In many experiments, some of the factors exist only within the level of another factor. Such factors are often called nested factors. A factor within which other factors are nested is called a branching factor. Suppose, for example, that we want to experiment with two processing methods. The factors involved in these two methods can be different. Thus in this experiment, the processing method is a branching factor, and the other factors are nested within the branching factor. The design and analysis of experiments with branching and nested factors are challenging and have not received much attention in the literature. Motivated by a computer experiment in a machining process, we have developed optimal Latin hypercube designs and kriging methods that can accommodate branching and nested factors. Through the application of the proposed methods, optimal machining conditions and tool edge geometry are attained, which resulted in a remarkable improvement in the machining process.

KEY WORDS: Finite element model; Kriging; Latin hypercube design.

1. INTRODUCTION

Nested factors are those factors that exist only within the level of another factor. A factor within which other factors are nested is called a branching factor. Suppose, for example, that we want to experiment with two surface preparation methods in printed circuit board (PCB) manufacturing: mechanical scrubbing and chemical treatment. Mechanical scrubbing can be optimized by changing the pressure of the scrub, and chemical treatment can be optimized by changing the micro-etch rate. Here the surface preparation method is the branching factor, and pressure and micro-etch rate are the nested factors. When designing an experiment, the two nested factors (pressure and micro-etch rate) are assigned to the same column in the design matrix. If we choose two levels for the pressure and micro-etch rate, then the experimental design will appear like that shown in Table 1. Because nested factors can differ with respect to the level of branching factor, designing and analyzing experiments with such factors is not trivial.

Taguchi (1987, p. 280) proposed an innovative approach to designing experiments with branching and nested factors. He called nested factors “pseudo-factors” and the resulting designs “pseudo-factor designs.” Phadke (1989, p. 168) called these “branching designs.” The core idea is to carefully assign branching and nested factors to the columns of orthogonal arrays using linear graphs in such a way that their interactions can be estimated. The interactions between branching and nested factors are important, because the nested factors differ with respect to the levels of the branching factors, and thus their effects can change depending on the level of the branching factors. Consider the PCB experiment, for example. As shown in Figure 1(a), the surface quality of the PCB may increase with increasing pressure but may decrease with increasing micro-etch rate. There is strong interaction between the branching and the nested factors. The interaction effect could be reduced if the effects of the nested factors were known before the experiment; for example, if the two pressure levels are interchanged, then the interaction becomes small [see Figure 1(b)]. In general, however, the effects of the nested factors are not known before the experiment, and thus this cannot be done. Furthermore, any factor correlated with a significant branching-by-nested interaction will be misspecified. Therefore, we should design experiments that are capable of estimating the potentially large branching-by-nested interactions. Although Taguchi’s approach using orthogonal arrays and linear graphs is very intuitive, it is not sufficiently general to apply to more complex situations, such as the design of computer experiments.

The designs that we consider here differ from the so-called “nested designs” reported in the literature (see, e.g., Hicks and Turner 1999, p. 190; Montgomery 2004, p. 525), in which the nested factors are assumed to be similar (e.g., different batches of material nested within different suppliers). Because
Table 1. Experimental design for the PCB experiment

<table>
<thead>
<tr>
<th>Run</th>
<th>Method (branching factor)</th>
<th>Nested factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>mechanical scrubbing</td>
<td>pressure_1</td>
</tr>
<tr>
<td>2</td>
<td>mechanical scrubbing</td>
<td>pressure_2</td>
</tr>
<tr>
<td>3</td>
<td>chemical treatment</td>
<td>micro-etch rate_1</td>
</tr>
<tr>
<td>4</td>
<td>chemical treatment</td>
<td>micro-etch rate_2</td>
</tr>
</tbody>
</table>

The nested factors are similar, the branching-by-nested interactions do not play a critical role, as they do in the present problem. Moreover, in nested designs, the nested factors are usually treated as random effects, and the main focus is on estimating the variance components, whereas in the present problem, the nested factors are treated as fixed effects, and the objective is to simultaneously identify the optimal settings of branching, nested, and other factors.

Our work is motivated by a computer experiment involving branching and nested factors in which the objective is to optimize a turning process for hardened bearing steel with a cBN cutting tool (see Figure 2). This process, commonly known as hard turning, is of considerable interest to bearing manufacturers as a potential replacement for the grinding process. Because the material being machined is very hard (hardness > 60 Rockwell C), the cutting tool is subjected to high forces, stresses, and temperatures during the operation. In practice, the tool’s cutting edge is shaped to withstand the severe conditions. Two commonly used cutting edge shapes, hone and chamfer, are shown in Figure 3. Note that Figure 3 represents the idealized view of the instantaneous cutting action in the cross-section “A–A” shown in Figure 2. These cutting edge shapes are intended to strengthen the cutting edge to bear the large tool stresses generated during cutting. The chamfer tool design can be changed using two factors, chamfer length and chamfer angle, whereas the hone design is fixed. In other words, the two factors length and angle are nested within the chamfer edge, and no factors are nested within the hone edge. In our terminology, the tool edge is a branching factor. Thus, when the branching factor takes the level chamfer, two additional factors are present in the experiment; but when the branching factor takes the level hone, no additional factors are present. A few other factors are common to both of the tool edges, including the cutting edge radius, tool nose radius, and rake angle. In addition, the machining parameters, such as cutting speed, feed, and depth of cut, also do not depend on the type of tool edge (see Figure 2). To distinguish these factors from the branching and nested factors, we call them “shared factors.” Table 2 lists all of the factors involved in this experiment and their allowable ranges. The experiments can be performed with computers using the commercially available finite element software AdvantEdge.

Latin hypercube designs (LHDs) are commonly used in computer experiments (McKay, Beckman, and Conover 1979). A desirable property of a LHD is its one-dimensional balance; that is, when an N-point design is projected onto any factor, there will be N different levels for that factor. Clearly, this cannot be satisfied for branching and nested factors. The branching factor is usually a qualitative factor, and thus the number of levels of a branching factor is fixed (i.e., it does not depend on the number of runs N). Moreover, the nested factors differ for different levels of the branching factor. Therefore, we need...
a one-dimensional balance for the nested factors within each level of the branching factor. As an example, consider an experiment with one branching factor $z_1$, one nested factor $v_1$, and two shared factors $x_1$ and $x_2$. Suppose that the branching factor has two levels and that we want to do the experiment in eight runs. Table 3 presents a possible design for this experiment. We can see that the shared factors have a one-dimensional balance, because they take eight different levels in the experiment. The nested factor has a one-dimensional balance within each level of the branching factor. Note that $v_1$ represents two different factors, one when $z_1 = 1$ and the other when $z_1 = 2$; therefore, $v_1 = 1$ in run 1 is not the same as $v_1 = 1$ in run 5. In the next section we discuss some general strategies for designing such experiments using LHDs.

It is well known that not all LHDs are "good." Most of the research in this area has focused on finding "good" LHDs based on some optimal design criteria (see Iman and Conover 1982; Tang 1993; Owen 1994; Morris and Mitchell 1995; Tang 1998; Ye 1998; Ye, Li, and Sudjianto 2000; Jin, Chen, and Sudjianto 2005; Joseph and Hung 2008). We need to extend those optimal design criteria for experiments with branching and nested factors. Take, for example, the case of maximin LHD proposed by Morris and Mitchell (1995), where the optimal design criterion is to maximize the intersite distance among the experimental points (runs). Now with the branching and nested factors, the notion of "distance" does not exist for all factors. Branching factors are qualitative and thus cannot be measured by distances. Moreover, for nested factors, the notion of "distance" exists only if the corresponding levels of the branching factor are the same. Another major factor that makes the design of experiment different from the usual designs is the importance of the interaction between branching and nested factors. As noted earlier, these interactions usually are not negligible; therefore, if any of the main effects is highly correlated with one of these interactions, then that main effect will be misspecified. Thus the optimal design criteria should be modified to capture the branching-by-nested interaction effects.

The remainder of the article is organized as follows. In Section 2 we discuss some general strategies for designing experiments with branching and nested factors and introduce the concept of branching LHD. In Section 3 we discuss three different criteria for finding an optimal branching LHD. We report an analysis of experiments with branching and nested factors in Section 4, and illustrate the proposed methods using the hard turning experiment in Section 5. We end with some concluding remarks and directions for future research in Section 6.

### 2. BRANCHING LATIN HYPERCUBE DESIGNS

In general, an LHD with $N$ runs and $p$ factors, denoted by $\text{LHD}(N, p)$, can be generated using a random permutation of $\{1, 2, \ldots, N\}$ for each factor. This cannot be done if the experiment involves branching and nested factors, however, as discussed earlier.

Consider a simple case in which the branching factor $z_1$ has only two levels and $m_1$ factors are nested within each level of the branching factor. Thus there are $m_1$ nested factors $(v_1^{z_1}, \ldots, v_{m_1}^{z_1})$, where each of the nested factors represents two different factors, depending on the two levels of the branching factor. In addition, there are $t$ more shared quantitative factors.

<table>
<thead>
<tr>
<th>Type of factor</th>
<th>Notation</th>
<th>Factor</th>
<th>Ranges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Branching factor</td>
<td>$z_1$</td>
<td>Cutting edge shape</td>
<td>hone or chamfer</td>
</tr>
<tr>
<td>Nested factors</td>
<td>$v_1</td>
<td>z_1 \leq \text{chamfer}$</td>
<td>Angle (degree)</td>
</tr>
<tr>
<td></td>
<td>$v_2</td>
<td>z_1 \leq \text{chamfer}$</td>
<td>Length ($\mu$m)</td>
</tr>
<tr>
<td></td>
<td>$v_1</td>
<td>z_1 \leq \text{hone}$</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>$v_2</td>
<td>z_1 \leq \text{hone}$</td>
<td>None</td>
</tr>
<tr>
<td>Shared factors</td>
<td>$x_1$</td>
<td>Cutting edge radius ($\mu$m)</td>
<td>$5 \sim 25$</td>
</tr>
<tr>
<td></td>
<td>$x_2$</td>
<td>Rake angle (degree)</td>
<td>$-15 \sim -5$</td>
</tr>
<tr>
<td></td>
<td>$x_3$</td>
<td>Tool nose radius (mm)</td>
<td>$0.4 \sim 1.6$</td>
</tr>
<tr>
<td></td>
<td>$x_4$</td>
<td>Cutting speed (m/min)</td>
<td>$120 \sim 240$</td>
</tr>
<tr>
<td></td>
<td>$x_5$</td>
<td>Feed (mm/rev)</td>
<td>$0.05 \sim 0.15$</td>
</tr>
<tr>
<td></td>
<td>$x_6$</td>
<td>Depth of cut (mm)</td>
<td>$0.1 \sim 0.25$</td>
</tr>
</tbody>
</table>

![Figure 3. Illustration of hone and chamfer tool edges.](image)
We discuss some strategies for constructing LHDs with branching and nested factors.

A naive strategy is to choose an LHD for the nested and shared factors and repeat it under each level of the branching factor; that is, first choose an LHD($n_0$, $m_1 + t$) that can accommodate the nested and shared factors, then repeat it for the two levels of the branching factor. The resulting design, shown in Table 4, is easy to construct. Moreover, optimal LHDs for the nested and shared factors can be readily chosen using the existing methods. In addition, because all of the combinations of nested and shared factors are repeated at each level of the branching factor, the interactions involving the branching factor can be estimated. This approach has two drawbacks, however. First, if the design matrix is projected onto one of those shared factors, then some replications occur; thus the design points are not spread out as uniformly as they could be. Second, the run size of these designs can be quite large.

The foregoing problems with the naive strategy can be easily overcome by using one LHD for all of the shared factors. By doing this, the design points are spread out more uniformly in the experimental region, and the required run size is comparatively smaller. As an example, for $m_1 = 3$ and $t = 5$, the run size of the naive approach ($2n_0$ in Table 4) should be at least 16 (because $n_0 \geq 8$). This can be reduced to 6 ($2n_1 \geq 6$) by the new design illustrated in Table 5. Following the terminology used by Phadke (1989, p. 168), we call a design with this structure a branching Latin hypercube design (BLHD).

Now consider a more general case with $q$ branching factors denoted by $z = (z_1, \ldots, z_q)$. Assume that all of these factors are qualitative by nature. For each branching factor, there are $k_u$ levels, and under each of these different levels, there are $m_u$ nested factors. Note that in general, the number of factors nested under each level of a branching factor can differ; for example, in the hard turning experiment, two factors are nested under chamfer but no factors are nested under hone. Here, for notational simplicity, we assume that the number of nested factors is the same (for a given branching factor) and develop the construction of the BLHD. Later we explain how this can be extended to deal with unequal numbers of nested factors.

Denote the nested factors by $v^u = (v_1^u, \ldots, v_{m_u}^u)$, $1 \leq u \leq q$. Again note that each nested factor corresponds to different factors, depending on the branching factor and its level; that is why we use a superscript to denote the branching factor level. In addition to the branching and nested factors, there are $t$ shared quantitative factors, $x = (x_1, \ldots, x_t)$. Let $v = ((v_1^1, \ldots, v_{m_1}^1), \ldots, (v_1^q, \ldots, v_{m_q}^q))$, and $w = (x', z', v')$ represent all of the $p$ factors involved in the experiment, where $p = t + q + \sum_{u=1}^{q} m_u$. A $N$-run BLHD then can be denoted by $W = (w_1, \ldots, w_N)$. In general, this consists of three parts. The first part is a design for branching factors. Because branching factors are qualitative factors, we can choose an orthogonal array of appropriate size depending on the number of levels of each branching factor. The second part comprises LHDs for the nested factors. Choose LHD($n_u$, $m_u$) for the $m_u$ nested factors under the branching factor $z_u$, $u = 1, 2, \ldots, q$. The third part is an LHD($N, t$) for all of the shared factors $x$. If there are $k_u$ levels for branching factor $z_u$, where $1 \leq u \leq q$, then clearly $N = k_1 n_1 = k_2 n_2 = \cdots = k_q n_q$. Thus, we have one orthogonal array for the branching factors, $q$ LHDs for the nested factors, and one LHD for the shared factors. These designs can be assembled to obtain a BLHD.

As an example, consider the case with two branching factors ($z_1$ and $z_2$) each at two levels. There are $m_1$ nested factors under $z_1$ and $m_2$ nested factors under $z_2$. Furthermore, there are $t$ shared factors. Table 6 illustrates a $N$-run BLHD for this example. The first part is a four-run orthogonal array for those two branching factors. For the second part, we choose LHD($n_1$, $m_1$) for the nested factors under $z_1$. Similarly, LHD($n_2$, $m_2$) is chosen for the nested factors under $z_2$. This LHD is divided into two halves and distributed among the two levels of $z_2$, as shown in the table. The third part consists of an LHD($N$, $t$) for the $t$ shared factors.

As is the case with LHDs, not all BLHDs are good. Some optimal design criteria are needed to ensure the best BLHD. We address this in the next section.

### Table 3. An example of a branching LHD

<table>
<thead>
<tr>
<th>Run</th>
<th>$z_1$</th>
<th>$v_1$</th>
<th>$x_1$</th>
<th>$x_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

### Table 4. Illustration of the naive strategy

<table>
<thead>
<tr>
<th>Run</th>
<th>$z_1$</th>
<th>$v_1^2 \cdots v_{m_1}^2$</th>
<th>$x_1$</th>
<th>$x_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>LHD($n_0$, $m_1 + t$)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$n_0$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$n_0 + 1$</td>
<td>2</td>
<td>LHD($n_0$, $m_1 + t$)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 5. Branching LHD with one branching factor

<table>
<thead>
<tr>
<th>Run</th>
<th>$z_1$</th>
<th>$v_1^2 \cdots v_{m_1}^2$</th>
<th>$x_1$</th>
<th>$x_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$n_1$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$n_1 + 1$</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$2n_1$</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3. OPTIMAL BRANCHING LATIN HYPERCUBE DESIGNS

As discussed in Section 1, several approaches to finding a good LHD are available. Using one of those approaches to generate \( q + 1 \) optimal LHDs for the nested and shared factors and assemble them to obtain the BLHD may be considered sufficient. Such an assembly of optimal LHDs may not lead to an optimal BLHD, however. Moreover, we need to make sure that in a BLHD, the correlation between the branching-by-nested interaction and any other main effect is small. In this section we propose three optimal design criteria for finding good BLHDs.

3.1 Maximin Branching Latin Hypercube Design

Morris and Mitchell (1995) proposed finding LHDs that maximize the minimum intersite distance. Let \( g \) and \( h \) be two design points (or sites or runs). Consider the distance measure \( d(g, h) = \sum_{j=1}^{p} |g_{j} - h_{j}|^{\frac{1}{\zeta}} \), in which \( \zeta = 1 \) and \( \zeta = 2 \) correspond to the rectangular and Euclidean distances. For simplicity, we consider only the rectangular distance (\( \zeta = 1 \)) hereinafter. For a given LHD, define a distance list \( (D_{1}, D_{2}, \ldots, D_{M}) \) in which the elements are the distinct values of the intersite distances, sorted from the smallest to the largest. Let \( J_{l} \) be the number of pairs of design points in the design separated by \( D_{l} \). Then a design is called a maximin design if it sequentially maximizes the \( D_{l}'s \) and minimizes the \( J_{l}'s \) in the following order: \( D_{1}, J_{1}, D_{2}, J_{2}, \ldots, D_{M}, J_{M} \). A scalar-valued function that can be used to rank competing designs in such a way that the maximin design receives the highest ranking is given by

\[
\phi_{\lambda} = \left( \sum_{l=1}^{M} J_{l} D_{l}^{-\lambda} \right)^{1/\lambda} = \left( \sum_{g \neq h} d(g, h)^{-\lambda} \right)^{1/\lambda}, \tag{1}
\]

where \( \lambda \) is a positive integer.

Extension of the maximin criterion to BLHDs is not straightforward. In contrast to LHDs, in which all factors can be measured by distances, BLHDs have branching factors that have no notion of distance and nested factors for which the definition of distance depends on the corresponding branching factors. Because of the different roles of factors, instead of calculating all of the pairwise distances over all factors, we need to consider branching and nested factors separately from shared factors. First, note that for BLHDs, not all factors are divided into the same number of levels as in LHDs; there are \( k_{u} \) levels for branching factor \( z_{u} \), \( n_{h} \) levels for nested factors \( v_{h} \), and \( N \) levels for \( x \). Therefore, before calculating the distances, the design matrix should be scaled to \((-1, 1)\).

Start with a simple case in which \( q = 1 \) and \( m_{1} = 1 \) and thus there are \( t + 2 \) factors in the experiment. Assume that the last two factors are the branching factor \( z_{1} \) and the nested factor \( v_{1}^{2} \). We need to define two types of intersite distances. The first type of distance focuses on all of the shared factors. It is the distance on the \( i \)-dimensional space \( (x) \), which can be defined by

\[
d_{i}(g, h) = \sum_{j=1}^{q} |g_{j} - h_{j}|, \quad g = (g_{1}, \ldots, g_{t+2}) \quad \text{and} \quad h = (h_{1}, \ldots, h_{t+2}).
\]

We can easily obtain

\[
d_{i}(g, h) = d_{x}(g, h) + d_{z_{1}}(g, h),
\]

where \( d_{z_{1}}(g, h) = |g_{t+1} - h_{t+1}| \) is the sum of \( \binom{N}{2} \) pairwise distances, in which \( g \) and \( h \) have the same level of branching factor and \( \sum_{g \neq h} d_{i}(g, h) = \sum_{i=1}^{q} \sum_{g_{t+1} = h_{t+1} = z_{1}} d_{x}(g, h) + \sum_{g_{t+1} \neq h_{t+1}} d_{i}(g, h) \). To illustrate this idea, consider the simple example given in Table 3. Assume that the branching factor \( z_{1} \) is qualitative.

The optimal design found by the modified maximin criterion (2) (with \( \lambda = 15 \)) is \( x_{1} = \{1, 5, 6, 4, 7, 3, 2, 8\} \), \( x_{2} = \{4, 8, 1, 5, 6, 2, 7, 3\} \), and \( z_{1} \) and \( v_{1} \) remain the same as in Table 3. This maximin BLHD is plotted in Figure 4, with the “×”s representing the design points with \( z_{1} = 1 \) and solid points representing those with \( z_{1} = 2 \). The first part in (2) tries to maximize the intersite distances in the space of \( x_{1} \) and \( x_{2} \) [Figure 4(a)], in which the “×” points and solid points are not dis-

<table>
<thead>
<tr>
<th>Run</th>
<th>( z_{1} )</th>
<th>( v_{1}^{2} )</th>
<th>( v_{2}^{1} )</th>
<th>( v_{2}^{2} )</th>
<th>( x_{1} )</th>
<th>( x_{2} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>( 1 )</td>
<td>( LHD(n_{1}, m_{1}) )</td>
<td>( 1 )</td>
<td>( LHD(n_{2}, m_{2}) ) first half</td>
<td></td>
</tr>
<tr>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
</tr>
<tr>
<td>\vdots</td>
<td>2</td>
<td>( LHD(n_{1}, m_{1}) )</td>
<td>( 1 )</td>
<td>( LHD(n_{2}, m_{2}) ) second half</td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>2</td>
<td>( LHD(n_{1}, m_{1}) )</td>
<td>( 1 )</td>
<td>( LHD(n_{2}, m_{2}) ) second half</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Throughout the design points are distributed evenly and Figure 5(a)] are isomorphic; however, these two isomorphisms are not equally good when the branching factor is distinguished. In contrast, the second part tries to maximize the intersite distances in the space of $x_1$, $x_2$, and $v_1$ [Figures 4(b) and (c)]. Moreover, because these distances are calculated only within the same level of the branching factor, the intersite distances among the “*” points and among the solid points are maximized.

If only the first part in (2) were used as the criterion, then the optimal design would be space-filling only over the shared factors. The design points can be structured with respect to the branching factor and the corresponding nested factors. As an extension of (2), the maximin criterion for BLHDs can be written as

$$
\phi_3 = \left( \sum_{g \neq h} \left[ \frac{t}{d_s(g, h)} \right]^\lambda \right) + \left[ \sum_{u=1}^{q} \sum_{g_u = h_{g_u} = z_{u,i}, i = 1}^{k_u} \left[ \frac{m_u + t}{d_s(g, h) + d_s(g, h)} \right]^\lambda \right]^{1/\lambda},
$$

where $d_s(g, h) = \sum_{j=1}^{l} |g_j - h_j|$ and the distance measure for the $u$th branching factor is denoted by $d_u(g, h) = \sum_{i=1}^{m_u} z_{u,i} |g_i - h_i|$. This criterion is general and includes some interesting special cases.

**Case 1.** If $q = 0$ and $m_u = 0$ for all $u$, then this will lead to the standard LHD($N, t$). Up to a constant, the maximin criterion (3) will be the same as (1) proposed by Morris and Mitchell (1995).

**Case 2.** If there is no nested factor corresponding to the branching factors (i.e., $m_u = 0$ for all $u$), then this can be considered an experiment with $q$ qualitative factors $z$ and $t$ quantitative factors $x$. As a special case of (3), the maximin criterion for experimental design with quantitative and qualitative factors can be written as

$$
\phi_3 = \left( \sum_{g \neq h} \left[ \frac{t}{d_s(g, h)} \right]^\lambda \right) + \left[ \sum_{u=1}^{q} \sum_{g_u = h_{g_u} = z_{u,i}, i = 1}^{k_u} \left[ \frac{t}{d_s(g, h)} \right]^\lambda \right]^{1/\lambda},
$$

where $d_s(g, h) = \sum_{j=1}^{l} |g_j - h_j|$ and the distance measure for the $u$th branching factor is denoted by $d_u(g, h) = \sum_{i=1}^{m_u} z_{u,i} |g_i - h_i|$. This criterion is general and includes some interesting special cases.

**Case 3.** Another special case is when $t = 0$. In this situation, experiments have branching factors and nested factors but no...
shared factors. Because $d_i(g, h) = 0$, for all $g$ and $h$, (3) can be simplified to

$$
\phi_\lambda = \left( \sum_{a=1}^{q} \sum_{i=1}^{k_a} \sum_{g_{hi}=h_{ai}=z_{ai}}^{N} \left[ \frac{d_i(g, h)}{m_a} \right]^{\lambda} \right)^{1/\lambda}.
$$

3.2 Minimum Correlation Branching
Latin Hypercube Design

Along with space-filling, another important issue in experimental designs is how to construct them such that the significant factors can be correctly identified. For LHDs, this can be achieved by minimizing the pairwise correlation among factors (Iman and Conover 1982; Owen 1994; Tang 1998). Owen (1994) proposed a performance measure $\rho^2$ for evaluating the goodness of an LHD with respect to pairwise correlations. For LHD($N$, $t$),

$$
\rho^2 = \frac{\sum_{i=2}^{t} \sum_{j=1}^{t-1} \rho_{ij}^2}{t(t-1)/2},
$$

(5)

where $\rho_{ij}$ is the linear correlation between columns $i$ and $j$.

In contrast to LHDs, in BLHDs orthogonality among main effects is not sufficient—considering the branching-by-nested interactions is equally important. Thus we propose a modified correlation criterion that minimizes the correlations among the main effects of all factors, as well as those between the main effect of a shared factor and a branching-by-nested interaction effect. We first enlarge the BLHDs by including two-factor interactions that represent the branching-by-nested interactions. There are $m_u$ such interactions for each branching factor $z_u$, where $m_u$ is the number of nested factors; therefore, the total number of branching-by-nested interactions is $s$, where $s = \sum_{u=1}^{q} m_u$. Thus the new criterion for BLHDs is given by

$$
\rho^2 = \frac{\sum_{i=2}^{t} \sum_{j=1}^{t-1} \rho_{ij}^2 + \sum_{i=1}^{t} \sum_{j=1}^{t} \rho_{ij}^2}{(\rho - 1)/2 + st},
$$

(6)

where $\rho_{ij}$ is the linear correlation between columns $i$ and $j$ in the design and $\rho_{ij}^2$ is the linear correlation between $x_i$ and the $j$th branching-by-nested interaction.

Consider the example used in Table 3. The optimal design that minimizes $\rho^2$ in (6) is $x_1 = (2, 8, 5, 3, 4, 7, 1, 6)$, $x_2 = (2, 8, 5, 3, 6, 1, 7, 4)$, and $z_1$ and $v_1$ remain the same as in Table 3. This design is plotted in Figure 6.

3.3 Orthogonal-Maximin Branching
Latin Hypercube Design

Maximizing minimum intersite distances does not ensure minimizing pairwise correlations, and vice versa. Therefore, Joseph and Hung (2008) proposed a multiobjective criterion for LHD that combines the maximin distance and the minimum correlation criteria. This criterion becomes even more important in the case of BLHDs, because it is important to ensure small correlations between the shared factors and the branching-by-nested interactions along with ensuring good space-filling properties. To extend the result of Joseph and Hung (2008) to BLHDs, we should scale $\phi_\lambda$ and $\rho^2$ to the same range, so that some meaningful weights can be assigned in the multiobjective function. The following result gives the lower and upper bounds for $\phi_\lambda$, which can be used for scaling it to $[0, 1]$. Here we consider only one of the case of a single branching factor; the results can be extended to include more than one branching factor, but then the expressions become more complicated.

Proposition 1. If there is only one branching factor, then $\phi_{\lambda,L} \leq \phi_\lambda \leq \phi_{\lambda,U}$, where

$$
\phi_{\lambda,L} = 3 \left[ \sum_{i=1}^{k_1} \sum_{g_{hi}=h_{ai}=z_{ai}}^{N} (m_1 + t)^{1/(p+1)} \right]^{(\lambda+1)/\lambda},
$$

and

$$
\phi_{\lambda,U} = \frac{N}{2} \left[ \sum_{i=1}^{k_1} \sum_{j=1}^{n_i-1} (n_1 - j)(t + m_1)^{1/(p+1)} + \sum_{j=1}^{N-1} \frac{N-j}{\lambda} \right]^{1/\lambda}.
$$

Thus the multiobjective criterion is to minimize

$$
\psi_\lambda = w \rho^2 + (1 - w) \frac{\phi_\lambda - \phi_{\lambda,L}}{\phi_{\lambda,U} - \phi_{\lambda,L}}.
$$

(7)

We usually take $w = 0.5$ and call the design that minimizes this criterion orthogonal-maximin BLHD.

[Figure 6. Minimum correlation BLHD. “X” represents $z_1 = 1$, and the solid points represent $z_1 = 2$.]

TECHNOMETRICS, NOVEMBER 2009, VOL. 51, NO. 4
The design matrix used in Table 3 is an orthogonal-maximin BLHD. This matrix is plotted in Figure 7. The optimal designs found by the forgoing three criteria (plotted in Figures 4, 6, and 7) are compared in Table 7. It can be seen that the correlation between the shared factor and the branching-by-nested interaction (denoted by INT) is 0 in the case of minimum correlation design; however, the points are much closer compared to the maximin BLHD. The orthogonal-maximin BLHD provides a good compromise between these two designs.

Because of the combinatorial nature of the optimization problem, finding the optimal BLHD for large dimensions is a difficult task. Several methods for finding the optimal LHD, including simulated annealing (Morris and Mitchell 1995), the columnwise-pairwise algorithm (Ye, Li, and Sudjianto 2000), the enhanced stochastic evolutionary algorithm (Jin, Chen, and Sudjianto 2005), and modified simulated annealing (Joseph and Hung 2008), have been proposed. These methods can be easily adapted for finding the optimal BLHD as well. A C++ code based on the algorithm of Joseph and Hung (2008) is available from the authors on request.

4. KRINGING WITH BRANCHING AND NESTED FACTORS

In this section we explain how branching and nested factors can be incorporated in kriging (Sacks et al. 1989). Although similar extensions can be made on other methods, such as linear regression and spline methods, here we focus on kriging because of its popularity in computer experiments (Santner, Williams, and Notz 2003, p. 86; Fang, Li, and Sudjianto 2006, p. 28). We note that if branching and nested factors are encountered in a physical experiment, then regression models and experimental designs based on orthogonal arrays should be considered.

The ordinary kriging model is given by $Y(w) = \mu + Z(w)$, where $Z(w)$ is a weakly stationary stochastic process with mean 0 and covariance function $\sigma^2\psi$ and $w \in \mathbb{R}^p$. The correlation function is defined as $\text{cor}(Y(w_1), Y(w_2)) = \psi(w_1, w_2)$. Usually a product correlation structure is assumed for the correlation function. Consider the example in Table 3. The correlation function between two points $w_1 = (x_{11}, x_{12}, z_{11}, v_{11}^1)$ and $w_2 = (x_{21}, x_{22}, z_{21}, v_{21}^1)$ can be described as a product of correlation functions of each factor ($\psi_i$) as follows: In the usual cases, a common correlation function is chosen for each factor; however, this cannot be done in the present problem, because of the different types of factors.

For the shared factors, a Gaussian correlation function may be used (Santner et al. 2003, p. 36),

$$\psi_i(x_{1i}, x_{2i}) = \exp\{-\alpha_i(x_{1i} - x_{2i})^2\},$$

whereas for a branching factor, an isotropic correlation function may be used (Joseph and Delaney 2007; Qian, Wu, and Wu 2008),

$$\xi_i(z_{1i}, z_{2i}) = \exp\{-\theta_i I_{[z_{1i} \neq z_{2i}]}\}.$$

Here $\alpha_i$ and $\theta_i$ are correlation parameters, and $I_A$ is an indicator function that takes value 1 when $A$ is true and 0 otherwise.

A new correlation function needs to be developed for nested factors. Assume that these are quantitative factors. We cannot use the Gaussian correlation function here, because a nested factor represents different factors depending on the level of the branching factor. Therefore, using one correlation parameter for a given nested factor is not reasonable. Instead, they should be different, depending on the level of branching factor. For the example in Table 3, the correlation function for $v_{1i}^1$ can be defined as follows. If two points have the same level in the branching factor (e.g., $z_{11} = z_{21} = 1$), then the correlation function will be $\exp(-\gamma_1(v_{11}^1 - v_{21}^1)^2)$. Similarly, if $z_{11} = 2$, then the correlation function will be $\exp(-\gamma_2(v_{11}^1 - v_{21}^1)^2)$. If the two points do not have the same level in the branching factor (i.e., $z_{11} \neq z_{21}$), then the correlation should be determined by

<table>
<thead>
<tr>
<th>Correlation</th>
<th>Maximin distance</th>
<th>Minimum correlation</th>
<th>Orthogonal-maximin</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi_k$</td>
<td>2.36</td>
<td>4.35</td>
<td>2.37</td>
</tr>
<tr>
<td>$D_1(J_1)$</td>
<td>$\frac{1}{2}$ (12)</td>
<td>$\frac{1}{4}$ (3)</td>
<td>$\frac{1}{4}$ (13)</td>
</tr>
<tr>
<td>$\rho^2$</td>
<td>0.0287415</td>
<td>0.000283</td>
<td>0.01445</td>
</tr>
<tr>
<td>($\text{cor}(x_1, \text{INT}), \text{cor}(x_2, \text{INT}))$</td>
<td>(0.195, 0)</td>
<td>(0, 0)</td>
<td>(−0.098, 0)</td>
</tr>
</tbody>
</table>
the branching factor, not the nested factor. Thus in this case, the correlation function for the nested factor will be equal to 1. The correlation function for the nested factor can be succinctly defined as

$$
\sigma_1(v_{11}^{21}, v_{21}^{21}) = \exp \left\{ - \sum_{j=1}^{2} \gamma_j (v_{11}^{21} - v_{21}^{21})^2 I_{[v_{11}^{21} = v_{21}^{21}]} \right\}.
$$

(8)

We can easily extend this to a more general situation. Assume that there are $q$ branching factors $z_1, \ldots, z_q$, and that for each branching factor $z_u$, there are $m_u$ nested factors. Also assume that $x_1 = (x_{11}, \ldots, x_{1p})$, $z_1 = (z_{11}, \ldots, z_{1q})$, $v_1 = (v_{11}, \ldots, v_{1m_1})$, and $v_1 = (v_{11}, \ldots, v_{1m_1})$. Similarly, $x_2 = (x_{21}, \ldots, x_{2p})$, $z_2 = (z_{21}, \ldots, z_{2q})$, $v_2 = (v_{21}, \ldots, v_{2m_2})$, and $v_2 = (v_{21}, \ldots, v_{2m_2})$. Given any two design points $w_1 = (x_1, z_1, v_1)$ and $w_2 = (x_2, z_2, v_2)$, the correlation function can be written as

$$
cor(Y(w_1), Y(w_2)) = \prod_{i=1}^{q} \psi_i(x_1, x_2) \times \prod_{u=1}^{m_u} [\xi_u(z_1, z_2) \prod_{j=1}^{m_u} \sigma_u(v_{1u}^{2u}, v_{2u}^{2u})],
$$

(9)

where $\psi_i(x_1, x_2) = \exp[-\alpha_i(x_{1i} - x_{2i})^2]$ is the correlation function for the shared factors, $\xi_u(z_1, z_2) = \exp[-\theta_u I_{[z_{1u} \neq z_{2u}]}]$ is the correlation function for the branching factors, and

$$
\sigma_u(v_{1u}^{2u}, v_{2u}^{2u}) = \exp \left\{- \sum_{j=1}^{k_u} \gamma_{uj}(v_{1u}^{2u} - v_{2u}^{2u})^2 I_{[z_{1u} = z_{2u} = z_{u}]} \right\}
$$

(10)

is the correlation function for the nested factors. Note that $z_{u,j}$, $1 \leq j \leq k_u$, are the $k_u$ levels for each branching factor $z_u$. Thus we obtain

$$
cor(Y(w_1), Y(w_2)) = \exp \left\{ - \sum_{i=1}^{p} \alpha_i(x_{1i} - x_{2i})^2 \right\} \times \prod_{u=1}^{m_u} \left[ \theta_u I_{[z_{1u} \neq z_{2u}]} + \sum_{j=1}^{k_u} \sum_{i=1}^{m_u} \gamma_{uj}(v_{1u}^{2u} - v_{2u}^{2u})^2 \right]
$$

$$
\times I_{[z_{1u} = z_{2u} = z_{u}]} \right\}. \right.
$$

(11)

Denote the correlation parameters by $\Theta = (\alpha', \theta', \gamma')$, where $\alpha = (\alpha_1, \ldots, \alpha_p)'$, $\theta = (\theta_1, \ldots, \theta_q)'$, and $\gamma = (\gamma_{11}, \ldots, \gamma_{qmq_k})'$. These parameters can be estimated from the data to obtain the ordinary kriging predictor, as we explain using an example in the next section.

5. HARD TURNING EXPERIMENT

The objective of our experiment is to optimize a hard turning process with respect to cutting forces. Hard turning is a metal cutting process that produces machined parts out of hard materials with good dimensional accuracy, surface finish, and surface integrity. Minimizing cutting forces will help reduce power requirements, elastic distortion of the workpiece, and tool wear; thereby reducing manufacturing costs and improving the quality of the machined part.

Nine factors are selected for experimentation, including one branching factor, two nested factors, and six shared factors. The factors and their ranges are given in Table 2. A 30-run orthogonal-maximin BLHD is generated by using the modified simulated annealing algorithm proposed by Joseph and Hung (2008). The optimal design matrix is given in Table 8. The branching factor (cutting edge shape, $z_1$), is labeled “1” for chamfer or “2” for hone. Two nested factors ($v_1$ and $v_2$) are nested within the cutting edge shape. Recall that if the cutting edge is chamfer, then $v_1$ represents chamfer angle and $v_2$ represents chamfer land length; otherwise, there is no factor.

The experiments are performed using the highly sophisticated finite-element–based machining simulation program AdvantEdge. This software models the underlying physics of metal cutting as a thermomechanical plastic deformation process and captures various material and geometric nonlinearities of the process. (The theoretical basis of the simulation model is provided in Marusich and Ortiz 1995.) The simulations are computationally intensive and require hours of running time (about 12–24 hours) to produce a single output. The simulation outputs are deterministic and incorporate all of the factors listed in Table 2. The software produces various responses, including temperature, residual stresses, and forces. A finite-element mesh and temperature distribution is shown in Figure 8. In this work, we chose to analyze only the resultant cutting force ($y$). The data are given in Table 8.

In this example, the number of nested factors is not the same for different levels of the branching factor. In the discussion in Section 3, we assumed these to be the same, for simplicity of notation. Here we explain how the criteria can be modified to deal with unequal numbers of nested factors. This is easily done. First, consider the maximin criterion $\phi_i$. In (3), use $m_1 = 2$ for the first 15 runs, and use $m_1 = 0$ for the last 15 runs. Now consider $\rho_i$ in (6). The pairwise correlations involving a nested factor should be calculated using the first 15 runs. Moreover, because there are no factors at a branching factor level of 2, we do not need to consider the branching-by-nested interaction.

Because the cutting forces are positive, we first apply a log transformation before fitting the ordinary kriging model. We also normalize all of the factor settings in Table 8 to $[-1, 1]$. The 30 design points after normalization are denoted by $\{w_1, \ldots, w_{30}\}$, where for all $1 \leq j \leq 30$, $w_j = (x_{j1}, \ldots, x_{j6}, z_{j1}, v_{j1}^{21}, v_{j1}^{22})$, $z_{j1} = -1$ represents the chamfer edge and $z_{j1} = 1$ represents the hone edge. The parameters in the kriging model can be estimated as (Santrner, Williams, and Notz 2003, p. 66)

$$
\hat{\Theta} = \arg \min_{\Theta} N \log \hat{\sigma}^2 + \log |\Psi|,
$$

$$
\hat{\mu} = (1^{\psi^{-1}})^{-1}1^{\psi^{-1}}y,
$$

$$
\hat{\sigma}^2 = \frac{1}{N} (y - \hat{\mu}1)^{\psi^{-1}}(y - \hat{\mu}1),
$$
where \( \mathbf{1} \) is a vector of 1’s with length 30, \( \mathbf{y} = (y_1, \ldots, y_{30})' \), and \( \Psi \) is a \( 30 \times 30 \) matrix whose \( nj \)th element is

\[
\exp \left\{ -\sum_{i=1}^{6} \hat{\alpha}_i (x_{ni} - x_{ji})^2 - \hat{\gamma}_1 \mathbf{1}_{[x_{ni} \neq x_{ji}]} \right. \\
- \hat{\gamma}_{111} (v_{n1} - v_{j1}) I_{[z_{ni} = z_{ji} = -1]} \\
- \hat{\gamma}_{121} (v_{n2} - v_{j2}) I_{[x_{ni} = x_{ji} = -1]} \}.
\]

We obtain \( \hat{\alpha} = (0.091, 0.014, 0.027, 0.009, 0.944, 1.082)' \), \( \hat{\beta} = \hat{\beta}_1 = 0.127 \), \( \hat{\gamma} = (\hat{\gamma}_{111}, \hat{\gamma}_{121})' = (0.140, 0.008)' \), \( \hat{\mu} = 5.124 \). Note that we do not need to estimate \( \gamma_{112} \) and \( \gamma_{122} \) in this example, because there no factors are nested within the hone edge. Thus the ordinary kriging predictor is given by (see, e.g., Joseph 2006)

\[
\hat{y}(\mathbf{w}) = 5.124 + \hat{\psi}(\mathbf{w}) \hat{\Psi}^{-1} (\mathbf{y} - 5.124 \mathbf{1}), \tag{12}
\]

where \( \mathbf{w} = (x_1, \ldots, x_6, z_1, v_{11}, v_{21})' \in [-1, 1]^9 \) and \( \hat{\psi}(\mathbf{w}) \) is a vector of length 30 with the \( j \)th element

\[
\exp \left\{ -\sum_{i=1}^{6} \hat{\alpha}_i (x_i - x_{ji})^2 - \hat{\gamma}_1 \mathbf{1}_{[x_i \neq x_{ji}]} \right. \\
- \hat{\gamma}_{111} (v_{i1} - v_{j1}) I_{[z_{ni} = z_{ji} = -1]} \\
- \hat{\gamma}_{121} (v_{i2} - v_{j2}) I_{[x_{ni} = x_{ji} = -1]} \}.
\]

To explore the effects of the factors, we applied the sensitivity analysis technique on the ordinary kriging predictor (see Welch et al. 1992). The main effects plot, given in Figure 9(a), shows that the cutting edge radius \( (x_1) \), feed \( (x_5) \), depth of cut \( (x_6) \), and chamfer angle \( (v_1) \) have significant effects on the cutting force. A significant interaction between cutting edge radius

\[
\hfill \text{Fig. 8. Finite-element mesh and temperature distribution.}
\]
and depth of cut also can be seen [Figure 9(b)]. The depth of cut has a positive effect on the cutting forces, but this effect is more significant when the cutting edge radius is smaller. This can be explained in physical terms as follows. For a small cutting edge radius, an increase in the depth of cut produces an increase in material deformation through shear, and thus a more significant effect on the force. For larger cutting edge radius values, the contribution of ploughing of material around the cutting edge to the cutting force is more pronounced, and consequently, an increase in depth of cut does not produce as significant a change in the cutting force.

The optimal setting of the factors can be found by minimizing the ordinary kriging predictor in (12). We obtain 
\[
(x_1, x_2, x_3, x_4, x_5, x_6, z_1, v_1, v_2) = (-1.00, -0.76, 0.69, 0.70, -0.66, -0.70, -1, 0.05, 0.16).
\]
In their original scales, the optimal setting for the shared factors is 
\[
(x_1, x_2, x_3, x_4, x_5, x_6) = (5, -13.80, 1.41, 222, 0.067, 0.123),
\]
and the optimal cutting edge geometry is chamfer, with an angle of 18.74 degrees and a length of 128.13 microns. The resulting cutting force predicted under this setting is 81 N, much smaller than the forces observed in the experiment. We also performed a new experiment at the optimal setting and obtained a resultant force of 79 N, confirming the validity of the optimal setting obtained with our model.

6. CONCLUSIONS

Surprisingly, the design and analysis of experiments with branching and nested factors have received scant attention in the literature. One possible reason for this could be that the experiments can be performed in two stages, with the first stage involving an experiment with the branching factors and shared factors. Because there are no nested factors, this experiment can be designed using existing methods. The data can be analyzed to determine the optimum level of the branching factor. Then a second stage of the experiment can be performed using only the nested factors under the optimum level of the branching factor. The design of this experiment also can be easily obtained using existing methods. Although this two-stage approach is quite intuitive, the final results may not be optimal. This is because a different level of the branching factor may be the true optimum, but it cannot be identified in the first stage of the experiment, because the nested factors under that branching level are not set at their optimal levels. This problem can be avoided by using branching designs, which determines the optimal settings of the branching factors, nested factors, and shared factors simultaneously.

Taguchi (1987, p. 280) and Phadke (1989, p. 168) have reported several case studies on experiments using branching designs; however, their approaches are not sufficiently general for more complex experiments, such as computer experiments. Moreover, the optimality properties of their approaches using orthogonal arrays are not known. In this article, we proposed BLHDs that are suitable for computer experiments involving branching and nested factors and discussed the optimal choice of such designs. We applied our proposed approach to the optimization of a machining process.

Although our primary focus here is on LHDs, some issues regarding the use of orthogonal arrays and their applications in physical experiments are important as well. Research on these topics is currently underway and will be reported elsewhere.

APPENDIX: PROOF OF PROPOSITION 1

If one branching factor \((q = 1)\) is present, then a BLHD will include \(k_1\) small LHDs \((n_1, m_1)\) for the \(k_1\) levels of the branching factor and a LHD\((N, r)\) for the shared factors. \(\phi_0\) can be...
written as
\[
\phi_{\lambda} = \left( \sum_{g \in h} \left[ \frac{t}{d_{i}(g, h)} \right]^{\lambda} \right)^{1/\lambda} + \sum_{i=1}^{k_{1}} \sum_{g_{i} = h_{i}, z_{1}} \left[ \frac{m_{1} + t}{d_{i}(g, h) + d_{i}(g, h)} \right]^{\lambda}.
\]
As shown by Joseph and Hung (2008), for a given LHD(\(N_{1}, t\)), the average intersite distance (rectangular measure) is \(N(\sum_{i=1}^{N_{1}} N_{1} - 1)/6\), which is a constant. With these constraints, finding a lower bound for \(\phi_{\lambda}\) can be formulated as a constraint minimization problem:

\[
\min \phi_{\lambda},
\]
subject to
\[
\sum_{g_{i} = h_{i}, z_{1}} d_{i}(g, h) = \frac{m_{1} n_{1} (N_{1} - 1)}{6}, \quad 1 \leq i \leq k_{1},
\]
\[
1 \leq i \leq k_{1},
\]
\[
\sum_{g \neq h} d_{i}(g, h) = \frac{t N(\sum_{i=1}^{N_{1}} N_{1} - 1)}{6},
\]
where \(\sum_{g_{i} = h_{i}, z_{1}} d_{i}(g, h)\) is the sum of intersite distances for those smaller LHDs \((m_{1}, m_{1})\) and \(\sum_{g \neq h} d_{i}(g, h)\) is the sum of intersite distances for LHD(\(N_{1}, t\)). Because
\[
\phi_{\lambda} \geq \phi_{\lambda}^{*} = \left( \sum_{g_{i} \neq h_{i}} \left[ \frac{t}{d_{i}(g, h)} \right]^{\lambda} \right)^{1/\lambda} + \sum_{i=1}^{k_{1}} \sum_{g_{i} = h_{i}, z_{1}} 2 \left[ \frac{m_{1} + t}{d_{i}(g, h) + d_{i}(g, h)} \right]^{\lambda},
\]
the lower bound can be found by minimizing \(\phi_{\lambda}^{*}\) with the same constraints as in (13). Thus the lower bound can be obtained using the Lagrange multiplier method. For the upper bound of \(\phi_{\lambda}\), the result for the BLHD is a simple extension of that for the LHD and thus can be proved by the same argument used by Joseph and Hung (2008).

ACKNOWLEDGMENTS

This research was supported by U.S. National Science Foundation grants CMMI-0654369 and CMMI-0620259. The authors thank the editor, an associate editor, and four referees for their helpful comments and suggestions.

[Received June 2007. Revised June 2008.]

REFERENCES


