Constructing Regular 2^{k-p} Designs by Bounding Length of Alias Chains

by

Robert W. Mee
University of Tennessee, Knoxville, TN 37996-0532

and

Robert M. Block
U.S. Air Force, Colorado Springs, CO 80914

Abstract

Minimum aberration and maximum number of clear two-factor interactions are two criteria for ranking resolution IV fractional factorial designs. A third criterion we recommend when many two-factor interactions are expected to be important is to minimize the maximum length of two-factor-interaction alias chains. Since minimum aberration resolution IV designs tend to have uniform length alias chains for two-factor interactions (Cheng, Steinberg and Sun 1999), this criterion would be expected to resemble aberration-based criteria. For designs of size 32 to 128, we explore whether these criteria coincide or not. For larger design situations where no complete enumeration of resolution IV designs exist, we use the notion of limiting the maximum length of alias chains as a basis for design construction. We tabulate results for n = 256 and n = 512, furnishing resolution IV designs that avoid long alias chains. When estimation of many two-factor interactions is intended, such designs reduce the number of follow-up runs required to de-alias two-factor interactions.

Key words and phrases: Alias length pattern, minimum aberration, resolution IV.

01/25/06

1. Introduction

Resolution IV designs are a compromise between resolution V designs, which are often too costly, and resolution III designs, which provide too little information to estimate main effects clear of two-factor interactions. Chen, Sun, and Wu (1993) constructed all regular 2_{IV}^{k-p} designs of size 16, 32, and 64; for n = 128, they were only able to enumerate all non-isomorphic designs up to k = 11 factors (Sun 2001). For n = 128, see Block (2003) and Block and Mee (2005) for an attempt at complete enumeration of resolution IV regular fractions. For the most extensive listings of minimum aberration designs in texts, see Wu and Hamada (2000, Appendix 4A) and Wu and Mukerjee (in press). For larger designs, except for special cases, one must resort to software to search for good designs. For example, SAS's (2002) Proc Factex will search for the best design for any specified k and $n = 2^{k-p}$.

Criteria for ranking resolution IV designs are based on either the word length pattern or the alias length pattern (Block and Mee 2003). Minimum aberration is based on sequentially minimizing elements of the word length pattern, wlp = (w_4, w_5, …), where w_j denotes the number of length j words in the defining contrast subgroup. Though popular, the minimum aberration criterion is not as intuitive to practitioners as are other criteria for ranking resolution IV designs. Several more intuitive criteria are all functions of the alias length pattern alp = (a_1, a_2, …, a_L), where a_1 is the number of clear two-factor interactions, a_2 = the number of pairs of aliased two-factor interactions, L is the length of the longest chain of aliased two-factor interactions, and a_L is the number of such chains. Four intuitive criteria based on alp are:

- Number of clear two-factor interactions, a_1 (Chen, Sun and Wu 1993)
- Maximum number of two-factor interactions that can be estimated in a single model, $M = a_1 + a_2 + \ldots + a_L$

- Estimation capacity sequence $(E_1^{'}, E_2^{'}, ..., E_M^{'})$ (Cheng, Steinberg, and Sun 1999), where $E_j^{'}$ is the proportion of models containing the *k* main effects and *j* two-factor interactions that can be estimated from the design
- The length of the longest chain of aliased two-factor interactions, *L* (Block 2003)

Note that L = 1 designs have resolution V or higher. For $L \geq 2$, minimizing *L* limits the maximum confusion caused by aliasing among two-factor interactions. *L* is a lower bound on the number of regular fractions from the same family that must be assembled to permit estimation of all two-factor interactions (Addelman 1969, Mee 2004). If augmenting an initial fraction with any set of follow-up runs, the smaller *L* is, the smaller the correlations will be among the resulting estimators after augmentation.

The next section will explore the coincidence, or lack thereof, between minimum aberration designs and minimum *L* designs for resolution IV designs of size 32, 64, and 128. For n = 256 and larger, we restrict *L* as a useful means for constructing good resolution IV designs. The connection between minimum aberration and uniform length alias chains does not appear to have been exploited previously in the search for good designs. We provide both an algorithm, implemented in MATLAB, and a table of the best minimum *L* designs we have found for *n* = 256 and 512. We conclude with an example involving 47 factors in a missile defense computer simulation study, were Proc Factex was used to generate an initial 2_{IV}^{47-38} design. The results of this paper make it possible to quickly find an alternative 2_{IV}^{47-38} with smaller *L* and w_4. Using the better initial design would have greatly simplified the follow-up work to separate aliased two-factor interactions.

2. Minimum Aberration and Minimum L Designs for Known Cases

Since several resolution IV designs may have the same minimum L, we now consider how a_1, a_L, M, and w_4 are interrelated for designs with the same L. For $L = 2$, these criteria coincide. That is, since in this case $a_2 = 3w_4 = \binom{k}{2} - M = \left[\binom{k}{2} - a_1\right]/2$, decreasing a_2 (or w_4) necessarily increases a_1 and M. For $L > 2$, the criteria do not coincide. At $L = 3$, if w_4 is fixed, decreasing a_3 by 1 increases a_1 by 3 but decreases M by 1. Thus, for various designs with $L = 3$ and the same number of length four words, one must choose between minimizing the number of length-three alias chains and maximizing the number of degrees of freedom for two-factor interactions. Which is preferred is not obvious. For L > 3, we cannot be as precise about the relationships among a_1, a_L, M, and w_4, but the required tradeoff between improving M versus improving a_L for fixed w_4 continues to arise.

Define a 2_{IV}^{k-p} design D* with alp = $(a_1^*, a_2^*, ..., a_{L^*}^*)$ to be L-optimal if:

- There does not exist a 2_{IV}^{k-p} design with $a_j = 0$ for all $j \geq L^*$, and
- For any resolution IV designs with alp = (a_1, a_2, …, a_{L^*}), $a_j^* < a_j$, where j is the largest integer such that $a_j^* \neq a_j$.

Any 2_{IV}^{k-p} design with $L = L^*$ will be called a minimum L design. All resolution IV designs with alp = $(a_1^*, a_2^*, ..., a_{L^*}^*)$ will be considered L-optimal. Note that two L-optimal designs necessarily have the same number of length four words, but may differ for the remainder of the wlp. L-optimality focuses on minimizing the number of long alias chains. If instead the priority is estimation capacity, then we want to maximize M and/or minimize w_4, subject to $L = L^*$, instead of minimizing a_L.

Given the varied criteria that may be relevant, Table 1 lists minimum L resolution IV designs for n of 32, 64, and 128, which achieve an optimum value on one or more of w_4, M, a_1, a_L. For $n \leq 64$, designs are identified by the numbering in Chen, Sun and Wu (1993). For n = 128, results are based on Block's (2003) enumeration of designs, where a complete isomorphism check was not performed; thus, while these n = 128 designs are the best regular designs known to date, they are not yet guaranteed to be optimal. Since the usefulness of the minimum-L criterion seems greatest for small to moderate L, Table 1 is restricted to cases where $L \leq 6$. Note that, for a given n, L is a non-decreasing function of k. This is not the case for minimum aberration designs; for instance, design 23-16.1 has $L = 7$, while design 24-17.1 has $L = 6$.

For the cases in Table 1, 61% of the minimum aberration designs have minimum L; in 73% of the cases, some minimum L design is at least weak minimum aberration. Especially for run size n = 128, there is close correspondence between L-optimal designs (with $L \leq 6$) and minimum aberration designs. However, the various criteria can differ dramatically for some cases. For instance, consider 2_{IV}^{17-11} designs that are optimal based on various criteria:

- Design 17-11.1 is minimum aberration, with $w_4 = 59$; however $L = 7$ and $M = 43$.
- Design 17-11.6 has the maximum number of clear two-factor interactions, with $a_1 = 31$ and maximum $M = 46$. However, $L = 7$ and $w_4 = 105$ make it less attractive.
- Design 17-11.2 is the only resolution IV design with $L = 4$. While it has $w_4 = 60$, one more than the minimum aberration design, it has the maximum M (with all its degrees of freedom used to estimate main effects and two-factor interactions) and it has the third most clear two-factor interactions, with $a_1 = 16$.

3. Minimum L Designs for $n \geq 256$

For cases where no complete enumeration of resolution IV designs exists, one may search for minimum L designs using a build-up method that randomly adds generators, and excludes them only if they would either lower the resolution or would add another two-factor interaction to an alias set that already contains the maximum number L. Regular resolution IV designs of size n with a specified L may be obtained as follows.

1. Identify the b basic columns for a full $2^b (= n)$ factorial as 1, 2, 4, …, 2^{b-1}. Set $s_j = 1$ for each of these columns, and $s_j = 0$ for all other columns ($j < n$).
2. Identify the column number for each of the $\binom{b}{2}$ two factor interactions of the basic columns. Set t_j =1 for each of these columns, and t_j =0 for all other columns
3. Create a random permutation of the n-1 column numbers $\boldsymbol{c} = \{1, 2, …, \text{n-1}\}$ and set $k = b$ and $r = 1$.
4. Consider adding the column c_r as a generator, where c_r is the r^{th} element of $\boldsymbol{c}$.
5. If $s_{c_r} + t_{c_r} = 0$, continue to step 6, since this column can be used as an additional generator without lowering the resolution. If $s_{c_r} + t_{c_r} > 0$, c_r is already occupied by a factor or one or more two-factor interactions; in this case, set $r = r + 1$. If $r < n$, return to step 4; if $r = n$, go to step 9.
6. Identify the columns containing the interactions of c_r and each of the k factors in the model. If any of these two-factor interactions fall on a column j where $t_j = L$, then adding c_r as a generator for a new factor would violate the upper bound L on the

length of one or more alias chains. In this case, set $r = r + 1$ and return to step 4 (or, if $r = n$, go to step 9). Otherwise, go to step 7.

7. Set $t_j = t_j + 1$ for each the k columns containing a two factor interaction involving the first k factors and c_r. Then set $s_{c_r} = 1$ and $k = k + 1$.
8. If k now equals the intended number of factors, then one has a candidate design satisfying the specified upper bound L, so stop. If not, set $r = r + 1$ and return to step 4 (or, if $r = n$, go to step 9).
9. If $r = n$ before we reach the desired number of factors, then it is not possible to achieve desired design with the current factors and value of L.

Whether one achieves the desired number of factors or not, one may repeat this process with a new permutation to see if one achieves a better design in terms of a_L (or w_4). If after many repeated tries, the desired number of factors is never achieved, one must increase L. For a Matlab implementation of such an algorithm, see the Appendix. A lower bound on L may be obtained by simply dividing the number of two-factor interactions by an upper bound on the number of degrees of freedom available for estimating two factor interactions, and then rounding up to the next integer. In particular,

$$L^* \geq \begin{cases} \left\lceil \dfrac{k(k-1)}{2(n-k-1)} \right\rceil & k \leq 5n/16 \\ \left\lceil \dfrac{k(k-1)}{n-2} \right\rceil & k > 5n/16 \end{cases}.$$

Since no designs of run size $n = 128$ from Table 1 attain this lower bound, one would not expect to attain this bound with our search algorithm for larger n (and small to moderate L).

Tables 2 and 3 present results from our search for run sizes $n = 256$ and 512, and $L = 2$, 3, 4 and 5. We list minimum L designs with small w_4, as well as a few designs with smaller w_4

but some chains longer than the minimum L. When we recognize that a listed design may be obtained by augmenting a design with smaller k, we indicate this in the list of generators.

4. Example with 47 Factors

Mitchell et al. (2000) report using an $n = 512$, $k = 47$ factor resolution IV design produced by SAS's (2002) Proc Factex to investigate a missile defense computer model. Following this 2_{IV}^{47-38} design, 17 smaller follow-up designs totaling 352 runs were used to de-alias the two-factor interactions considered of interest. This laborious task led the research team subsequently to use a resolution V design when exploring a second scenario, even though the resolution V design used required 4096 runs.

Although we do not have access to the initial 2_{IV}^{47-38} design used, if one requests such a design from SAS's Proc Factex using the minimum aberration criterion and a 20 minute search, a typical outcome has $L = 19$ and $w_4 = 1601$. No wonder estimating individual interactions via follow-up designs was problematic for Mitchell et al (2000). (Running Proc Factex for two evenings we found a much better design with $L = 7$ and $w_4 = 342$.) Using our algorithm, in less than 10 minutes one can obtain designs with $L = 5$ and $w_4 < 360$. Our best design, found by carefully augmenting a minimum L design with fewer factors, has alp = (96, 167, 136, 52, 7) and $w_4 = 319$ (refer to Table 3). With $k = 47$ and $M = 458$, all but 6 degrees of freedom are used for main effects and two-factor interactions.

5. Conclusion

We have defined minimum L resolution IV designs, and provided tables of these designs for run sizes up to 128. For $n = 256$ and 512, we have provided tables of the best designs known to date for $L \leq 5$. The minimum-L criterion makes the most sense for small L where there is some expectation of needing to estimate many two-factor interactions. For large n, enumerating all the possible resolution IV designs is infeasible. In such cases, the algorithm we have proposed is not only a help for constructing designs with minimal L, but also in finding designs with low aberration, regardless of the value for L.

Appendix: MATLAB code for Random Search of Minimum *L* Designs

The following MATLAB code may be used to search for an L-optimal design with specified k, L and n. In this algorithm, columns are identified by their Yates column number. Basic columns are numbered 1, 2, 4, 8, …; every other column is a product of the basic columns, and its number is defined by the sum of basic column numbers. For example, the product of the first, second, and fourth basic factors is column 11, since 1 + 2 + 8 = 11.

```
% This program computes minimum L designs of size 512
b=9
n=2^b
nminus1=n-1
rand('state',sum(100*clock))
% Specify values for longest alias chain L and number of factors K (<=60)
L=5
K= 47
Lplus1=L+1;
% Specify the number of iterations for the search itmax
itmax= 10000
amin=100
for it=1:itmax
G=[1 2 4 8 16 32 64 128 256];
G=[G zeros(1,51)];
C=zeros(60,b);
A=zeros(1,nminus1);
for i=1:b-1
   for j = i+1:b
      A(1,G(i)+G(j))=1;
   end
end
E=A;
```

```
for i=1:b
   C(i,i)=1;
   E(G(i))=1;
end
k=b;
  V=randperm(nminus1);
% Begin adding factors
for i=1:nminus1
   if E(V(i))==0
   new=V(i);
   bnew=zeros(1,b);
   rnew=new;
   while rnew>=256
      bnew(1,9)=1;
      rnew=rnew-256;
   end
   while rnew>=128
      bnew(1,8)=1;
      rnew=rnew-128;
   end
    while rnew>=64
      bnew(1,7)=1;
      rnew=rnew-64;
   end
   while rnew>=32
      bnew(1,6)=1;
      rnew=rnew-32;
   end
     while rnew>=16
      bnew(1,5)=1;
      rnew=rnew-16;
   end
     while rnew>=8
      bnew(1,4)=1;
      rnew=rnew-8;
   end
     while rnew>=4
      bnew(1,3)=1;
      rnew=rnew-4;
   end
     while rnew>=2
      bnew(1,2)=1;
      rnew=rnew-2;
   end
   bnew(1,1)=rnew;
   for j=1:k
```

```
            cold=C(j,:);
        ij=abs(cold-bnew);
newint(j)=ij(1,1)+2*ij(1,2)+4*ij(1,3)+8*ij(1,4)+16*ij(1,5)+32*ij(1,6)+64*ij(1,7)+128*ij(1,8)+
256*ij(1,9);
    end
    big=0;
    for j=1:k
     big=max(big,1+A(newint(j)));
 end
 if big<Lplus1
    k=k+1;
    G(k)=new;
    C(k,:)=bnew;
    E(new)=1;
    for j=1:k-1
        A(newint(j))=1+A(newint(j));
        E(newint(j))=1+E(newint(j));
    end
end
end
if k == K
    i=n;
end
end
if k == K
    alp=zeros(1,L);
    M=0;
    for i=1:nminus1
        a=A(i);
        if a > 0
            M=M+1;
            alp(a)=alp(a)+1;
        end
    end
    if alp(L) < amin
    it
    G(1:k)
    alp
    M
    amin=alp(L);
    w4=trace(A'*A)/6-(k-1)*k/12
    end
end
end
```

Whenever the algorithm finds a design that improves on the value of a_L, it prints out the iteration number, the columns numbers for the k factors, the alp, and w_4 for the new design. One could alter the code to check for improvement in w_4 if one preferred a minimum L design with the least aberration.

To create the listing of designs in Tables 2 and 3, we used a slight modification of this program, where we did not specify k, and allowed each search to add the maximum number of generators for the specified value of L. We then took naïve projections from the best of these designs, deleting the factor which reduced a_L (or w_4) the most, to find good designs with smaller k. We then consider augmenting these designs to find improved designs with larger k.

To search for designs of size 256, simply change line 2 to "b=8", drop 256 from the vector G, and remove the last term in the calculation of newint(j). To search for designs of size $n > 2^9$, one must make the following changes in addition to increasing the value of b:

- Add additional basic columns to the vector G
- Insert after line 35 additional loops such as

```
while rnew>=512
bnew(1,10)=1;
rnew=rnew-512;
end
```

- Add additional terms in the sum computing newint(j)
- If more than 60 factors are desired, increase the number of columns allocated to the matrices C and G.

References

Addelman, S., 1969. Sequences of two-level fractional factorial plans. *Technometrics* 11, 477-509.

Block, R.M., 2003. *Theory and Construction Methods for Large Resolution IV Designs.* Univ. of Tennessee Ph.D. dissertation: Knoxville [http://etd.utk.edu/2003/BlockRobert.pdf]

Block, R.M., Mee, R.W., 2003. Second order saturated resolution IV designs. *Journal of Statistical Theory and Applications* 2, 96-112.

Block, R.M., Mee, R.W., 2005. Resolution IV designs with 128 runs. *Journal of Quality Technology* 37, 282-293.

Chen, J., Sun, D.X., Wu, C.F.J., 1993. A catalogue of two-level and three-level fractional factorial designs with small runs. *International Statistical Review* 61, 131-145.

Cheng, C.W., Steinberg, D.M., Sun, D.X., 1999. Minimum aberration and model robustness for two-level fractional factorial designs. *Journal of the Royal Statistical Society, ser. B* 61, 85-93.

Mee, R.W., 2004. Efficient two-level designs for estimating all main effects and two-factor interactions. *Journal of Quality Technology* 36, 400-412.

Mitchell, B.L., Spriesterbach, T.P., Telford, J.K., West, R.L., 2000. The identification of ballistic missile defense system needs via experimental design, U. SECRET, SSD-POR-00-7285, Johns Hopkins University Applied Physics Laboratory, Baltimore, MD.

SAS Institute, Inc., 2002. *SAS/QC User's Guide*, Release 9.0. SAS Institute, Inc., Cary, NC.

Sun, D.X., 2001. Private communication. Catalog of n = 64, 128 two-level designs.

Wu, C.F.J., Hamada, M., 2000. *Experiments: Planning, Analysis, and Parameter Design Optimization*. John Wiley & Sons, New York.

Wu, C.F.J., Mukerjee, R., in press. *Theory of Factorial Design.* John Wiley & Sons, New York.

Table 1. Minimum L 2_{IV}^{k-p} Designs of size n = 32, 64, 128 with $L \leq 6$

n	Design	alp	w_4	Among Minimum L Designs				Minimum Aberration
				Min w_4	Max M	Max a_1	Min a_L	
32	7-2.1	15,3	1	Yes	Yes	Yes	Yes	Yes
	8-3.1	13,6,1	3	Yes	Yes	Yes	Yes	Yes
	9-4.2	15,0,7	7	Yes	Yes	Yes	Yes	$L = 4$, $w_4 = 6$
	10-5.2	0,0,15	15	Yes	Yes	-	Yes	$L = 5$, $w_4 = 10$
	11-6.1	0,0,5,10	25	Yes	Yes	-	Yes	Yes
	12-7.1	0,0,0,9,6	38	Yes	Yes	-	Yes	Yes
64	9-3.1	30,3	1	Yes	Yes	Yes	Yes	Yes
	10-4.1	33,6	2	Yes	Yes	Yes	Yes	Yes
	11-5.2	25,15	5	Yes	Yes	Yes	Yes	$L= 3$, $w_4 = 4$
	12-6.1	36,12,2	6	Yes	Yes	Yes	Yes	Yes
	13-7.2	36,0,14	14	Yes	Yes	Yes	Yes	$L= 4$, $w_4 = 14$
	14-8.7	13,15,12,3	23	Yes	Yes	No	Yes	$L = 5$, $w_4 = 22$
	14-8.10	16,9,15,3	24	No	Yes	Yes	Yes	
	15-9.3	14,6,17,7	33	Yes	Yes	Yes	Yes	$L = 5$, $w_4 = 30$
	16-10.2	15,0,15,15	45	Yes	Yes	Yes	Yes	$L = 6$, $w_4 = 43$
	17-11.2	16,0,0,30	60	Yes	Yes	Yes	Yes	$L = 7$, $w_4 = 59$
	18-12.4	0,0,1,3,24,3	102	Yes	Yes	-	Yes	$L = 8$, $w_4 = 78$
128	12-5.1	60,3	1	Yes	Yes	Yes	Yes	Yes
	13-6.1	66,6	2	Yes	Yes	Yes	Yes	Yes
	14-7.1	73,9	3	Yes	Yes	Yes	Yes	Yes
	15-8.1	63,21	7	Yes	Yes	Yes	Yes	Yes
	16-9.1	60,30	10	Yes	Yes	Yes	Yes	Yes
	17-10.1	46,45	15	Yes	Yes	Yes	Yes	Yes
	18-11.1	33,60	20	Yes	Yes	Yes	Yes	Yes
	19-12.1	36,54,9	27	Yes	No	No	Yes	Yes

	19-12.2	45,42,14	28	No	Yes	Yes	No	
	20-13.1	24,60,14,1	36	Yes	No	No	Yes	Yes
	20-13.2	41,39,21,2	38	No	Yes	No	No	
	20-13.20	46,27,26,3	41	No	No	Yes	No	
	21-14.2	28,51,12,11	51	Yes	No	No	No	$L = 5$, $w_4 = 51$
	21-14.6	36,36,22,9	52	No	Yes	Yes	Yes	
	22-15.3	21,52,12,15,2	66	Yes	No	No	Yes	$L = 6$, $w_4 = 65$
	22-15.9	32,30.28,10,3	68	No	Yes	Yes	No	
	23-16.2	14,54,11,17,6	83	Yes	Yes	No	No	$L = 7$, $w_4 = 83$
	23-16.24	24,28,24,19,5	88	No	No	Yes	Yes	
	24-17.2	7,57,9,17,12	102	Yes	Yes	No	Yes	$L = 6$, $w_4 = 102$
	24-17.13	16,48,0,26,12	108	No	No	Yes	Yes	
	25-18.1	0,64,0,18,20	124	Yes	Yes	-	Yes	Yes
	26-19.1	0,29,41,4,16,8	152	Yes	Yes	No	Yes	Yes
	27-20.1	0,15,55,0,12,16	180	Yes	Yes	No	Yes	Yes
	28-21.1	0,0,70,0,0,28	210	Yes	Yes	-	Yes	Yes

Table 2. Good Minimum *L* Designs of Size n = 256

L	*k*	alp	w_4	Generators
2	18	135, 9	3	39 57 81 95 107 168 179 190 198 201
	19	147, 12	4	Add 218
	20	160, 15	5	Add 244
	21	138, 36	12	31 39 90 111 118 123 150 152 169 179 188 203 245
	22	135, 48	16	Add 193
	23	127, 63	21	Add 221
	24	120, 78	26	Add 240
3	25	108, 90, 4	34	Add 19
	26	94, 102, 9	43	Add 57
	27	54, 135, 9	54	23 27 37 46 84 90 103 107 118 121 166 171 203 211 213 222 225 226 232
	28	63, 117, 27	66	Add 183
	29	73, 99, 45	78	Add 141
4	30	61, 102, 46, 8	96	Add 239
	31	49, 105, 46, 17	115	Add 156
	32	54, 66, 74, 22	140	26 29 35 41 63 76 102 113 116 119 122 137 143 145 158 167 180 198 202 217 223 225 236 242
	33	64, 36, 88, 32	164	7 29 51 53 56 76 79 87 91 101 107 113 142 145 155 161 191 195 210 216 230 239 245 249 252
	34	45, 72, 44, 60	188	21 43 45 50 77 78 83 103 106 108 113 116 123 137 142 154 175 182 198 203 208 213 217 223 232 243
5	33*	31, 89, 73, 20, 4	156	35 61 77 81 84 94 98 100 111 120 123 137 142 149 150 152 155 161 164 175 178 211 231 232 253
	35	34, 71, 53, 45, 16	220	15 19 35 46 59 87 89 92 97 107 109 112 117 137 138 159 182 184 189 197 203 206 209 210 236 247 251
	36	23, 71, 53, 34, 34	258	21 30 39 42 60 76 84 95 97 110 119 121 122 134 155 157 162 175 179 181 182 184 208 211 218 224 235 237

* While not a minimum *L* design, it is listed because it has smaller w_4.

Table 3. Good Minimum *L* Designs of Size n = 512

L	*k*	alp	w_4	Generators
2	24	264, 6	2	31 46 85 103 156 171 207 242 301 308 348 383 465 474 489
	25	276, 12	4	Add 420
	26	289, 18	6	Add 185
	27	291, 30	10	Add 90
	28	294, 42	14	Add 200
	29	264, 57	19	Add 407
	30	262, 72	24	Add 446
	31	273, 96	32	45 63 83 114 116 137 163 186 197 223 230 287 302 314 327 336 347 390 424 432 459 510
	32	262, 117	39	Add 194
	33	246, 141	47	Add 216
	34	153, 204	68	54 56 67 150 163 200 217 223 228 234 285 305 308 330 336 355 366 414 421 428 455 459 460 481 502
3	31*	294 81 3	30	Add 324 to k=30, L=2 design
	32*	286 99 4	37	Add 385
	33*	270 120 6	46	Add 402
	34*	264 132 11	55	Add 423
	35	250 150 15	65	Add 504
	36	234 165 22	77	Add 176
	37	219 171 35	92	Add 161
	38	205 174 50	108	Add 327
	39	192 159 77	130	13 55 90 113 118 134 189 190 196 211 238 267 279 301 306 312 326 328 338 349 372 408 419 420 442 455 465 490 493 511
	40	135 195 85	150	28 55 86 122 125 134 172 185 190 202 207 213 220 229 233 305 310 326 333 351 357 372 389 404 414 416 445 449 462 471 492
4	38*	219 174 44 1	104	Add 105 to the 37-factor, L=3 design
	39*	207 183 52 3	119	Add 331
	40*	189 192 61 6	137	Add 275
	41	175 195 73 9	156	Add 494
	42	158 195 87 13	178	Add 142

	43	142 195 97 20	202	Add 460
	44	111 192 105 34	237	57 73 87 92 100 107 126 135 147 157 169 172 209 226 255 273 298 303 307 308 317 350 379 385 395 414 438 443 452 462 475 483 489 497 504
	45	103 180 113 47	267	Add 103
	46	93 168 114 66	302	11 54 83 84 94 122 127 131 205 206 210 227 234 239 245 269 273 283 286 297 302 307 308 326 355 391 394 403 405 417 431 438 440 456 463 474 503
5	44*	126,194,109,25,1	227	Add 228 to 43-factor, L=4 design with w4=202
	45*	115,184,123,32,2	255	Add 115
	46*	100,185,123,44,4	286	Add 182
	47	96,167,136,52,7	319	Add 358
	48	89,158,135,67,10	355	Add 255
	49	72,150,142,72,18	396	54 106 131 157 172 175 207 209 226 244 248 270 308 328 347 349 353 389 414 422 442 445 458 491 493 202 338 365 214 335 312 347 73 226 420
	50	68,142,125,77,38	453	15 22 35 63 71 78 91 105 116 157 165 166 171 177 186 200 203 205 213 217 243 248 277 280 290 293 297 303 304 325 343 371 376 381 402 407 428 441 454 463 501
	51	69,127,117,84,53	504	7 11 19 45 84 93 142 154 159 164 167 170 184 193 208 217 233 246 266 279 285 292 302 310 315 326 337 344 357 362 372 389 395 414 418 447 457 471 475 487 488 497

* While not a minimum *L* design, it is listed because it has smaller w_4.