

Chapter 4

LOW-PROBABILITY HIGH CONSEQUENCE CONSIDERATIONS IN A MULTIOBJECTIVE APPROACH TO RISK MANAGEMENT.

4.1 Introduction

As our society becomes increasingly aware of environmental and public safety concerns, the shipment of hazardous materials (hazmat) via highways is becoming an issue of major importance. In particular, of utmost concern in hazmat transportation safety is the avoidance of accidents involving multiple fatalities. Such a scenario might occur if a tank truck rolled over in a heavily populated area exposing people to a lethal toxic vapor cloud, an explosion, or a fireball. Thus, the major purpose of route selection should be to minimize the risk of multiple-fatality accidents by avoiding situations where the most hazardous combination of risk factors is present.

Accordingly, this study presents a route-finding method that addresses the concern of minimizing the risk of low probability-high consequence (**LPHC**) accidents that can potentially result in multiple fatalities. Our approach differs from that of Sivakumar et al. (1993, 1995) (as discussed in Section 2.2) in that we consider catastrophic accidents in the objective function along with the stated side-constraints in our model, and we develop a special tight representation of the corresponding discrete fractional programming model as well as design a branch-and-bound algorithm to solve this problem. Issues dealing with data acquisition, multi-objective analyses, implementation, and computational performance are also addressed.

The remainder of this section is organized as follows. The next section discusses modeling considerations and motivates the proposed model and suggests possible approaches. Section 4.2 develops a specific discrete fractional programming optimization approach to solve the model, and the proposed algorithm is illustrated in Section 4.3 using a numerical example. Some

computational experience on randomly generated test problems is presented in Section 4.4. Section 4.5 describes a realistic application and discusses implementation considerations. Finally, Section 4.6 presents conclusions and recommendations for future research.

4.2 Modeling Considerations

Let the transportation network under consideration be represented by a directed graph $G = (N, A)$, where N is the set of nodes of the graph and A is the set of arcs. Let the probability of an accident occurring on link $a \in A$ be denoted by p_a , and assume that these probabilities are of small magnitudes (Abkowitz and Cheng, 1988, report that these probabilities are typically of the order 10^{-6} to 10^{-8} per mile traveled). Also, let c_a denote the consequence (cost) incurred if an accident occurs on link $a \in A$. We define a catastrophic accident as one in which the consequence is greater than or equal to C^* , a predetermined value known as the critical value.

Consider some origin-destination pair (O, D) , and let \wp be a path from O to D , where $\wp = \{a_1, a_2, \dots, a_L\}$, $a_i \in A$ for $i = 1, 2, \dots, L$, is represented by the set of unidirectional links in this path \wp . Accordingly, let \wp_c denote the set of critical links in \wp for which the associated consequence exceeds C^* , and let \wp_{nc} denote the remaining (non-critical) links. (We assume that each path in G has at least one such critical link. If paths having no critical links exist, these can be handled separately via a similar approach applied to a network that deletes all the critical links.) The expected consequence E_\wp on any such path \wp is given by

$$E_\wp = \sum_{i=1}^L \left\{ \prod_{j<i} (1 - p_{a_j}) \right\} p_{a_i} c_{a_i}. \quad (4.1)$$

The probability $P(\wp)$ of an accident occurring on path \wp , or the probability $P^*(\wp)$ of a catastrophic accident occurring on that path, are given as follows:

$$P(\wp) = \sum_{i=1}^L p_{a_i} \prod_{j<k} (1 - p_{a_j}) \quad \text{and} \quad P^*(\wp) = \sum_{k \in \wp_c} p_{a_k} \prod_{j<k} (1 - p_{a_j}). \quad (4.2)$$

We now define the conditional expectation of the consequence on path \wp , given that a catastrophic accident has occurred, as

$$CE_{\varphi} = \sum_{i=1}^L c_{a_i} \cdot P[\text{catastrophic accident on link } a_i \mid \text{catastrophic accident on } \varphi]$$

where the stated conditional probability is 0 if $i \in \varphi_{nc}$, and is given by $p_{a_i} \prod_{j<i} (1-p_{a_j}) / P^*(\varphi)$ if $i \in \varphi_c$. Hence, we obtain

$$CE_{\varphi} = \frac{\sum_{i \in \varphi_c} p_{a_i} c_{a_i} [\prod_{j<i} (1-p_{a_j})]}{P^*(\varphi)}. \quad (4.3)$$

Because of the relatively small magnitudes of the link accident probabilities, we can ignore terms that involve products of two or more probabilities and arrive at the following approximations for E_{φ} , $P(\varphi)$, $P^*(\varphi)$, and CE_{φ} , respectively:

$$\hat{E}_{\varphi} = \sum_{i=1}^L p_{a_i} c_{a_i}, \quad \hat{P}(\varphi) = \sum_{i=1}^L p_{a_i}, \quad \hat{P}^*(\varphi) = \sum_{i \in \varphi_c} p_{a_i}, \quad \text{and} \quad \hat{CE}_{\varphi} = \frac{\sum_{i \in \varphi_c} p_{a_i} c_{a_i}}{\sum_{i \in \varphi_c} p_{a_i}}. \quad (4.4)$$

Now, instead of simply adopting the traditional objective of minimizing the expected consequence or the accident probability over all possible paths, let us also examine an objective criterion that attempts to minimize the conditional expectation of the consequence, given that a catastrophic accident has occurred. The motivation here is that a path φ may have very low link-probabilities of an accident occurring, resulting in a small expected consequence E_{φ} on this path. However, if the actual consequences associated with this path are high, then the conditional expectation CE_{φ} of the consequence on this path given that a catastrophic accident has occurred can be relatively large. We, therefore, may not wish to choose this path for routing. Although this motivation is sound, as we shall presently see, sole consideration of this conditional objective function is not a suitable criterion for hazmat routing.

To illustrate, consider the example shown in Figure 4.1, where the tuple (probability of accident, consequence) is shown against each link, and where all consequences are assumed to exceed the critical level. Letting $O \equiv 1$ and $D \equiv 4$, we compute the following using (4.1) and (4.3).

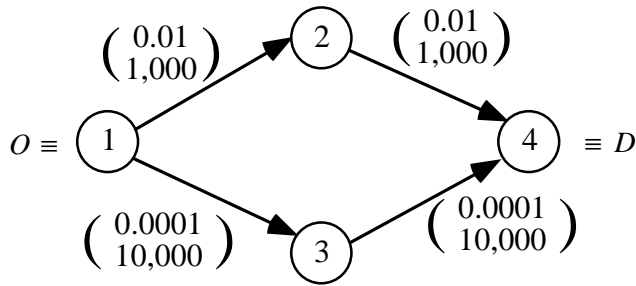


Figure 4.1. Example to Illustrate the Concept of Conditional Expectation.

Paths	E_{φ}	CE_{φ}
$1 \rightarrow 2 \rightarrow 4$	19.9	1,000
$1 \rightarrow 3 \rightarrow 4$	1.9999	10,000

With respect to E_{φ} , path $1 \rightarrow 3 \rightarrow 4$ is optimal, because the arcs involved have low *expected* consequences. However, with respect to CE_{φ} , the path $1 \rightarrow 2 \rightarrow 4$ is optimal due to the relatively high consequences associated with the arcs that belong to the path $1 \rightarrow 3 \rightarrow 4$. Thus, the issue that arises is evident. Should the path with the smallest expected consequence of an accident be chosen regardless of the consequences, or should we primarily consider the degree of the consequences?

To address this question, consider now the same example of Figure 4.1, but with an additional arc from node 2 to 3, having $p_{(2,3)} = 0.9$ and $c_{(2,3)} = 100$. For this case, we obtain the following computations.

Paths	E_{φ}	CE_{φ}
$1 \rightarrow 2 \rightarrow 4$	19.9	1,000
$1 \rightarrow 3 \rightarrow 4$	1.9999	10,000
$1 \rightarrow 2 \rightarrow 3 \rightarrow 4$	99.199	101

Referring to these results, if a path is chosen so as to simply minimize CE_{φ} , the path $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$ would be selected. Choosing this particular path may not be advisable, because we would be selecting a route on which the risk of an accident is relatively high. Although the expected consequence is relatively small given that an accident occurs, this path should not be chosen under most circumstances because of its high probability of causing an

accident. Hence, if conditional expectation is used as the sole criterion, the network should be preprocessed by eliminating links that might have an unacceptably high probability of an accident. However, a subjective deletion of links might eliminate paths that could have been the "lesser of two evils," given the alternatives. Note that the consideration of catastrophic accidents in formulating the conditional risk objective function somewhat ameliorates this effect. For example, if the accident on link (2, 3) is designated to be non-catastrophic, then for the path $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$, we obtain $CE_{\wp} = 1008.9011$ via Equation (4.3), hence yielding the path $1 \rightarrow 2 \rightarrow 4$ again as the optimum in this case. In general, a better alternative might be to impose certain restrictions on expected consequences and path accident probabilities and consider the problem to

$$\text{Minimize } \{\hat{CE}_{\wp} : \wp \in G, \hat{E}_{\wp} \leq v, \hat{P}(\wp) \leq \eta\} \quad (4.5)$$

for some specified values of v and η . For example, we can define v to be $v = (1 + q)v^*$, for various desired values of $q = 0$, where $v^* = \min\{\hat{E}_{\wp} : \wp \in G\}$ is the optimal value for the least expected cost (shortest) path problem, and η can be similarly selected.

Before proceeding to develop this proposed approach, let us comment on the model (4.5) and some alternative approaches. First, note that as discussed by Sivakumar et al. (1993, 1995), the conditional expectation objective function used in (4.5) can be interpreted as selecting a path, subject to the stated constraints, that minimizes the expected consequences in a scenario in which shipments are made on the selected path until the first catastrophic accident occurs. In other words, since $1 / P^*(\wp)$ in (4.3) has the interpretation of the number of trips on path \wp per (catastrophic) accident, the quantity CE_{\wp} intuitively denotes the expected cost per (catastrophic) accident over a great number of trips. Hence, the conditional objective function essentially attempts to minimize the expected cost per accident, which is appealing from a "damage control" viewpoint. However, as noted above, and as seen in Glickman and Sherali (1991), as well as through the examples presented by Erkut (1995), this criterion by itself might be unsuitable in that it could select paths that have high consequences or accident probabilities, but for which the

expected cost per accident is relatively low. Hence, the compromise suggested in (4.5) attempts to eliminate such undesirable paths by minimizing the risk per accident, while placing suitable upper bounds on the total expected consequence as well as on the relative frequency of accidents. In this context, for example, note that if two paths \wp_1 and \wp_2 have the same value of the expected consequence, but the probability of an accident on \wp_1 is more than that on \wp_2 , then assuming that all accidents are catastrophic, as seen from (4.4), the conditional expectation objective function would select \wp_1 over \wp_2 . While this might appear to be counterintuitive, observe that the condition $\hat{E}_{\wp_1} = \hat{E}_{\wp_2}$ while $\hat{P}(\wp_1) > \hat{P}(\wp_2)$ asserts that the average weighed consequence (weighted by its corresponding relative probability of occurrence), which is \hat{CE}_{\wp} in (4.4), is in fact smaller for path \wp_1 than for path \wp_2 . Hence, given that both the total expected consequence and the path accident probabilities are within acceptable limits, the decision maker might prefer the path \wp_1 that has the smaller expected consequence per accident in order to contain the extent of damage when an accident occurs. Moreover, as indicated above, since our objective function focuses on catastrophic accidents, it reflects the expected consequence per catastrophic accident or the average weighed catastrophic consequence on a path. Hence, it tends to better promote the appeal of the conditional risk objective function while somewhat ameliorating its counterintuitive effects.

The approach suggested in (4.5) is recommended to be just one other viable option that a decision maker might wish to consider in selecting a route for hazmat shipment. As an alternative to solving (4.5), the decision maker might wish to solve the complimentary problem of minimizing the expected consequence or the path accident probability subject to a constraint on the conditional risk being no more than a specified value, where this value itself might be based on the optimal objective value for (4.5). As another alternative, considering the bicriterion problem of minimizing the expected and the conditional risks, for example, the decision maker could rank some k best solutions to the shortest path problem of minimizing the approximation \hat{E}_{\wp} to E_{\wp} given by (4.4) over $\wp \in G$, using a k -shortest path algorithm (see Yen, 1971, Shier, 1979, or Skiscim and Golden, 1989), and then evaluate these paths with

respect to the (approximate or even exact) conditional risk objective function in order to select a route. Note that this strategy could itself be used to solve (4.5), by ranking or enumerating all possible paths that satisfy the side-constraints. However, depending on ν and η , this could be computationally prohibitive as compared to an *implicit* enumeration branch-and-bound approach as proposed herein.

Yet another approach would be to design an iterative routine that would actually generate some efficient solutions (see Steuer, 1986) for the bicriterion problem to minimize $\{\hat{E}_{\wp}, \hat{CE}_{\wp}\}$, using the previous remarks. For example, consider the problem (4.5), and let us assume that $\eta = \infty$. If we set $\nu = \nu^* = \min\{\hat{E}_{\wp} : \wp \in G\}$, the solution to this problem gives an efficient solution. Now, by considering the ranked k -best objective values of this latter shortest path problem, in order $\nu_1 < \nu_2 < \dots < \nu_k$, we can solve (4.5) with ν set at these values sequentially. Then, so long as the values of \hat{CE}_{\wp} drop sequentially, we would be tracing solutions on the efficient frontier. Solutions on the efficient frontier can also be generated by solving the problem (4.5) using any value of ν , and then considering the complimentary problem to

Minimize $\{\hat{E}_{\wp} : \wp \in G, \hat{CE}_{\wp} \leq \nu'\}$, where ν' is the optimal objective value obtained for the problem (4.5).

Hence, Problem (4.5) could be used as one in a repertoire of models that a decision maker might use in composing a strategy for selecting a path to route hazmat. We now proceed to present a suitable mathematical representation for (4.5), and develop an algorithm to solve this problem.

4.3 A Discrete Fractional Programming Optimization Approach

We begin our analysis of Problem (4.5) by presenting a basic formulation for it. For each node $k \in N$, define $F(k)$ to be the set of arcs emanating out of node k , i.e., the "forward arc set"

for node k , and similarly, define $R(k)$ to be the "reverse arc set" for node k . Throughout, we assume that the network has been preprocessed so that

$$R(O) \equiv F(D) \equiv \emptyset \quad (4.6)$$

Also, define the decision variables as follows:

$$x_a = \begin{cases} 1, & \text{if arc } a \text{ is in the prescribed O - D path} \\ 0, & \text{otherwise} \end{cases} \quad \forall a \in A.$$

Furthermore, denote by A_c the set of critical links in A that have consequences exceeding some specified critical level C^* . The *conditional expectation problem (CEP)* given by (4.5) can then be formulated as follows:

$$\text{CEP: Minimize } \frac{\sum_{a \in A_c} p_a x_a c_a}{\sum_{a \in A_c} p_a x_a} \quad (4.7a)$$

$$\text{subject to: } \sum_{a \in F(k)} x_a - \sum_{a \in R(k)} x_a = \begin{cases} 1, & \text{if } k \equiv O \\ -1, & \text{if } k \equiv D \\ 0 & \forall k \text{ otherwise} \end{cases} \quad (4.7b)$$

$$\sum_{a \in A} p_a x_a c_a \leq \nu \quad (4.7c)$$

$$\sum_{a \in A} p_a x_a \leq \eta \quad (4.7d)$$

$$x \equiv (x_a, a \in A) \in X \quad (4.7e)$$

$$x \text{ binary} \quad (4.7f)$$

where the set X in (4.7e) represents a suitable set of (linear) subtour elimination constraints as described below. Observe that the objective function in (4.7a) is both pseudoconvex and pseudoconcave. Thus, when (4.7f) is relaxed to $0 \leq x \leq e$, where e is a vector of ones, a local

minimum would also be a global minimum, and given feasibility, an optimum would be attained at an extreme point solution (see Bazaraa et al., 1993).

We will now analyze the subtour elimination constraints (4.7e). First of all, note that without such constraints, it is entirely possible that an optimal solution contains cycles of the types C_1 and C_2 shown in Figure 4.2, where, by the nature of the fractional objective function, these cycles help to reduce (4.7a) below the corresponding value of the simple (i.e., loopless) O-D path.

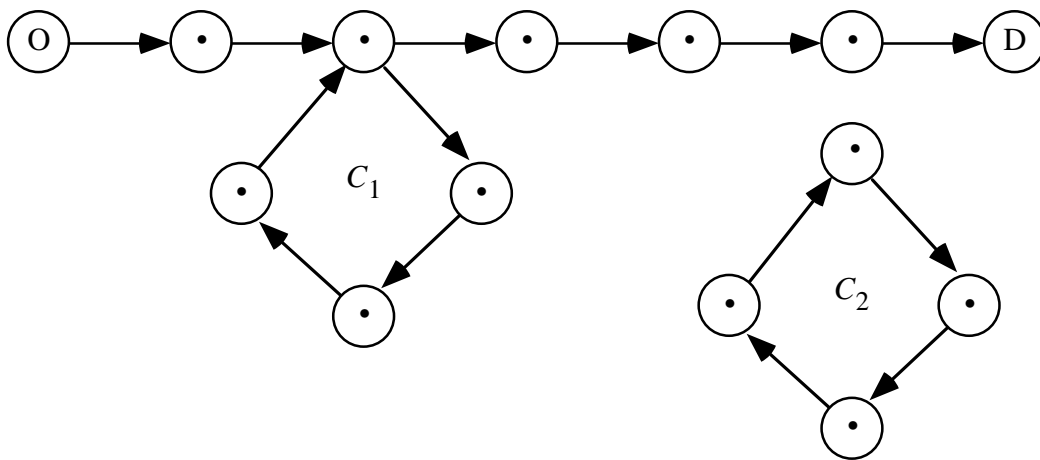


Figure 4.2. Cycles among arcs with unit flows.

The subtour elimination constraints can take on several forms. One simple form of these constraints that can be used are the Miller-Tucker-Zemlin (MTZ) constraints (see Nemhauser and Wolsey, 1988). The standard MTZ constraints yield weak continuous relaxations in the context of Traveling Salesman problems (TSP), but as shown by Desrochers and Laporte (1991), these constraints can be tightened by lifting them into facets for the underlying TSP polytope. These lifted constraints are applicable in our situation as well, and are incorporated as constraints (4.9e), (4.9i), and (4.9j) in the model representation given below, where y_k , for any $k \in N$, denotes the rank-order in which node k is visited (being arbitrary if it is not visited), with $y_O \equiv 1$. Here we have adopted the notation that $n \equiv |N|$, each link $a \equiv (t(a), h(a))$ has been identified by its tail node $t(a)$ and its head node $h(a)$, and for each such link, \bar{a} denotes the reverse link $(h(a), t(a))$, if it exists. (If the latter link does not exist, then $x_{\bar{a}} \equiv 0$. (Note that the

arcs in A specifically include both the oppositely directed arcs corresponding to bidirected links.) Also, x_{Ok} denotes the variable x_a for $a \equiv (O, k)$, $\forall k \in N - O$.

A more effective form of subtour elimination constraints are the Dantzig-Fulkerson-Johnson constraints (see Nemhauser and Wolsey, 1988), given as follows (noting (4.6)):

$$X = \{x: \sum_{a \in A_S} x_a \leq |S| - 1 \quad \forall S \subset N, \{O, D\} \cap S = \emptyset, |S| \geq 2\}, \quad (4.8)$$

where $A_S = \{a \in A: \{t(a), h(a)\} \subseteq S\}$. There are, however, an exponential number of constraints within (4.8). To remedy this situation, only a subset of these constraints are included in the model as constraints (4.9f), where the set $S^* \subseteq \{S: S \subset N, \{O, D\} \cap S = \emptyset, |S| \geq 2\}$ is prescribed by the algorithmic statement given below. In addition, we include another set of constraints given by (4.9g) that, in particular, are specifically designed to eliminate type C_1 cycles (see Figure 4.2). Note that similar to (4.9f), although these constraints are implied by the lifted MTZ constraints in the *integer* sense, they might serve to tighten the *continuous relaxation* of the problem. This can indeed be the case as illustrated in the application described in Section 4.4. We remark here that the symmetric constraints $\sum_{a \in F(k)} x_a \leq 1 \quad \forall k \in N - \{D\}$ are implied in the continuous sense by (4.9g) and (4.9b), and so need not be included in the model formulation. In this connection, note that as illustrated in Section 4.3, even if η is selected so that $\hat{P}(\emptyset) \leq \eta$ is implied (in the discrete sense) by $\hat{E}_{\emptyset} \leq \nu$ for the chosen value of ν over all $\emptyset \in G$, the constraint $\hat{P}(\emptyset) \leq \eta$ might not be implied in the continuous sense and hence, might aid in further tightening the relaxation to (4.5).

The resulting proposed model is stated below. For a problem having n nodes and m arcs, this problem has $m + n - 1$ variables and $4n + m + |S^*| - 1$ constraints, in addition to the variable-bounding restrictions (4.9h). This model is a valid representation of (4.7), and hence of (4.5), since the lifted MTZ constraints (4.9e), (4.9i), and (4.9j) provide a formulation for (4.7e) as established by Desrochers and Laporte (1991), and constraints (4.9f) and (4.9g) are additional

implied inequalities that aid in tightening the continuous relaxation of the model (when (4.9k) is deleted).

$$\text{CEP:} \quad \text{Minimize} \quad \frac{\sum_{a \in A_c} p_a c_a x_a}{\sum_{a \in A_c} p_a x_a} \quad (4.9a)$$

$$\text{subject to:} \quad \sum_{a \in F(k)} x_a - \sum_{a \in R(k)} x_a = \begin{cases} 1, & \text{if } k = O \\ -1, & \text{if } k = D \\ 0 \quad \forall k, & \text{otherwise} \end{cases} \quad (4.9b)$$

$$\sum_{a \in A} p_a c_a x_a \leq v \quad (4.9c)$$

$$\sum_{a \in A} p_a x_a \leq \eta \quad (4.9d)$$

$$y_{h(a)} \geq y_{t(a)} + 1 + (x_a - 1)(n - 1) + (n - 3)x_{\bar{a}} \quad \forall a \in A \quad (4.9e)$$

$$\sum_{a \in A_s} x_a \leq |S| - 1 \quad \forall S \in S^* \quad (4.9f)$$

$$\sum_{a \in R(k)} x_a \leq 1 \quad \forall k \in N - \{O\} \quad (4.9g)$$

$$0 \leq x \leq e, \quad y_O \equiv 1 \quad (4.9h)$$

$$(3 - x_{OD}) \leq y_D \leq 2 + (n - 2)(1 - x_{OD}) \quad (4.9i)$$

$$(3 - x_{Ok}) \leq y_k \leq 2 + (n - 3)(1 - x_{Ok}) \quad \forall k \in N - \{O, D\} \quad (4.9j)$$

$$x \text{ binary.} \quad (4.9k)$$

We now design a branch-and-bound algorithm to solve Problem CEP. In this approach, each node of the branch-and-bound tree represents two segments of a partial simple path, one originating at node O and the other terminating at node D . Consequently, each branch-and-bound tree node q has a *shifted origin*, designated $SO(q)$, and a *shifted destination*, designated $SD(q)$, where the partial path is composed of the segment starting at O and terminating at $SO(q)$,

along with the segment starting at $SD(q)$ and terminating at D . (For the sake of simplicity, whenever the dependence on q is clear from the context, we will simply write $SO(q)$ and $SD(q)$ as SO and SD , respectively.) Note that if $|F(SO)| = 1$, then we can fix $x_{F(SO)} = 1$ and shift the origin to the head-node of the arc $F(SO)$ and repeat this check. Likewise, if $|R(SD)| = 1$, then we can set $x_{R(SD)} = 1$ and shift the effective destination to the tail-node of the arc $R(SD)$. Furthermore, note that each time some variable x_a is set at one, the variables associated with all the *other* arcs in $R[h(a)]$ and in $F[t(a)]$ can be set at zero.

We also perform another preprocessing operation as follows, to possibly eliminate additional network nodes and arcs from any given branch-and-bound tree node subproblem. Suppose that SO and SD are, respectively, the shifted origin and destination for any such subproblem, and that the residual network at hand (corresponding to the free variables) satisfies (4.6) for $O \equiv SO$ and $D \equiv SD$. Then, by sequentially tracing forward stars starting from SO (see Bazaraa et al., 1990, for standard network-related definitions), we can easily find the set of nodes that are reachable from SO in the residual network. Likewise, we can sequentially scan the reverse star lists starting from node SD to find the set of nodes in the residual network that can reach node SD . Denote the intersection of these two node sets as $REACH(SO \rightarrow SD)$. Observe that this intersection represents the set of all those nodes that could possibly lie on paths connecting SO and SD . All other nodes could then be eliminated, along with their incident arcs. Such a preprocessing can substantially tighten the bounds derived via \overline{CEP} , the continuous relaxation of CEP , possibly eliminating, in particular, cycles of type C_2 that can arise which are not associated with the SO to SD path, and sometimes even cycles of type C_1 . The proposed branch-and-bound algorithm is now described in detail below. Finite termination is guaranteed by the finiteness of O-D paths in the network.

Branch-and-Bound Algorithm

INITIALIZATION. Let the set of active nodes $Q = \{0\}$ contain the starting node (zero) of the branch-and-bound (**B&B**) tree, and let its corresponding shifted origin and shifted

destination be $SO(0) = O$ and $SD(0) = D$, respectively. The partial path comprised of nodes and arcs corresponding to this B&B node is $\wp(0) = \{O, D\}$. Let the nodes in this partial path be denoted by $N(0) = \{O, D\}$, and let $L(0) \equiv 2$ denote the cardinality of $N(0)$. Let $q = 0$ represent the selected active node, record the incumbent solution $x^* = \emptyset$ of objective value $v^* = \infty$, and choose a termination tolerance $\varepsilon > 0$. Construct the set S^* for (4.9f) as follows. First, solve the continuous relaxation \overline{CEP} of CEP without the constraints (4.9f), and hence obtain a flow \bar{x} that satisfies the conservation constraints (4.9b). (The algorithm of Charnes and Cooper, 1962, can be used to solve \overline{CEP} as an equivalent linear program.) Using the procedure in Sherali et al. (1994), decompose this (possibly fractional) flow into O-D path flows and cycle flows. For each cycle thus detected, if any, include the corresponding constraint (4.9f) in the problem. This formulates the set S^* to be used for the remainder of the algorithm. Proceed to Step 1.

STEP 1. (Preprocessing and Bounding Step).

Step 1(a). (Preprocessing): For the branch-and-bound tree node q , given $SO(q) \equiv SO$, and $SD(q) \equiv SD$, and given the partial path $\wp(q)$ having $\lambda(q)$ nodes comprising the set $N(q)$ along with the connecting arcs that define the two segments of $\wp(q)$, set $x_a = 1 \ \forall \ a \in \wp(q)$, and set

$$x_a = 0 \ \forall \ a \notin \wp(q) \text{ such that } h(a) \in [N(q) - \{SD\}] \text{ or } t(a) \in [N(q) - \{SO\}]. \quad (4.10)$$

If $|F(SO)| = 1$, then set $x_{F(SO)} = 1$, shift the origin to the node $h[F(SO)]$, and accordingly, update $\wp(q)$, $N(q)$ and $\lambda(q)$. Similarly, if $|R(SD)| = 1$, then set $x_{R(SD)} = 1$, shift the destination to the node $t[R(SD)]$, and accordingly, update $\wp(q)$, $N(q)$ and $\lambda(q)$. For each shift, eliminate arcs by applying (4.10). Continue until no further shifts are possible. If $SO=SD$ in this process, then $\wp(q)$ represents an optimal path for this node subproblem. Compute its objective value, update the incumbent x^* and its value v^* , if necessary, and go to Step 4.

Next, scan the forward and reverse stars of the residual network to determine the set $REACH(SO \rightarrow SD)$ as described above. If $REACH(SO \rightarrow SD) = \emptyset$, proceed to Step 4.

Otherwise, update the residual network by eliminating all nodes not belonging to $REACH(SO \rightarrow SD)$ along with any incident arcs. If any further origin or destination shifts are possible, then repeat this preprocessing step. Having done this, solve the shortest path problems on the residual network to minimize the left-hand sides of (4.9c) and (4.9d), in turn, in order to determine if these constraints have a feasible completion. If these subproblems yield feasible solutions to Problem CEP, update the incumbent x^* and its value v^* . If either of these constraints cannot be satisfied, then transfer to Step 4. Otherwise, proceed to Step 1(b).

Step 1(b). (Lower Bounding Problem): Construct the node subproblem CEP corresponding to the available residual network, with (N, A) representing the revised node and arc sets of the residual network, $R(k)$, $F(k)$, and A_s all being defined with respect to this residual network, and with $n \equiv |N|$, $O \equiv SO$, and $D \equiv SD$. Also, incorporate the appropriate constants in the numerator and denominator of (4.9a), as well as in (4.9c) and (4.9d), corresponding to the variables $x_a = 1$ for $a \in \mathcal{P}(q)$. Furthermore, retain in (4.9f) only those constraints from S^* that correspond to sets $S \subseteq [N - \{SO, SD\}]$. Formulate and solve the associated relaxation \overline{CEP} , and let (\bar{x}, \bar{y}) be the solution obtained of objective value \bar{v} . If \bar{x} is integral, then it is feasible to the original problem and optimal for the node subproblem. Update x^* and v^* , if necessary, and then fathom this node by proceeding to Step 4. Otherwise, attempt to determine an improved solution via the following heuristic.

Step 1(c). (Upper Bounding Heuristic): Having solved the continuous relaxation \overline{CEP} for the residual network as an equivalent linear program (see Charnes and Cooper, 1962), dualize all the constraints using the corresponding optimal Lagrange multipliers except for the network flow conservation constraints (4.9b) along with (4.9h). Solve this network flow problem and augment the resulting SO - SD path solution using the appropriate fixed values of zero or one for the arcs not in the residual network to obtain a complete solution \hat{x} . If \hat{x} is feasible to (4.9c) and (4.9d), compute its objective value in (4.9a), and update the incumbent

solution x^* and its value v^* , if necessary. If $\bar{v} \geq v^* - \varepsilon$, go to Step 4. Otherwise, proceed to Step 2.

Step 2. (Branching Step). Remove q from Q . Create as many new nodes as there are successors of $SO(q)$ in the residual network. For each such branch-and-bound node q' created, put q' in Q , let $SO(q')$ be the particular successor node of $SO(q)$ that corresponds to q' , let $SD(q') \equiv SD(q)$, $N(q') = N(q) \cup \{SO(q')\}$ and $L(q') = L(q) + 1$, let $\wp(q') = \wp(q) \cup \{(SO(q), SO(q')), SO(q')\}$, let $v_0(q') = \bar{v}$, set indicators $ind_1(q') = \bar{x}_{[SO(q), SO(q)]}$ and $ind_2(q') = P_{[SO(q), SO(q')] }^c_{[SO(q), SO(q)]}$ to assist in selecting branch-and-bound tree nodes, and put $x(q') = \bar{x}$ if $\bar{x}_{[SO(q), SO(q)]} = 1$, and let $x(q') = \emptyset$ otherwise. Proceed to Step 3.

Step 3. (Node Selection Step). If $Q = \emptyset$, terminate the algorithm with x^* as an optimal solution. If $Q \neq \emptyset$, pick a node

$$q \in \underset{t \in Q}{\operatorname{arglexmin}} \{v_0(t), -L(t), -ind_1(t), ind_2(t)\}.$$

If $x(q) = \emptyset$, go to Step 1. If $x(q) \neq \emptyset$, and this solution is feasible to the node q subproblem that has been revised as described at Step 1, then $x(q)$ continues to be optimal to this subproblem. Hence, put $\bar{x} = x(q)$, $\bar{v} = v_0(q)$, and return to Step 2. Else, return to Step 1.

Step 4. (Fathoming Step). Remove node q from Q and transfer to Step 3.

4.4 Illustrative Example

Consider the following example shown in Figure 4.3, where the tuple against each link $a \in A$ represents the data (p_a, c_a) . The variables x_1, x_2, x_3, x_4 , and x_5 represent the flows on the links (1, 2), (2, 4), (1, 3), (3, 4), and (2, 3), respectively. Also, the variables y_1, y_2, y_3 , and y_4 relate to nodes 1, 2, 3, and 4, respectively.

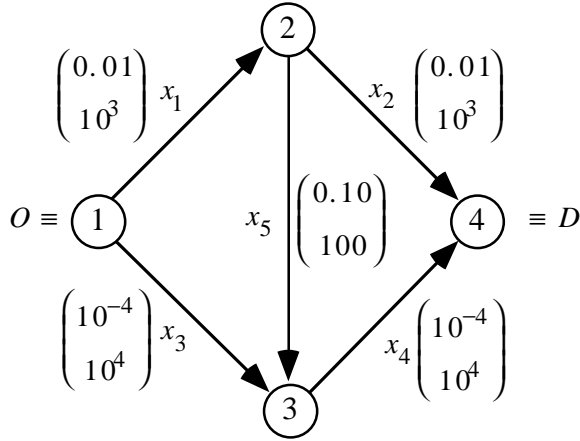


Figure 4.3. Illustrative Example of Section 4.3.

For the sake of exposition, we enumerate below the path information for this problem (all accidents are assumed to be catastrophic).

Path \wp	\hat{E}_{\wp}	\hat{CE}_{\wp}	$\hat{P}(\wp)$
1 \rightarrow 2 \rightarrow 4	20	10^3	0.02
1 \rightarrow 3 \rightarrow 4	2	10^4	0.0002
1 \rightarrow 2 \rightarrow 3 \rightarrow 4	21	190.736	0.1101

Hence, Path 1 \rightarrow 3 \rightarrow 4 is optimal for $2 = v < 20$ and $\eta = 0.0002$, Path 1 \rightarrow 2 \rightarrow 4 is optimal for $20 = v < 21$ and $\eta = 0.02$, and Path 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 is optimal for $v = 21$ and $\eta = 0.1101$. Furthermore, since the paths ranked in increasing values of \hat{E}_{\wp} sequentially drop in their respective values of \hat{CE}_{\wp} , all three paths lie on the efficient frontier in this case for the underlying bicriterion problem to minimize $\{\hat{E}_{\wp}, \hat{CE}_{\wp} : \wp \in G\}$.

Now, let us pick $v = 20$ and $\eta = 0.02$, and solve the corresponding problem using the proposed branch-and-bound algorithm.

Initialization and Step 1. Beginning with $Q = \{0\}$, and selecting $q = 0$, since no preprocessing operation is applicable, we first solve \overline{CEP} at Step 1(b). This yields $\bar{x} = (0.180, 0, 0.82, 1, 0.180)$, and $\bar{y} = (1, 2.82, 2.180, 3.180)$, with objective value (lower bound) equal to 271.156. Note that although the constraint (4.9d) is implied (for the discrete problem) by the other constraints in the model for the chosen values of v and η as seen from the above table, if this constraint is deleted, the resulting continuous relaxation yields the solution $\bar{x} = (0.947, 0,$

0.053, 1, 0.947), and $\bar{y} = (1, 2.05, 2.947, 3.947)$ with an objective value of 191.726. Note that $\sum_{a \in A} p_a \bar{x}_a = 0.104 > 0.02 \equiv \eta$ for this solution, i.e., (4.9d) is violated. Hence, the constraint (4.9d) indeed serves to tighten the continuous relaxation, although it is redundant in the discrete sense. Furthermore, observe that evidently, the lower bound is being driven by the path $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$ which is marginally infeasible since $v = 20$ and $\hat{E}_\varnothing = 21$ for this path. Also, in this example, the only admissible constraint in (4.8) or (4.9f) is $x_5 \leq 1$ corresponding to $S = \{2, 3\}$, which is implied by (4.9h), and so, the constraint set (4.9f) is null.

Now, instead of using the heuristic of Step 1(c), let us derive feasible solutions directly via the branch-and-bound algorithm for the sake of illustration. (Note that since the lower bound is lesser than the optimal value in this case, the procedure would be unaffected by using Step 1c.) Hence, proceeding to Step 2, we create nodes 1 and 2 as follows.

Step 2: Branch on node 0, creating nodes 1 and 2 corresponding to the branching restrictions $x_1 = 1$ and $x_3 = 1$, respectively (see Figure 4.3). This gives $Q = \{1,2\}$. For $q' = 1$, we get $SO(1) = 2$, $SD(1) = 4$, $N(1) = \{1,2,4\}$, $\lambda(1) = 3$, $\varnothing(1) = \{1, (1,2), 2,4\}$, $v_0(1) = 271.156$, $ind_1(1) = \bar{x}_1 = 0.180$, $ind_2(1) = 10$, and $x(1) = \emptyset$. Similarly, for $q' = 2$, we get $SO(2) = 3$, $SD(2) = 4$, $N(2) = \{1,3,4\}$, $\lambda(2) = 3$, $\varnothing(2) = \{1, (1,3), 3,4\}$, $v_0(2) = 271.156$, $ind_1(2) = \bar{x}_3 = 0.82$, $ind_2(2) = 1$, and $x(2) = \emptyset$.

Step 3: Since $v_0(1) = v_0(2)$, $\lambda(1) = \lambda(2)$, and $-ind_1(1) > -ind_1(2)$, we select B&B node 2 and return to Step 1.

Step 1: Fixing $x_3 = 1$ implies by (4.10) that $x_1 = 0$ and $x_5 = 0$. Also, with $SO = 3$, we find that $|F(SO)| = 1$ which means that we can fix $x_{(3,4)} = 1$, and so, $x_2 = 0$ by (4.10). Now, $SO = SD = 4$ and so an optimal solution for this node is obtained via the preprocessing step 1(a) as $\bar{x} = (0, 0, 1, 1, 0)$ having an optimal objective function value of 10^4 . Node 2 is fathomed because this solution is all integer, and we now have an incumbent solution $x^* = (0, 0, 1, 1, 0)$, with $v^* = 10^4$.

Step 4: Fathoming node 2 $\Rightarrow Q = \{1\}$.

Step 3: Select B&B node 1.

Step 1: Fixing $x_1 = 1$ implies by (4.10) that $x_3 = 0$. The residual network has $N = \{2, 3, 4\}$ and $A = \{(2,3), (2,4), (3,4)\}$. Also, $SO = 2$ and $SD = 4$. Incorporating this in \overline{CEP} and re-solving, yields the complete x -solution $\bar{x} = (1, 1, 0, 0, 0)$ having an objective function value of $10^3 < v^* = 10^4$. Hence, we update the incumbent solution to $x^* = (1, 1, 0, 0, 0)$ with $v^* = 10^3$.

Step 4: Fathoming node 1 $\Rightarrow Q = \{\emptyset\}$.

Step 3: Since $Q = \emptyset$, the algorithm terminates. The optimal solution, found at B&B node 1, has an objective function value of 10^3 . The corresponding path is $1 \rightarrow 2 \rightarrow 4$.

4.5 Computational Experience

To illustrate the performance of the proposed branch-and-bound algorithm for solving Problem (4.5), we conducted the following experiment. We randomly generated layered networks having $p+2$ layers of nodes for different values of p where the layers are numbered $0, 1, \dots, p, p+1$, and where nodes O and D lie respectively at layers 0 and $p+1$, with the number of nodes at layer λ being given by 2^λ for $\lambda = 1, \dots, p$. The network arcs are as depicted in Figure 4.4, where the forward star of each node at layers $0, 1, \dots, p - 1$ has cardinality r , for different values of r , and the nodes at layer p are all individually connected to the destination node D . Hence, there exist r^p possible paths in the network, and there are $2 + r(r^p - 1) / (r - 1)$ nodes and $r^p + r(r^p - 1) / (r - 1)$ arcs in the network. The values of c_a and p_a for each arc in the network were uniformly generated on the intervals $[10^2, 10^4]$ and $[10^{-6}, 10^{-4}]$, respectively. Using this data, we computed the minimal possible values of the expected consequence and the accident probability on a path, and accordingly, we let v and η be 10% above these respective values.

Table 4.1 presents the results obtained upon applying the proposed branch-and-bound algorithm for some sample problem sizes as determined by p and r . The results exhibit that fairly good lower bounds are obtained for these test problems, resulting in a reasonable solution effort. Note that we have used a very stringent termination tolerance of $\varepsilon = 10^{-6}$ for these test runs. Hence, for example, even though the initial lower bound and the incumbent solution values for the last problem in Table 4.1 differ by only 0.79 (or 0.029%), the algorithm still enumerates 15 branch-and-bound nodes. In practice, a percentage-based termination tolerance (e.g., $\varepsilon = 0.1\%$ of the incumbent value) could be more appropriately used. Note also that the purpose of this experiment was merely to illustrate the performance of the algorithm on some sample test cases. For deducing definitive conclusions, a more detailed computational study would be required. This is beyond the scope of the present study and will be pursued in further research.

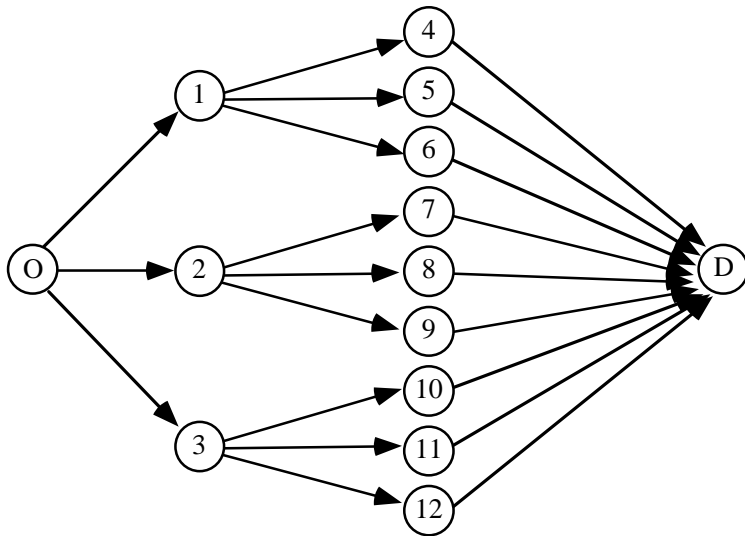


Figure 4.4. Sample Test Network for $p = 2$ and $r = 3$.

Table 4.1. Computational Results for Randomly Generated Test Problems.

Network Parameters (p, r)	# Nodes	# Arcs	Node Zero LB	Optimal Objective Values	# B&B Nodes Enum	cpu (secs)*
(3, 2)	16	22	3848.32	4034.19	3	0.386
(4, 2)	32	46	6681.04	7432.32	5	0.864
(5, 2)	64	94	2753.78	2753.78	1	0.553
(6, 2)	128	190	4347.13	4581.66	6	2.378
(7, 2)	256	382	2673.77	2673.77	1	0.253

(2, 3)	14	21	1297.37	1297.37	1	0.182
(3, 3)	41	66	3196.79	3395.08	4	0.715
(4, 3)	122	201	1760.19	1760.19	1	1.151
(5, 3)	365	606	1877.82	1944.98	8	7.484
(6, 3)	1094	1821	2444.01	2756.15	10	35.89
(7, 3)	3281	5466	2738.16	2738.95	16	268.37

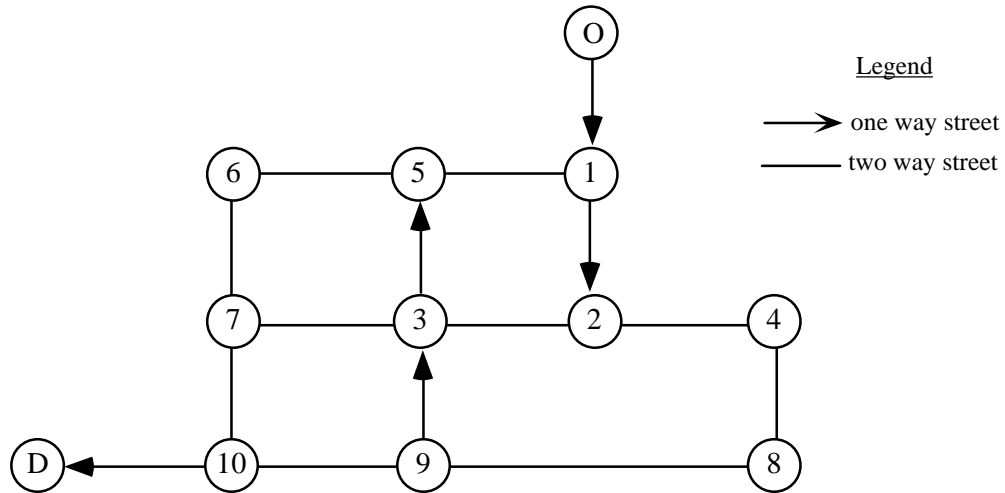
* The runs were made on a SunSparc 10-41 workstation.

4.6 Case Study

The proposed methodology was tested using a network representing the city streets of Bethlehem, Pennsylvania. This section describes how the data were procured and how the information required by the model was extracted therefrom, and analyzes one particular scenario using a specified origin-destination pair.

In this case study, we examined accidents involving gases in bulk, and in particular, chlorine gas. Over a five-year period spanning from 1983-1988, data were provided for the Bethlehem city by the Center for Transportation Research at Virginia Polytechnic Institute and State University. For every arc, we had available the daily average truck traffic (DATF), the number of truck accidents, the residential and employed populations, and the length of the arc segment. The remainder of the data pertaining to accident probabilities and consequences as discussed below, were found in the Midwest Research Institute (1989) report, and in a paper by Saccomanno et al. (1990). Details on these data are given in Brizendine (1994). The original street network was comprised of 47 nodes and 97 arcs and is depicted in Brizendine (1994). Here, in Figure 4.5, we present a collapsed representation of this network that is obtained by aggregating nodes and links that are connected in series. Note that this network has been screened to omit roadways that violate regulatory guidelines. (The referenced U.S. Department of Transportation publication (1994) provides further information in this regard.) Such guidelines advise against hazmat being transported on streets that contain, for example, a school or hospital or a tunnel.

To construct our model representation, we need values of the accident probability p_a and the consequence c_a for each arc $a \in A$. These are discussed in turn below.



Link Index #	Link	p_a $\cdot (10^{-6})$	c_a
1	(O, 1)	1.00	1.5720
2	(1, 2)	1.00	2.0342
3	(1, 5)	1.00	1.5720
4	(2, 3)	1.00	2.7286
5	(2, 4)	1.00	3.6151
6	(3, 5)	6.85	0.4466
7	(3, 7)	1.00	3.3672
8	(4, 8)	5.37	1.6728
9	(5, 6)	1.00	2.2552
10	(6, 7)	1.00	3.7312
11	(7, 10)	4.74	2.6746
12	(8, 9)	16.62	3.7516

Path Index #	Path \wp	\hat{E}_{\wp} $\cdot (10^{-6})$	$\hat{P}(\wp)$ $\cdot (10^{-6})$	\hat{CE}_{\wp}
1	{O,1,5,6,7,10,D}	24.73	12.39	1.99
2	{O,1,2,3,5,6,7,10,D}	30.98	20.24	1.53
3	{O,1,2,3,7,10,D}	25.30	12.39	2.04
4	{O,1,2,4,8,9,3,5,6,7,10,D}	125.41	50.73	2.47
5	{O,1,2,4,8,9,3,7,10,D}	119.73	42.88	2.79

13	(9, 3)	8.50	2.6130	6	{O,1,2,4,8, 9,10,D}	97.06	39.60	2.45
14	(9, 10)	10.96	1.42188					
15	(10, D)	3.65	0.7994					

Figure 4.5. Illustration and Computations for the Case Study.

4.6.1 Computation of Accident Probabilities

For deriving the accident probability p_a for each $a \in A$, we used the formula

$$p_a = \frac{\text{Number of hazmat truck accidents on arc } a \text{ over the five year period}}{\text{Number of hazmat truck shipments on arc } a \text{ over the five year period}}, \forall a \in A, \quad (3.11)$$

where the denominator in (4.11) was computed using the daily average truck traffic data. Zero values for p_a were replaced by relatively small values equal to 1.00×10^{-6} to avoid misleading results, and to ensure that the model objective function was well defined. Alternatively, to evaluate (4.11), we could have used the national default values that are given in the Midwest Research Institute (1989) report. Table 20 from Task B of this document contains truck accident rates (i.e., number of truck accidents per truck-mile), given by highway type.

4.6.2 Computation of Consequences

We previously defined c_a to be the expected consequence (cost) incurred if an accident occurs on link $a \in A$. For the present application, the “consequence” is considered to be the expected number of deaths resulting from an accident on link a . If we define $\beta_a(m)$ to be the probability of m fatalities, given that a truck accident has occurred on arc $a \in A$, then $c_a =$

$$\sum_m m\beta_a(m). \text{ To compute } \beta_a(m) \text{ for all possible values of } m, \text{ we used the relationship}$$

$$\beta_a(m) = P(m \text{ fatalities} \mid \text{a truck accident and a release of noxious gases on arc } a) \times$$

$$P(\text{release of noxious gases} \mid \text{truck accident on arc } a), \quad (4.12)$$

noting that the probability of a release-related fatality when no release of noxious gases occurs can be assumed to be zero. The value of the second term in Equation (4.12) can be found in

Table 42 of the main volume of the Midwest Research Institute (1989) report. Since our focus is on chlorine gas, this value could have been taken as 0.081, but using the same value for each arc would not have distinguished between the different types of roadways contained in the network, e.g., city streets versus interstate highways. An adjustment was made accordingly by using the multiplicative adjustment factors given in Table 62 of the Midwest Research Institute (1989) report. Using that data and multiplying the corresponding factors by 0.081, we obtain

$$\begin{array}{ll}
 \text{Interstate highway:} & \text{P}(\text{release} \mid \text{truck accident}) = 0.093 \\
 \text{U.S. or state route:} & \text{P}(\text{release} \mid \text{truck accident}) = 0.106 \\
 \text{Country road:} & \text{P}(\text{release} \mid \text{truck accident}) = 0.105 \\
 \text{City street:} & \text{P}(\text{release} \mid \text{truck accident}) = 0.027 .
 \end{array} \quad (4.13)$$

As far as the first term in Equation (4.12) is concerned, we consider the event of a fatality, given that exposure to chlorine gas has occurred, to be a Bernoulli event with probability p , and let the corresponding exposed population on link $a \in A$ be M_a . (Note that this is an approximation in the sense that the probability of a fatality given an exposure to chlorine gas would actually depend on the distance from the accident site.) Then we have

$$\text{P}(m \text{ fatalities} \mid \text{truck accident and release on arc } a) = \binom{M_a}{m} p^m (1-p)^{M_a-m} \quad (4.14)$$

and so, using (4.12) - (4.14), we get

$$\begin{aligned}
 c_a &= \sum_m m \beta_a(m) = \text{P}(\text{release} \mid \text{truck accident on arc } a) \sum_m m \cdot \binom{M_a}{m} p^m (1-p)^{M_a-m} \\
 &= [\text{Equation (13) value for arc } a] M_a p.
 \end{aligned} \quad (4.15)$$

The value of p can be derived using Table 5 from Saccomanno et al. (1990). If we use the median fatality rate for an instantaneous release of chlorine, then we have a rate of $p = 0.0827$ fatalities per capita among the exposed population. Then, for example, if the exposed population (motorists plus surrounding public) is $M_a = 2000$ persons for some city street a , we would compute $c_a = (0.027) (2000) (0.0827) = 4.4658$ using Equation (4.15).

In our study, the exposed population M_a , for each arc $a \in A$, was computed using 1980 U.S. Census data in the following manner. A map of the city street network was overlaid onto a map of the neighboring census tracts. For each arc, noting that an area of 6.0 square kilometers will be affected by a chlorine gas spill/leak (Saccomanno et al., 1990), we need to use a scaled circle having that area and slide it along the desired arc to find the population that could be potentially affected. The values of exposed population provided by the Center for Transportation Research considered an area covered by 0.33 km on either side of each arc in computing these values. For simplicity, we assumed that this same local density uniformly covers the corresponding area affected by a chlorine gas spill/leak, and so, we multiplied the given exposed population values by the factor $6/(0.66 L_a)$, where L_a is the length (km) of link a , in order to compute M_a . The values c_a , $a \in A$, were thus obtained using Equation (4.15). The corresponding aggregated values for the two-tuple (p_a, c_a) for the arcs in the equivalent collapsed network are given in Figure 4.5.

4.6.3 Analysis of the Case Study

For the given instance depicted in Figure 4.5, having computed the values (p_a, c_a) $\forall a \in A$ as above, we explicitly enumerated all the six possible O-D paths for the sake of illustration, and computed the corresponding values of \hat{E}_{\wp} , $\hat{P}(\wp)$ and \hat{CE}_{\wp} for each such path \wp . The results obtained are tabulated in Figure 4.5.

Note that the minimum value of the expected consequence is $24.73 \cdot 10^{-6}$ and is attained by the first path in Figure 4.5. The minimum value of the conditional expectation is 1.53 and is attained by the second path in Figure 4.5. (The exact computation of CE_{\wp} for this path yields a value which differs only in the sixth decimal place, and is therefore quite close to the stated approximate value. Also, Figure 4.5 gives values that are truncated to two decimal places.) Both these paths are efficient solutions with respect to the stated multicriteria problem, while all

other paths are dominated in this respect. For $\nu \in [24.73, 30.98) * 10^{-6}$, which represents a 25.27% variation over the minimal expected consequence, and/or for $\eta \in [12.39, 20.24) * 10^{-6}$, which represents a 63.3% variation over the minimal path accident probability, Path 1 is optimal for Problem CEP. For these parameter values, the other feasible path, Path 3, is uniformly dominated with respect to each of the expected consequence and conditional risk criteria. However, if $\nu \geq 30.98 * 10^{-6}$ and $\eta \geq 20.24 * 10^{-6}$, then Path 2 becomes the optimum solution for Problem CEP. Note that the difference between Paths 1 and 2 is that the direct route from node 1 to node 5 in Path 1 is replaced by the route segment $1 \rightarrow 2 \rightarrow 3 \rightarrow 5$ in Path 2. On this latter detour, the effective accident probability is $6.85 * 10^{-6}$, compared with $p_{(1,5)} = 1.00 * 10^{-6}$, but the effective consequence (which yields the same value of the expected consequence on this segment) is only 0.88, compared with $c_{(1,5)} = 1.57$. If the increase in the accident probability on this segment of (or on the overall) Path 2 is considered acceptable, then this reduced expected consequence, given that an accident occurs, might make Path 2 worth considering for further evaluation.

Next, let us examine the sensitivity of the optimal solution for Problem CEP to the value of the critical index C^* that is used for ascertaining whether an accident is catastrophic or not. For $\nu = 30.98$ and $\eta = 20.24$, so that Paths 1, 2, and 3 are feasible, the following table presents computations of the conditional risks for $C^* = 2, 2.5,$ and 3 fatalities.

	Conditional Risk \hat{CE}_{φ}		
C^*	Path 1	Path 2	Path 3
2	2.77	2.68	2.69
2.5	2.86	2.84	2.69
3	3.73	3.73	3.37

For $C^* = 2$, the conditional risks for the three paths are quite close to each other, although Path 2 is still optimal. However, for $C^* = 2.5$ and 3 fatalities, we now see that Path 3 becomes optimal to Problem CEP. Note that this path has the minimum of the maximum link consequences, and this feature becomes increasingly dominant in determining the relative values of the conditional risk as C^* increases. For these values of C^* , Path 2 is now no longer efficient, and is uniformly dominated with respect to all the three criteria of \hat{E}_φ , $\hat{P}(\varphi)$, and \hat{CE}_φ . Hence, this is another factor that the decision maker can explore in evaluating the different viable paths.

4.6.4 Model Results

Observe that if we were to apply the k -shortest path procedure to solve Problem CEP for different values of ν , we would enumerate the paths in the sequence 1, 3, 2, 6, 5, and 4 of their index numbers, as evident from Figure 4.5. These paths could then be explicitly evaluated with respect to their values of $\hat{P}(\varphi)$ and \hat{CE}_φ in order to determine an optimum to Problem CEP for specified values of ν and η .

To illustrate the application of the branch-and-bound algorithm of Section 4.2, we used $C^* = 0$, $\nu = 31 * 10^{-6}$ and $\eta = 21 * 10^{-6}$ and ran this procedure. Note that for the given O and D , the shifted origin and destination nodes are given by $SO = 1$ and $SD = 10$, respectively. Solving \overline{CEP} without (4.9f), we obtained the solution

$$x_{O,1} = \bar{x}_{1,2} = \bar{x}_{2,3} = \bar{x}_{3,5} = \bar{x}_{5,6} = \bar{x}_{6,7} = \bar{x}_{7,10} = \bar{x}_{10,D} = 1, \text{ and } \bar{x}_{ij} = 0 \text{ otherwise.}$$

This solution corresponds to Path 2 in Figure 4.5. The linear programming problem used for solving \overline{CEP} took 32 iterations using CPLEX 2.0. Since an integer answer is obtained, this solution is itself optimal to the overall problem.

For the sake of interest, the model \overline{CEP} was also run without (4.9g), which imposes a set of additional constraints that were added to tighten the LP relaxation. The following solution was obtained in this case:

$\bar{x}_{O,1} = \bar{x}_{1,2} = \bar{x}_{2,3} = \bar{x}_{3,5} = \bar{x}_{5,6} = \bar{x}_{6,7} = \bar{x}_{7,10} = \bar{x}_{10,D} = 1$ and $\bar{x}_{3,7} = \bar{x}_{7,3} = 0.421$,
with $\bar{x}_{ij} = 0$ otherwise.

Note that a cycle of type C_1 (see Figure 4.2) can be identified in this solution as $\{3 \rightarrow 7 \rightarrow 3\}$ via the fractional positive variable values. This fractional solution is infeasible to the constraint (4.9g) written for the node $k = 7$, for which the left-hand side of the inequality is 1.421. Hence, (4.9g) can indeed be useful in tightening the relaxation.

4.7 Summary, Conclusions and Recommendations for Future Research

This study has examined a routing model that attempts to reduce the risk of low probability-high consequence accidents. The model formulated minimizes the conditional expectation of a catastrophic outcome, i.e., the expected consequence given that a catastrophic accident has occurred, subject to the side-constraints that the expected value of the consequence (expected risk) is lesser than or equal to some specified value ν , and that the probability of an accident on the selected path is no more than some value η . The motivation for using this model, as well as cautionary remarks when doing so, have been explored. A suitable discrete fractional programming model that incorporates various classes of constraints that serve to tighten its continuous relaxation has been developed for this problem, and an iterative branch-and-bound solution algorithm has been proposed. This branch-and-bound algorithm employs various preprocessing techniques along with methods for deriving suitable lower and upper bounds on the problem, in order to compose a viable solution strategy. The approach has been illustrated on a numerical example, tested on some randomly generated problems, and has been applied to a real-life problem associated with a test case concerned with transporting hazmat through the city roadways of Bethlehem, Pennsylvania, for which, the issue of acquiring relevant data has also been addressed. The results and illustrations indicate that this approach is a viable

tool, and can provide some useful information to the decision maker in selecting a route for transporting hazmat material.

The proposed model can be enhanced along several directions. For example, uncertainties in data related to consequences, or the possibility of being diverted away from a selected route due to an incident can be considered. Also, congestion-related travel times can be modeled, and accident probabilities and consequences can be assessed based on an estimate of the time-of-day that an arc is likely to be traversed. This might also provide improved guidelines on prescribing departure times as well as routes that would be sensitive to time-dependent exposed populations. Such considerations, along with more detailed computational studies, are recommended for future research.