Generating (n,2) De Bruijn Sequences with Some Balance and Uniformity Properties

Yi-Chih Hsieh[1], Han-Suk Sohn[2], and Dennis L. Bricker[2]

[1]Department of Industrial Management, National Huwei Institute of Technology, Huwei, Yunlin 632, Taiwan
[2]Department of Industrial Engineering, The University of Iowa, Iowa City, IA 52242, USA

(August, 2001)

Abstract

This paper presents two new algorithms for generating (n,2) de Bruijn sequences which possess certain properties. The sequences generated by the proposed algorithms may be useful for experimenters to systematically investigate intertrial repetition effects. Characteristics are compared with those of randomly sampled (n,2) de Bruijn sequences.

Keywords: De Bruijn sequence; Algorithms; Balance criterion; Uniformity criterion

---

Correspondence to : *Dr. Yi-Chih Hsieh, Dept. of Industrial Management, National Huwei Institute of Technology, Huwei, Yunlin 632, Taiwan; yhsieh@sparc.nhit.edu.tw, Fax: +886-5-6311548.*

1. Introduction

Assume n symbols which, without loss of generality, we denote by 1, 2,..., n-1, n, with the natural order $1<2< \cdots <n-1<n$. A n-symbol m-tuple de Bruijn sequence (or (n,m) de Bruijn sequence), is a string of n^m symbols $s_1 s_2 \ldots s_{n^m}$ such that each substring of length m,

$$s_{i+1} s_{i+2} \ldots s_{i+m}, \tag{1}$$

is unique with subscripts in (1) taken modulo n^m. For $m \geq 2$ and $n \geq 2$, there are $N = \left[(n-1)!\right]^{n^{m-1}} \cdot n^{n^{m-1}-m}$ (n,m) de Bruijn sequences (Fredricksen [3]). For example, there are 36 pairs (i.e., m=2) which may be formed using n=6 symbols. The number of (6,2) de Bruijn sequences is over 3.87×10^{15}.

During the 70s-80s, de Bruijn sequences were well studied and several algorithms have been proposed for generating such sequences, e.g., Fredricksen and Kessler [4], Fredricksen and Maiorana [5], and Ralston [10]. Most of those proposed algorithms use concepts of either finite field theory or combinatorial theory to generate a single (n,m) de Bruijn sequence, but in addition, there is a well-known algorithm to sample, with equal probability, "random" (n,m) de Bruijn sequences [1]. An excellent survey has been provided by Fredricksen [3].

Owing to the special properties of de Bruijn sequences, there are various recent applications of (n,m) de Bruijn sequences, such as the planning of reaction time experiments (Emerson and Tobias [2] and Sohn et al. [11]) and 3-D pattern recognition (Griffin et al. [6], Hsieh [7-9], and Yee and Griffin [12]). Hsieh [8] describes some of these recent applications.

In the reaction time experiment problems in which n stimuli are used and the effect of the preceding stimulus is considered, the (n,2) de Bruijn sequences represent the order of the stimuli (Emerson and Tobias [2], Sohn et al. [11]). The subject selects and executes a response depending upon the identity of each stimulus and his or her reaction times (RTs) are recorded and analyzed. Such a (n,2) de Bruijn sequence should have the properties that (i) each stimulus appears equally often and (ii) is preceded equally often by itself and by the other stimuli. Sohn et al. [11]

presented sequences of trials that exhibit two characteristics which are intended to balance out practice effects and/or intertrial repetition effects in experiments.

The present paper, after reviewing the two criteria of Sohn et al., proposes two new and efficient algorithms to generate (n,2) de Bruijn sequences which are judged favorably by these criteria.

2. Two Criteria for (n,2) de Bruijn Sequences

Sohn et al. [11] defined two criteria for measuring the quality of various (n,2) de Bruijn sequences, namely, *balance* and *uniformity.* The balance criterion measures the extent to which the average positions of the stimuli differ; balance aims to avoid the influence of practice effects during the blocks of trials. The uniformity criterion measures the interval between appearances of each stimulus condition in the sequence; uniformity aims to avoid intertrial repetition effects. For a given (n,2) de Bruijn sequence s, let $P(i,\bullet)$ be the sum of positions of component i and $I(i,j)$ be the interval between the j^{th} presentation of a symbol i and the $(j+1)^{\text{st}}$ presentation of that same i. We also define the *max norm* as $\|u\|_\infty = \max\{\ |u_1|,|u_2|,\ldots,|u_n|\ \}$ for $u\in R^n$, and $\|u\|_\infty = \max_{i,j}\{\ |u_{ij}|\ \}$ for $u\in R^{m\times n}$.

Definition 1. The *balance* of a (n,2) de Bruijn sequence s is

$$t(s) \equiv \left\| P(i,\cdot) - n(n^2+1)/2 \right\|_\infty .$$

(Note that $n(n^2+1)/2$ is the average of the sum of the positions for the components in sequence s)

Definition 2. Sequence s is said to be *more balanced* than sequence s' if $t(s)<t(s')$, and sequence s is said to be *perfectly balanced* if $t(s)=0$.

Remark. Perfectly balanced sequences do not always exist, e.g., the minimum balance of the (3,2) de Bruijn sequences *is 2* (Sohn et al. [11]).

Definition 3. The *uniformity* of a (n,2) de Bruijn sequence s is

$$s\ (s) \equiv \|I - n\|_{\infty} .$$

Example 4. The balance and uniformity of the (3,2) de Bruijn sequence s=1-2-3-1-3-3-2-2-1 are

$t(s)$=max{|1+4+9-15|, |2+7+8-15|, |3+5+6-15|}=2, and

$s\ (s) = \underset{ij}{\mathrm{Max}}\{|I(i,j) - n|\}$ =Max{|3-3|,|5-3|,|5-3|,|1-3|,|2-3|,|1-3|}=2, respectively.

3. New Algorithms

This paper is concerned with the generation of (n,2) de Bruijn sequences with desirable balance and uniformity. We will define a sequence by constructing a square matrix A, where $A_{ij} = k \Leftrightarrow$ substring (i,j) is in the k^{th} position of the sequence, i.e., $s_k = i$ and $s_{k+1} = j$.

Algorithm I:
Step 0. $k \leftarrow 2$, $r \leftarrow 2$, $i \leftarrow 0$, $A=[0_{ij}] \in R^{n \times n}$, $A[1,1] \leftarrow 1$
Step 1. While ($i+1 \leq n$) do
begin
$i \leftarrow i+1$, $j \leftarrow i+1$
while ($i+j \leq 2n$) do
begin
if $j=n$, then
$A[i,j] \leftarrow k$, $A[j,r] \leftarrow k+1$, $A[r,r] \leftarrow k+2$, $k \leftarrow k+3$, $r \leftarrow r+1$
else $A[i,j] \leftarrow k$, $A[j,i] \leftarrow k+1$, $k \leftarrow k+2$, $j \leftarrow j+1$
end (while)
end (while)
$A[i,j\text{-}1] \leftarrow n^2\text{-}1$, $A[n,1] \leftarrow n^2$

Example 5. For n=5, Algorithm I generates the matrix A:

$$
\begin{array}{c c}
 & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{bmatrix} 1 & 2 & 4 & 6 & 8 \\ 3 & 10 & 11 & 13 & 15 \\ 5 & 12 & 17 & 18 & 20 \\ 7 & 14 & 19 & 22 & 23 \\ 25 & 9 & 16 & 21 & 24 \end{bmatrix}
\end{array}
$$

Thus the (5,2) de Bruijn sequence is given by 1-1-2-1-3-1-4-1-5-2-2-3-2-4-2-5-3-3-4-3-5-4-4-5-5 (i.e., the first subsequence is 1-1, and the second subsequence is 1-2 etc.) with $t(s)$=44 and $s(s)$=4. It is clear that matrix *A* is used to represent the order of subsequences appearing in the (*n*,2) de Bruijn sequence.

Theorem 6. *For any n³3, Algorithm I generates a* (*n*,2) *de Bruijn sequence s. Moreover, the uniformity of this sequence is* $s(s)$=*n*–1.

Proof. Generating a (*n*,2) de Bruijn sequence is equivalent to assigning each of the numbers 1,2,..., n^2 to the elements of matrix *A* in the Algorithm I, which is further equivalent to finding an Eulerian circuit in the corresponding de Bruijn digraph whose vertex set is {1,2,...,*n*} and edge set is {1,2,...,*n*}×{1,2,...,*n*} (Chartrand and Oellermann [1]). Since Algorithm I starts the Eulerian circuit from vertex one, it will stop at vertex one if the algorithm cannot travel further (because the indegree and outdegree are both equal to *n* for each vertex in the de Bruijn digraph). However, the last cell to be assigned is (*n*,1) with value n^2 by the algorithm, which further implies that the proposed algorithm can generate a (*n*,2) de Bruijn sequence. More specifically, Algorithm I assigns 1,2,..., n^2 to matrix *A* as shown in the Appendix. By examination of the matrix, it is straightforward to prove that the uniformity of the sequence is $s(s)$=*n*–1. €

Theorem 7. *For any n≥3, the minimum uniformity measure* $s(s)$ *for all* (*n*,2) *de Bruijn sequences s is* s*=*n*–1.

Proof. Firstly, we prove that, for any (*n*,2) de Bruijn sequence *s*, $s(s)$≥*n*–1. There are *n* diagonal cells of *A*, which implies that there is more than one row with the consecutive assignments of either (*A*[*i*,*i*]=*k* and *A*[*i*,*i*+*j*]=*k*+1) or (*A*[*i*,*i*]=*k* and *A*[*i*,*i*-*j*]=

k+1) for some j. This further implies that $I(i,k)$=1. Following Definition 3, we have $s(s) \geq n-1$. However, Theorem 6 states that Algorithm I generates a (n,2) de Bruijn sequence s with $s(s)=n-1$. This implies that the minimum value of s is $n-1$. €

Algorithm I generates a sequence with uniformity $s^*(s)=n-1$ for n³3; however, the balance $t(s)$ of the sequence is rather large. The second algorithm uses a particular Latin square L to assign priority to each subsequence (i,j), and is intended to generate a more balanced (n,2) de Bruijn sequence s, i.e, with lower $t(s)$.

Algorithm II.

Step 0. Construct the Latin square $L=[L_{ij}]$ as shown in Figure 1. $A=[0_{ij}]\in R^{n\times n}$. $k\leftarrow 1$, $i\leftarrow 1$.

Step 1. While (k£n^2-2n) do

begin

assign k to the cell of $A[i,j]$, where

$j = \underset{j}{\operatorname{argmin}} \{ L[i,\ j] \mid A[i,\ j\ ] = 0, L[i,\ j] \neq 0 \}$

$i\leftarrow j$, $k\leftarrow k+1$

end (while)

Step 2. $A[j,n]\leftarrow k$, $j\leftarrow n$, $k\leftarrow k+1$

While k£n^2 do

begin

$A[j,j]\leftarrow k$, $A[j,j\text{-}1]\leftarrow k+1$, $k\leftarrow k+2$, $j\leftarrow j\text{-}1$

end (while)

$i\backslash j$	1	2	3	…	n-2	n-1	n
1	0	1	2	…	n-3	n-2	n-1
2	n-1	0	1	…	n-4	n-3	n-2
3	n-2	n-1	0	…	n-5	n-4	n-3
:	:	:	:	…	:	:	:
n-2	3	4	5	…	0	1	2
n-1	2	3	4	…	n-1	0	1
n	1	2	3	…	n-2	n-1	0

Figure 1. The Latin square L.

Theorem 8. *Algorithm II generates a* (n,2) *de Bruijn sequence s for n*³3.

Proof. Firstly, we prove that the last assignment is either $A[3,1]=n^2-2n$ (n is odd) or $A[4,2]=n^2-2n$ (n is even) in Step 1. Let $G(v)\equiv\{[i,j] \mid L[i,j]=v,\ v=0,1,2,\ldots,n\text{-}1\}$, thus we have $|G(v)|=n$.

(i) When n is odd, we have $G(n\text{-}2)=\{[1,n\text{-}1],[2,n],[3,1],[4,2],\ldots,[n\text{-}1,n\text{-}3],[n,n\text{-}2]\}$. This implies that all the pairs in this set construct a circuit starting at [1,n-1] and ending at [3,1]. For each row of matrix A, the cell with smaller $L[i,j]$ (except zero) has higher priority to be assigned. This further implies that when the cells in $G(n\text{-}2)$ are assigned, all cells $A[i,j]$, where $[i,j]\in G(v)$, v=1,2,…, n-2, have been assigned. Since $\sum_{v=1}^{n-2}|G(v)| = n^2 - 2n$, thus we have $A[3,1]=n^2-2n$.

(ii) When n is even, we have $G(n\text{-}2)=\{[1,n\text{-}1],[2,n],[3,1],[4,2],\ldots,[n\text{-}1,n\text{-}3],[n,n\text{-}2]\}$. This implies that all the pairs in this set construct two circuits, one starting at [1,n-1] and ending at [3,1], and another starting at [2,n] and ending at [4,2]. Note that the second circuit travels [n, n-2]. Thus Step 1 of Algorithm II travels the first circuit in $G(n\text{-}2)$, follows the pair [1,n] in $G(n\text{-}1)$, then travels the second circuit in $G(n\text{-}2)$. Similarly, since for each row of matrix A, the cell with smaller $L[i,j]$ (except zero) has higher priority to be assigned. This further implies that when the cell [4,2] is assigned, all cells $A[i,j]$ (except for [2,n]), where $[i,j]\in G(v)$, v=1,2,…, n-2, have been assigned. Since $1+(|G(n-2)|-1)+\sum_{v=1}^{n-3}|G(v)| = n^2 - 2n$, thus we have $A[4,2]=n^2-2n$.

Therefore, in Step 2, we assign n^2-2n+1 to either cell $A[1,n]$ (n is odd) or cell $A[2,n]$ (n is odd), and n^2-$2n$ +2 to n^2 to the other shaded cells of Figure 1 in the order specified by Step 2. €

Example 9. For n=5, Algorithm II generates matrix A:

	1	2	3	4	5
1	25	1	6	11	16
2	24	23	2	9	13
3	15	22	21	3	7
4	10	12	20	19	4
5	5	8	14	18	17

Thus, the (5,2) de Bruijn sequence is 1-2-3-4-5-1-3-5-2-4-1-4-2-5-3-1-5-5-4-4-3-3-2-2-1, i.e., the first subsequence is (1,2), and the second subsequence is (2,3) etc., with balance and uniformity $t(s)$=6 and $s(s)$=5, respectively.

If we compare Example 5 with Example 9, it is clear that Algorithm II can reduce $t(s)$ significantly (from 44 to 6) with a slight increase of $s(s)$ (from 4 to 5). To further reduce the values of $t(s)$ and $s(s)$, we may employ a simple one-to-one mapping of the set $\{1,2,\ldots,n\}$ onto itself in the elements of its corresponding matrix A.

Example 10. Consider Example 9 again. If we apply the mapping

$$p_{20}(t) = \begin{cases} 6+t & if \quad t<20 \\ t-19 & if \quad 20 \le t \le 25 \end{cases} \qquad \text{for } 1 \le t \le 25$$

to the elements of matrix A, then we have

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 6 & 7 & 12 & 17 & 22 \\ 5 & 4 & 8 & 15 & 19 \\ 21 & 3 & 2 & 9 & 13 \\ 16 & 18 & 1 & 25 & 10 \\ 11 & 14 & 20 & 24 & 23 \end{bmatrix} \end{array}.$$

The new sequence has t=27 and s=4.

Based on this concept, in addition to the original sequence by Algorithm II, we may employ the other n^2-1 possible mappings of elements for 1,2,..., n^2 in the matrix A. More specifically, the i^{th} mapping, $2 \le i \le n^2$, is defined as:

$$p_i(t) = \begin{cases} (n^2 - i + 1) + t & \text{if } \ t<i \\ t - i + 1 & \text{if } \ i \le t \le n^2 \end{cases} \qquad \text{for } 1 \le t \le n^2 .$$

Among all the possible n^2 sequences obtained by this mapping of the sequence constructed by Algorithm II, the sequences with minimal values of s and t are both reported in Table 1.

4. Numerical Results

Table 1 compares the balance and uniformity (t,s) of $(n,2)$ de Bruijn sequences generated by Algorithms I and II with the “random” $(n,2)$ de Bruijn sequences generated by Emerson and Tobias [2] for 3≤n≤30. In the case of Algorithm II, the characteristics of the sequences obtained by the optimal mappings with respect to both balance and uniformity criteria are shown.

Table 1 goes here

In Table 1, we observe that:

1. Although Algorithm I generates sequences with optimal uniformity (minimum s), the corresponding balance criterion t is rather large.
2. The average values of (t,s) for the random (n,2) de Bruijn sequences generated by the computer program of Emerson and Tobias are relatively large for each n≥3. For example, when n=9, the mean values of (t,s) for these random sequences are (122.76, 31.35). Algorithm II generates a sequence which, by a mapping p described above, has characteristics (t,s) = (32,11) if the balance criterion s is optimized, and (t,s) = (88,10) if the uniformity criterion t is optimized. The approach of Sohn et al. yielded (t,s) = (21,32) and (50,8), when applying the balance and uniformity criteria, respectively.
3. The integer linear programming approach of Sohn et al., because it enumerates all feasible (n,2) de Bruijn sequences, is very time consuming and therefore not practical in general, e.g., for n≥10. Even when the Lagrangian relaxation methodology is used for the reduction of constraints in the original integer linear programming, the CPU time increases dramatically as the problem size increases. However, Algorithms I and II above are simple and efficient to implement.
4. Note that the sequences constructed by Algorithm II exhibit low values for both t and s simultaneously, especially for those sequences chosen to minimize t. For example, when n=9, (t,s)=(32,11) is obtained by Algorithm (II). On the other hand, the results of Sohn et al. show that (t,s)=(21,32) with respect to balance criterion and (t,s)=(50,8) with respect to uniformity criterion. It is clear that if both balance and uniformity are important in the design of the experiment, then

constructing a sequence using Algorithm II and selecting the best mapping with respect to the balance criterion might be a better alternative, since for balance t we have 32<50 and for uniformity s we have 11<32.

5. Conclusions

In this paper:

1. we have presented two new algorithms for generating (n,2) de Bruijn sequences with desirable balance and uniformity characteristics. As shown, both algorithms can be easily implemented for large values of n.
2. we have reported numerical results for $n \leq 30$ and compared the balance and unifority of the generated sequences with randomly sampled (n,2) de Bruijn sequences. The results show that the sequences generated by the new algorithms possess very good characteristics for both criteria simultaneously.

The sequences generated by the proposed algorithms might be useful for experimenters who wish to systematically investigate intertrial repetition effects. If one wishes to construct a longer sequence by repeating stimulus conditions, e.g., doubling or tripling the length of sequences, the strategies of Sohn et al. [11] can be employed to extend the sequences obtained by the new approach. Moreover, following Hsieh's [8] approach, one can generate a class of (n,2) de Bruijn sequences based upon the (seed) sequence generated by either one of the new algorithms.

Algorithm I generates a sequence with optimal value of the uniformity criterion s. It would be interesting to find optimal values or upper and lower bounds for the balance criterion t. We offer the following conjecture:

Conjecture. For every $n \geq 4$, there exists a perfectly balanced (n,2) de Bruijn sequence s, i.e., a sequence s for which $t(s)=0$.

Acknowledgements. This research was partially supported by National Science Council, Taiwan, under grant No. NSC 89-2213-E-150-023.

REFERENCES

[1] G. Chartrand and O.R. Oellermann, *Applied and Algorithmic Graph Theory,* McGraw-Hill, Singapore, 1993.

[2] P.L. Emerson and R.D. Tobias, Computer program for quasi-random stimulus sequences with equal transition frequencies, *Behavior Research Methods, Instruments, & Computers*, **27** (1995), 88-98.

[3] H. Fredricksen, A survey of full length nonlinear shift register cycle algorithms, *SIAM Review* **24** (1982), 195-221.

[4] H. Fredricksen and I. Kessler, Lexicographic compositions and de Bruijn sequences, *Journal of Combinatorial Theory* **2** (1981), 63-76.

[5] H. Fredricksen and J. Maiorana, Necklaces of beads in *k* colors and *k*-ary de Bruijn sequences, *Discrete Mathematics* **23** (1978), 207-210.

[6] P.M. Griffin, L.S. Narasimhan, and S.R. Yee, Generation of uniquely encoded light patterns for range data acquisition, *Pattern Recognition* **25** (1992), 609-616.

[7] Y.C. Hsieh, A note on the structured light patterns for three-dimensional imaging systems, *Pattern Recognition Letters*, **19** (1998), 315-318.

[8] Y.C. Hsieh, De Bruijn Sequences.Generation, Reproduction and Applications, *ARS Combinatoria*, (2000), (Accepted, to appear).

[9] Y.C. Hsieh, Decoding structured light patterns for three-dimensional imaging systems, *Pattern Recognition*, **34** (2001), 343-349.

[10] A. Ralston, A new memoryless algorithm for de Bruijn sequences, *Journal of Algorithms* **2** (1981), 50-62.

[11] H.S. Sohn, D.L. Bricker, J.R. Simon and Y.C. Hsieh, Optimal sequences of trials for balancing practice and repetition effects, *Behavior Research Methods, Instruments, & Computers*, **29** (1997), 574-581.

[12] S.R. Yee, and P.M. Griffin, Three-dimensional imaging system, *Optical Engineering* **33** (1994), 2070-2075.

Table 1. Comparison of results for various approaches.

n	Emerson and Tobias [2] (1995)		Sohn et al. [11] (1997)		New Algorithms			
	t (M,SD)	s (M,SD)	w.r.t. t (t,s)	w.r.t s (t,s)	I (t,s)	II (t,s)	II w.r.t. t (t,s)	II w.r.t. s (t,s)
3	(4.98,2.26)	(1.94,0.44)	(2*,2*)	(4,2*)	(8,2*)	(2*,2*)	(2*,2*)	(2*,2*)
4	(11.88,4.52)	(4.10,1.29)	(0*,4)	(5,3*)	(21,3*)	(6,5)	(6,5)	(13,3*)
5	(23.04,7.32)	(7.19,1.94)	(0*,9)	(20,4*)	(44,4*)	(6,5)	(6,5)	(24,4*)
6	(38.14,10.81)	(11.60,2.84)	(0*,10)	(20,5*)	(80,5*)	(20,8)	(12,8)	(33,5*)
7	(59.01,14.80)	(16.43,3.64)	(1,13)	(44,6*)	(132,6*)	(15,8)	(15,8)	(49,6*)
8	(88.35,18.90)	(23.43,4.46)	(15,21)	(44,7*)	(203,7*)	(35,11)	(21,11)	(67,9)
9	(122.76,23.44)	(31.35,5.28)	(21,32)	(50,8*)	(296,8*)	(40,11)	(32,11)	(88,10)
10	(163.94,28.64)	(40.69,6.51)	N/A	N/A	(414,9*)	(64,14)	(27,14)	(110,13)
11	(209.52,33.52)	(52.96,7.34)	N/A	N/A	(560,10*)	(45,14)	(45,14)	(123,10*)
12	(271.53,38.49)	(65.84,8.16)	N/A	N/A	(737,11*)	(111,21)	(28,21)	(28,21)
13	(344.46,44.54)	(76.51,9.34)	N/A	N/A	(948,12*)	(66,17)	(66,17)	(172,12*)
14	(410.51,49.90)	(94.35,10.34)	N/A	N/A	(1196,13*)	(132,20)	(50,20)	(217,17)
15	(500.37,56.58)	(117.45,11.72)	N/A	N/A	(1484,14*)	(155,22)	(55,22)	(55,22)
16	(579.23,62.67)	(133.65,11.87)	N/A	N/A	(1815,15*)	(175,23)	(65,23)	(65,23)
17	(702.03,69.03)	(148.53,13.76)	N/A	N/A	(2192,16*)	(120,23)	(120,23)	(294,16*)
18	(814.57,75.19)	(173.59,14.39)	N/A	N/A	(2618,17*)	(260,27)	(83,27)	(83,27)
19	(939.09,81.94)	(192.32,16.25)	N/A	N/A	(3096,18*)	(153,26)	(153,26)	(367,18*)
20	(1078.62,89.15)	(224.18,17.04)	N/A	N/A	(3629,19*)	(321,37)	(90,37)	(90,37)
21	(1241.96,95.22)	(253.75,17.74)	N/A	N/A	(4420,20*)	(310,32)	(110,32)	(110,32)
22	(1396.91,102.54)	(276.34,18.91)	N/A	N/A	(4872,21*)	(340,32)	(122,32)	(539,29)
23	(1586.52,109.52)	(320.10,19.52)	N/A	N/A	(5588,22*)	(231,32)	(231,32)	(537,22*)
24	(1781.80,117.16)	(351.72,20.85)	N/A	N/A	(6371,23*)	(505,45)	(135,45)	(135,45)
25	(1995.04,123.47)	(381.81,22.24)	N/A	N/A	(7224,24*)	(356,37)	(244,37)	(244,37)
26	(2189.07,123.47)	(410.06,23.83)	N/A	N/A	(8150,25*)	(480,41)	(170,41)	(170,41)
27	(2439.65,138.25)	(482.74,23.83)	N/A	N/A	(9152,26*)	(481,43)	(221,43)	(221,43)
28	(2684.85,147.00)	(502.71,26.34)	N/A	N/A	(10233,27*)	(635,53)	(173,53)	(173,53)
29	(2969.65,155.80)	(583.63,26.55)	N/A	N/A	(11396,28*)	(378,41)	(378,41)	(852,28*)
30	(3227.50,160.60)	(580.44,28.06)	N/A	N/A	(12644,29*)	(836,57)	(225,57)	(225,57)

*optimal.
N/A: not available by integer programming approach due to CPU time limit 86,400 seconds.
w.r.t.: with respect to.
t (M,SD): mean and standard deviation of t (based upon 1000 random sequences).
s (M,SD): mean and standard deviation of s (based upon 1000 random sequences).

Appendix

Matrix A constructed by Algorithm I.

	1	2	3	4	…	i-1	i	i+1	…	n-2	n-1	n
1	1	2	4	6	…	$2(i-2)$	$2(i-1)$	$2i$	…	$2(n-3)$	$2(n-2)$	$2(n-1)$
2	3	$2n$	$2n+1$	$2n+3$	…	$2n+2(i-4)+1$	$2n+2(i-3)+1$	$2n+2(i-2)+1$	…	$4n-9$	$4n-7$	$4n-5$
3	5	$2n+2$	$4n-3$	$4n-2$	…	$4n+2(i-6)$	$4n+2(i-5)$	$4n+2(i-4)$	…	$6n-14$	$6n-12$	$6n-10$
4	7	$2n+4$	$4n-1$	$6n-8$	…	$6n-8+2(i-5)-1$	$6n-8+2(i-4)-1$	$6n-8+2(i-3)-1$	…	$8n-21$	$8n-19$	$8n-17$
:	:	:	:	:	…	:	:	:	…	:	:	:
i-1	$2(i-1)-1$	$2n+2(i-3)$	$4n+2(i-6)+1$	$6n-8+2(i-5)$	…	$2(i-2)n-(i-2)^2+1$	$2(i-2)n-(i-2)^2+2$	$2(i-2)n-(i-2)^2+4$	…	$2(i-1)n-(i-1)^2-5$	$2(i-1)n-(i-1)^2-3$	$2(i-1)n-(i-1)^2-1$
i	$2i-1$	$2n+2(i-2)$	$4n+2(i-5)+1$	$6n-8+2(i-4)$	…	$2(i-2)n-(i-2)^2+3$	$2(i-2)n-(i-1)^2+1$	$2(i-2)n-(i-1)^2+2$	…	$2in-i^2-5$	$2in-i^2-3$	$2in-i^2-1$
i+1	$2(i+1)-1$	$2n+2(i-1)$	$4n+2(i-4)+1$	$6n-8+2(i-3)$	…	$2(i-2)n-(i-2)^2+5$	$2(i-2)n-(i-1)^2+3$	$2in-i^2+1$	…	$2(i+1)n-(i+1)^2-5$	$2(i+1)n-(i+1)^2-3$	$2(i+1)n-(i+1)^2-1$
:	:	:	:	:	…	:	:	:	…	:	:	:
n-2	$2(n-2)-1$	$4n-8$	$6n-13$	$8n-20$	…	$2(i-1)n-(i-1)^2-4$	$2in-i^2-4$	$2(i+1)n-(i+1)^2-4$	…	n^2-8	n^2-7	n^2-5
n-1	$2(n-1)-1$	$4n-6$	$6n-11$	$8n-18$	…	$2(i-1)n-(i-1)^2-2$	$2in-i^2-2$	$2(i+1)n-(i+1)^2-2$	…	n^2-6	n^2-4	n^2-2
n	n^2	$2n-1$	$4n-4$	$6n-9$	…	$2(i-2)n-(i-2)^2$	$2(i-1)n-(i-1)^2$	$2in-i^2$	…	n^2-9	n^2-4	n^2-1