Stochastic Traveling Salesperson and Shortest Route Models with Safety Time

Lilit Mazmanyan[1], Dan Trietsch[2]

[1] College of Engineering, American University of Armenia, Yerevan, Armenia.

[2] College of Engineering, American University of Armenia, Yerevan, Armenia.

5 August 2013

Abstract

We address the stochastic traveling salesperson problem (TSP) with distances measured by travel time. We study how to select the best tour and due date for the minimization of fundamental safe scheduling objectives. *Model 1* requires minimizing the due date subject to a service level constraint. *Model 2* addresses a weighted trade-off between the due date and the expected tardiness. Both models require safety time and therefore the distribution of the tour length is important. In alternate formulations the due date is given and we maximize the service level for Model 1 or minimize the expected tardiness for Model 2. In an unpublished working paper (available as a web resource), we addressed normal travel times. In this paper we recap some of those results and extend them to lognormal travel times. In general, we show that Alternate 1 is equivalent to Model 1, but Alternate 2 is different to Model 2. For the normal distribution, we solve optimally for Model 2 and for Model 1 with service level 50% and higher by solving few deterministic TSP derived models. For other instances, including all lognormal models, we provide effective heuristics and tight performance guarantee certificates. As a by-product of our TSP analysis, we obtain comparable results for the shortest route problem.

Keywords: Traveling Salesperson Problem, Shortest Route Problem, Stochastic Scheduling, Safe Scheduling, Lognormal Scheduling, Safety Time

1. Introduction

We address the stochastic traveling salesperson problem (TSP) with distances measured by travel time and with tour due dates. For this purpose we need specialized shortest routes to populate the TSP distance matrix. An important by-product we thus obtain is an application of the same analysis for shortest routes. Initially, we assume stationary and independent travel time distributions. As a first approximation, we assume either that they are normally distributed or that the number of cities is large enough to justify using the central limit theorem for the tour length distribution. In what follows, for brevity, we use the term *normal* to refer to such models. We note, however, that the sum of several positive random variables can also be approximated by the lognormal random variable [1], which is more appropriate when dealing with strictly positive random variables and allows for instances with high coefficient of variation (*cv*). We refer to these models simply as *lognormal*. For both distributions, we study how to select the best tour and due date for the minimization of fundamental safe scheduling objectives [2, 3]. Our lognormal analysis includes models with linear association. (Linear association occurs when positive independent random variables are all multiplied by an additional positive independent random variable [1, 3]. Linearly associated random variables are positively correlated.) There is ample empirical evidence that the lognormal is realistic and that linear association is an adequate model for various scenarios [1]. We reported theoretical and computational results for the normal distribution earlier in an unpublished working paper, which is available on the web [4]. Here, we repeat some of those results for completeness, add a couple of new normal insights, and extend the analysis to the lognormal case.

A key element of typical safe scheduling models is the *service level*, defined as $SL = \Pr\{M \le d\}$, the probability the tour completes by its due date, d, where the tour length, M, is a random variable. Because the TSP can also be used for production applications, we also refer to M as the *makespan*. Let b denote a given target probability, $0 \le b \le 1$, then in *Model 1* the objective is minimizing d subject to a service level constraint $SL \ge b$. This formulation yields different due dates for different tours so the problem requires finding the tour with the minimal optimal d. Similarly, in *Model 2* we minimize the sum of d and the expected weighted tardiness, namely $d + \gamma \mathrm{E}(T)$, where $T = (M - d)^+ = \max\{(M - d), 0\}$ denotes tardiness and $\gamma > 0$ is a weight parameter. In this case the optimal service level is determined as part of the solution. Both models involve typically positive (but possibly negative) safety time—defined as $(d - \mathrm{E}[M])$—and therefore the makespan distribution is important. We also consider alternate models in which the due date is given and we either maximize SL (Alternate 1) or minimize $\mathrm{E}(T)$ (Alternate 2). For the normal distribution, we provide theoretical and empirical results indicating that our

models are tractable in the practical sense, especially when safety time is positive (although formally they are all NP-hard). For all other instances, including lognormal times, we provide effective heuristics and show how to calculate performance guarantee certificates. Our numerical experience suggests that these certificates are very tight but usually our heuristics identify the optimal solution to begin with.

The deterministic TSP is strongly NP-hard, but instances with thousands of cities have been solved since the mid-1980s [5]. More recently, an evolving state-of-the-art branch-and-cut symmetric TSP solver, Concorde [6], has been associated with several breakthrough milestones; e.g., a 24,978-city instance in 2004. (Many important transportation and production applications are asymmetric but any n-city asymmetric TSP instance can be readily transformed to an equivalent $2n$-city TSP [7], so the distinction is not crucial.) On the one hand, such large instances require decades of CPU time (using parallel processing to compress the actual duration). On the other hand, most practical applications—especially for actual travel applications—are smaller by several orders of magnitude. Hence, we consider the deterministic TSP tractable.

Conventional stochastic scheduling models are often based on deterministic templates, optimizing deterministic objectives by expectation. We call them *stochastic counterparts*. The *deterministic counterpart* of a stochastic model is obtained by using means in deterministic algorithms. Some deterministic counterpart solutions are optimal for the stochastic counterpart. The stochastic TSP counterpart is such a case (because the total distance is a simple sum). In addition, there is an association between Model 2 and the stochastic counterpart of the E/T problem [8], but Model 2 is more general: in the stochastic counterpart of the E/T problem with normal processing times the optimal sequence depends on variances alone but in Model 2 means are crucial [2]. Furthermore, these stochastic counterparts are not equivalent to their deterministic counterparts.

In general, the literature on stochastic versions of the TSP is sparse, perhaps because the stochastic counterpart is solved by the deterministic counterpart. One challenging stochastic adaptation of the TSP involves optimizing a predetermined sequence to visit a random subset of cities [9]. That model exploits shortcuts to bypass cities that do not require service but without re-optimizing the sequence. Thus, it concerns a dynamic problem but restricts the solution to a static sequence. Stochastic models of the related vehicle routing problem are more akin to the safe scheduling approach, although they consider other sources of variation. For a survey, see Gendreau et al. or Bertsimas & Simchi-Levi [10, 11]. Typically they assume stochastic demands on vehicles' capacities, which are only revealed dynamically, so there is an overload risk,

requiring expensive corrective actions. The objective is to minimize the expected total cost of scheduled operations and corrective actions. Travel-time randomness is usually excluded, but we mention some exceptions later. Dynamic models that rely on real-time information for minimizing mean travel time are also relevant [12]. Our own models are static, but static models can support dynamic decisions on a real-time basis, which is a useful dynamic heuristic in general [13].

In Model 1 and Model 2 we treat the due date as a decision [14], but alternate related models involve a given due date. The Model 1 alternate, *Alternate 1*, calls for maximizing *SL*. Alternate 1 was introduced by Kao [15] with independent normal travel times. Sniedovich [16] exposed a flaw in Kao's solution. Carraway et al. [17] presented a generalized dynamic programming approach whose *average* complexity exceeds $O(n^2 2^n)$. Kenyon and Morton [18] show that when the due date exceeds the deterministic counterpart duration (so $SL \geq 0.5$), the model is convex. For that case they propose solving the first few members of an infinite series of deterministic nonlinear integer programs. (One of our contributions is a much more efficient solution.) The Model 2 alternate minimizes the expected tardiness [19].

In Section 2 we discuss generic depictions of the set of all possible tours or routes. We address the normal Models 1 and 2 in Section 3. In Section 4 we consider the normal alternates. We introduce lognormal models in Section 5. Section 6 is the conclusion.

2. Planar Analysis of Routes and Tours with Independent Travel Times

Formally, the TSP problem forbids visiting the same city twice. To allow that, we populate the TSP distance matrix by shortest routes. In our context, that process must take into account stochastic parameters. We present the analysis for shortest routes and for the optimal TSP together. For this purpose, we reserve the term *tour* to a TSP closed path and the term *route* to a path between cities *i* and *k*, perhaps through other cities but without cycles. Henceforth, we reserve the term *path* to statements that apply to both tours and routes.

Let the travel time between cities *i* and *k* have mean μ_{ik} and standard deviation σ_{ik}, and assume independent travel times. For finding a route, these parameters apply to the direct connection between the two cities whereas for the purpose of finding a tour they represent shortest-routes. We use up to one index for paths; we also take the liberty of allowing real numbers as path indices. In both the normal and the lognormal instances, we require two parameters for each path: the mean μ (given by $\Sigma\mu_{ik}$ along the path) and the standard deviation σ (given similarly by $[\Sigma\sigma_{ik}^2]^{0.5}$). Our basic approach for all models is to select the best path we can identify efficiently by solving standard deterministic TSP or shortest route sub-problems. For

some normal models, this leads to the optimal path in time comparable with the deterministic TSP or shortest route. In all other instances (including all lognormal models) it provides an effective heuristic with similar time consumption.

2.1. Identifying Candidate Solutions by Deterministic Applications

We refer to the optimal deterministic counterpart solution as *path* μ (with mean μ_μ and variance σ_μ^2). By the independence assumption, $\sigma^2 = \Sigma\sigma_{ik}^2$, where the summation is along a path; so we can also find *path* σ—the path that minimizes σ (with mean μ_σ and variance σ_σ^2)—by solving a deterministic model for σ_{ik}^2. Similarly, we can find the path that maximizes μ (called *path* $-\mu$—because it requires solving a model with distances of $-\mu_{ik}$) and the path that maximizes σ^2 (*path* $-\sigma$). To clarify, solving a TSP with negative distances requires adding a sufficiently large positive constant to each distance first. The result will be larger by n times the constant. The minimal route problem is known to be polynomial only if no negative cycles exist, and it is also easy to verify whether that is the case [20]. With negative cycles, it is NP-hard [21]. Thus, finding route $-\mu$ or route $-\sigma$ is NP-hard but identifying routes μ and σ is easy. (In likely practical applications we may not encounter such NP-hard shortest route problems.) For presentation purposes, we assume that tours μ and σ are distinct, so $\mu_\mu < \mu_\sigma$ and $\sigma_\mu > \sigma_\sigma$.

For both routes and tours, there are exponentially many choices, with parameters μ_j and σ_j. By complete enumeration, we can depict all paths as points on a Euclidean plane where μ is measured along the horizontal axis and σ, the vertical. Alternatively, we can substitute σ by σ^2. We refer to the former as the μ−σ *plane* (or *depiction*), and to the latter as the μ−σ^2 *plane* (or *depiction*). Figure 1 shows a network with travel time parameters given in Table 1. The set of possible routes from *S* to *T* is depicted in Figure 2: the left depiction is in the $\mu-\sigma$ plane and the right, $\mu-\sigma^2$. We may refer to paths and to the points representing them interchangeably. Furthermore, we consider any point in the positive quadrant of either depiction as a [virtual] path. Contours (with constant objective function) can be drawn through any path in either depiction. Our problems can be solved by finding the best contour through a real path.

Figure 1. Network structure

Table 1. Travel time parameters

Figure 2. All possible routes in the $\mu-\sigma$ and $\mu-\sigma^2$ *planes*

A path is *efficient* if it minimizes the linear function $\alpha\mu + \beta\sigma$ for $|\alpha| + |\beta| > 0$. The vertices of the convex hull of all paths in the $\mu-\sigma$ depiction are efficient. In Figure 2 we completed the

convex hull between tours μ and σ, where $\alpha, \beta \geq 0$. In this case, the same points are on the convex hulls of both depictions. Nonetheless, the two convex hulls are not identical. Whereas for some models our interest is confined to the μ–σ convex hull, we need the other depiction because its extreme points are *identifiable* without complete enumeration (as we show presently). Proposition 1 clarifies the distinction between the depictions.

Proposition 1: Let paths i and k satisfy $\mu_i \neq \mu_k$ and $\sigma_k < \sigma_i$. Any path with $\min\{\mu_i, \mu_k\} < \mu < \max\{\mu_i, \mu_k\}$ and $\sigma_k < \sigma < \sigma_i$ that lies on or below the segment connecting tours i and k in the μ–σ depiction is strictly below the segment connecting them in the μ–σ^2 depiction.

(For any missing proof, see [4].)

Proposition 1 has two corollaries. One is that all the efficient tours between tours μ and σ are identifiable, but not all identifiable tours are efficient. Accordingly, we refer to these identifiable tours as the *superset*. Non-efficient superset tours are *superfluous* for the normal Models 1 and 2 (but they are Pareto optimal and relevant for other models). The other corollary is that members of the top part of the μ–σ convex hull may be *obscure*; i.e., not identifiable. We refer to the identifiable top set tours as the *subset*. The corollaries are due to continuity (a point just above the μ–σ segment may fall below the μ–σ^2 counterpart segment).

We now present an elementary procedure for generating the μ–σ^2 convex hull, starting with the left part between paths $-\sigma$ and σ through path μ. This part is characterized by weighing μ positively. Define *path* λ as the path that minimizes $(1 - |\lambda|)\mu + \lambda\sigma^2$ for $-1 \leq \lambda \leq 1$. Paths $-\sigma$, μ and σ correspond to $\lambda = -1$, 0, and 1. For a while, we confine ourselves further to $0 \leq \lambda \leq 1$, so tour λ minimizes $(1 - \lambda)\mu + \lambda\sigma^2$. We find path λ by solving a deterministic instance with distances of $(1 - \lambda)\mu_{ik} + \lambda\sigma_{ik}^2$ for all pairs $i, k = 1, 2, \ldots, n$. For the route problem, because $0 \leq \lambda \leq 1$, the problem is in P. For the tour problem, conceptually, we start by finding the shortest route λ for every pair of cities. For each such route we use the optimal value to populate the TSP distance matrix. Next we solve the TSP for these values by a deterministic solver. It is then straightforward to calculate the true mean and variance of any resulting path. By construction, when viewed as step functions of λ, μ_λ is monotone nondecreasing and σ_λ^2 is monotone nonincreasing. Suppose that the left side of superset has m members, each associated with a range of possible λ arguments. To identify all the members of this set, assume we already

identified $k \geq 2$ members (starting with paths μ and σ). Using any two adjacent members as *endpoints*, we connect them by a *search segment*. Let λ be the argument for which $(1 - \lambda)\mu + \lambda\sigma^2$ is constant along the search segment. Path λ is either a new member or it merges with an endpoint, signaling that the search segment is a side of the μ–σ^2 convex hull polygon, on the chain of sides between tours μ and σ. When that chain is complete, so is the left side of the superset. Generating the left side of the subset is similar, substituting path $-\sigma$ for path σ. For that purpose, $\lambda < 0$ is required, so $(1 - |\lambda|)\mu + \lambda\sigma^2$ weighs μ positively and σ^2 negatively. (In such case, it is not guaranteed that we can find shortest routes in polynomial time.) To generate the right side of the convex hull we use $(|\lambda| - 1)\mu + \lambda\sigma^2$, for $-1 \leq \lambda \leq 1$, which weighs μ negatively.

As an example, we return to Figure 2 and generate the left superset routes. To identify route μ, we ignore the variances and minimize the route with distances from the second row of Table 1, by Dijkstra's algorithm [20]. The minimal route is S−2−3−T, with mean 16 + 4 + 6 = 26 and variance 12 + 4 + 10 = 26. This result, (26, 26), is depicted at the top of the chain of linear segments in Figure 2. Next, using the third row of Table 1, we find route σ, namely S−1−4−T, with mean 36 and variance 17, as depicted at the bottom end of the chain. To calculate the λ value for which two paths are equivalent, let g be the slope of the search segment that connects them, then (for the left part of the convex hull),

$$\lambda = \frac{-SIGN(g)}{|g| + 1}$$

where SIGN(x) = −1 if $x < 0$ or 1 if $x > 0$. Because $|\lambda| = 1/(|g| + 1)$,

$$1 - |\lambda| = \frac{|g|}{|g| + 1}$$

In our example, $g = (26 - 17)/(26 - 36) = -0.9$ so $\lambda = 1/(0.9 + 1) = 0.5263$ and $1 - |\lambda| = 0.4737$. Edge S−1 now becomes 0.4737×8 + 0.5263×4 = 5.895; S−2, 13.895; etc. With these new values, we find the shortest route is S–1–2–4–T, with $\mu = 31$ and $\sigma^2 = 20$. We proceed to check whether there are other identifiable routes between this new route and each of the two former ones. These are associated with $g = (26 - 20)/(26 - 31) = -1.2$, yielding $\lambda = 1/2.2 = 0.4545$ and $g = (20 - 17)/(31 - 36) = -0.6$, yielding $\lambda = 1/1.6 = 0.625$. The former yields the route S–1–2–3–T, with $\mu = 29$, $\sigma^2 = 22$. The latter yields S–1–3–T, with $\mu = 34$, $\sigma^2 = 18$. Each new insertion requires two subsequent trial insertions. In the example, no additional tours emerge. Table 2 gives the full list.

Table 2: The identifiable left superset tours

Regenerating the TSP distance matrix for every λ we encounter can be time-consuming, especially if negative cycles occur. But we can generate a table of relevant convex hull shortest routes in advance, and maintain lists of threshold λ values for which successive members of these routes are equivalent. For instance, the λ value for which routes 0 and 0.4545 are equivalent is 3/7; routes 0.4545 and 0.5263 are equivalent for $\lambda = 1/2$. Hence, for any $\lambda < 3/7$ we would select tour 0; for $3/7 < \lambda < 1/2$, tour 0.4545; etc. In case of a tie, it does not matter which one we use for the TSP, because they have the same scalar value, but if such a connection is used in the optimal tour for that λ, we should list both options as parts of our efficient frontier later.

2.2 On Computational Complexity and the Shape of the Convex Hull

Suppose that we can solve n-city instances in TSP(n) time units on average, and denote the cardinality of the convex hull in the μ–σ^2 plane by m, then we can find the vertices of this convex hull in $O(m)$ steps. So the expected time to generate the convex hull is $O(m)$TSP(n). We strongly suspect that m is polynomial in n.

Conjecture: The cardinality of the convex hull of all TSP tours with n cities in the μ–σ^2 plane cannot exceed $n(n-1)$; for a symmetric TSP it cannot exceed $n(n-1)/2$.

In [4] we argue that it is practically impossible to construct a counterexample because the distance matrix has a limited number of values we can control. A similar conjecture would fail for the μ–σ plane, however. One counterexample occurs if all distances have the same *variance ratio*—defined as the variance divided by the mean (i.e., $\sigma_{ik}^2/\mu_{ik} = const.$ for all i, k). If so, all tours lie on a straight segment in the μ–σ^2 plane connecting tour μ and tour $-\sigma$, and thus $m = 2$. All these tours are extreme in the μ–σ plane. A similar counterexample occurs if $\sigma_{ik}^2 + \mu_{ik} = const.$ for all i, k.

Whereas conceptually we could formulate a similar conjecture for routes, the argument given in [4] would not apply directly and we made no attempt to adapt it. If such a result can be proved for routes, however, it will imply that generating the left part of the superset for routes is in P. We leave this question open, as it is tangential to our main thrust.

In [4] we report simulations designed to determine the shape of the convex hull and to test the conjecture. We generated distance matrices in two distinct approaches. Under the *random* approach, we used the same distribution for all distances and generated the variances similarly and independently. Because the random approach is not appropriate for simulating geographical distances, the second, *Euclidean*, approach starts by locating cities randomly within a unit square. The resulting Euclidean distances compose $\{\mu_{ik}\}$. Next, a random coefficient of variation was generated for each distance, yielding $\{\sigma_{ik}^2\}$. That induces a positive correlation between μ_{ik} and σ_{ik}, but one would expect correlation between them in practice. The results for the two approaches are remarkably different (Figure 3). The random approach yields symmetric convex hulls whereas the Euclidean approach yields a distinct but irregular shape. In both cases, however, the cardinality of the convex hull appeared linear in n; that is, we could not reject the hypothesis that it is linear, with about $4n$ tours on the convex hull. (Recall that the conjecture allows up to $O(n^2)$, so these results support it. However, the empirical results we obtained were likely influenced by the distributions we used in our simulation, so we cannot claim in general that the empirical cardinality is linear.) We utilized Concorde and conducted experiments with n between 20 and 100 cities. (For other purposes, we experimented with up to n = 1000.) Notably, we could not check the Euclidean cardinality for $n > 50$ because finding the tours on the right side of the convex hull took inordinately long time. This can be explained by the empirical observation that these tours are arranged in an almost vertical pattern. Therefore, they each have very close neighbors that make it difficult for Concorde to fathom branches. On the one hand, such tours often took hours where tours on the left took seconds. For example, typical 100-city TSP applications designed to find tour μ typically took between ten and twenty seconds (on a Pentium 4, 2.00 GHz, 256 MB RAM PC) but in the first Euclidean case we tested, tour $-\mu$ required 14 hours. For n = 50 with 100 repetitions the program execution for tour $-\mu$ took on average about 600 times longer than for tour μ (3.6 minutes vs. 0.36 seconds). On the other hand, the right side is irrelevant to our main models and we could get excellent tours with upper bound values very close to the lower bound right away: the bulk of the time was spent for minute improvements and optimality proofs. In a particular but typical 100-city case, we obtained a certificate of 0.08% for the initial upper bound within five seconds. It then took about 300 seconds to close the gap to below 0.01% and 2900 additional seconds for final convergence and proof of optimality. But in practice the original certificate would be good enough.

Figure 3. Typical $\mu-\sigma^2$ convex hull polygons under the two approaches

3. Models 1 and 2 with Independent Normal Travel Times

As noted in Section 2, we proceed by selecting the best path among the identifiable paths of the convex hull. That yields the optimal path for Model 1 with $SL \geq 0.5$ and for all instances of Model 2. We also discuss how to find that best path without generating the complete convex hull.

3.1 Contour Analysis

We use the notation d_j (j = 1, 2) to distinguish between the path due dates of Model 1 and Model 2; i.e., Model 1 requires minimizing d_1 subject to $SL \geq b$ and Model 2 requires minimizing $d_2 + \gamma \mathrm{E}(T)$, both by optimizing d_j for the best path.

Given a path we can compute its length distribution and then it is straightforward to set d_1: if the cumulative distribution function (cdf) of the path length is $F(t)$, then $SL(d) = F(d)$, so the optimal d_1 for this path is $F^{-1}(b)$—the argument for which $F(d_1) = b$. For the normal case, let $z_j = (d_j - \mu)/\sigma$, let $\varphi(z)$ denote the standard normal density function, and let $\Phi(z)$ denote the standard normal cdf. Using an asterisk to denote optimality, $z_1^* = \Phi^{-1}(b)$ and $d_1^* = \min_k\{\mu_k + z_1^*\sigma_k\}$ among all paths k.

For Model 2, under any distribution, if $\gamma \leq 1$ then $d_2^* = 0$ and the deterministic counterpart path is optimal; henceforth, we assume $\gamma > 1$. Minimizing $d_2 + \gamma \mathrm{E}(T)$ is then identical to minimizing $\mu + \mathrm{E}(E) + (\gamma - 1)\mathrm{E}(T)$, where $E = (d_2 - M)^+$ is the earliness. This formulation generalizes the stochastic early/tardy makespan problem by including μ [2]. Therefore, for any given makespan distribution, the problem can be solved by applying the critical fractile model, which implies a service level target of $b = (\gamma - 1)/\gamma$. For the normal case, $z_2^* = \Phi^{-1}[(\gamma - 1)/\gamma]$. Furthermore, the objective function for $d_2^* = \mu + z_2^*\sigma$ equals $\mu + \gamma\varphi(z_2^*)\sigma$ [2]. Thus, the normal Model 2 requires selecting the path k that minimizes $\mu_k + \gamma\varphi(z_2^*)\sigma_k$.

Both normal models involve minimizing a linear function of μ and σ, $\mu + \pi\sigma$, where $\pi = z_1^*$ or $\gamma\varphi(z_2^*)$. Whereas $\gamma\varphi(z_2^*) > 0$ even if $z_2^* < 0$, in Model 1 $\pi < 0$ for $SL < 0.5$. Nonetheless, we address π as the *price* of each standard deviation unit relative to μ. Our two models are similar, but not equivalent.

Proposition 2: For the normal case, when models 1 and 2 require the same service level (i.e., $z_1^* = z_2^*$), $d_1^* \leq d_2^*$.

That is, when the same service level is required, the prices are different so the optimal paths may not coincide and $d_1^* \leq d_2^*$; e.g., for a service level of 0.9, Model 1 yields $\pi = 1.28$ and Model 2, $\pi = 1.75$. Hence Model 2 may favor a longer path with a smaller variance, which may lead to a later due date.

Proposition 3: For the normal case, when models 1 and 2 have the same price (i.e., $z_1^* = \gamma\varphi(z_2^*)$), $d_1^* > d_2^*$.

For instance $\pi = 1.5$ implies service levels of $b = 0.933$ and $(\gamma - 1)/\gamma = 0.833$, so Model 2 requires less safety time and thus a strictly smaller due date. But the same path is optimal for both models.

Because the objective function for either model is linear, contours for any fixed value are rays emanating upwards from that value on the μ-axis in the $\mu-\sigma$ depiction. For $\pi \neq 0$, the slope of a contour is $-1/\pi$, whereas for $\pi = 0$ the contour is vertical. Thus, different rays reflect different prices, from low to high in the anticlockwise direction. If $\pi \geq 0$ (i.e., for Model 1 with $SL \geq 0.5$ and for Model 2), we can ignore rays with positive slopes. That implies that only paths on the left part of the superset should be considered. For a given positive π, conceptually, the optimal objective function is obtained by shifting the corresponding ray in parallel until it touches the convex hull (in the $\mu-\sigma$ depiction) from the left. For Model 1, d_1^* is then determined by the point at which the optimal contour intersects the μ-axis. For Model 2 the intersection with the μ-axis gives the objective function value, but we still need to calculate d_2^*, which by Proposition 3 will be to the left of d_1^* (for the same nonnegative price). This process identifies at least one optimal path that the ray touches (if the ray coincides with a side of the convex hull, two paths are optimal).

3.2. A Heuristic Solution for Model 1 with $\pi < 0$

If $\pi < 0$, $d_1^* < \mu_\mu$, and may even be negative, thus highlighting a shortcoming of the normal model. (We'll revisit the issue in Sections 4 and 5.) Furthermore, the optimal path may be obscure. However, selecting the best identifiable path is a simple and efficient heuristic. Fortuitously, it is not necessary to generate the full subset to find it. Since $\pi < 0$, the slope of the tangent to the (parabolic) $\mu-\sigma^2$ contour through any tour is positive, and the contour is to the left of the tangent. Furthermore, no better tour can be to the right of the contour or the tangent. We

can use such a tangent in lieu of a search segment, in which case we refer to it as a *search tangent*. Search tangents induce steepest descent searches. Let tour τ be identified by the search tangent of tour μ. If tour μ is also tour τ, then it must be optimal. Otherwise, it is enough to conduct the search between tours μ and τ. To see that, draw a parallel to the tangent of tour μ through tour τ and draw the tangent for tour τ (which has a steeper slope). No tour resides to the left of the parallel and the optimal tour cannot be to the right of the tangent. That excludes any efficient tour (identifiable or obscure) except between tours μ and τ (inclusive). In our experiments, the two tours were very often identical, indicating that tour μ is optimal; otherwise only a small subset of the subset had to be generated. In addition, [4] provides a sufficient optimality condition for the best identifiable tour, assuming tours μ and τ are distinct. If this optimality condition does not hold, we show there how to obtain performance guarantee certificates that are usually very tight.

3.3. Finding the Optimal Tour for $\pi \geq 0$ Directly

Figure 4. Direct search procedures

Although our results here still apply to routes as well as to tours, and for positive prices it is easy to find optimal routes, our main motivation is to find optimal tours. So, for presentation purposes, we discuss finding an optimal tour using a TSP solver such as Concorde. As a rule, the time to solve a TSP is considerably longer than the time to find a lower bound for it. For instance, Concorde starts off by generating an LP-based lower bound in negligible time relative to the full TSP solution. Similarly, calculations of the type we describe below also take relatively negligible time. Therefore our focus is on minimizing the number of full-fledged TSP solutions. We also use an empirical observation: tour μ tends to be better than tour σ (indeed, it is often optimal). We start by identifying tour μ and a lower bound for the minimal standard deviation of tour σ, denoted σ_{LB}. Tour μ provides an initial upper bound, denoted UB. Any point on the contour of tour UB is a virtual UB tour. Initially, one virtual UB tour—denoted tour σ_{LB}—is defined by the standard deviation σ_{LB} and the mean $\mu_\mu + \pi(\sigma_\mu - \sigma_{LB})$. In the $\mu-\sigma$ plane, the contour is a straight line but in the $\mu-\sigma^2$ plane, it is parabolic. We can also define virtual tours associated with lower bounds, denoted LB tours. Initially, the tour with mean μ_μ and standard deviation σ_{LB}, at the left hand side corner, is an LB tour. By assumption, LB tours are strictly super-optimal. In the $\mu-\sigma$ plane, the optimal (real) tour must reside on or within the triangle defined by the two initial UB tours and the initial LB tour: any tour above that triangle must have a higher objective function

value and no tour exists below or to the left. In the $\mu-\sigma^2$ plane, a similar observation holds because the relevant part of the parabolic contour is strictly within the $\mu-\sigma^2$ triangle. We can only conduct searches in the $\mu-\sigma^2$ plane so superfluous tours may exist in the triangle. The direct search aims to identify new tours within the initial triangle so that progressively better tours may be identified and the total search domain is decreased. Any identifiable tours outside the triangle can be ignored.

Our basic search procedure is recursive and generates a binary tree representing right triangles nested within larger (parent) right triangles (see the left side of Figure 4). We call them *search triangles*. Each search triangle has a horizontal side at the lowest possible variance in it and a vertical side at the minimal mean. The hypotenuse is a search segment between two UB tours. For instance, the triangle we identifies for the first UB tours is the root of the tree, and its sides are at μ_μ and σ_{LB}. Using such search segments, if no new tour is identified strictly within a triangle, the triangle is fathomed. Otherwise, the new tour is either the new UB tour or superfluous. In the former case, all UB tours should be updated. To that end, draw a new parabolic UB contour and update the former UB tours (real and virtual) by reducing the variance of any tour associated with the minimal mean (of its triangle) and reducing the mean of the tour associated with the corresponding minimal variance. This is done in every open search triangle. In addition, we draw a vertical segment from the new tour downwards until it intersects the horizontal side of its parent triangle. By that we create a new nested right triangle below and to the right of the new tour. Similarly, we draw a horizontal segment from the new UB tour towards the vertical side of the initial triangle, creating a new nested right triangle above and to the left of the new tour (the first two nested triangles are shaded in Figure 4). No tour exists below and to the left of the new tour or it would have been identified instead. The search then continues in each of the two new triangles recursively. If the new tour is superfluous, the procedure is modified in two minor ways: we do not update the UB tours and after drawing the two new vertical and horizontal segments we mark on them two new UB tours (below and to the left of the new tour, respectively).

By observing the contours, it is obvious that the potential benefit of our models increases with b for Model 1 and with γ for Model 2. In [4] we report experiments we conducted for π = 1.5, using the random generation approach. In most instances where tour μ was not optimal, the optimal tour was identified in the second search, but it took up to two more searches to verify it. In all cases, any improvement beyond the second search was very small (0.02% on average). Tour μ was optimal in between 64% and 16%. Interestingly, for large n tour μ was rarely optimal but not far from optimum whereas for small n it was often optimal (56% for n = 10, 64% for n =

20), but the potential improvement when it was not optimal was highest: about 6% (thus, the roughly 40% of cases where the second tour was optimal contributed about 2.5% to the average). We observed that as n grows the probability that the second search finds the optimum is reduced significantly, but the additional possible improvement decreases. Figure 5 shows the improvement obtained by the second search, on average. In a practical sense we conclude that it is important to perform the second TSP search, especially for practical sizes of n (that is, small n) but there is no practical incentive to run any further searches.

Figure 5. Average improvement obtained by the second search ($\pi = 1.5$)

We can streamline the search further if we reduce the size of one of the search segments that emerge when we identify a new tour by utilizing the observation that no attractive tour may reside below the parallel to the search segment through the new tour. That also leads to tighter lower bounds. The first two reduced search triangles are shaded on the right side of Figure 4, where dashed lines show the boundaries of the right triangles. When the lower and upper bounds are sufficiently close, we can stop the search.

4. Alternate Normal Models

4.1. Alternate 1

Alternate 1 maximizes the service level for a given due date, d. To solve Model 1 we conceptually shift a contour with slope $-1/\pi$ (or a vertical for $\pi = 0$) in parallel until it touches the convex hull (from the left). Similarly, to solve Alternate 1, we select the contour for d that touches the convex hull; i.e., we rotate a ray pivoted at (d, 0) in the clockwise direction until it touches the convex hull. For the resulting SL, this ray constitutes a contour that also solves Model 1. Therefore, the two models are equivalent duals. However, to employ the direct search procedure for Alternate 1, new UB contours are drawn as parabolas through the new UB tour that touch the μ-axis at d, whereas for Models 1 and 2 new UB contours are shifted to the left from the previous contour.

If $d \geq \mu_\mu$, the contour will have a negative slope, g, corresponding to $\pi \geq 0$; otherwise, $g > 0$ and the optimal solution may be obscure. In this connection, recall that solving the normal Model 1 may lead to $d_1^* < 0$. Alternate 1 may resolve the issue. In Alternate 1, d is given and by assumption it is positive. Accordingly, one way to address negative d_1^* is to impose a nonnegative lower bound on d_1^*, d_{LB}, and switch to Alternate 1 with $d = d_{LB}$ if the bound is

breached. (Otherwise we accept a low *SL* without achieving a legitimate due date in return.) If we adopt this approach and search for tour τ using the contour through tour μ that corresponds to d_{LB}, we can ignore any tours beyond tour τ for any *SL*.

4.2. Alternate Model 2

Alternate 2 minimizes the expected tardiness, E(*T*), given *d*. For any virtual tour at (μ, σ),

$$E(T) = \sigma\int_{z}^{\infty}(x-z)\varphi(x)dx = \sigma\left[\varphi(z) - z\Phi(-z)\right]$$

The partial derivatives are $\partial E(T)/\partial\mu = \Phi(-z)$ and $\partial E(T)/\partial\sigma = \varphi(z)$. Both are positive. The Hessian is,

$$\begin{bmatrix} \dfrac{\partial^2 E(T)}{\partial^2\mu}, & \dfrac{\partial^2 E(T)}{\partial\sigma\partial\mu} \\ \dfrac{\partial^2 E(T)}{\partial\mu\partial\sigma}, & \dfrac{\partial^2 E(T)}{\partial^2\sigma} \end{bmatrix} = \frac{\varphi(z)}{\sigma}\begin{bmatrix} 1, & z \\ z, & z^2 \end{bmatrix}$$

The Hessian is positive semi-definite (the diagonal elements are positive and the determinant is 0) and therefore the E(*T*) function is convex. We refer to the virtual tour with $\mu = d$ and $\sigma = 0$ as the *focal point*, and it is immediate that E(*T*) = 0 on any point between the origin and the focal point and rises with a directed derivative of 1 along the μ-axis to the right. Similarly, E(*T*) increases linearly along any ray emanating upwards from the focal point (those rays are contours of Models 1 and 2). Figure 6 depicts E(*T*) contours in the $\mu-\sigma$ plane (generated for $d = 10$). We can view each contour as a function of μ, $h_E(\mu)$—where the subscript denotes the value of E(*T*) and may be omitted. Each contour is monotone decreasing and concave. Specifically, because $\partial E(T)/\partial\mu = \Phi(-z)$ and $\partial E(T)/\partial\sigma = \varphi(z)$, *h'*—the first derivative—is $-\Phi(-z)/\varphi(z)$. The marginal rate at which *z* changes as a function of μ along the contour is

$$\frac{-1}{\sigma}\left(1 + zh'\right) = \frac{-1}{\sigma}\left(1 - z\frac{\Phi(-z)}{\varphi(z)}\right) = \frac{-E(T)}{\sigma^2\varphi(z)}$$

It follows that the second derivative is

$$h'' = \frac{-E(T)}{\sigma^2\varphi(z)}\left(\frac{\varphi(z) - z\Phi(-z)}{\varphi(z)}\right) = \frac{-1}{\sigma^3}\left(\frac{E(T)}{\varphi(z)}\right)^2$$

Both h' and h'' are strictly negative. The optimal contour is the lowest one that touches any tour, and any tour that touches the optimal contour is optimal. Because the contour is monotone and concave, the optimal tour must be Pareto optimal; i.e., no suboptimal tour can have a strictly smaller μ unless its σ is strictly larger and vice versa. But we have no efficient way to generate all these Pareto optimal tours. So we resort to a heuristic solution (which is very often optimal, however).

Figure 6. $\mu-\sigma$ normal contours for Alternate 2 (d = 10, E(T) = 1, 2, ..., 7)
Figure 7. $\mu-\sigma^2$ normal contours for Alternate 2 (d = 10, E(T) = 1, 2, ..., 7)

Transforming the E(T) contours to the $\mu-\sigma^2$ plane—see Figure 7—they are no longer concave everywhere. Each contour has one inflection point, which satisfies

$$\Phi(-z) = \frac{E(T)}{\sigma} \Leftrightarrow (z+1)\Phi(-z) = \varphi(z)$$

That implies $z = -0.481$. Therefore, the inflection point is a virtual tour with $\mu = d + 0.703 \times \mathrm{E}(T)$ and $\sigma = 1.460 \times \mathrm{E}(T)$. Consider the transformed contour of the best identifiable tour. Due to the concave region to the right of $d + 0.703 \times \mathrm{E}(T)$, it is plausible that the contour will protrude into the area of the convex hull. When that occurs, there is an area below the contour inside the convex hull. Any tour that might reside within that area has a lower E(T) than the best identifiable tour. Furthermore, if such tours exist, at least one of them is Pareto optimal. Upon such protrusion, we can conduct a numerical search and find the virtual tour on that segment that has the minimal E(T). That yields a valid lower bound on E(T) because (i) for any tour within the enclosed area there must be better virtual tours on the segment and (ii) no tour is below the segment. Such lower bounds are the basis of our performance guarantees.

4.2.1. Identifying the Best Alternate 2 Solution Directly

Although we cannot always guarantee that the optimal tour is identifiable, we can adapt the direct search procedure of Subsection 3.4 as a heuristic. That leads to an optimality guarantee often (in our experiments, it always did). Again we start by identifying tour μ and its contour. If $\sigma_{LB} \geq 1.460 \times \mathrm{E}(T_\mu)$, all potentially optimal tours reside above the UB inflection point and the search is restricted to the convex part of the contour, leading to optimality. Otherwise, modify the

search segment between tours μ and σ_{LB} by making it tangent to the contour just above tour σ_{LB}. No optimal tour resides above the modified search segment. Any emerging new tour is in the *concave region* if it resides below the inflection point of the updated UB contour; otherwise, it is in the *convex region*. When a concave region tour emerges, the next search segment must connect it to the adjacent identified real tour (initially, tour μ). When an emerging concave tour is the new UB tour *and* the parallel to the previous search segment through it intersects the new UB contour to the right, we add a search tangent through the new UB tour. At the end of the procedure, if any segment connecting adjacent identified real tours intersects the UB contour, it signals the potential existence of a superior but obscure tour in the area captured between the segment and the contour. The best virtual tour on this segment then provides a lower bound and a performance guarantee certificate.

5. Models with Lognormal Travel Times

Trietsch et al. (2012) show that the lognormal distribution provides excellent fits for various processing time field data (with matched mean and variance). Although travel time data is not included in their analysis, we assume that the lognormal is also a better fit for travel time than the normal. Furthermore, the lognormal distribution can provide a better fit for the convolution of positive random variables than the normal: as n grows large, the distribution of the sum of n independent, positive random variables tends to lognormal [1, 3], and see also [22]. [1] and [3] refer to that as the *lognormal central limit theorem*. More formally, the lognormal distribution provides an ε-close fit to the sum distribution of a large number of positive random variables, but it is not a proper central limit distribution [23]. Henceforth, we call it the *lognormal sum approximation*. This approximation is more effective than the normal for small n too, especially if the components' distributions are skewed to the right, such as the lognormal itself. Using the lognormal to approximate convolutions of few lognormal elements is known as the Fenton-Wilkinson approximation [24]. Moreover, [1] demonstrate that field data often exhibits positive correlation that can be modeled effectively by linear association using the lognormal. Such correlation is likely to manifest in traffic data due to various causes; e.g., weather. We first address independent lognormal travel times and then we consider linear association.

The lognormal distribution is the exponent of a core normal with mean m and standard deviation s. These parameters can be uniquely determined by μ and σ,

$$s = \sqrt{\ln\left(1 + \frac{\sigma^2}{\mu^2}\right)} \quad ; \quad m = \ln(\mu) - \frac{s^2}{2}$$

For tour k (with μ_k and σ_k) and a given due date, d, the service level is

$$SL_k = \Phi(z_k) = \Phi\left(\frac{\ln(d) - m_k}{s_k}\right)$$

To avoid confusion with z_1* and z_2*, we only use the notation z_k for the alternate models, where d is given, so z_k is a parameter of tour k, whereas z_1* and z_2* are not tour specific. Because the service level for any z is identical to the normal case, z_1* and z_2* still apply. For Model 1,

$$d_1^* = min_k\{\exp(m_k + z_1^* s_k)\}$$

Equivalently, Model 1 seeks to minimize

$$m_k + z_1^* s_k = m_k + \Phi^{-1}(b) s_k$$

The expected tardiness is

$$E(T_k) = \mu_k \Phi(s_k - z_k) - d\,\Phi(-z_k)$$

By substituting z_2* into this expression and optimizing d, Model 2 is solved by

$$min_k\{C_k\} = min_k\{\gamma \mu_k \Phi(s_k - z_2^*)\}$$

Where C_k is the total cost associated with tour k. (We may omit the subscript and treat C as a continuous function of μ and σ^2. Because $\Phi(s - z_2^*) \geq \Phi(-z_2^*) = 1 / \gamma$, $C \geq \mu$, as required.) Unfortunately, none of these expressions is linear or convex in either the μ–σ or the μ–σ^2 depiction. To illustrate, Figure 8 shows contours of the lognormal Alternate 1 for various *SL* levels such that $d = 1$. As for the normal distribution, these contours still form a fan shape, but the spokes are curved. To generate similar contours for any d, we use polar coordinates from the origin. Draw a ray from the origin, and let r_1 be the distance from the origin where the ray intersects a contour for some given *SL*. (As the figure shows, all contours for *SL* > 0.5 are vertical

at some minimal μ value and then curl backwards. Nonetheless, for every contour, even when $SL << 0.5$, there is exactly one such intersection.) The contour for d is at a distance of $r_1 d$ from the origin on the same ray. By repeating this a sufficient number of times we can trace the new contour (which touches the μ-axis at d). Therefore, the same figure also provides contours for the lognormal Model 1. Accordingly, as in the normal case, the lognormal Model 1 and Alternate 1 are equivalent dual problems.

Figure 8. $\mu-\sigma$ lognormal contours for Model 1 or Alternate 1 (z = 3.0, 2.5, ..., −1.0)
Figure 9. $\mu-\sigma$ lognormal contours for Model 2 (z_2* = 2.5, 2.0, ..., −1.5)
Figure 10. $\mu-\sigma$ lognormal contours for Alternate 2 (E(T) = 5%, 10%, ..., 50%)

Regarding Model 2, Figure 9 provides contours for C = 1 and various z_2* values, ranging from −1.5 to 2.5 (corresponding to SL between 6.7% and 99.4%), in the $\mu-\sigma$ depiction. Again, we can generate contours for any C and any γ by extending a ray from the origin through the contour and marking a point at a distance of $r_1 C$ from the origin. Here, all contours start at μ = 1 on the μ-axis and go up and to the left as σ increases. As $\Phi(s - z_2^*)$ approaches 1 when σ is increased, each contour tends to the value 1 / γ asymptotically (from above). That is counterintuitive but note that d_2* is monotone decreasing nonetheless, thus allowing E(T) to increase; it is just the sum $d_2^* + \gamma E(T)$ that tends to the constant 1 / γ. As we increase γ, the contours shift to the left, which indicates that we require a lower μ value to achieve the same constant C. Most notable in the figure is the behavior of the contour for z_2* = 2.5, which—similarly to the case of z_1* = 3.0 in Figure 8—has a distinct hump on a roughly horizontal section. We also note that the contours of Model 1 and Model 2 are not identical (unlike the normal case); e.g., the Model 2 contours do not curl back.

Figure 10 concerns the lognormal Alternate 2. For very low σ values and low E(T), the contours start as in Figure 6—which should not be surprising given that the normal and the lognormal are ε-close when σ/μ is small—but they are dramatically different for higher σ values and especially near the σ-axis, to which they tend asymptotically.

Even in the $\mu-\sigma^2$ plane, all those contours are not convex in general. Therefore, we cannot guarantee that the optimal tour is identifiable. In principle, we can adapt the same approach we did for the normal Alternate 2. Thus we are essentially limited to heuristics for all lognormal models. Our numerical experience suggests, however, that we can still safely limit ourselves to identifiable tours: we tested hundreds of instances with up to 11 cities by complete enumeration without a single exception (we did not include instances with z > 2.5, however). For any

parameter, as the number of cities increases, performance guarantees tend to improve (because identifiable tours tend to be close to each other). In the following subsections we develop an effective and efficient steepest descent search heuristic for Model 2. The other three models can be addressed by the same approach, but we omit details for brevity. Conceptually, the same heuristic could also be used for normal models but we did not test it for that purpose.

5.1. Independent Lognormal Travel Times

The proposed heuristic starts by finding a good heuristic solution for tour μ, and then proceeds to determine a steepest descent search direction in which to search for a better tour. That is, we use search tangents. It is based on two key ideas. First, if we could guess a λ value within the range for which the best identifiable tour prevails, we could then identify the tour in one TSP application. Therefore, it is a good bet that if we can guess λ at least near the range, we'll obtain a good heuristic solution in one TSP application. Second, it is much easier to obtain reasonably tight bounds for the TSP than to solve it: e.g., in Concorde, the major time consuming task is guaranteeing convergence of upper and lower bounds. So we may settle for a solution with a sufficiently small optimality gap without completing the TSP application.

To find the approximate λ, we need to calculate the partial derivatives of the model we solve (here, Model 2) by μ and by σ^2. Since $C = \gamma\mu\Phi(s - z_2^*)$,

$$\frac{\partial C}{\partial \mu} = \gamma\Phi(s - z_2^*) - \gamma\frac{\sigma^2\varphi(s - z_2^*)}{s(\mu^2 + \sigma^2)} = \gamma\frac{s(\mu^2 + \sigma^2)\Phi(s - z_2^*) - \sigma^2\varphi(s - z_2^*)}{s(\mu^2 + \sigma^2)}$$

$\partial C/\partial\mu < 0$ is possible. Nonetheless, until further notice, we assume $\partial C/\partial\mu > 0$ (which is true for $z_2^* \leq 2$; see Figure 9). Next,

$$\frac{\partial C}{\partial(\sigma^2)} = \gamma\frac{\mu\varphi(s - z_2^*)}{2s(\mu^2 + \sigma^2)} > 0$$

The contour's slope is

$$g = \frac{-\left[\frac{\partial C}{\partial \mu}\right]}{\left[\frac{\partial C}{\partial(\sigma^2)}\right]} = \frac{2\sigma^2\varphi(s - z_2^*) - 2s(\mu^2 + \sigma^2)\Phi(s - z_2^*)}{\mu\varphi(s - z_2^*)}$$

which is negative if $\partial C/\partial\mu$ is positive. We use search tangents and $g < 0$, so $\lambda = |g| / (|g| + 1)$. That has to be calculated for the current best tour. We choose to implement this result by finding tour μ heuristically (applying Concorde for a limited time), calculating λ for it, and then conducting a single full TSP application for tour λ. We also recommend to bound the TSP search time since numerical experience suggests that in hard TSP instances the bulk of computation time is expended for tiny improvements and for confirming optimality.

For $\partial C/\partial\mu < 0$, the steepest descent direction becomes $-(1 - |\lambda|)\mu + |\lambda|\sigma^2$. (In such case, we may expect $|\lambda| \approx 1$, so tour σ becomes attractive.) Such a search may identify a superset tour to the right of tour σ. Nonetheless, the optimal tour is still likely to be one of the identifiable superset tours. (Model 1 can also have such a solution; e.g., for $z_1^* = 3$. For the same *SL*, however, the optimal tour may be above tour μ, because of the curl-back feature.)

Our experimentation with this heuristic were limited to Model 2 with $z = 1.5$, for which case the contours are relatively well behaved. The heuristic identified the optimal tour very frequently, and overall provided results within 0.2% of optimality.

5.2. Linearly Associated Lognormal Travel Times

We say that n positive random variables, Y_j, are linearly associated if $Y_j = BX_j$, where $\{X_j\}$ and B are $n + 1$ independent positive random variables. If we relax the independence assumption in our TSP and replace it by linear association, the lognormal sum approximation can still be used: we just have to add up the X_j elements before multiplying by B. In practice, B is likely to be influenced by numerous additive and multiplicative causes, so B is likely to be approximately lognormal too. Therefore, if X_j is lognormal, $Y_j = BX_j$ is also lognormal. Furthermore, by Cramér's Theorem—that the sum of two independent random variables cannot be normal unless both are normal—if Y_j is lognormal then both X_j and B must be lognormal. To apply linear association to our problem, we use the independent model as the base, but we multiply the final result by a lognormal random variable, B. Without loss of generality, we assume that $E(B) = 1$, and let μ, σ^2, s_{ind}, and m_{ind} denote the parameters of a tour *before* multiplying by B. Let s_b denote the s parameter of B, and m_b is given by $-s_b^2/2$. After the multiplication, the lognormal parameters become s_{comb}, and m_{comb}, given by

$$m_{comb} = m_{ind} - \frac{s_b^2}{2} \quad ; \quad s_{comb} = \sqrt{s_b^2 + \ln\left(1 + \frac{\sigma^2}{\mu^2}\right)}$$

where μ^2 and σ^2 refer to pre-multiplication values. To search in the pre-multiplication steepest descent direction, we modify *g* by identifying *s* explicitly,

$$g = \frac{2\sigma^2\varphi(s_{ind} - z_2^*) - 2s_{comb}(\mu^2 + \sigma^2)\Phi(s_{ind} - z_2^*)}{\mu\varphi(s_{ind} - z_2^*)}$$

Our experimentation with this heuristic were limited as above. It identified the optimal tour frequently, and overall provided results within 1% of optimality.

6. Conclusion

We formulated safe stochastic versions of the TSP and the shortest route problem with normal or lognormal travel times. All versions can be solved very quickly by applying deterministic models iteratively, but not always to optimality. In practice, it is sufficient to run a deterministic solver twice to obtain a nearly-optimal (and often optimal) solution. We showed that Model 1 and Alternate 1 are equivalent, and we can solve them crisply for the normal $SL \geq 0.5$. We can also solve any normal Model 2 instance. Conceptually, we do that by selecting from a finite set of identifiable extreme paths. For the normal, Alternate 2 requires considering a larger set of Pareto optimal solutions. Therefore, Model 2 and its alternate are not equivalent. Our solution approach becomes an effective heuristic for the normal Model 1 or Alternate 1 with $SL < 0.5$ and for Alternate 2. Our results for the normal case constitute significant progress relative to previously published ones, but their practical value is limited to instances where travel time has relatively low standard deviation. Therefore, we also consider lognormal travel time, including the case of linear association (which is a basic but validated model that accounts for correlation among travel times by applying a multiplicative factor to all travel times). In the lognormal case we must resort to heuristics for all models. In particular, we developed a steepest descent heuristic that can be applied for any instance and any distribution. For the TSP, this heuristic requires a single full TSP search and yields excellent results (usually optimal, with the maximal error below 1% even when linear association is involved). Especially when using the heuristic, solving our TSP is practically as tractable as solving the deterministic TSP. Indeed, we were able to solve TSP instances with up to $n = 1000$, using Concorde on an Intel Core 2 Duo E4300, 1.8 GHz, 1Gb of RAM PC. (This particular heuristic is not tailored for the shortest path when we can solve it in polynomial time, but it is also less important in that context.)

There are several possible extensions of this research. Combining our results with stochastic vehicle routing models is a natural step, although it is known that similar combinations

are challenging [10]. Similarly, it might be interesting to combine our approach with the stochastic TSP version that requires visiting a random subset of the *n* cities in a prescribed sequence [9]. Yet another safe scheduling approach is based on the notion that we can deselect customers and optimize the tour for the remaining ones. This is a generalization of the stochastic *U*-problem [25, 26]. In a sense, the vehicle routing problem already includes a selection of customers for each truck tour, and conceptually we could assign all rejected customers to a dummy truck.

Acknowledgement: We are grateful to the developers of the Concorde software for making it freely available for academic research. In particular, Bill Cook has generously provided useful advice and insights. We also wish to thank Ken Baker for detailed and helpful comments.

References

[1] Trietsch D, Mazmanyan L, Gevorgyan L, Baker KR. Modeling Activity Times by the Parkinson Distribution with a Lognormal Core: Theory and Validation. Eur J Oper Res 2012; 216: 386-396.

[2] Baker KR, Trietsch D. Safe scheduling: Setting due dates in single-machine problems. Eur J Oper Res 2009; 196: 69-77.

[3] Baker KR, Trietsch D. Principles of Sequencing and Scheduling. Wiley; 2009.

[4] Mazmanyan L, Trietsch D. Stochastic traveling salesperson models with safety time. 2009. URL: http://ac.aua.am/trietsch/web/SafeTSP_Mazmanyan_Trietsch.pdf.

[5] Laporte G. The traveling salesman problem: An overview of exact and approximate algorithms. Eur J Oper Res 92; 59: 231-247.

[6] Applegate D, Bixby R, Chvatal V, Cook W.
URL (accessed June 2008): http://www.tsp.gatech.edu/concorde

[7] Jonker R, Volgenant T. Transforming Asymmetric into Symmetric Traveling Salseman Problems. Oper Res Letters 1983; 2: 161-3.

[8] Soroush HM. Sequencing and due-date determination in the stochastic single machine problem with earliness and tardiness costs. Eur J Oper Res 1999; 13: 450-468.

[9] Jaillet P. A Priori Solution of a Traveling Salesman Problem in which a Random Subset of the Customers are Visited. Oper Res 1988; 36: 929-936.

[10] Gendreau M, Laporte G, Séguin R. Stochastic vehicle routing. Eur J Oper Res 1996; 88(1): 3-12.

[11] Bertsimas DJ, Simchi-Levi D. A New Generation of Vehicle Routing Research: Robust Algorithms, Addressing Uncertainty. Oper Res 1996; 44: 286-304.

[12] Gao S, Chabini I. Optimal routing policy problems in stochastic time-dependent networks. Transp Res Part B 2006; 40: 93-122.

[13] Bertsimas DJ, Jaillet P, Odoni A. A Priori Optimization. Oper Res 1990; 38(6): 1019-1033.

[14] Baker KR, Bertrand JWM. A Comparison of Due Date Selection Rules. AIII Transactions 1981; 13: 123-131.

[15] Kao HPC. A preference order dynamic program for a stochastic traveling salesman problem. Oper Res 1978; 26: 1035-1045.

[16] Sniedovich M. Analysis of a preference order traveling salesman problem. Oper Res 1981; 29: 1234-7.

[17] Carraway RL, Morin TL, Moskowitz H. Generalized dynamic programming for stochastic combinatorial optimization. Oper Res 1989; 37: 819-829.

[18] Kenyon AS, Morton DP. Stochastic Vehicle Routing with Random Travel Times. Transportation Sci 2003; 37: 69-82.

[19] Laporte G, Louveaux FV, Mercure H. The vehicle routing problem with stochastic travel times. Transportation Sci 1992; 26: 161-170.

[20] Dreifus SE. An appraisal of some shortest-path algorithms. Oper Res 1969; 17: 385-412.

[21] Garey MR, Johnson DS. Compters and Intractability: A Guide to the Theory of NP-Completeness. WH Freeman; 1979.

[22] Robb DJ. Scheduling in a management context: Stochastic tasks and non-regular reward functions. Unpublished Doctoral Dissertation, Faculty of Management, University of Calgary.

[23] Paul AA, Trietsch D. Correlation sensitivity analysis in stochastic project networks. Working Paper, November 2012.
URL: http://ac.aua.am/trietsch/web/PaulTrietschWorkingPaperNovember2012.pdf

[24] Fenton LF. The sum of lognormal probability distributions in scatter transmission systems. IRE Trans Commun Syst 1960; 8: 57-67.

[25] Akker van den JM, Hoogeveen JA. Minimizing the number of late jobs in case of stochastic processing times with minimum success probabilities. J Scheduling 2008; 11: 59-69.

[26] Trietsch D, Baker KR. Minimizing the number of tardy jobs with chance constraints and stochastically ordered processing times. J Scheduling 2008; 11: 71-73.

Table 1: Travel time parameters

Edge	S−1	S−2	1−2	1−3	1−4	2−3	2−4	3−4	3−T	4−T
Mean	8	16	11	20	24	4	8	6	6	4
Variance	4	12	4	4	5	4	4	6	10	8

Table 2: The identifiable left superset tours

λ	0.0	0.4545	0.5263	0.625	1.0
Path	S−2−3−T	S−1−2−3−T	S−1−2−4−T	S−1−3−T	S−1−4−T
Mean	26	29	31	34	36
Variance	26	22	20	18	17

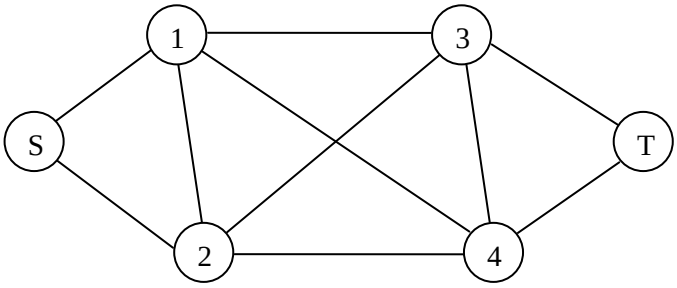

Figure 1. Network structure

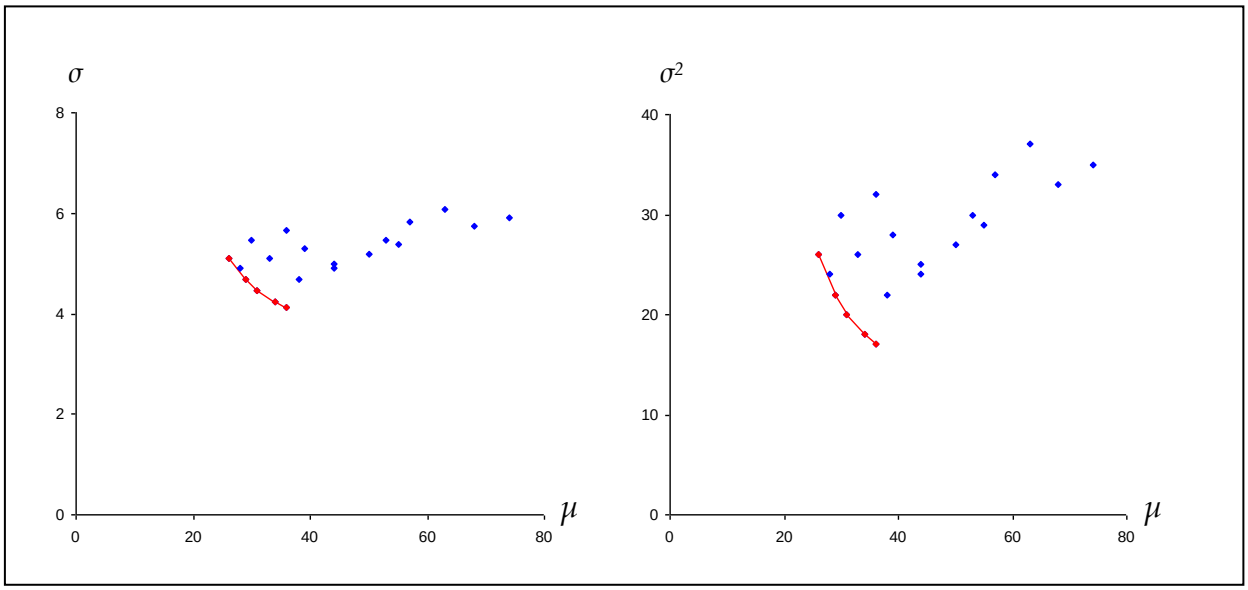

Figure 2. All possible routes in the $\mu-\sigma$ and the $\mu-\sigma^2$ *planes*

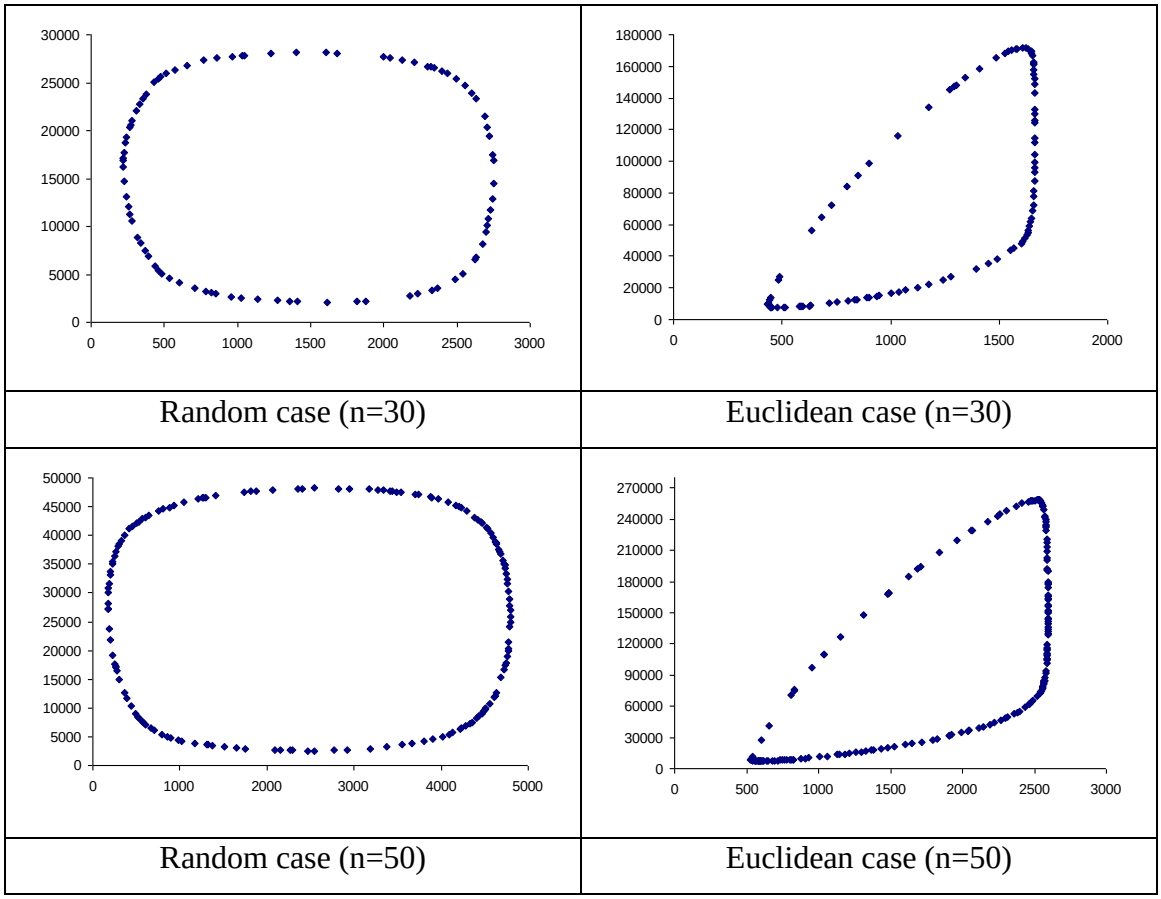

Figure 3. Typical $\mu-\sigma^2$ convex hull polygons under the two approaches

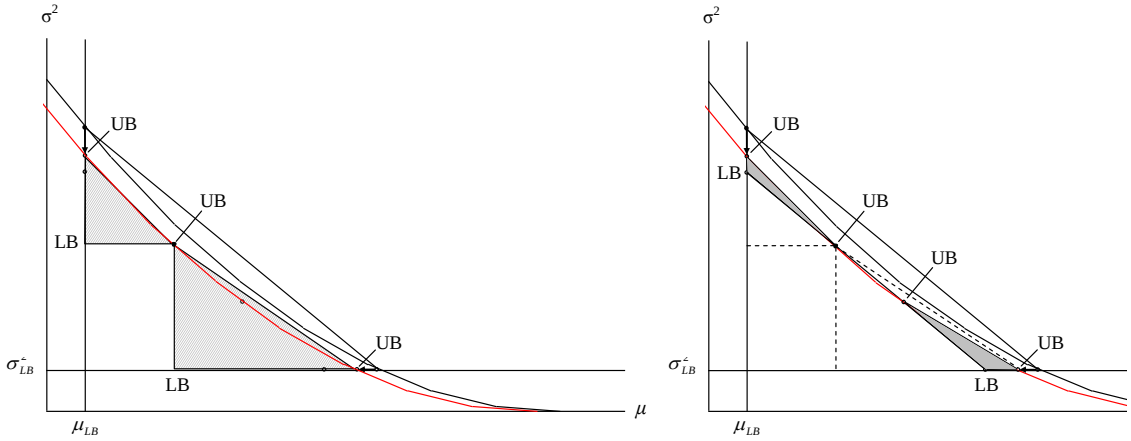

Figure 4. Direct search procedures

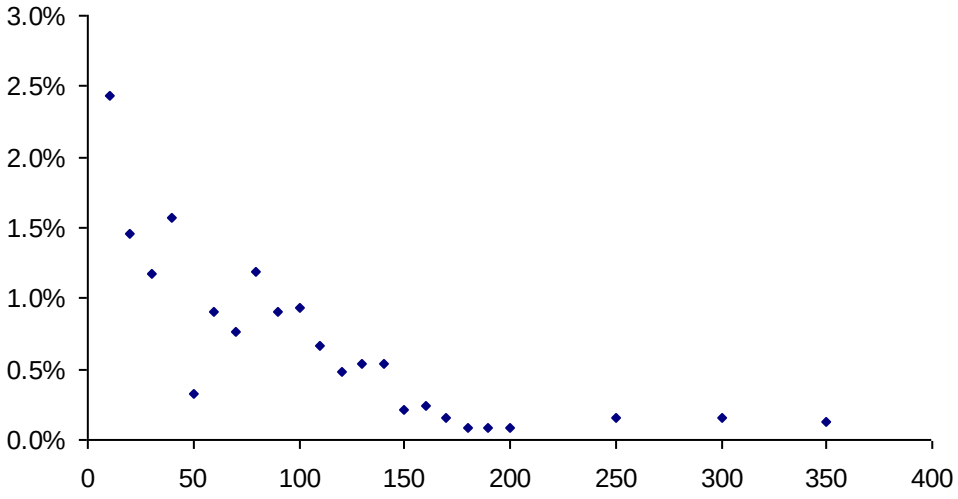

Figure 5. Average improvement obtained by the second search (π = 1.5)

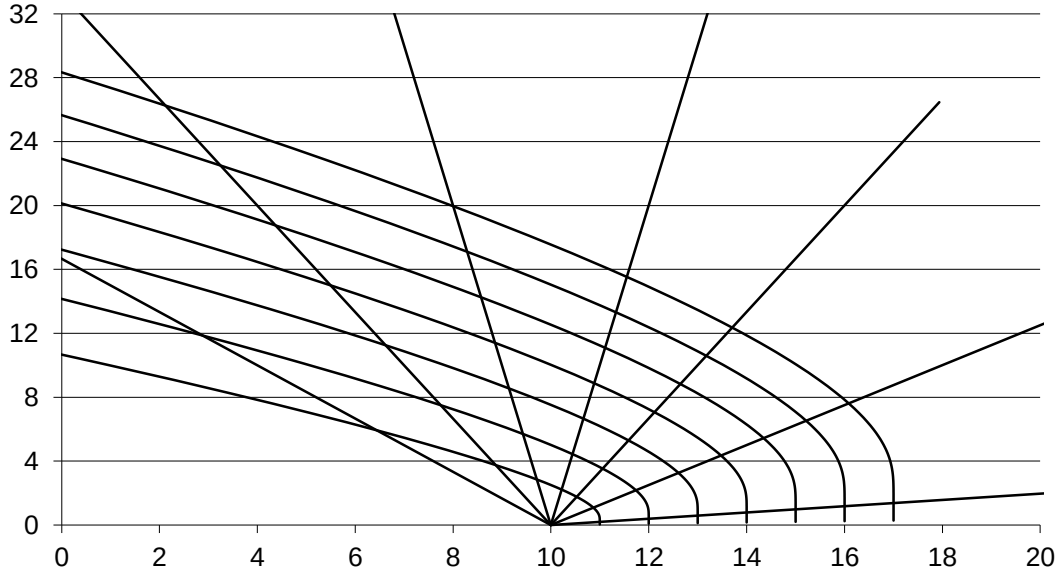

Figure 6. $\mu-\sigma$ normal contours for Alternate 2 (d = 10, E(T) = 1, 2, ..., 7)

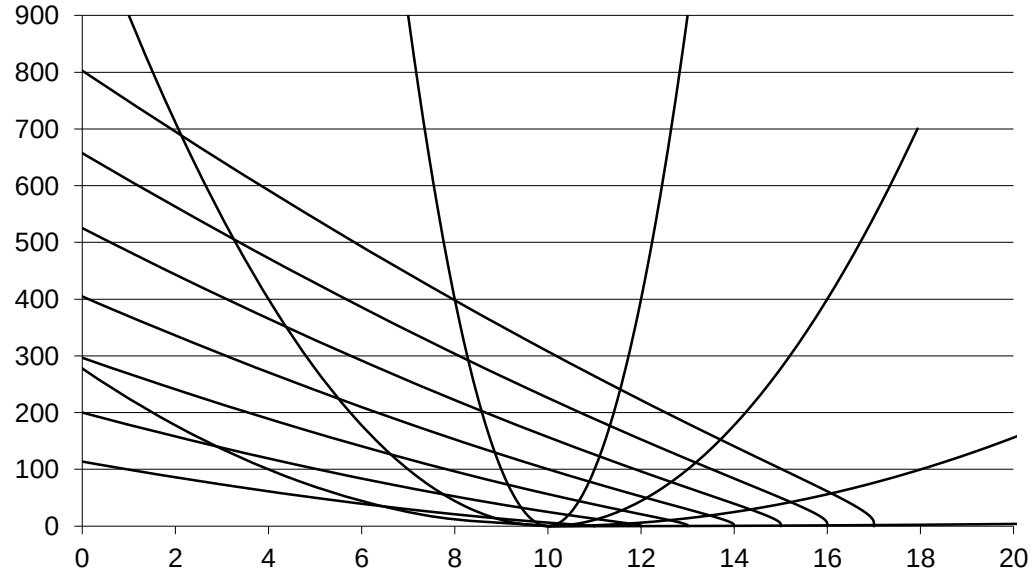

Figure 7. $\mu-\sigma^2$ normal contours for Alternate 2 (d = 10, E(T) = 1, 2, ..., 7)

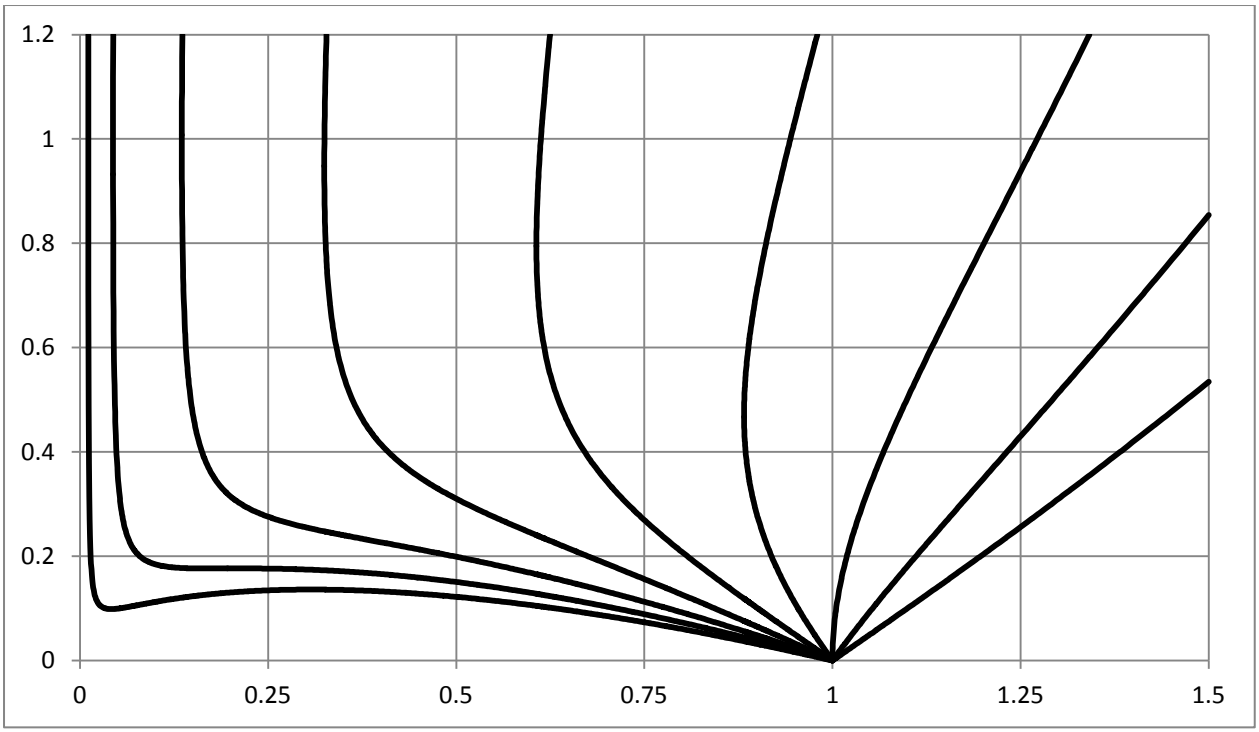

Figure 8. $\mu-\sigma$ lognormal contours for Model 1 or Alternate 1 (z = 3.0, 2.5, ..., −1.0)

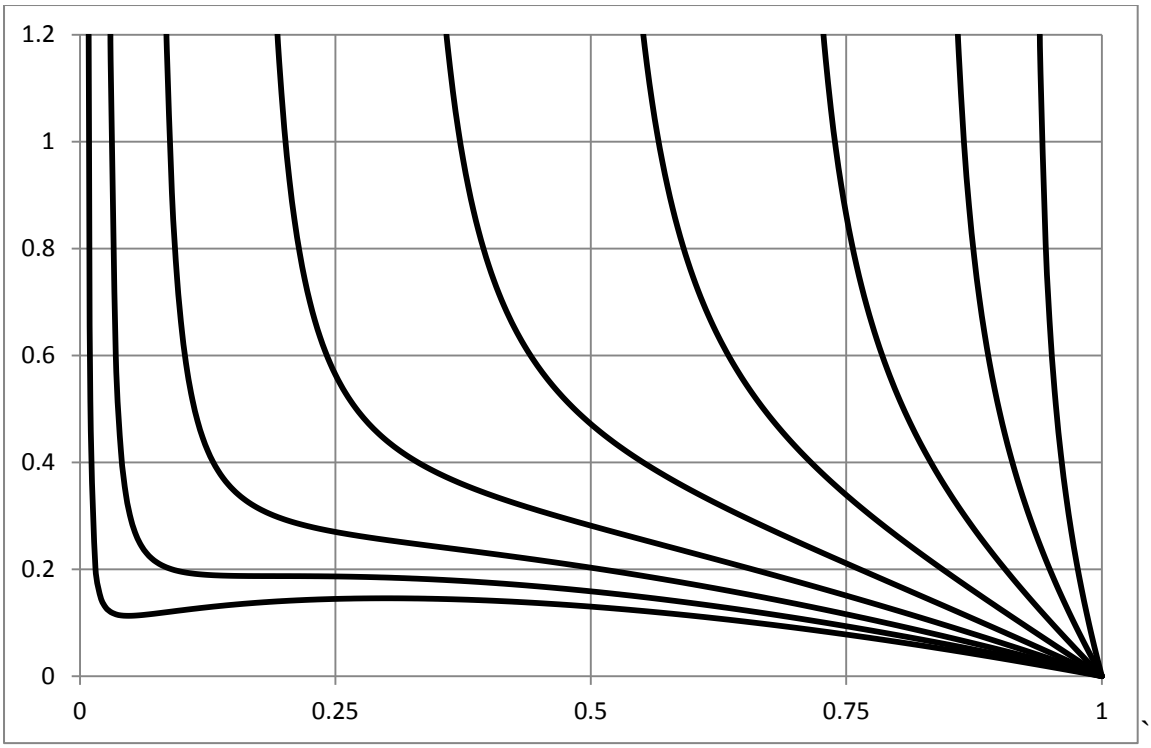

Figure 9. $\mu-\sigma$ lognormal contours for Model 2 (z_2* = 2.5, 2.0, ..., −1.5)

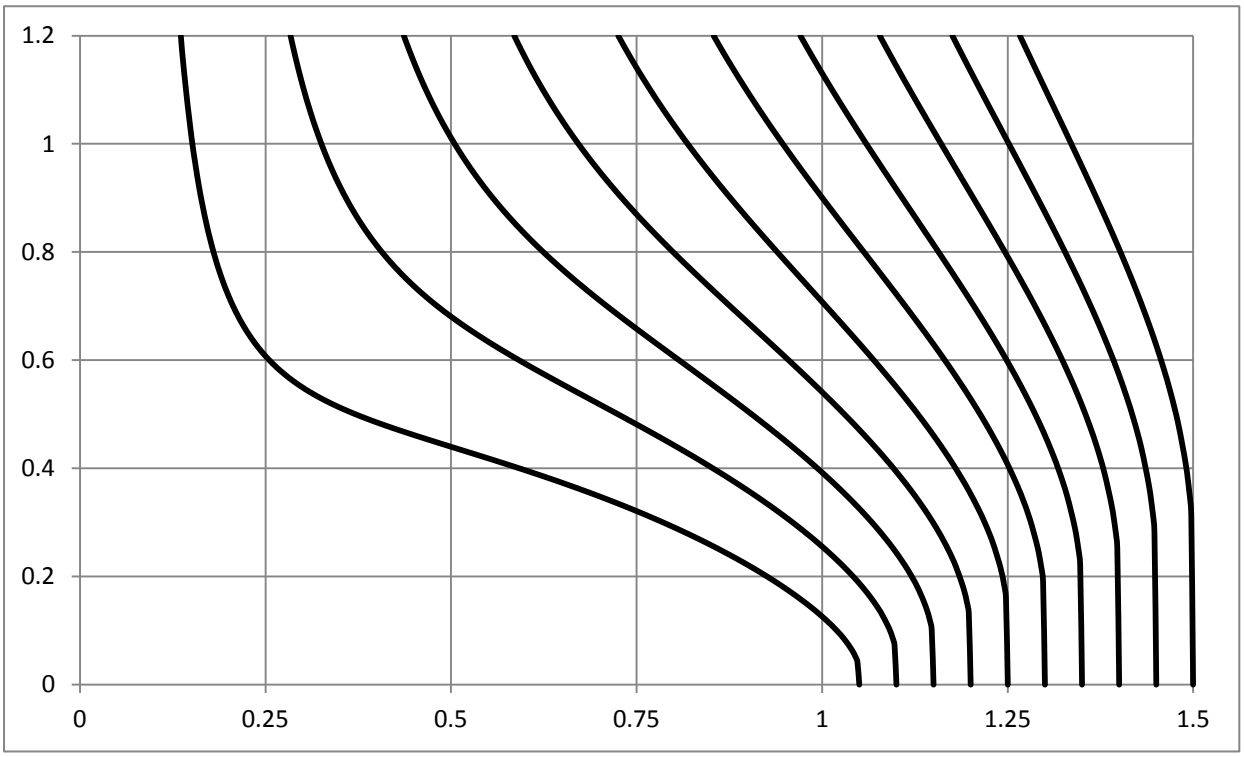

Figure 10. $\mu-\sigma$ lognormal contours for Alternate 2 (E(T) = 5%, 10%, ..., 50%)