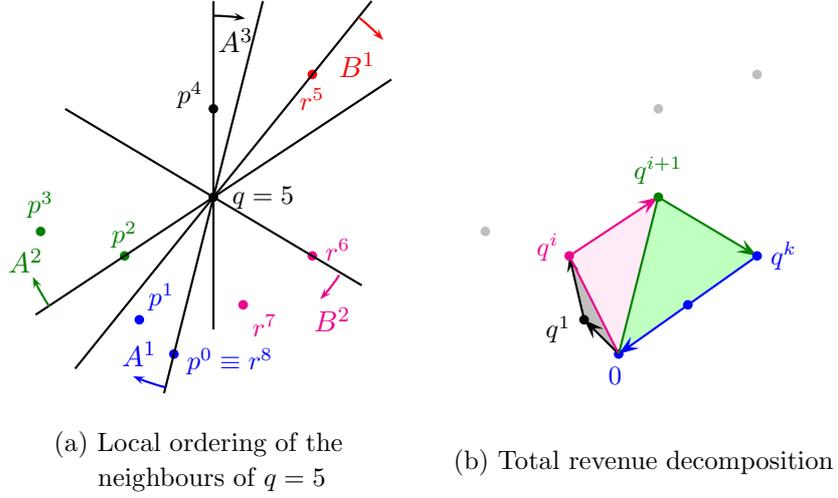


Figure 5: A 9-point instance (continued).

where $A_q^{t,k}$ are the neighbours of q in the k^{th} subsequence of predecessors, and $B_q^{t,\ell}$ are the neighbours in the ℓ^{th} subsequence of successors.

As an illustration, Figure 4(b) shows an instance in which the $n = 9$ nodes of X^t are labeled in clockwise order, whereas in Figure 5(a) (where subscripts q and superscripts t are omitted for simplicity) they are ordered relative to node $q = 5$. The nodes are hit in the order $p^0 p^1 r^5 p^2 p^3 r^6 r^7 p^4 r^8 = 026318749$ leading to the sets $A_5^{t,1} = \{p^0, p^1\} = \{0, 2\}$, $B_5^{t,1} = \{r^5\} = \{6\}$, $A_5^{t,2} = \{3, 1\}$, $B_5^{t,2} = \{8, 7\}$, $A_5^{t,3} = \{4\}$ and $B_5^{t,3} = \{9\}$. The neighborhood of this node $q = 5$ in the expanded acyclic digraph \overline{D}^t is the one shown in Figure 2(a) above.

In order to complete our turn-restricted shortest path formulation we need to define the revenues of the arcs in the network (V^t, A^t) , so that the value of every feasible $(0, n_t)$ -path equals the total weight of the corresponding convex polygon. For every candidate edge $(p, r) \in A^t$ with $0 < p$ and $r < n_t$ let

$$T^t(p, r) = \text{conv}(x^t, p, r) \setminus \text{conv}(r, x^t)$$

denote the *right-open triangle* defined by vertices $0, p, r$ and with its rightmost edge $E^t(r) = \text{conv}(r, x^t)$ removed. As observed in [8] the total weight of a convex polygon with boundary³ $0 q^1 \dots, q^k n_t$ is the total weight, for $i = 1, \dots, k-1$, of the right-open triangles $T^t(q^i, q^{i+1})$, their rightmost edges being removed to avoid double counting, plus the weight of the closing edge $E^t(q^k)$; see Figure 5(b). Thus, letting as usual $f(S) = \sum_{s \in S} f_s$ denote the

³Recall that both nodes 0 and n_t represent the point x^t .

total weight of any set S , define the revenue of each arc $(p, r) \in A^t$ as

$$c^t(p, r) = \begin{cases} 0 & \text{if } 0 = p \text{ and } r < n_t; \\ f(T^t(p, q) \cap X^t) & \text{if } 0 < p \text{ and } r < n_t; \\ f(E^t(p) \cap X^t) & \text{otherwise.} \end{cases}$$

From the preceding discussion it follows that the subproblem associated with point x^t may be formulated as an optimum turn-restricted path problem with nested compatibilities in the acyclic digraph D^t just constructed. It can thus be solved by the recursion (5)–(7), which gives the DP algorithm of [8] as adapted in [3] to the case where the number of edges of the polygon is not prescribed.

Remark. Eppstein et al. [8, Section 2] also give an efficient data structure that can be set up, as a preprocessing step for the whole problem, in $O(m^2)$ time and $O(m^2)$ space, and allows querying each revenue $c^t(p, r)$ in constant time. As a result, each subproblem may be solved in $O(m^2)$ time. Choosing the best subproblem solution then solves the overall Optimum Convex Polygon problem in $O(m^3)$ time [3].

For each x^t it follows from Section 3 that, using indicator variable

$$y_{pq}^t = 1 \text{ if } pq \text{ is an edge of the polygon, and } 0 \text{ otherwise,}$$

the subproblem defined by x^t may be solved as a linear program

$$\max_{y^t \in P^t(j)} \sum_{(p,q) \in A^t} c^t(p, q) y_{pq}^t$$

where $P^t(j)$ is the polytope $P(j)$ associated with the t^{th} subproblem and $j = 1, 2$ or 3 .

We may also formulate this subproblem with additional indicator variables

$$z_q^t = 1 \text{ if point } q \in X^t \text{ is in the polygon, and } 0 \text{ otherwise,}$$

by using the definitional equations,

$$z_q^t = \sum_{(p,r) : q \in T^t(p,r)} y_{pr}^t + \sum_{r : q \in E^t(r)} y_{r,n_t}^t \quad \text{for all } q \in X^t, \quad (18)$$

which correspond to the definition of the arc revenues $c^t(p, r)$ above. These definitional equations do not affect the integrality of the formulation, and the

bounds $0 \leq z^t \leq 1$ are superfluous. Thus for any $j = 1, 2, 3$, the subproblem defined by x^t may be solved as the linear program

$$\max_{y^t, z^t} \left\{ \sum_{q \in X^t} f_q z_q^t : y^t \in P^t(j) \text{ and (18)} \right\} + f_{x^t} .$$

Finally, since the best solution among all subproblems yields an optimum convex polygon, we may apply Balas's Disjunctive Programming principle [1] and derive a compact extended formulation. For this, we use the indicator variable

$$w^t = \begin{cases} 1 & \text{if } x^t \text{ is the base vertex of the polygon;} \\ 0 & \text{otherwise,} \end{cases}$$

and obtain the main geometric result of this paper:

Theorem 4. *For any instance (X, f) of the Optimum Convex Polygon problem with at least one nonnegative weight (i.e., with $f_i \geq 0$ for some $x^i \in X$), the following linear program has an optimum solution which is integer:*

$$\max_{w, y, z} \sum_{t=1}^n \left(f_{x^t} w^t + \sum_{q \in X^t} f_q z_q^t \right) \quad (19)$$

subject to

$$\sum_{x^t \in X} w^t = 1; \quad (20)$$

$$y_{0, n_t}^t + \sum_{p \in X^t} y_{0, p}^t = w^t \quad \text{for } x^t \in X; \quad (21)$$

$$\sum_{p: (p, q) \in A^t} y_{p, q}^t = \sum_{r: (q, r) \in A^t} y_{q, r}^t \quad \text{for } q \in X^t, x^t \in X; \quad (22)$$

$$\sum_{\ell \geq k} \sum_{p \in A_q^{t, \ell}} y_{p, q}^t \leq \sum_{\ell \geq k} \sum_{r \in B_q^{t, \ell}} y_{q, r}^t \quad \text{for } 2 \leq k \leq K_q^t, q \in X^t$$

$$\text{and } x^t \in X; \quad (23)$$

$$z_q^t = \sum_{(p, r): q \in T^t(p, r)} y_{pr}^t + \sum_{r: q \in E^t(r)} y_{r, n_t+1}^t \quad \text{for } q \in X^t, x^t \in X; \quad (24)$$

$$w, y, z \geq 0. \quad (25)$$

This LP formulation has $O(m^3)$ variables and $O(m^3)$ constraints.

4.2 Decomposable Objectives and Constraints

Theorem 4 extends to any optimum convex polygon problem with an objective that can be expressed as a linear combination of the base-vertex variables w , the edge variables y and the point-polygon membership variables z , i.e.,

$$\max_{w,y,z} \sum_{t=1}^n \left(\omega^t w^t + \sum_{(p,q) \in A^t} \eta_{p,q}^t y_{p,q}^t + \sum_{q \in X^t} \zeta_q^t z_q^t \right). \quad (26)$$

This applies, in particular, to such “decomposable” objectives as

- maximum (or minimum) area, by letting $\eta_{p,q}^t$ denote the area of the triangle $\text{conv}(x^t, p, q)$ (or its negative), and all other objective coefficients in (26) equal to zero; and
- perimeter, by similarly letting $\eta_{p,q}^t$ denote the length of the line segment $\text{conv}(p, q)$.

Such problems are discussed in [8, 3] in the context of a generalization of their DP recursions to so-called *decomposable and monotone weight functions*. Here we only consider objectives that can be formulated as linear programs without losing the integrality property of (20)–(25).

Similarly, a constraint on convex polygon C (with base x^t) is *decomposable* if it is satisfied by C if and only if it is satisfied by every edge of C , or by every triangle $\text{conv}(x^t, p, q)$ defined by an edge pq of C ($p \neq x^t \neq q$). For example,

- C is an *empty* convex polygon [8] if there are no points of X *inside* C . Since we do not require the points in X to be in general position, “inside C ” may be interpreted as either “in the interior of C ” (allowing points of X on the edges of C), or “in C except for its vertices”. Either interpretation defines a decomposable constraint, as C is empty if and only if all such edge-defined triangles $\text{conv}(x^t, p, q)$ are.
- In clustering and classification applications, each point in X may have a given *color*, say, blue, red, etc. As in [3], convex polygon C is a *blue island* if it only contains blue points (and thus no point of any other color). Being a blue island is also a decomposable constraint. Furthermore, one may restrict attention to base points x^t that are blue.

- Given colored points, a *monochromatic island* [3] is a convex polygon which only contains points of a single (unspecified) color. This also defines a decomposable constraint. Furthermore, an optimum monochromatic island may be determined by finding, for each base point, an optimum island of the same color as the base point, as in the preceding example, and then choosing the best of these m candidate solutions.
- Decomposable constraints directly associated with the edges of C include bounds or other restrictions on its edge lengths $\|pq\|$, e.g., $\ell \leq \|pq\| \leq u$ for each edge pq ; on the number of points of X on its edges, e.g., $\ell \leq |\text{conv}(p, q) \cap X| \leq u$ for each edge pq ; etc.

A decomposable constraint may be enforced by simply excluding from the arc sets A^t every pair (p, q) that defines an edge pq or triangle $\text{conv}(x^t, p, q)$ which violates the constraint. Indeed, the nested compatibility property is preserved under arbitrary edge deletions; in other words, the staircase structure of the adjacency matrix of the compatibility graphs $G_{L,q}$ is preserved under row and column deletions, see Figure 1(b).

4.3 Optimum Convex K -gons and K -Point Sets

The optimum convex polygon problems considered in [8] concern K -gons, i.e., convex polygons with a specified number K of edges. Such a constraint is clearly not decomposable, but it may be enforced in a Dynamic Programming approach by expanding the state space to include as an additional dimension the number of edges in the partial boundary constructed so far.

More generally, and as is well known, we can solve an optimum (s, t) -path problem with a constrained number of arcs by using as DP functional the maximum net value $F(q, h)$ of a feasible (s, q) -path with (exactly) h arcs and ending with node q . For turn-restricted optimum path problems with nested compatibilities, let $G^k(q, h)$ denote the maximum net value of a feasible $(0, q)$ -path with h arcs and ending with arc (p_q^k, q) . If the constraint is that the number $|P|$ of arcs in the path P satisfy $a \leq |P| \leq b$ for given integers

$b \geq a \geq 0$, then the resulting extension of the DP recursion (5)–(7) is:

$$G^1(0, 0) = 0; \quad G^k(p, 0) = -\infty \quad \text{for all } p \neq 0 \text{ and } 1 \leq k \leq K_p;$$

for $h = 1, \dots, b$,

$$G^k(q, h) = \max \left\{ \max_{p \in A_q^k} \left[G^{\ell(p,q)}(p, h-1) + c(p, q) \right], G^{k-1}(q, h) \right\}$$

for $1 \leq k \leq K_q$ and $1 \leq q \leq n-1$;

$$z = \max_{h=\max\{a,1\}, \dots, b} \max_{q:(q,n) \in A} \max_{k=1, \dots, K_q} \left(G^k(q, h-1) + c(q, n) \right)$$

where, as before, $\ell(p, q)$ is the unique integer ℓ for which $q \in B_p^\ell$. This recursion corresponds to finding an optimum path in an acyclic digraph with $O(n^2b)$ nodes and $O(n^2b)$ arcs, and thus it runs in $O(n^2b)$ time. When applied to optimum K -gon problems (where $a = b = K$ and $m = |X|$), it yields the $O(m^3K)$ algorithm of [8] after enumerating all $O(m)$ choices of the base vertex x^t .

A similar approach applies to *convex K -point set* problems [8] of finding an optimum “convex” subset $S \subset X$, i.e., satisfying $X \cap \text{conv } S = S$, with (exactly) K elements (i.e., $|S| = K$). A similar recursion applies where, omitting for simplicity the base node superscripts t , $G^k(q, h)$ now denotes the optimum value of a (clockwise) right-open polygon $C \setminus \text{conv}(q, 0)$ based at node 0, with vertices in $X = \{0, 1, \dots, n-1\}$, containing exactly h points from X , and ending with arc (p_q^k, q) :

$$G^k(p, h) = -\infty \quad \text{for all } p \in X, \quad -n \leq h \leq 0 \text{ and } 1 \leq k \leq K_p;$$

$$G^1(0, 0) = 0;$$

for $h = 1, \dots, K$,

$$G^k(q, h) = \max \left\{ \max_{p \in A_q^k} \left[G^{\ell(p,q)}(p, h - \tau(p, q)) + c(p, q) \right], G^{k-1}(q, h) \right\}$$

for $1 \leq k \leq K_q$ and $1 \leq q \leq n-1$;

$$z = \max_{q:(q,n) \in A} \max_{k=1, \dots, K_q} \left(G^k(q, K - \tau(q, n)) + c(q, n) \right)$$

where $\tau(p, q)$, resp., $c(p, q)$, denotes the number of points in the right-open triangle $T(p, q) = \text{conv}(0, p, q) \setminus \text{conv}(q, 0)$, resp., its contribution to the objective value, if $0 < q < n$; and those of the closing edge $E(p) = \text{conv}(p, 0)$ if $q = n$. This recursion also corresponds to finding an optimum path in an acyclic digraph with $O(n^2K)$ nodes and $O(n^2K)$ arcs, and runs in $O(n^2K)$ time, yielding a corresponding $O(m^3K)$ algorithm in [8].

Theorem 4 extends to these problems, by replacing the edge variables $y_{p,q}^t$ with “extended” variables $y_{p,q,h}^t$ associated with the arcs in the acyclic digraphs corresponding to the recursions described above:

Corollary 5. *The Optimum Convex K -gon and Optimum Convex K -Point Set problems with decomposable objective and constraints admit integral extended formulations with $O(m^3K)$ constraints and $O(m^3K)$ variables.*

The details of these extended formulations are left to the interested reader.

5 Concluding Remarks

In this paper we have drawn and exploited connections between optimum turn-restricted path problems and optimum convex polygon problems. Dynamic Programming algorithms for the latter geometric optimization problems may be interpreted as sequences of optimum turn-restricted path problems satisfying a nested compatibility condition. As is well known, many optimum path problems have integral extended (network flow) formulations with variables associated with the arcs of the corresponding network. This is the case here, and the nested compatibility condition allows a reduction by a factor m (the number of nodes, or of points in the geometric problems) in the size of the networks, and thus also in the size of the extended formulations, for each choice of the polygon base point. Combining these with Balas’s Disjunctive Programming principle yields integral extended formulations with $O(m^3)$ variables and constraints for the Optimum Convex Subset problem (and several of its extensions), and $O(m^3K)$ variables and constraints for optimum convex K -gon and K -point subset problems.

This approach may also extend to other geometric optimization problems. For instance, one may consider strengthened or relaxed notions of “convexity” by imposing a positive or negative minimum angle between successive edges of the polygon (instead of a zero minimum angle, as for ordinary convexity). Such extensions may modeled by appropriately modifying the compatibility list L , and this will lead to similar results provided the nested compatibility property is preserved.

References

- [1] Egon Balas, *Disjunctive programming: Properties of the convex hull of feasible points*, GSIA Management Science Research Report MSRR 348, Carnegie Mellon University (1974), published as invited paper in: *Discrete Applied Mathematics* **89** (1998), no. 1, 3–44.

- [2] Hannah Bast, Daniel Delling, Andrew Goldberg, Matthias Müller-Hannemann, Thomas Pajor, Peter Sanders, Dorothea Wagner, and Renato F Werneck, *Route planning in transportation networks*, arXiv preprint arXiv:1504.05140 (2015).
- [3] Crevel Bautista-Santiago, José Miguel Díaz-Báñez, Dolores Lara, Pablo Pérez-Lantero, Jorge Urrutia, and Inmaculada Ventura, *Computing optimal islands*, *Operations Research Letters* **39** (2011), no. 4, 246–251.
- [4] Tom Caldwell, *On finding minimum routes in a network with turn penalties*, *Communications of the ACM* **4** (1961), no. 2, 107–108.
- [5] Michele Conforti, Gérard Cornuéjols, and Giacomo Zambelli, *Extended formulations in combinatorial optimization*, *4OR* **8** (2010), no. 1, 1–48.
- [6] Daniel Delling, Andrew V Goldberg, Thomas Pajor, and Renato F Werneck, *Customizable route planning*, *Experimental algorithms*, Springer, 2011, pp. 376–387.
- [7] Jean-Paul Doignon, André Ducamp, and Jean-Claude Falmagne, *On realizable biorders and the biorder dimension of a relation*, *Journal of Mathematical Psychology* **28** (1984), no. 1, 73–109.
- [8] David Eppstein, Mark Overmars, Günter Rote, and Gerhard Woeginger, *Finding minimum areak-gons*, *Discrete & Computational Geometry* **7** (1992), no. 1, 45–58.
- [9] Robert Geisberger and Christian Vetter, *Efficient routing in road networks with turn costs*, *Experimental Algorithms*, Springer, 2011, pp. 100–111.
- [10] Eliécer Gutiérrez and Andrés L Medaglia, *Labeling algorithm for the shortest path problem with turn prohibitions with application to large-scale road networks*, *Annals of Operations Research* **157** (2008), no. 1, 169–182.
- [11] Bernhard Korte and László Lovász, *Polyhedral results for antimatroids*, *Annals of the New York Academy of Sciences* **555** (1989), no. 1, 283–295.
- [12] Bernhard Korte, László Lovász, and Rainer Schrader, *Greedoids*, vol. 4, Springer, 1991.
- [13] Gordon T Wilfong, *Motion planning for an autonomous vehicle*, *Autonomous Robot Vehicles*, Springer, 1990, pp. 391–395.

Appendix

Proof of Lemma 1. Consider any $q \in \{1, \dots, n - 1\}$. First, note that by assumptions (i)–(iii) we may assume that $p_q^0 = 0$ and $r_q^{\sigma(q)} = n$. Note also that $\pi(q) = |\text{Pred}(q)|$ and $\sigma(q) = |\text{Pred}(q)| + |\text{Succ}(q)|$. Let $A_q^1 = \{p \in \text{Pred}(q) : (p, q, r) \in L \text{ for all } r \in \text{Succ}(q)\}$. If $A_q^1 = \text{Pred}(q)$ then the Lemma holds with $B_q^1 = \text{Succ}(q)$. Otherwise, we construct B_q^1, A_q^2, B_q^2 , etc., inductively as follows. Assume we

have identified nonempty disjoint subsets A_q^1, \dots, A_q^h of $\text{Pred}(q)$ and, if $h > 1$, B_q^1, \dots, B_q^{h-1} of $\text{Succ}(q)$ such that, for all $p \in \bigcup_{j \leq h} A_q^j$ and all $r \in \text{Succ}(q)$, $(p, q, r) \in L$ if and only if $(p, r) \in \bigcup_{k=1}^h (A_q^k \times (\text{Succ}(q) \setminus \bigcup_{\ell < h} B_q^\ell))$, i.e., the conclusion of the Lemma holds for all such p and r . By assumption (iii) and by induction, A_q^h is a consecutive set of predecessors of q , i.e., $A_q^h = \{p_q^i : \alpha(h-1)+1 \leq i \leq \alpha(h)\}$ for some $\alpha(h) \in \{\alpha(h-1)+1, \dots, \pi(q)\}$ where, by assumption (i) we may assume $\alpha(0) = -1$ (so $A_q^1 = \{p_q^i : 0 \leq i \leq \alpha(1)\}$). As for the case $h = 1$ above, if $\bigcup_{k=1}^h A_q^k = \text{Pred}(q)$ then the Lemma holds with $B_q^h = \text{Succ}(q) \setminus \bigcup_{\ell < h} B_q^\ell$ and we are done. Else $\alpha(h) < \pi(q)$. Let $B_q^h = \{r \in \text{Succ}(q) : (p_q^{\alpha(h)+1}, q, r) \notin L\}$. By assumption (iii) and induction, B_q^h is a consecutive set of successors of q , i.e., $B_q^h = \{p_q^i : \beta(h-1)+1 \leq i \leq \beta(h)\}$ for some $\beta(h) \in \{\beta(h-1)+1, \dots, \sigma(q)\}$ where, by assumption (ii) we may assume $\beta(0) = \pi(q)$ (so $B_q^1 = \{p_q^i : \pi(q)+1 \leq i \leq \beta(1)\}$). Letting $A_q^{h+1} = \{p \in \text{Pred}(q) \setminus \bigcup_{k \leq h} A_q^k : (p, q, r) \in L \text{ for all } r \in \text{Succ}(q) \setminus \bigcup_{\ell \leq h} B_q^\ell\}$ now satisfies the inductive assumption with h replaced by $h+1$. This completes the inductive proof of the Lemma. \square