AN ADDITIVE BOUNDED PROCEDURE FOR COMBINATORIAL
OPTIMIZATION PROBLEMS

MATTEO FISCHETTI and PAOLO TOTH
University of Bologna, Bologna, Italy
(Received December 1986; revision received December 1987; accepted January 1988)

We know that the effectiveness of the branch-and-bound algorithms proposed for the solution of combinatorial
optimization problems greatly depends on the tightness of the available bounds. In this paper, we consider optimization
problems with a linear objective function. We propose an additive approach for computing lower bounds that yields an
increasing sequence of values. An application to the traveling salesman problem with precedence constraints is presented
to exemplify the technique.

Many NP-hard combinatorial optimization problems are successfully approached through branch-and-bound algorithms. Assuming a minimization problem \( P \) of the form \( v(P) = \min \{ f(x) : x \in F(P) \} \), a basic ingredient of a branch-and-bound method is a lower bounding procedure, i.e., a procedure for finding a lower bound on \( v(P) \).

A widely used approach for computing lower bounds is to solve a relaxation of problem \( P \), i.e., a problem \( R \) of the form \( v(R) = \min \{ g(x) : x \in F(R) \} \) with \( F(R) \supseteq F(P) \) and \( g(x) \leq f(x) \) for each \( x \in F(P) \). In fact, it is well known that \( v(R) \leq v(P) \). Effective bounds can be computed through Lagrangian relaxation by removing some of the constraints that define \( F(P) \) and by embedding them in the objective function through appropriate multipliers. The tightness of the bounds thus provided greatly depends on the value of the multipliers. Subgradient optimization techniques have been used successfully in many applications to find good values for the multipliers. Other relaxations can be obtained by simply removing a set of constraints, or by surrogating them through a linear combination. For more information on Lagrangian relaxation techniques, see, for instance, Held and Karp (1970), Held, Wolfe and Crowder (1974), Geoffrion (1974), Sandi (1979), Shapiro (1979) and Fisher (1981).

In this paper, we consider a problem \( P \) of the form \( v(P) = \min \{ cx : x \in F(P) \} \). In Section 1, we propose an additive approach for computing lower bounds that yields a sequence of relaxations of problem \( P \), each producing an improved lower bound. An application to the traveling salesman problem with precedence constraints and a numerical example that exemplify the new approach are given in Section 2.

1. An Additive Lower Bounding Technique

Let us consider the following problem.

**Problem P**

\[
v(P) = \min \ cx
\]

subject to

\[
x \in F(P)
\]

where \( x \) and \( c \) are, respectively, a solution column vector and a cost row vector, both having \( n \) elements, and \( F(P) \subseteq \{ x \in \mathbb{R}^n : x \geq 0 \} \). For the sake of simplicity, we assume that all the problems considered in this paper are feasible and bounded.

In many situations, several different lower bounding procedures for \( P \) are available; each exploits a different substructure of the problem. (For example, in the 3-matroid intersection problem, three different lower bounds can be obtained by solving the corresponding, well-structured 2-matroid intersection problem for each pair of matroids.) Clearly, a lower bound for \( P \) could be obtained by applying each single bounding procedure and taking the maximum of the values computed. In this way, however, only one substructure is fully exploited, while all the others are lost completely. Thus, we propose an additive approach, which partially overcomes this drawback.

Let \( D^{(1)}, \ldots, D^{(r)} \) be the lower bound procedures available for problem \( P \). Also suppose that, for \( k = 1, \ldots, r \), procedure \( D^{(k)}(x) \)—when applied to the
instance of problem $P$ with cost vector $\delta$—returns a lower bound value $\delta^{(k)}$ as well as a residual cost row vector $c^{(k)} \in \mathbb{R}^n$ such that $c^{(k)} \geq 0$, and

$$\delta^{(k)} + c^{(k)}x \leq \bar{c}x \quad \text{for each } x \in F(P).$$

The additive approach generates a sequence of instances of problem $P$, each obtained from the previous one by considering the corresponding residual costs and applying a different bounding procedure.

Initially, procedure $\mathcal{L}^{(1)}(c)$ is applied to obtain a value $\delta^{(1)}$ and the corresponding residual cost vector $c^{(1)}$. We consider the following problem.

**Problem $R^{(1)}$**

$$v(R^{(1)}) = \delta^{(1)} + \min_{x \in F(P)} c^{(1)}x.$$ 

One can easily verify that $R^{(1)}$ is a relaxation of $P$, since the two problems have the same feasible set, $F(P)$, and their objective functions are such that:

$$\delta^{(1)} + c^{(1)}x \leq cx \quad \text{for each } x \in F(P).$$

Note that in problem $R^{(1)}$ a residual instance

$$v(P^{(1)}) = \min_{x \in F(P)} c^{(1)}x$$

of $P$ has been introduced.

Because of our hypotheses, $v(P^{(1)})$ is nonnegative, so any lower bound on its value can be added to $\delta^{(1)}$ to obtain a tighter bound for $P$. To this end, we apply procedure $\mathcal{L}^{(2)}(c^{(1)})$, which yields a value $\delta^{(2)}$ and the corresponding residual costs $c^{(2)}$. Now, a new relaxation

$$v(R^{(2)}) = \delta^{(1)} + \delta^{(2)} + \min_{x \in F(P)} c^{(2)}x$$

of problem $P$ is available. In fact, one can verify easily that

$$\delta^{(1)} + \delta^{(2)} + c^{(2)}x \leq \delta^{(1)} + c^{(1)}x \leq cx$$

for each $x \in F(P)$.

As before, the current lower bound for $P$, given by $\delta^{(1)} + \delta^{(2)}$, can be further strengthened by applying procedure $\mathcal{L}^{(3)}(c^{(2)})$. The approach can be iterated by applying the remaining bounding procedures, in sequence, as outlined in the following Pascal-like description.

**Algorithm ADD**

- **Input:** cost vector $c$; $r$ bounding procedures $\mathcal{L}^{(1)}$, $\ldots$, $\mathcal{L}^{(r)}$ for problem $P$;
- **Output:** lower bound $\delta$; final residual cost vector $c^{(r)} \geq 0$;

Begin

initialize $c^{(0)} := c$; $\delta := 0$;

for $k := 1$ to $r$ do

begin

apply $\mathcal{L}^{(k)}(c^{(k-1)})$, thus obtaining value $\delta^{(k)}$ and the residual cost vector $c^{(k)}$;

$\delta := \delta + \delta^{(k)}$

end

End.

In this way, it is possible to take into account the different substructures of problem $P$ exploited by the available bounding procedures. However, note that during execution of the algorithm, the current residual costs are progressively weakened. Consequently, the gap between $v(P)$ and the value $\delta + \min\{c^{(k)}x : x \in F(P)\}$ of the current relaxation could increase at each iteration.

A comparison with subgradient optimization techniques shows that the new approach is advantageous because no decrease in the value of the lower bounds can occur, and no parameter tuning is required. However, due to its greedy nature, the new approach does not assure consistently better bounds. Therefore, an improvement could be obtained, in some cases, by embedding subgradient optimization techniques in one or more bounding procedures $\mathcal{L}^{(k)}$.

The additive approach is related to the so-called Restricted Lagrangian Relaxation introduced by Balas and Christofides (1981), which also leads to a sequence of nondecreasing lower bounds. A comparison between the two techniques for the Asymmetric Traveling Salesman Problem is reported by Fischetti and Toth (1987a), who show that the additive approach allows the design of bounding procedures which dominate some of those proposed by Balas and Christofides, both for the value of the lower bound and the overall time complexity.

The additive approach also has been applied successfully to the Multiple Depot Vehicle Scheduling Problem (Carpaneto, Dell'Amico, Fischetti and Toth 1987), to the Prize-Collecting Traveling Salesman Problem (Fischetti and Toth 1988), and to the Symmetric Traveling Salesman Problem (Carpaneto, Fischetti and Toth 1987).

Algorithm ADD returns the residual cost vector $c^{(r)}$ in addition to $\delta$. These final costs are not used to determine $\delta$, so their computation may be avoided. However, they can be used to reduce the size of the problem by determining an upper bound on the value of each variable. In fact, let $\bar{c}$ be the value of a heuristic
solution for $P$. Since
\[ \delta + c_j x_j \leq \delta + \sum_{i=1}^{n} c_i x_i \]
\[ \leq \sum_{i=1}^{n} c_i x_i < \tilde{z} \]
must hold for any feasible solution $(x_j)$ whose value is less than $\tilde{z}$, we can impose the additional constraints:
\[ \delta + c_j x_j < \tilde{z} \quad \text{for } j = 1, \ldots, n. \]

In particular, if $x_j$ is restricted to being binary, we can fix $x_j = 0$ if $c_j \geq \tilde{z} - \delta$.

The key step of the new approach is the computation of the residual costs in the lower bound procedures. In the following, we propose three different methods related to three possible ways of computing lower bounds (with applications in Sections 2.1, 2.2, and 2.3). In addition, in Section 1.4, we show how to compute valid residual costs associated with bounding procedures based on Lagrangian relaxation.

**1.1. Bounds from Linear Programming Réglaxation**

Let us suppose that
\[ F(P) \subset \{ x \in \mathbb{R}^n : x \geq 0, Ax \geq b \}. \]

A well known lower bound $\delta$ on $v(P)$ can be computed by considering the following linear programming relaxation of $P$.

**Problem LP**

\[ \delta = v(LP) = \min cx \]

subject to
\[ Ax \geq b \]
\[ x \geq 0. \]

To compute both $\delta$ and the corresponding residual cost vector $\tilde{c}$, we solve the dual problem of LP defined as follows.

**Problem D**

\[ v(D) = \max ub \]

subject to
\[ c - uA \geq 0 \]
\[ u \geq 0. \]

Let $u^*$ be an optimal solution to $D$, so $\delta = v(D) = u^*b$. It can be verified easily that the reduced cost vector $\tilde{c} = c - u^*A$ is a valid residual cost vector associated with lower bound $\delta$. Whenever the dual problem does not have an unique optimal solution, $u^*$ could be chosen to determine reduced costs, which allows the following bounding procedures to be more effective.

It is worth noting that any feasible solution $\tilde{u}$ for $D$ can be used instead of $u^*$ to produce a valid lower bound $\tilde{\delta} = \tilde{u}b$ and its corresponding residual cost vector $\tilde{c} = c - \tilde{u}A$. This can be performed through any heuristic procedure for $D$, for example, by applying a dual ascent technique as suggested by, among others, Bilde and Krarup (1977), Erlenkotter (1978), Guignard and Spielberg (1979), Fisher and Hochbaum (1980), Wong (1984), and Fisher, Jaikumar and Van Wassenhove (1986). Although the lower bound $\tilde{\delta} = \tilde{u}b$ is weaker than $u^*b$, the corresponding residual cost vector $c - \tilde{u}A$ is generally “stronger” than $c - u^*A$, and therefore, the following bounding procedures could obtain higher bound improvements in some cases.

**1.2. Bounds from Variable Decomposition**

Let us suppose that it is possible to define $q$ sets $Y^{(1)}, \ldots, Y^{(q)}$ with
\[ Y^{(h)} \subset \{ y \in \mathbb{R}^n : y \geq 0 \} \quad \text{for } h = 1, \ldots, q, \]
with the property that each $x \in F(P)$ can be decomposed into $q$ column vectors $y^{(1)}, \ldots, y^{(q)}$ as
\[ x = \sum_{h=1}^{q} y^{(h)} \text{ with } y^{(h)} \in Y^{(h)} \quad \text{for } h = 1, \ldots, q. \]

For example, let $F(P)$ be the set of all the Hamiltonian circuits in a given digraph. Each feasible solution can be decomposed into two (vertex disjoint) paths between two given vertices $s$ and $t$. Hence, it is possible to define $q = 2$ sets
\[ Y^{(1)} = \{ y : y \text{ is the 0-1 incidence vector corresponding to a path from } s \text{ to } t \} \]
and
\[ Y^{(2)} = \{ y : y \text{ is the 0-1 incidence vector corresponding to a path from } t \text{ to } s \}. \]

Now consider the following equivalent formulation of $P$.

**Problem EP**

\[ v(EP) = \min c(y^{(1)} + y^{(2)} + \ldots + y^{(q)}) \]

subject to
\[ y^{(h)} \in Y^{(h)}, \quad \text{for } h = 1, \ldots, q \]
\[ (y^{(1)} + y^{(2)} + \ldots + y^{(q)}) \in F(P) \]
and define the **partial problem** for \( h = 1, \ldots, q \), as follows.

**Problem PP\(^{(h)}\)**

\[
\nu(PP^{(h)}) = \min c y^{(h)}
\]

subject to

\[
y^{(h)} \in Y^{(h)}.
\]

Let us suppose that, for each partial problem PP\(^{(h)}\), a bounding procedure is available that produces a lower bound \( \theta^{(h)} \) on \( \nu(PP^{(h)}) \) and a residual cost vector \( \gamma^{(h)} \geq 0 \) such that

\[
\theta^{(h)} \gamma^{(h)} + \gamma^{(h)} y^{(h)} \leq c y^{(h)}
\]

for each \( y^{(h)} \in Y^{(h)} \).

It is now possible to define the following problem.

**Problem ER**

\[
\nu(ER) = \sum_{h=1}^{q} \theta^{(h)} + \min \sum_{h=1}^{q} \gamma^{(h)} y^{(h)}
\]

subject to (1) and (2), which is clearly a relaxation of EP, since

\[
\sum_{h=1}^{q} (\theta^{(h)} + \gamma^{(h)} y^{(h)}) \leq c \sum_{h=1}^{q} y^{(h)}
\]

for each feasible solution to EP.

Problem ER can be relaxed further into R as follows.

**Relaxation R**

\[
\nu(R) = \delta + \min \tilde{c} \sum_{h=1}^{q} y^{(h)}
\]

subject to (1) and (2), where

\[
\delta = \sum_{h=1}^{q} \theta^{(h)}
\]

and

\[
\tilde{c} = \min\{|\gamma^{(h)}|: h = 1, \ldots, q\} \quad \text{for } j = 1, \ldots, n.
\]

Finally, relaxation R can be reformulated as

\[
\nu(R) = \delta + \min \tilde{c} x
\]

subject to

\[
x \in F(P).
\]

Hence, \( \delta \) is a valid lower bound for P, and \( \tilde{c} \) is a valid residual cost vector associated with \( \delta \).

A bounding procedure based on the variable decomposition approach can be outlined as follows.

**Step 1.** Find a possible decomposition of problem P into q partial problems PP\(^{(1)}\), \ldots, PP\(^{(q)}\).

**Step 2.** For \( h = 1 \) to \( q \), apply a bounding procedure to partial problem PP\(^{(h)}\) to obtain a lower bound \( \theta^{(h)} \) and a residual cost vector \( \gamma^{(h)} \).

**Step 3.** Compute the overall bound \( \delta \) by adding the single lower bounds \( \theta^{(h)} \), and the overall residual costs \( \tilde{c} \) by taking the minimum of all the single residual costs \( \gamma^{(h)} \).

### 1.3. Bounds from Disjunction

Let us suppose that

\[
F(P) \subseteq \bigcup_{h=1}^{p} W^{(h)}
\]

with \( W^{(h)} \subseteq \{ x \in \mathbb{R}^n : x \geq 0 \} \) for \( h = 1, \ldots, p \). In other words, for each \( x \in F(P) \), the disjunction

\[
(x \in W^{(1)}) \text{ or } (x \in W^{(2)})
\]

or \( \cdots \)

or \( (x \in W^{(p)}) \)
holds.

Now consider the following **restricted problem** for \( h = 1, \ldots, p \).

**Problem RP\(^{(h)}\)**

\[
\nu(RP^{(h)}) = \min c x
\]

subject to

\[
x \in F(P) \cap W^{(h)}
\]

for which a bounding procedure is supposed to be available, that produces a lower bound \( \theta^{(h)} \) on \( \nu(RP^{(h)}) \) and a residual cost row vector \( \gamma^{(h)} \geq 0 \) such that

\[
\theta^{(h)} + \gamma^{(h)} x \leq c x
\]

for each \( x \in F(P) \cap W^{(h)} \).

A valid lower bound \( \delta \) for P is then

\[
\delta = \min \{ \theta^{(h)} : h = 1, \ldots, p \}
\]

and its associated residual costs are

\[
\tilde{c} = \min \{ \gamma^{(h)} : h = 1, \ldots, p \} \quad \text{for } j = 1, \ldots, n.
\]

In fact, for each \( \tilde{x} \in F(P) \), let \( \tilde{h} \) be such that \( \tilde{x} \in W^{(\tilde{h})} \). We then have

\[
\delta + \tilde{c} \tilde{x} \leq \theta^{(\tilde{h})} + \gamma^{(\tilde{h})} \tilde{x} \leq c \tilde{x}.
\]

### 1.4. Bounds from Lagrangian Relaxation

Let us suppose that

\[
F(P) \subseteq \{ x \in \mathbb{R}^n : A x \geq b \}
\]

with \( b \in \mathbb{R}^r \).

Let \( \tilde{u} \in \mathbb{R}^r \) be a row vector of nonnegative multipliers, and consider the Lagrangian cost vector.
\[ \tilde{c} = c - \tilde{u}A. \] Then, inequality \( \tilde{u}b + \tilde{c}x \leq cx \) holds for each \( x \in F(P) \).

Now apply any bounding procedure to the instance of \( P \) defined by cost vector \( \tilde{c} \) to obtain a lower bound value \( \theta \) and the corresponding residual cost vector \( \gamma \). So inequalities
\[ \tilde{u}b + (\theta + \gamma x) \leq \tilde{u}b + \tilde{c}x \leq cx \]
hold for each \( x \in F(P) \). Hence
\[ \delta = \tilde{u}b + \theta \]
is a valid lower bound for the instance of \( P \) defined by cost vector \( c \) (since \( \gamma \) is nonnegative), and
\[ \tilde{c} = \gamma \]
is the corresponding residual cost vector.

A bounding procedure based on Lagrangian relaxation can then be outlined as follows:

**Step 1.** Determine a convenient vector \( \tilde{u} \geq 0 \) of Lagrangian multipliers;

**Step 2.** Apply a bounding procedure to the instance of \( P \) defined by cost vector \( \tilde{c} = c - \tilde{u}A \) to obtain a lower bound value \( \theta \) and a corresponding residual cost vector \( \gamma \);

**Step 3.** Compute a lower bound \( \delta \) as \( \tilde{u}b + \theta \) (the corresponding residual cost vector is \( \gamma \)).

Vector \( \tilde{u} \) can be heuristically determined to produce a "good" value of bound \( \delta \) by iterating Steps 2 and 3, and updating multipliers according to subgradient-based techniques.

### Problem TSP-PC

\[
v(\text{TSP-PC}) = \min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{i,j}x_{i,j} \tag{3}
\]

subject to
\[
\sum_{j=1}^{n} x_{i,j} = 1, \quad i = 1, \ldots, n \tag{4}
\]

\[
\sum_{i=1}^{n} x_{i,j} = 1, \quad j = 1, \ldots, n \tag{5}
\]

\[
\sum_{i \in V \setminus S} \sum_{j \in S} x_{i,j} \geq 1, \quad S \subset V: 1 \notin S, S \neq \emptyset \tag{6}
\]

\[
\begin{cases}
\text{the path from vertex 1 to vertex } h \text{ implied by } (x_{i,j}) \text{ must visit all the vertices in } P_h \\
h = 2, \ldots, n
\end{cases} \tag{7}
\]

\[
x_{i,j} \geq 0, \quad i = 1, \ldots, n; \quad j = 1, \ldots, n \tag{8}
\]

\[
x_{i,j} \text{ integer, } \quad i = 1, \ldots, n; \quad j = 1, \ldots, n \tag{9}
\]

where \( x_{i,j} = 1 \) if and only if arc \( (i, j) \) is in the optimal tour.

Different substructures can be pointed out and exploited to produce lower bounds for TSP-PC. In particular, any bounding procedure for the Asymmetric Traveling Salesman Problem (ATSP) defined by (3)–(6) and (8)–(9), can be applied to TSP-PC as well.

### 2.1. Bounds from Linear Programming Relaxations

Two basic relaxations for ATSP are the Assignment Problem (AP) defined by (3), (4), (5) and (8), and the Shortest Spanning 1-Tree Problem (1-SSAP) defined by (3), (5), (6) and (8). Since both relaxations are linear programming problems, the corresponding bound values and residual costs can be obtained, as described in Section 1.1, by solving the associated dual problems.

### Problem D-AP

\[
v(\text{D-AP}) = \max \sum_{i=1}^{n} (u_i + v_i) \tag{D-AP}
\]

subject to
\[
c_{i,j} - u_i - v_j \geq 0, \quad i = 1, \ldots, n; \quad j = 1, \ldots, n. \tag{D-AP}
\]

### Problem D-1-SSAP

\[
v(\text{D-1-SSAP}) = \max \sum_{S \subset V, S \neq \emptyset, 1 \in S} w(S) + \lambda \tag{D-1-SSAP}
\]
subject to
\[ c_{i,j} - \sum_{\substack{S \subset V: 1 \in S, \ i \in S \setminus j \in S}} w(S) \geq 0, \]
\[ i = 1, \ldots, n; \ j = 2, \ldots, n \]
\[ c_{i,1} - \lambda \geq 0, \ i = 1, \ldots, n \]
\[ w(S) \geq 0, \ S \subset V: 1 \in S, \ S \neq \emptyset. \]

Since constraints 5 with \( j \neq 1 \) are not active for 1-SSAP when nonnegative costs are considered, the corresponding dual variables are neglected in D-1-SSAP. Both the values and the reduced costs of the two dual problems can be found efficiently through the Hungarian or the Shortest Augmenting Path \( O(n^3) \) algorithms (see, for instance, the survey of Martello and Toth 1987) for AP, and through the Tarjan (1977) \( O(n^2) \) implementation of the Edmonds algorithm for 1-SSAP (Edmonds 1967), as corrected by Camerini, Fratta and Maffioli (1979) (see also, Fischetti and Toth (1987b) for an efficient \( O(n^2) \) computation of the reduced costs).

2.2. A Bound from Variable Decomposition

In order to take into account the precedence constraints, consider a pair of distinct vertices \((a, b)\), with \( a, b \neq 1 \), such that \( a \in P_h \), and for each vertex \( h \neq 1 \), define the set \( S_h = \{i \in V:\ h \in P_i\} \) containing all the vertices that must be visited after vertex \( h \). Clearly, any feasible tour can be decomposed into three disjoint paths \( P_{a,b} \), \( P_{a,b} \) and \( P_{a,b} \), connecting, respectively, vertex 1 to vertex \( a \), vertex \( a \) to vertex \( b \), and vertex \( b \) to vertex 1. Moreover, the three paths satisfy the following properties:

1. path \( P_{a,b} \) does not visit any vertex \( i \in Q^{(1,a)} = S_a; \)
2. path \( P_{a,b} \) does not visit any vertex \( i \in Q^{(a,b)} = P_a \cup \{1\} \cup S_b; \)
3. path \( P_{a,b} \) does not visit any vertex \( i \in Q^{(6,1)} = P_b. \)

Hence, according to Section 1.2, we can point out three partial problems. Each problem has to determine a path that connects a given pair of vertices, say \( s \) and \( t \), without visiting a given vertex subset \( Q^{(s,t)} \). This problem can be formulated as follows.

**Problem PP**

\[ v(PP^{(a,b)}) = \min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{i,j}y_{i,j} \]

subject to
\[ \sum_{i=1}^{n} y_{i,h} - \sum_{j=1}^{n} y_{h,j} = \begin{cases} -1, & \text{for } h = s, \\ 1, & \text{for } h = t, \\ 0, & \text{for } h \in V \setminus \{s, t\}; \end{cases} \]
\[ y_{i,j} \geq 0, \ i \in V, j \in V; \]
\[ y_{i,j} \geq 0, \ i \in V, j \in V; \]

where \( y_{i,j} = 1 \) if and only if arc \((i, j)\) is in the optimal solution.

In this case, problem PP can be solved exactly by setting \( \tilde{c}_{i,j} = c_{i,j} + M \) for \( i \in Q^{(s,t)}, j \in V \), and \( \tilde{c}_{i,j} = c_{i,j} \) for \( i \in V \setminus Q^{(s,t)}, j \in V \) (where \( M \) is an arbitrarily large positive value), and by applying any shortest path algorithm with respect to costs \( \tilde{c}_{i,j} \) to determine the cost \( \tilde{L_i} \), of the shortest path from vertex \( s \) to vertex \( i \) (for each \( i \in V \)). It is known that \( \theta^{(s,t)} = v(PP^{(s,t)}) = \tilde{L_i} \). To compute the corresponding reduced costs, let us consider the associated dual problem.

**Problem D-PP**

\[ v(D-PP^{(s,t)}) = \max(L_i - L_j) \]

subject to
\[ c_{i,j} + L_i - L_j \geq 0, \ i \in V \setminus Q^{(s,t)}, j \in V; \]
\[ c_{i,j} + L_i - L_j + \sigma_{i,j} \geq 0, \ i \in Q^{(s,t)}, j \in V. \]

It is known that \( L_i = \tilde{L_i} \) (for \( i \in V \)) and \( \sigma_{i,j} = M \) (for \( i \in Q^{(s,t)} \) and \( j \in V \)); determine an optimal solution for D-PP. The reduced costs corresponding to this solution are \( \tilde{c}_{i,j} + \tilde{L_i} - L_j \geq 0 \). An alternative optimal solution, generally leading to stronger reduced costs, can be obtained by setting \( L_i = \min[L_i, 1] \) for \( i \in V \). In fact, the value of the two optimal solutions is the same. As for the dual feasibility of the alternative solution, suppose, absurdly, that there exists an arc \((i, j)\) with \( \tilde{c}_{i,j} + L_i - L_j < 0 \); we would have \( L_j > L_i \)

\( \text{(since } c_{i,j} \geq 0), \) so \( L_i - L_j \geq \tilde{L_i} - \tilde{L_j} \) (from the definition of the \( L_i \)'s), and finally, the contradiction \( \tilde{c}_{i,j} + L_i - L_j \geq \tilde{L_i} - \tilde{L_j} \geq 0 \). Then, the reduced costs associated with \( PP^{(s,t)} \) are

\[ \gamma_{i,j}^{(s,t)} = \tilde{c}_{i,j} + L_i - L_j \geq 0 \quad \text{for each } i, j \in V. \]

A valid overall bound, \( \delta \), can be computed as

\[ \delta = \theta^{(1,a)} + \theta^{(a,b)} + \theta^{(b,1)} \]

and the corresponding residual costs are

\[ \tilde{c}_{i,j} = \min\{\gamma_{i,j}^{(1,a)}, \gamma_{i,j}^{(a,b)}, \gamma_{i,j}^{(b,1)}\} \quad \text{for each } i, j \in V. \]

2.3. A Bound from Disjunctions

Another lower bound (valid for the ATSP too) can be derived from the following considerations. Let
$(a, b, r)$ be any triplet of distinct vertices of $G$: in any
tour, two paths connecting $a$ and $b$ exist, one is not
visiting $r$. Hence, the following disjunction holds for
each $(x_{i,j})$ corresponding to a tour:

(a path $P^{(1)}$ from $a$ to $b$
not visiting $r$ is implied by $(x_{i,j})$)
or
(a path $P^{(2)}$ from $b$ to $a$
not visiting $r$ is implied by $(x_{i,j})$).

According to Section 1.3, it is possible to introduce
two restricted problems, $RP^{(1)}$ and $RP^{(2)}$, one for each
clause of the disjunction, where, for $h = 1, 2$:

\[ v(RP^{(h)}) = \min \sum_{i=1}^n \sum_{j=1}^n c_{i,j}x_{i,j} \]
subject to

$(x_{i,j})$ defines a tour containing a path $P^{(h)}$.

Each problem $RP^{(h)}$ $(h = 1$ or $2$) can be viewed as a
simple case of $TSP-PC$ where the starting vertex is $r$,
instead of $1$, and vertex $a$ must precede vertex $b$ (when
$h = 1$) or vice versa (when $h = 2$). So a lower bound
$\theta^{(h)}$ and the corresponding residual costs $\gamma^{(h)}_{i,j}$ for
problem $RP^{(h)}$ can be computed, as outlined, to obtain the
bound based on variable decomposition.

The overall lower bound for ATSP is then

$\delta = \min[\theta^{(1)}, \theta^{(2)}]$ and the corresponding residual costs are

$\widetilde{c}_{i,j} = \min[\gamma^{(1)}_{i,j}, \gamma^{(2)}_{i,j}]$ for each $i, j \in V$.

2.4. An Additive Lower Bound

According to the general framework of our approach,
an additive lower bound for $TSP-PC$ can be obtained by
applying all the previously defined bounding procedures
in a given sequence. A possible choice leads to the
following global bounding procedure.

Let $PATH(n, c, s, t, Q^{(5,t)}, \theta^{(5,t)}, \gamma^{(5,t)})$ be any $O(n^2)$
shortest path procedure that solves problem $PP^{(5,t)}$,
defined above, and returns the optimum value $\theta^{(5,t)}$
and the corresponding reduced costs ($\gamma^{(5,t)}_{i,j}$).

Step 1. Solve the Assignment Problem with respect to the original
cost matrix $c$ and let $v(AP)$ and $\tilde{c}$ be the
optimum value and the corresponding reduced

\[ \delta := v(AP). \]

Step 2. Solve the Shortest Spanning 1-Arborescence
Problem with respect to the cost matrix $\tilde{c}$. Let

$\nu(1-SSAP)$ be the optimum value and store the corresponding reduced cost matrix in $\tilde{c}$

\[ \delta := \delta + \nu(1-SSAP). \]

Step 3. Apply the lower bound procedure based on
disjunctions as follows:

for any proper triplet of vertices $(a, b, r)$ do begin

3.1. comment: compute a lower bound $\theta^{(1)}$
and the corresponding residual costs $\gamma^{(1)}_{i,j}$ for

\[ PATH(n, \tilde{c}, r, a, \{b\}, \theta^{(1,a)}, \gamma^{(1,a)}_{i,j}); \]
\[ PATH(n, \tilde{c}, \{b\}, a, \{r\}, \theta^{(b,a)}, \gamma^{(b,a)}_{i,j}); \]
\[ PATH(n, \tilde{c}, b, \{a\}, \{r\}, \theta^{(b,r)}, \gamma^{(b,r)}_{i,j}); \]
\[ \theta^{(1)} := \theta^{(r,a)} + \theta^{(a,b)} + \theta^{(b,r)}; \]
for each $i, j \in V$
do $\gamma^{(1)}_{i,j} := \min[\gamma^{(1,a)}_{i,j}, \gamma^{(b,a)}_{i,j}, \gamma^{(b,r)}_{i,j}]$;
end.

3.2. comment: compute a lower bound $\theta^{(2)}$
and the corresponding residual costs $\gamma^{(2)}_{i,j}$ for

\[ PATH(n, \tilde{c}, r, b, \{a\}, \theta^{(r,b)}, \gamma^{(r,b)}_{i,j}); \]
\[ PATH(n, \tilde{c}, b, \{r\}, \theta^{(b,a)}, \gamma^{(b,a)}_{i,j}); \]
\[ PATH(n, \tilde{c}, b, \{r\}, \theta^{(b,r)}, \gamma^{(b,r)}_{i,j}); \]
\[ \theta^{(2)} := \theta^{(r,b)} + \theta^{(a,b)} + \theta^{(r,a)}; \]
for each $i, j \in V$
do $\gamma^{(2)}_{i,j} := \min[\gamma^{(r,b)}_{i,j}, \gamma^{(b,a)}_{i,j}, \gamma^{(b,r)}_{i,j}]$;
end.

3.3. $\delta := \delta + \min[\theta^{(1)}, \theta^{(2)}]$;
for each $i, j \in V$
do $\tilde{c}_{i,j} := \min[\gamma^{(1)}_{i,j}, \gamma^{(2)}_{i,j}]$;
end.

Step 4. Apply the lower bound procedure based on
variable decomposition as follows:

for any proper pair of vertices $(a, b)$ with $a \in P_b$
do begin

4.1. $PATH(n, \tilde{c}, 1, a, S_a, \theta^{(1,a)}, \gamma^{(1,a)}_{i,j});$
\[ PATH(n, \tilde{c}, a, b, P_a \cup \{1\} \cup S_b, \theta^{(a,b)}, \gamma^{(a,b)}_{i,j}); \]
\[ PATH(n, \tilde{c}, b, 1, P_b, \theta^{(b,1)}, \gamma^{(b,1)}_{i,j}); \]
4.2. $\delta := \delta + \theta^{(1,a)} + \theta^{(a,b)} + \theta^{(b,1)}$
for each $i, j \in V$
do $\tilde{c}_{i,j} := \min[\gamma^{(1,a)}_{i,j}, \gamma^{(a,b)}_{i,j}, \gamma^{(b,1)}_{i,j}]$
end.

Because of Step 3, the overall time complexity is
$O(n^2)$. However, it is possible to reduce the global
computational effort by exploiting parametric tech-
niques for the shortest path computations and heu-
ristically choosing proper triplets and pairs of vertices
at Steps 3 and 4.

Steps 1 to 3 compute a lower bound $\delta$ that is valid
for the ATSP as well. If, during their execution, a zero
residual cost tour is detected, a jump to Step 4 can be
performed, since no further improvement can be obtained by exploiting the ATSP substructure. Analogously, as soon as a feasible solution for TSP-PC is detected in the zero residual cost graph, and the procedure can be stopped. In this case, a heuristic solution for TSP-PC has been found, whose optimality can be checked by comparing its value (computed by considering the original cost matrix $c$) with $\delta$.

### 2.5. A Numerical Example

Let us consider the following instance of TSP-PC: $n = 12$, $P_2 = P_4 = P_6 = P_7 = P_9 = P_{11} = \emptyset$, $P_3 = \{5, 7, 8\}$, $P_8 = \{5, 7\}$, $P_{10} = \{3, 5, 7, 8\}$, and $P_{12} = \{5, 7, 8\}$; the corresponding cost matrix $c$ is given in Figure 1.

We apply the additive bounding procedure to this numerical example.

#### Step 1.
We have $\delta = v(\text{AP}) = 216$ and the dual solution is $u = (3, 5, 10, 5, 48, 52, 10, 35, 5, 7, 8, 10)$, $v = (3, 0, 0, 2, 4, 0, 2, 4, 0, 0, 2, 1)$. The corresponding reduced cost matrix $\bar{c}$ is given in Figure 2. The zero reduced cost arcs of the graph are drawn in Figure 3 (the arcs belonging to the optimal solution of AP are given in boldline).

#### Step 2.
We have $v(1-\text{SSAP}) = 86$ and $\delta = 216 + 86 = 302$. The optimal dual solution is $w(\{3, 4\}) = 28$, $w(\{5, 6\}) = 10$, $w(\{7, 8, 9\}) = 20$, $w(\{10, 11, 12\}) = 21$, $w(\{5, 6, 7, 8, 9\}) = 5$, $w(\{5, 6, 7, 8, 9, 10, 11, 12\}) = 2$ ($w(S) = 0$ for all the other subsets $S$), $\lambda = 0$. The corresponding reduced cost matrix is given in Figure 4. The zero reduced cost arcs are drawn in Figure 5 (the arcs belonging to the optimal solution of 1-SSAP are given in boldline). The figure also shows the subsets $S$ having positive multipliers $w(S)$ in the optimal dual solution.

#### Step 3.
Consider the vertex triplet $(a, b, r) = (6, 8, 9)$.

##### Step 3.1
We have $\theta^{(r,a)} = 0$; the corresponding residual cost matrix $\gamma^{(r,a)}$ is obtained through the dual vector $L(c,a) = (0, \ldots, 0)$. We also have $\theta^{(a,b)} = 5$, with $L(a,b) = (5, 5, 5, 5, 5, 0, 0, 5, 5, 5, 5, 5)$, and $\theta^{(b,a)} = 0$, with $L(b,a) = (0, \ldots, 0)$. Hence, $\theta^{(1)} = 0 + 5 + 0 = 5$.

##### Step 3.2
We have $\theta^{(r,b)} = 0$, with $L(c,b) = (0, \ldots, 0)$, $\theta^{(b,a)} = 10$, with $L(b,a) = (10, 10, 10, 10, 10, 10, 0, 0, 10, 10, 10, 0)$, and $\theta^{(a,r)} = 0$, with $L(a,r) = (0, \ldots, 0)$. Hence, $\theta^{(2)} = 0 + 10 + 0 = 10$. 

---

**Figure 1.** The input cost matrix.

$$
\begin{array}{cccccccccccccccc}
\infty & 0 & 31 & 48 & 39 & 37 & 35 & 50 & 50 & 47 & 44 & 45 & 23 \\
0 & \infty & 28 & 50 & 35 & 38 & 47 & 48 & 47 & 61 & 51 & 25 \\
27 & 29 & \infty & 0 & 20 & 59 & 56 & 53 & 56 & 48 & 49 \\
28 & 25 & 0 & \infty & 17 & 44 & 52 & 60 & 55 & 46 & 49 & 56 \\
13 & 9 & 35 & 35 & \infty & 0 & 28 & 20 & 31 & 28 & 30 \\
9 & 7 & 38 & 35 & 0 & \infty & 25 & 25 & 28 & 30 & 29 \\
5 & 10 & 30 & 32 & 15 & 10 & \infty & 0 & 6 & 24 & 31 & 23 \\
51 & 33 & 61 & 62 & 51 & 45 & 16 & \infty & 0 & 31 & 62 & 54 \\
26 & 23 & 49 & 53 & 37 & 33 & 0 & 4 & \infty & 21 & 45 & 47 \\
5 & 10 & 30 & 30 & 20 & 19 & 27 & 33 & 25 & \infty & 0 & 9 \\
2 & 4 & 31 & 38 & 21 & 17 & 27 & 30 & 35 & 20 & \infty & 0 \\
0 & 5 & 30 & 33 & 25 & 20 & 28 & 31 & 29 & 0 & 20 & \infty \\
\end{array}
$$

**Figure 2.** The residual cost matrix after Step 1.

$$
\begin{array}{cccccccccccccccc}
\infty & 0 & 3 & 34 & 53 & 46 & 40 & 55 & 57 & 50 & 47 & 50 & 27 \\
0 & \infty & 33 & 57 & 44 & 43 & 54 & 57 & 52 & 66 & 58 & 31 \\
40 & 39 & \infty & 12 & 34 & 60 & 71 & 70 & 63 & 66 & 60 & 60 \\
36 & 30 & 5 & \infty & 26 & 49 & 59 & 69 & 60 & 51 & 56 & 62 \\
64 & 57 & 83 & 85 & \infty & 48 & 85 & 80 & 68 & 79 & 78 & 79 \\
64 & 59 & 90 & 89 & 85 & \infty & 79 & 91 & 77 & 80 & 84 & 82 \\
18 & 20 & 40 & 44 & 29 & 20 & \infty & 14 & 16 & 34 & 43 & 34 \\
89 & 68 & 96 & 90 & 90 & 90 & \infty & 35 & 66 & 99 & 90 \\
34 & 26 & 54 & 60 & 40 & 46 & 38 & 7 & \infty & 26 & 52 & 53 \\
15 & 17 & 55 & 49 & 31 & 36 & 66 & 44 & 32 & \infty & 9 & 17 \\
13 & 12 & 39 & 48 & 35 & 25 & 37 & 42 & 43 & 28 & \infty & 9 \\
13 & 15 & 40 & 45 & 39 & 30 & 40 & 45 & 39 & 10 & 32 & \infty \\
\end{array}
$$

**Figure 3.** The zero residual cost arcs after Step 1.

$$
(\bar{c}_{i,j}) =
\begin{array}{cccccccccccccccc}
\infty & 0 & 3 & 20 & 22 & 20 & 23 & 20 & 21 & 21 & 22 & 0 \\
0 & \infty & 0 & 22 & 18 & 21 & 20 & 21 & 20 & 38 & 28 & 2 \\
27 & 29 & \infty & 0 & 33 & 32 & 29 & 26 & 33 & 25 & 26 \\
28 & 25 & 0 & \infty & 0 & 27 & 25 & 33 & 28 & 23 & 26 & 33 \\
13 & 9 & 7 & 7 & \infty & 0 & 15 & 8 & 0 & 10 & 7 & 9 \\
9 & 7 & 10 & 7 & 0 & \infty & 5 & 15 & 5 & 7 & 9 & 8 \\
5 & 10 & 2 & 4 & 5 & 0 & \infty & 0 & 6 & 3 & 10 & 2 \\
31 & 33 & 33 & 34 & 41 & 35 & 16 & \infty & 0 & 10 & 41 & 33 \\
26 & 23 & 21 & 25 & 27 & 23 & 0 & 4 & \infty & 0 & 24 & 26 \\
5 & 10 & 20 & 2 & 5 & 4 & 2 & \infty & 0 & 0 & 0 & 9 \\
2 & 4 & 3 & 10 & 6 & 2 & 2 & 5 & 10 & \infty & 0 & 0 \\
0 & 5 & 2 & 5 & 10 & 5 & 3 & 6 & 4 & 0 & 20 & \infty \\
\end{array}
$$

**Figure 4.** The residual cost matrix after Step 2.
Step 3.3. \( \delta = 302 + \min(5, 10) = 307 \). The corresponding residual cost matrix is given in Figure 6. (Note that reducing \( \theta^{(a, 5)} \) from 10 to 5, and consequently, \( L^{(b, a)} \) to \( (5, 5, 5, 5, 5, 5, 0, 0, 5, 5, 5) \), does not affect the value of \( \delta \) while producing better residual costs.) The zero residual cost arcs are drawn in Figure 7 (the arcs of the path from vertex 6 to 8, and not visiting vertex 9, are given in boldline; those of the path from vertex 8 to 6, and not visiting vertex 9, are in dashed boldline). Note that in the partial graph that corresponds to the zero residual cost arcs, a tour exists that visits, in sequence, vertices 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12. Its cost, computed by considering the original cost matrix, is equal to the current lower bound \( \delta = 307 \). Since this bound is valid for ATSP as well, the tour is optimal for ATSP. However, it does not satisfy the precedence constraints: in particular, vertex 3 is visited before its predecessor 8. So we jump to Step 4.

Step 4. Consider the vertex pair \((a, b) = (8, 3)\), for which \( S_a = \{3, 10, 12\} \), \( S_b = \{10\} \).

![Figure 5. The zero residual cost arcs after Step 2.](image)

\[
(e_{i,j}) = \begin{bmatrix}
\infty & 0 & 3 & 20 & 22 & 20 & 23 & 22 & 20 & 21 & \infty & 0 \\
0 & \infty & 0 & 22 & 18 & 21 & 20 & 21 & 20 & 28 & 28 & 2 \\
27 & 29 & \infty & 0 & 3 & 32 & 32 & 29 & 26 & 33 & 25 & 26 \\
28 & 25 & \infty & 0 & 27 & 25 & 33 & 28 & 23 & 26 & 33 & 25 \\
8 & 4 & 2 & 2 & \infty & 0 & 10 & 3 & 0 & 5 & 2 & 4 \\
4 & 2 & 5 & 2 & \infty & 0 & 10 & 5 & 2 & 4 & 3 \\
5 & 10 & 2 & 4 & 5 & \infty & 0 & 6 & 3 & 10 & 2 \\
41 & 23 & 23 & 24 & 31 & 25 & 6 & \infty & 0 & 31 & 23 \\
26 & 23 & 21 & 25 & 27 & \infty & 0 & 4 & \infty & 0 & \infty & 24 & 26 \\
5 & 10 & 20 & 2 & 5 & 4 & 2 & 8 & 0 & \infty & 0 & \infty & 9 \\
2 & 4 & 3 & 10 & 6 & 2 & 2 & 5 & 10 & 20 & \infty & 0 \\
0 & 5 & 2 & 5 & 10 & 5 & 3 & 6 & 4 & 0 & \infty & \infty & \infty \\
\end{bmatrix}
\]

![Figure 6. The residual cost matrix after Step 3.](image)

\[
\begin{bmatrix}
\infty & 0 & 3 & 2 & 4 & 2 & 5 & 2 & 3 & 4 & 0 \\
0 & \infty & 0 & 4 & 0 & 3 & 2 & 2 & 20 & 10 & 2 \\
4 & 6 & \infty & 0 & 3 & 10 & 9 & 6 & 3 & 10 & 2 & 3 \\
5 & 2 & 0 & \infty & 0 & 4 & 2 & 10 & 5 & 0 & 3 & 10 \\
26 & 22 & 20 & 2 & \infty & 0 & 10 & 3 & 0 & 5 & 2 & 22 \\
4 & 2 & 5 & 2 & \infty & 0 & 10 & 5 & 2 & 4 & 3 \\
23 & 28 & 20 & 4 & 5 & \infty & 0 & 6 & 3 & 10 & 20 \\
20 & 2 & 2 & 3 & 10 & 4 & 6 & \infty & 0 & 10 & 2 \\
5 & 2 & 0 & 4 & 6 & 2 & 0 & \infty & 0 & 3 & 5 \\
5 & 10 & 43 & 25 & 28 & 4 & 2 & 8 & 0 & \infty & 0 & 9 \\
2 & 4 & 3 & 10 & 6 & 2 & 2 & 5 & 10 & 20 & \infty & 0 \\
0 & 5 & 2 & 5 & 10 & 5 & 3 & 6 & 4 & 0 & \infty & \infty \\
\end{bmatrix}
\]

![Figure 7. The zero residual cost arcs after Step 3.](image)

Step 4.1. We have \( \theta^{(1, a)} = 18 \), with \( L = (0, 0, 0, 18, 18, 18, 18, 18, 18, 18, 18, 18, 0) \); \( \theta^{(a, b)} = 21 \), with \( L = (21, 21, 21, 21, 21, 21, 21, 0, 0, 0, 0, 0, 21, 21) \); \( \theta^{(b, 1)} = 23 \), with \( L = (23, 23, 0, 0, 0, 23, 23, 23, 23, 23, 23) \).

Step 4.2. \( \delta = 307 + 18 + 21 + 23 = 369 \). The corresponding residual cost matrix is given in Figure 8. The zero residual cost arcs are drawn in Figure 9 (the arcs of the path from vertex 1 to vertex 8, not visiting vertices 3, 10, 12, are given in boldline; those of the path from 8 to 3, not visiting 1, 5, 7, 10, in dashed boldline; those of the path from 3 to 1, not visiting 5, 7, 8, in dotted boldline). Note that in the partial graph corresponding to the zero residual cost arcs, a feasible tour exists, visiting, in sequence, vertices 1, 2, 5, 6, 7, 8, 9, 3, 4, 10, 11, 12. This tour represents a heuristic solution for TSP-PC. Its cost, computed by considering the original cost matrix, is 371, with a gap equal to 2 with respect to the current lower bound \( \delta = 369 \). However, all the nonzero residual costs are not less than the gap, so the corresponding
Figure 9. The final zero residual cost arcs.

arcs cannot be used to obtain a better feasible solution. On the other hand, it can be verified easily, by inspection, that no other feasible tour exists with zero residual cost. Hence, the previous feasible tour is optimal.

Acknowledgment

This work was supported by the Ministero Pubblica Istruzione, Italy. We thank two anonymous referees for helpful comments.

References


