國立中山大學　應用數學研究所

碩士論文

# 題目：具有 MA 誤差之簡單線性迴歸模型下之穩健實驗順序設計

研究生：邱國輝 撰

指導教授：羅夢娜 教授

中華民國　九十三　年 六 月

# Robust Run Order for Experimental Designs in Simple Linear Regression with MA Errors


by

G. H. Chiou

Advisor

M.-N. L. Huang

Department of Applied Mathematics,
National Sun Yat-sen University
Kaohsiung, Taiwan 804 R.O.C.


June 2004

# Contents

# Abstract

In this work, a method to choose the best run order for a given experimental design is proposed, for the simple linear regression model with MA errors. More specifically we investigate the best run order of an uniform design when errors follow a MA(1) or a subset MA($k$) process where $k$ is a positive integer. The correlation matrix $P$ resulting from the errors is usually difficult to obtain a good estimate. Using the change of variance function($CVF$) to see the relation of the uncorrelated and the serially correlated errors. Criterion proposed by Zhou (2001), we find the best run order of the uniform design on $[-1, 1]$ to minimize the robust criterion, $|CVF|$. We will display the permutation of a run order after rearrangement by our method and show how the structure is decomposed into three categories to solve the problem.

**Keywords :** Best run order, Change of variance function, Correlation matrix, MA process, Uniform design.

# List of Tables

# List of Figures

# 1 Introduction

Consider a multiple linear regression model,

$$Y_i = \mathbf{x}_i^T \beta + \varepsilon_i, \qquad i = 1, ..., n \ ,$$

the response $Y_i$ is observed at $\mathbf{x}_i$ in a $p$ - dimensional space $S$, having expected value $\mathbf{x}_i^T \beta$, where $\beta$ is a $p \times 1$ unknown vector parameter and the $\varepsilon_i^{\prime} s$ are possibly serially correlated with covariance matrix $\sigma^2 P$ for some correlation matrix $P$.

Most robust designs against small departures from uncorrelated errors under certain robust criteria are intended to yield efficient estimates, so these robust designs usually minimize some scalar functions of the information matrix of the unknow parameter estimators. The scalar functions such as determinant, trace, and largest characteristic root correspond to classical $D$-, $A$-, and $E$- optimal criterion respectively. Robust designs against small departures from uncorrelated errors have been investigated by many authors such as, Sacks and Ylvisaker (1966, 1968, 1970), Bickel and Herzberg (1979), Wu (1981), Zergaw (1988) Constantine (1989), Bischoff (1992, 1993), Su and Cambanis (1994), Wiens and Zhou (1996, 1997, 1999), Hughes-Oliver, Jacqueline (1998), and Angelis, Bora-Senta, and Moyssiadis (2001).

A special robust criterion which selects the run order of design points $\mathbf{x}_i$'s in an optimal manner for the purpose of making accurate inferences about an unknown parameter $\beta$ has been proposed by Zhou (2001). According to this robust criterion we can form the confidence intervals for linear functions of $\beta$, say $a^T \beta$, where $a^T$ is a vector for $\beta$. The coverage of these confidence intervals based on the change of variance function ($CVF$) can be possibly close to the nominal level as $CVF$ is close to 0. We find the best run order of a given design that is robust against possible correlation among the errors.

In particular we focus on constructing the run order of an uniform design $(x_1, ..., x_n)$

for a simple linear regression model $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, with design point $x_1 = -1$, $x_n = 1$, $x_i$ is equally spaced on interval $[-1, 1]$. In this work the errors are assumed to follow a MA process. We will discuss the design problem for errors with MA(1) process and subset MA process in the following sections. It is known that in MA(1) process, $\varepsilon_i = w_i + \phi_1 w_{i-1}$, where $\phi_1$ is the MA(1) parameter and the $w_i$ is white noise. The correlation matrix $P = (p_{i,j})$ has elements $p_{i,i} = 1$, $p_{i,i-1} = p_{i-1,i} = \rho_1 = \phi_1/(1+\phi_1^2)$, and $p_{i,j} = 0$ otherwise. Without knowing the exact value of $P$ in this case, the experimenter tends to estimate $\beta$ by the unweighted least square estimate $\widehat{\beta}$. Then covariance matrix for $\widehat{\beta}$ is

$$Var(\widehat{\beta}) = V(P) = \sigma^2 (X^T X)^{-1} X^T P X (X^T X)^{-1},$$

where $X = (\mathbf{x}_1, ..., \mathbf{x}_n)^T$ is a $n \times p$ matrix.

To make inference about $\beta$, we need an estimate for the correlation matrix $P$. To avoid estimation of $P$, the experimenter may prefer to use the covariance matrix under uncorrelated errors, $V(I) = \sigma^2 (X^T X)^{-1}$, for inference purpose. The inference based on $V(P)$ and $V(I)$ can be quite different with a given design, especially if the ordering of the design points are not properly arranged.

In the following we first introduce the robust criterion called $CVF$, proposed by Zhou (2001), for describing the influence by using $V(I)$ instead of $V(P)$ in estimation of the linear function of $\beta$. Then optimal designs are investigated by finding the best run order to minimize the absolute value of the change of variance function.

## 1.2 Change of variance function and robust criterion.

As introduced above, the experiment designs model is $Y_i = \mathbf{x}_i^T \beta + \varepsilon$, suppose the errors $\varepsilon_i$ are normally distributed with covariance matrix $\sigma^2 P$. We can construct a $100(1-\alpha)\%$

confidence interval for $a^T\beta$ with $V(P)$. Usually $P$ is unknown in practice and needs to be estimated. In many situations the correlation in the errors may be ignored and a $100(1-\alpha)\%$ confidence interval would be constructed based on the variance for uncorrelated errors; that is $V(P)$ is replaced by $V(I)$. We use the change of variance function ($CVF$) for $a^T\beta$ at the uncorrelated errors in the direction of correlation matrix $P$ to investigate the coverage for the interval constructed by $V(I)$. The relation of the $CVF$ for a given vector $a$, $a \neq 0$, is defined as

$$CVF_a(\xi, P) = \frac{(\partial a^T V((1-t)I + tP)a)/\partial t|_{t=0}}{a^T V(I)a} \ , \quad a \neq 0 \ ,$$

where $\xi$ denotes the design points $(x_1, \ldots, x_n)$ of their ordering and $\partial/\partial t$ denotes the partial derivative with respect to $t$. The $CVF_a(\xi, P)$ represents the change of the variance due to small correlation among the errors and normalizes finally. The form of the $CVF$ may be rewritten as

$$CVF_a(\xi, P) = \frac{a^T(V(P) - V(I))a}{a^T V(I)a} \ , \quad a \neq 0 \ .$$

The $CVF$ provides a natural mechanism for assessing coverage under deviations from the idealized model. As noted earlier, ignoring the error correlations, i.e. setting $P = I$, gives coverage for the confidence intervals which are vary different from the normal levels even if the correlation is very weak. We use the $CVF$ when there is a small departure from uncorrelated errors. The $CVF$ is a very useful tool to study robustness of a procedure and indicates the local robustness when there is a small departure from the underlying model assumptions.

By Zhou (2001) it is noted there that the coverage for the confidence interval of $a^T\beta$ constructed by $V(I)$ with $a^T\widehat{\beta} \pm z_{\alpha/2}\sqrt{a^T V(I)a}$ is actually $100(1-\alpha\prime)\%$ under correlated errors, where $\alpha' = 2 \cdot P(Z > z_{\alpha/2} \ / \sqrt{1 + CVF_a(\xi, P)} \ )$ , and $Z$ is the standard normal random variable. To improve the coverage for the interval constructed by $V(I)$ by the choice of the design in Zhou (2001), it is proposed to find a design under the robust criterion which minimizes $|CVF_a(\xi, P)|$ with respect to certain correlation matrix $P$, where the minimum is over all permutations of the order of the design points in $\xi$. The resulting design, say $\xi^*$ , would be useful for making more accurate inference on $a^T\beta$.

Given a design $\xi$, a robust run order design $\xi^*$ which minimizes the robust criterion minimizes the difference between the coverage level for the interval constructed by $V(I)$ and the nominal level among all permutations of the order of the design points in $\xi$. The $CVF$ indicates the local robustness of the variance of $a^T\widehat{\beta}$ against possible correlation among the errors. If the errors are uncorrelated, the coverage for the interval constructed by $V(I)$ is the same as that constructed by $V(P)$ and the coverage is exactly the same as the nominal. A design with a robust run order would be quite useful. Now we give a formal definition about an optimal design with a best run order for a given $\xi$.

**Definition 1.** *A permutation of design points $\mathcal{X}_n = (x_1, ..., x_n)$ in a design $\xi$ is defined to be a best run order for estimating $a^T\beta$, if it minimizes the absolute value of the $CVF_a(\xi, P)$ with respect to the correlation matrix $P$, i.e.*

$$\min_{\xi} |CVF_a(\xi, P)| \ .$$

In the following sections we consider the case with $a^T = (0, 1)$ to find the best run order in a simple linear regression with MA errors, and without confusion write the corresponding change of variance function as $CVF(\xi, P)$.

This paper is organized as follows. In Section 2, we propose a special method to arrange the run order of an uniform design which is optimal in the sense that the inferences based on $V(P)$ and $V(I)$ have minimal difference in the simple linear regression with MA(1) error. In Section 3, according to the method used in Section 2, we find some rules which may be generalized to apply to that for errors with the subset MA process. Finally we conclude with a few remarks and discuss possible future work in MA(2) process in Section4.

## 2 Robust run order for MA(1) process in the simple linear regression model

From this section on, we focus our attention on the simple linear regression model, where the design matrix is $X = (1\ \mathcal{X}_n^T)$. Assume $\varepsilon$ follows a MA(1) process with $\rho_1 = \phi_1/(1+\phi_1^2)$, where $\phi_1$ is the MA(1) parameter. Let $\mathcal{X}_n = (x_1, x_2, \ldots, x_n)$ be equally spaced on $[-1, 1]$ with $-1 = x_1 < x_2 < \ldots < x_n = 1$. For convenience, we first adjust the design points as $y_i = (1+\delta_n)(-1-n+2i)/2$ for $i = 1, 2, \ldots, n$. For $Y_n = (y_1, y_2, \ldots, y_n)$, let

$$Y_n = (1+\delta_n)\frac{(n-1)}{2}\mathcal{X}_n, \quad \delta_n = \begin{cases} 0 & \text{if } n \text{ is odd} \\ 1 & \text{if } n \text{ is even.} \end{cases}$$

After changing the form of the design expression from $\mathcal{X}_n$ to $Y_n$, the elements in $Y_n$ are all integers. Note that $Y_n$ is just a run order with the elements arranged from the smallest to the largest. If we rearrange the run order by some other permutation, it will be in a different form. Let $Z_n = (z_1, z_2, ..., z_n)$, where $z_i = Z_n(i)$ for $i = 1, 2, ..., n$, be an one to one and onto mapping from $\{1, 2, ..., n\}$ to $\{y_1, ..., y_n\}$. Now we define the set of all permutations on $Y_n$ as follows.

**Definition 2.** *Let $\mathbb{Z}_n$ be the set of all permutations on $Y_n$, that is*

$$\mathbb{Z}_n = \{Z_n \mid Z_n \text{ is a permutation on } Y_n\}\ ,$$

*and $S(Z_n)$ be the summation of the product by the neighboring two elements of $Z_n$ as follows,*

$$S(Z_n) = \sum_{i=1}^{n-1} z_i z_{i+1}\ .$$

Let $\mathbb{U}_n$ be the set of the best run orders on $\mathbb{Z}_n$, that is $\mathbb{U}_n = \{U_n \mid U_n$ is a best run order on $\mathbb{Z}_n\}$. In order to find the optimal robust design to minimize the $|CVF(\xi, P)|$, we go a step further to decompose the structure of the run order according to the given criterion. The minimization of $|CVF(\xi, P)|$ related to $V(I) = \sigma^2(X^TX)^{-1}$ and $V(P) = \sigma^2(X^TX)^{-1}X^TPX(X^TX)^{-1}$ is equivalent to make $V(I)$ and $V(P)$ as close as possible in some sense, i.e. to minimize $|X^TPX - X^TX| = |2\rho_1 \sum_{i=1}^{n-1} x_i x_{i+1}|$, it is also equivalent to minimize $|S(Z_n)|$ for $Z_n$ belongs to $\mathbb{Z}_n$.

Before investigating some properties of a run order, first note that from Definition 2, we have the following relation for $S(Z_n)$, where

$$S(Z_n) = S((z_1, ..., z_n)) = S((z_1, ..., z_m)) + S((z_m, ..., z_n)), \quad 1 < m < n \ .$$

Now we divide the whole sequence of a run order into three categories. The first category is an unit with three elements. Then there is always an element between neighboring units, all kinds of these elements in the original sequence are classified as links and referred them as in the second category. The remnant members of the sequence are put in the ending as a coda, which is the third category. After arranging the whole sequence by this method, the structure of a run order becomes one which connects elements from these categories, one is an unit, followed by a link then an unit etc, and leave all the rest from the third category in a coda.

In summary, the basic components in the three categories of a sequence is expressed as follows.

**(i) An unit**: a partial sequence with three elements.

**(ii) The link**: elements arranged between units.

**(iii) The coda**: the last few elements after those already arranged in the units and links.

The following figure is an example of the possible structure of a run order.

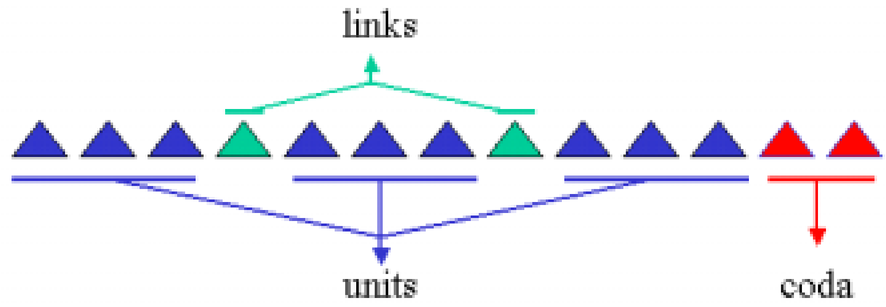



**Figure 1.** Three categories of a run order.

Now we will explain in a more detail of our method. In the first category, a partial sequence with three elements are treated as an unit. For an unit which is on the $r^{th}$ location with a special permutation form is as $u_r(i,j) = (y_i, y_j, y_{n-i+1})$ for $i = 1,2,...,n, \quad j \neq i,\ n-i+1, \quad r = 1,2,...,r(n)$, where $r(n)$ is an upper bound depending on $n$, this will result in $S(u_r(i,j)) = 0$. Because if the sum of the first and the last members in an unit is zero, then no matter what value is in the second member, the result after operation of summation will be zero. This is summarized in Lemma 1.

**Lemma 1.** *For an unit with a special form as* $u_r(i,j) = (y_i, y_j, y_{n-i+1})$*, we have* $S(u_r(i,j)) = 0\ ,\ \forall\ i = 1,2,\ldots,n\ ,\ j \neq i,\ n-i+1\ ,\ r = 1,2,...,r(n).$

**Proof.** Because $\mathcal{X}_n$ is an uniform design on $[-1,1]$ with $-1 = x_1 < x_2 < \ldots < x_n = 1$ $x_i = -x_{n-i+1}$, $i = 1,2,\ldots,n$, then $y_i = -y_{n-i+1}$. Therefore $S((y_i, y_j, y_{n-i+1})) = y_i y_j + y_j y_{n-i+1} = 0$. $\square$

In the second category, let every four continuous elements on $s^{th}$ location with a special form be $l_{s,m}(i,j,v) = (y_i, -y_{i+j}, -y_{i+j+v}, y_{i+2j+v})$ for $i,j,v = 1,2,...,n, \quad i+2j+v \leq n \quad s = 1,2,...,s(n), \ m = 1,2,3,4$, where $s(n)$ is an upper bound depending on $n$, it in turns makes $y_i - y_{i+j} - y_{i+j+v} + y_{i+2j+v} = 0$. Note that $m$ is denoted as the $m^{th}$ element in the vector $(y_i, -y_{i+j}, -y_{i+j+v}, y_{i+2j+v})$. In particular we will use this property as $v = -2$ to infer Lemma 2.

Lemma 1 deals with the partial order of the sequence and the above equation deals with the order of a subsequence of the links of a given run order. According to these results, as long as $n$ is larger, say $n \geq 17$, the best run order can be found immediately when the best run order for $n < 17$ has been found. In the following lemma, we will show that if $n \geq 17$, the first seventeen members will result in zero.

**Lemma 2.** *If $n \geq 17$, let the partial sequence $(z_1, ..., z_{17})$ of $Z_n$ be ( $u_1(1,2)$, $l_{1,1}(n-1,-2,-2)$, $u_2(3,n-3)$, $l_{1,2}(n-1,-2,-2)$, $u_3(5,n-5)$, $l_{1,3}(n-1,-2,-2)$, $u_4(7,8)$, $l_{1,4}(n-1,-2,-2)$, $y_9$ ) $= (y_1, y_2, y_n, y_{n-1}, y_3, y_{n-3}, y_{n-2}, y_4, y_5, y_{n-5}, y_{n-4}, y_6, y_7, y_8, y_{n-6}, y_{n-7}, y_9)$, then $S((z_1, ..., z_{17})) = 0$.*

**Proof.** By the properties of the special units and links as given above, it is easy to see the sums for the units are all zero, i.e. $S(u_1(1,2)) = S(u_2(3,n-3)) = S(u_3(5,n-5)) = S(u_4(7,8)) = 0$. So we compute the first seventeen partial sequence in $(z_1, ..., z_n)$ as,

$S((z_1, ..., z_{17})) = S((y_n, y_{n-1}, y_3)) + S((y_{n-2}, y_4, y_5)) + S((y_{n-4}, y_6, y_7)) + S((y_{n-6}, y_{n-7}, y_9))$

$= y_{n-1}(y_n + y_3) + y_4(y_{n2} + y_5) + y_6(y_{n-4} + y_7) + y_{n-7}(y_{n-6} + y_9)$

$= (2\delta_n + 2)(-y_2 + y_4 + y_6 - y_8)$

$= -(2\delta_n + 2)(l_{1,1}(n-1,-2,-2) - l_{1,2}(n-1,-2,-2) - l_{1,3}(n-1,-2,-2) + l_{1,4}(n-1,-2,-2))$

$= 0$ . □

In Lemma 2 we know that as $n \geq 17$, the first seventeen members result in zero. When

$n$ is large enough, we want to find a best run order $U_n$, then $|S(U_n)| = |S(U_{n-16})|$, where we can arrange the first seventeen elements by Lemma 2. The discussion of the remnants are similar with the case that the sequence is $(z_1, ..., z_{n-16})$. If $n - 16 > 16$ , on the same way we have $|S(U_{n-16})| = |S(U_{n-2\times 16})|$, it may continue until $|S(U_n)| = |S(U_{n-k\times 16})|$, where $n - k \times 16 \leq 16$, $k$ is a positive integer. Every seventeen members will move in cycles which results in zero. Because $n$ and $n - k \times 16 \leq 16$ are both odd or both even, therefore $|S(U_n)| = |S(U_{n-k\times 16 \leq 16})| = min|S(Z_n)|$. But we should be careful when $n = 20$, because when $n = 4$, $Y_4 = (-3, -1, 1, 3)$ then $|S(U_4)| = 3$, however if we revise the elements in the coda and the second member in the last unit, we can get $U_{20}$ such that $|S(U_{20})| = 1$. Here let $U_{20} = (y_1, y_2, y_{20}, y_{19}, y_3, y_{17}, y_{18}, y_4, y_5, y_{15}, y_{16}, y_6, y_7, y_{12}, y_{14}, y_{13}, y_9, y_{11}, y_{10}, y_8)$ then $|S(U_{20})| = |4(y_{19} + y_4 + y_6 + y_{13}) + S((y_9, y_{11}, y_{10}, y_8))| = |S((y_9, y_{11}, y_{10}, y_8))| = 1$. The first seventeen members of $U_{20}$ are according to Lemma 2, but we change the elements $y_8$ and $y_{12}$ with each other and adjust slightly on the last four members, $|S((z_{17}, z_{18}, z_{19}, z_{20}))| = 1$. This is the only exception case.

Because any size of run orders can be divided into three categories. The structure of a run order can be considered as to connect adjacent units by a link, and with elements at the last few positions. According to the possible number of the elements in the third category, the observations of size $n$ will be divided into four groups. A key point that we need to pay more attention is that we always adjust the last few elements to get the best run order. Let the few elements, which take the absolute value will become the smallest, leave in the coda will convenient us to revise. When $n$ is odd we usually leave five smallest ones, else we leave six smallest ones in the coda. However, when $n$ is not large enough, the number of the elements in the coda may have a small difference. For example when $n = 9$, $U_9 = (-4, -3, 4, 3, -2, 2, -1, 0, 1)$, there is a special unit $(-4, -3, 4)$ in the first. Because the number of the remnant elements is not large enough, we can not arrange the second unit. There has six elements in the coda.

When using the methods as given above, we still have to concern about some rules as follows.

**(i)** The arrangement of the elements are from the absolute value large to small in the units, links, and the coda as they possible. Arrange the elements in the units from the absolute value large to small continuously, this will make the arrangement in the links more easily to adjust.

**(ii)** We can use the second member, which can be put in by arbitrary element, in the units to easily revise the arrangement in the coda.

In Theorem 1 we will provide details on how we arrange the run order to find the best one.

**Theorem 1.** *If n=4, $min|S(Z_4)| = 3$; else when n is even, $min|S(Z_n)| = 1$ and n is odd, $min|S(Z_n)| = 0$, for $Z_n$ belongs to $\mathbb{Z}_n$ .*

**Proof.** Because the number of the elements in the coda will be restricted by the number of units and links, every three continuous members in the coda may be an unit, an element connected after an unit may be a link, so we divide the proof into 4 cases.

Case 1: $4|(n-1)$

We can take the last five members, $(z_{n-4}, z_{n-3}, z_{n-2}, z_{n-1}, z_n)$, of $Z_n$ as $Z_5$, which is one of the permutations on $Y_5 = (-2, -1, 0, 1, 2)$ for $Z_5 \in \mathbb{Z}_5$. The values of any kind of permutation may be $S(Z_5) \in \{-7, -6, \ldots, -2,\ 0,\ 2,\ 3,\ 4\}$.

$(i) n = 5$, we can choose $U_5$ quickly, s.t. $S(U_5) = 0$.

$(ii) n = 9$, we take the first five members of $U_9$ as $(-4, -3, 4, 3, -2)$, then $S((z_1, z_2, z_3, z_4, z_5)) = 6$, therefore choose the last five members such that $S((z_5, z_6, z_7, z_8, z_9)) = (-6)$, and we can get $S(U_9) = 0$.

$(iii)$otherwise, we make the form of $U_n$ as $(z_1, z_2, z_3)$, $(z_4)$, $(z_5, z_6, z_7)$, $(z_8)$, ..., $(z_n)$ . By Lemma 1, $S((z_{4r-3}, z_{4r-2}, z_{4r-1})) = S(u_r(i, j)) = 0$ for $i = 1, 2, ..., n$, $j \neq i, n-i+1$, $r = 1, 2, ...r(n)$, where $r(n)$ is an upper bound depending on $n$, then arrange the elements in the second category as $l_{s,m}(i, j, v)$ for $i, j, v = 1, 2, ..., n$, $\quad i + 2j + v \leq n \quad s = 1, 2, ..., s(n)$ , $m = 1, 2, 3, 4$, where $s(n)$ is an upper bound depending on $n$, we adjust the last five members of $U_n$, such that $S(U_n) = 0$.

Case 2: $4|(n + 1)$

According to Case1.(iii), we make $U_n$ as $(z_1, z_2, z_3)$, $(z_4)$, $(z_5, z_6, z_7)$, $(z_8)$, ... , $(z_{n-2}, z_{n-1}, z_n)$. Concerning about each element $z_{4i}$ in the links for $i = 1, 2, ..., i(n)$, where $i(n)$ is an upper bound depending on $n$, and the last unit $(z_{n-2}, z_{n-1}, z_n)$, if the resulting value of the summation of $z_{4i-1}z_{4i} + z_{4i}z_{4i+1}$ for $i = 1, 2, ..., i(n)$ is zero, then taking $S((z_i, z_{i+1}, z_{i+2})) = 0$ for $i = 4k + 1$, $k = 0, 1, 2, ..., k(n)$, where $k(n)$ is an upper bound depending on $n$, else we adjust the last unit such that $S(U_n) = 0$.

Case 3: $4|(n + 2)$

We can take the last six members of $Z_n$ as one of the permutation of $Y_6 = (-5, -3, -1, 1, 3, 5)$, all the discussions are similar as in Case 1, therefore $|S(U_n)| = 1$.

Case 4: $4|n$

$(i)n = 4$, because $n$ is not large enough, all the possible permutations make $|S(U_n)| = 3$.

$(ii)n = 8, 12$, which are similar with Case 2, but the arrangement have weaker relation of each other, we should adjust the permutation farther.

$(iii)$*otherwise*, all the discussions are similar with Case 2.

When $n \geq 17$ , we may use Lemma 2 to simplify the selection in by the above discussions. $\square$

We give two examples to illustrate our procedure as follows.

***Example*** 1: To find the best run order when the design points are $(x_1, ..., x_6) = (-1, -0.6, -0.2, 0.2, 0.6, 1)$. $Y_6 = (y_1, ..., y_6) = (-5, -3, -1, 1, 3, 5)$, and $Z_6$ is one of the possible permutations on $Y_6$. From the methods proposed before, we can get $U_6 = (-5, -1, 5, 1, 3, -3)$.

If $n = 22$, the elements after taking the absolute value from largest to smallest are $|-21|,\ 21, |-19|,\ 19, ..., |-1|,\ 1$. Let the first four units be $u_1(1,2) = (-21, -19, 21)$, $u_2(3, 19) = (-17, 15, 17)$, $u_3(5, 17) = (-13, 11, 13)$, $u_4(7, 8) = (-9, -7, 9)$, where $(19, -15, -11, 7)$ will be arranged in the first link set. Therefore the first seventeen elements $(z_1, ..., z_{17}) = (u_1(1,2), l_{1,1}(21, -2, -2), u_2(3, 19), l_{1,1}(21, -2, -2), u_3(5, 17), l_{1,1}(21, -2, -2), u_4(7, 8), l_{1,1}(21, -2, -2), y_9)$ . The best run order is $U_{22} = (-21, -19, 21, 19, -17, 15, 17, -15, -13, 11, 13, -11, -9, -7, 9, 7, -5, -1, 5, 1, 3, -3)$, $S(U_{22}) = S((z_1, ..., z_{17})) + S(U_6) = S(U_6)$.

Figure 2 is given for the illustration.

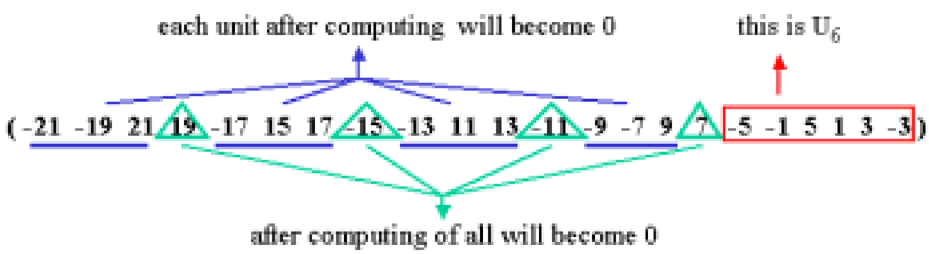



**Figure 2.** The structure of the best run order $U_{22}$.

***Example*** 2: To find the best run order when the design points are $(x_1, ..., x_5) = (-1, -0.5, 0, 0.5, 1)$. $Y_6 = (y_1, ..., y_5) = (-2, -1, 0, 1, 2)$, and $Z_5$ is one of the possible permutations on $Y_5$. From the methods proposed before, we can get $U_5 = (-2, -1, 2, 0, 1)$.

If $n = 21$, the elements after taking absolute value from largest to smallest are $|-10|,\ 10, |-9|,\ 9, ..., |-1|,\ 1, 0$. Let the first four units be $u_1(1, 2) = (-10, -9, 10)$,

$u_2(3,18) = (-8,7,8)$, $u_3(5,16) = (-6,5,6)$, $u_4(7,8) = (-4,-3,4)$, where $(9,-7,-5,3)$ will be arranged in the first link set. Therefore the first seventeen elements $(z_1,...,z_{17}) = (u_1(1,2), l_{1,1}(20,-2,-2), u_2(3,18), l_{1,1}(20,-2,-2), u_3(5,16), l_{1,1}(20,-2,-2), u_4(7,8), l_{1,1}(20,-2,-2), y_9)$ . The best run order is $U_{21} = (-10,-9,10,9,-8,7,8,-7,-6,5,6,-5,-4,-3,4,3,-2,-1,2,0,1)$, $S(U_{21}) = S((z_1,...,z_{17})) + S(U_5) = S(U_5)$.

Figure 3 is given for the illustration.

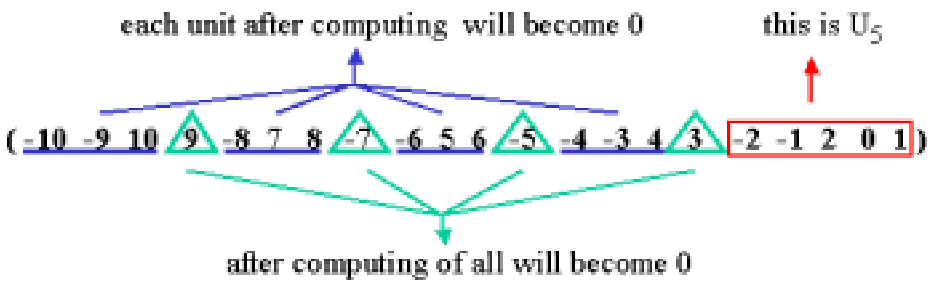



**Figure 3.** The structure of the best run order $U_{21}$.

## 3 Robust run order for subset MA process in the simple linear regression model

After studying the MA(1) process in a simple linear regression model, we are interested in the other type of MA processes. Now we investigate the case with MA(2) process. Consider a simple linear regression model with MA(2) error, $\varepsilon_i = w_i + \phi_1 w_{i-1} + \phi_2 w_{i-2}$, where $\phi_1$ and $\phi_2$ are the MA(2) parameters and the $w_i$ is white noise. The correlation matrix $P = (p_{i,j})$ has elements $p_{i,i} = 1$, $p_{i,i-1} = p_{i-1,i} = \rho_1 = (\phi_1 + \phi_1\phi_2)/(1 + \phi_1^2 + \phi_2^2)$, $p_{i,i-2} = p_{i-2,i} = \rho_2 = \phi_2/(1 + \phi_1^2 + \phi_2^2)$, and $p_{i,j} = 0$ otherwise. Using the robust criterion introduced in the previous section, minimizing $|CVF(\xi, P)|$ is equivalent to minimizing

$|2\rho_1 \sum_{i=1}^{n-1} x_i x_{i+1} + 2\rho_2 \sum_{i=1}^{n-2} x_i x_{i+2}|$. Because in a subset MA($k$) process, $k$ is a positive integer, it may put more emphasis on one of the MA($k$) parameters, that is we only deal with one parameter each time. In this situation consider the case with the correlation matrix $P = I + M_t$ , $I$ is the identity matrix and $M_t$ is the matrix where $(M_t)_{ij} = \rho_t$ only if $|i-j| = t$, for $t = 1, 2, \ldots, n-1$. Then minimizing $|CVF(\xi, P)|$ is equivalent to minimizing $|2\rho_t \sum_{i=1}^{n-t} x_i x_{i+t}|$, for $t = 1, 2, \ldots, n-1$.

We will provide some steps to explain the process in dealing with the subset MA($k$) process. By the discussion in these steps, we may obtain a best run order for any special order $t$.

***Step*** 1: When $t = 1$, the correlation matrix $P$ is $I + M_1$, which is the same in the MA(1) process. So the discussion is just as in Section 2.

***Step*** 2: When $t = n-1$, the correlation matrix $P$ is $I + M_{n-1}$. The robust criterion can be simplified as $min|2\rho_{n-1} \cdot x_1 x_n|$. The solution is trivial for this case, we will not discuss it further.

***Step*** 3: When $t = 2, 3, ..., n-2$, the correlation matrix $P$ is $I + M_t$. The robust criterion can be simplified to minimize $|2\rho_t \sum_{i=1}^{n-t} x_i x_{i+t}|$. It may be divided into $4 \times t$ cases to find the best run order.

### 3.1 Run order for the subset MA(2) process

We first investigate the case with $t = 2$. In order to apply the method used in MA(1) case, we rearrange the structure of the run order. As $t = 2$, we take product of $z_i$ and $z_{i+2}$, across $z_{i+1}$ for $i = 1, 2, \ldots, n-2$ in the run order $Z_n$. This will divide a run order into two sequences, one is with the odd members and the other is with the even members from the original run order. Each sequence is just as a new run order. We treat these two sequences as Sequence 1 and Sequence 2. Now we may use the same method used in MA(1) process to arrange the two new sequences, but with some modifications such that it is a little bit different from that in the MA(1) process, in order to get the minimum value of the summation after taking the absolute value. We should make some revisions at the last part of the two new sequences. Here we draw a plan for the subset MA($k$) process when $t = 2$ as in Figure 4.

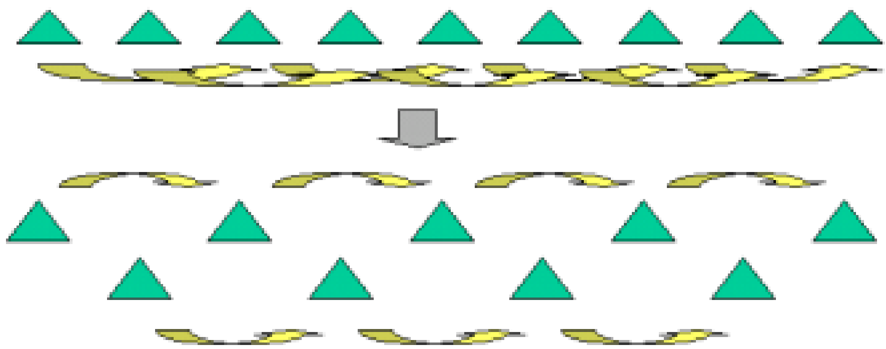
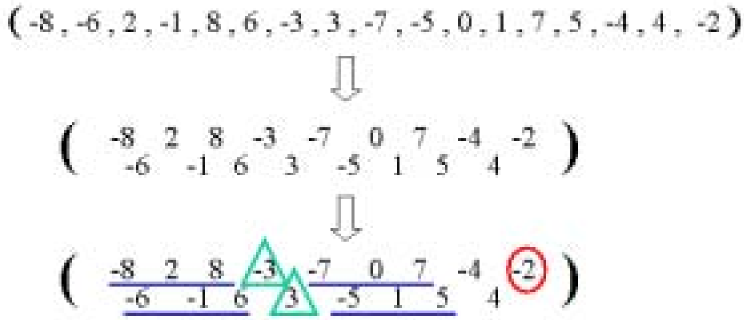


**Figure 4.** Plan for the change of the run orders.

In the subset MA($k$) process we may divide $n$ into eight groups. Because of the number of the elements in the third category of a run order have the recursive property. We use the same rule for each group to permute the run orders, but when $n$ is not large enough we may make some revision in the permutation. In the following we will illustrate how to arrange sequences in each group.

(1)$8|n-1$ : We should fill in the element in the last one of Sequence 1, whose value plus the third last one of Sequence 1 must be equal to the last one of Sequence 2. Later filling in $k$ and $-k$ in the second last elements of Sequence 1 and Sequence 2 will get the minimum result. An example is given for illustration.

***Example*** 3: When $n = 17$, we show the run order for a given design by our methods as follows.




Again as discussed above, the steps of trying to find a best run order can be summarized that

($i$) Each three members in the same baseline group an unit.

($ii$) The location marked with circles are the key of arranging the whole run order.

($iii$) Finally, revising the elements marked with triangles will get the minimum result.

Therefore this run order is a best one which minimizes $|CVF|$. Similar to the discussion above, we just give the run order by our method in the following examples and skipping the proof.

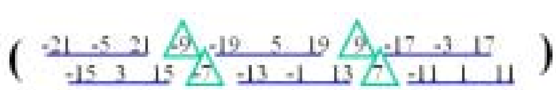
(2)$8|n-2$ : Consider the last two elements of Sequence 1 and Sequence 2, let the last one of the two sequences be filled in so that it will make the sum of the last one and the third last elements of the two sequences equal. Then filling in $k$ and $-k$ in the second last elements of Sequence 1 and Sequence 2 will get the minimum result.


(3)$8|n-3$ : We should fill in 0 in the second last element of Sequence 1, and fill in the element in the last one of Sequence 2, whose value plus the third last one of Sequence 2 must be equal to the fourth last of Sequence 1. Then filling in $k$ in the third last element of Sequence 1 and $-k$ in the second last element of Sequence 2 will get the minimum result.

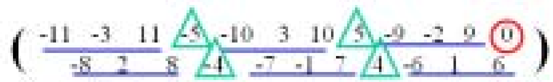

(4)$8|n-4$ : Consider the last three elements in Sequence 1 and Sequence 2, let the second last element of the two sequences be filled in so that it will make the sum of the second last and the fourth last elements of the two sequences equal. Then filling in $k$ and $-k$ in the third last elements of Sequence 1 and Sequence 2 will get the minimum result. The value of the second last elements of two sequences should not only follow the rule described above but also make the value of the mutual difference be the multiple of 4.


(5)$8|n-5$ : Consider the last three elements in Sequence 2, let the third last and the second last elements be filled in 1 and -2 in turns. When $n > 13$, filling in $(n-5)/8$ in the last one of Sequence 2 will yield a best run order.


(6)$8|n-6$ : Two sequences in this group just obey the method we used in MA(1) process will yield a best run order.

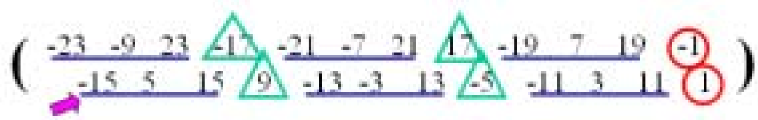



(7)$8|n-7$ : We just need to fill in 0 in the last one of Sequence 1 in this group.




(8)8$|n$ : When $n > 8$, fill in -1 and 1 in the last one of Sequence 1 and Sequence 2. In order to eliminate the remaining value, let the value of the mutual difference in the second last elements of the two sequences be the multiple of 4.


Concluding the method we illustrated above, if we want to fill in -1 and 1 in the last one of Sequence 1 and Sequence 2 respectively, we must be careful that the mutual difference in the second last elements of two sequences should be a multiple of 4. In this situation if the mutual difference in the second last elements of the two sequences is not a multiple of 4, we reduce to a lower rank of Sequence 2 and use the same method.




The method described above has the feature that it always tries to adjust the last few elements and make some revisions in units and links. Actually in the beginning of the permutation, we always follow the rule to arrange a run order which is used in MA(1) process.

## 3.2 Run order for the subset MA(3) process

When $t = 3$, the structure of the subset MA($k$) process is similar to the case with $t = 2$, but not exactly the same. The minimum of the criterion is equivalent to minimize $|2\rho_3 \sum_{i=1}^{n-3} x_i x_{i+3}|$.

We take product of $z_i$ and $z_{i+3}$, across $z_{i+1}$ and $z_{i+2}$ for $i = 1, 2, \ldots, n-2$ in the run order $Z_n$. This will divide a run order into three subsequences, each subsequence is a family with $(z_{3i+1})$ for $i = 0, 1, 2, \ldots i(n)$, where $i(n)$ is an upper bound depending on $n$, from the original sequence $Z_n$. Then we may use the same method which described above to arrange these three new sequences as Sequence 1, 2, 3.

In order to get the minimum value of the summation after taking the absolute value, again we should make some revisions at the last part of the three new sequences. Here we draw a plan for the subset MA($k$) process when $t = 3$ as in Figure 5.

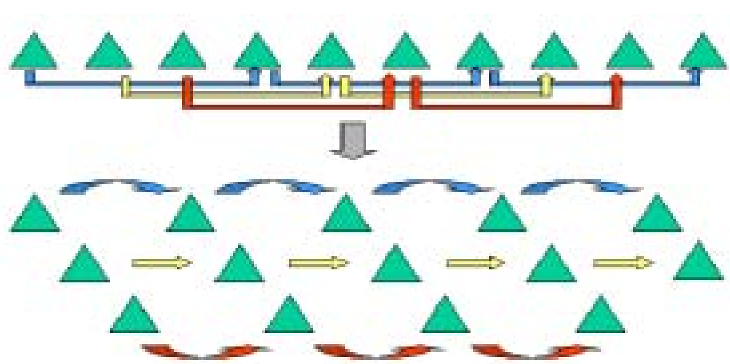
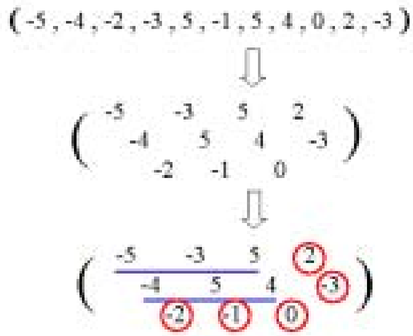


**Figure 5.** Plan for the change of the run orders.

In this subset MA(3) process the observations of size $n$ will be divided into twelve groups. We use the similar methods for each sequence to arrange the run order, but when $n$ is not large enough we may make some revisions in the arrangement. In the following, we will illustrate how to arrange the sequences.

(1)$12|n-1$ : We should fill in the element in the last one of Sequence 1, whose value plus the third last one of Sequence 1 must be equal to the last one of Sequence 2. Therefore filling in 0 in the last one of Sequence 3 will get the minimum result. An example is given for illustration.

***Example*** 4: When $n = 13$, we show the run order for a given design by our methods as follow.




Again as discussed above, the steps of trying to find a best run order can be summarized that

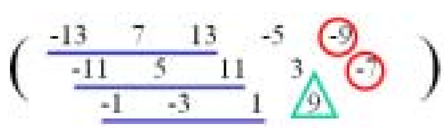
$(i)$ Each three members in the same baseline group an unit.

$(ii)$ The location marked with circles are the key of arranging the whole run order.

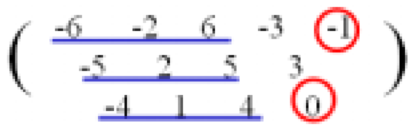
$(iii)$ Finally, revising the elements marked with triangles will get the minimum result.

Therefore this run order is a best one which minimizes $|CVF|$. Similar to the discussion above, we just give the run order by our method in the following examples and skipping the proof.

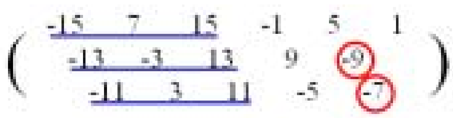
(2)$12|n-2$ : For $n \neq 14$, let the last element in each of the Sequence 1 and Sequence 2 be filled in so that it will make the sum of the last one and the third last element of these two sequences equal. Then we fill in 1 in the last one of Sequence 3. When $n = 14$, there are only four elements in Sequence 3, so we should revise Sequence 3 properly.



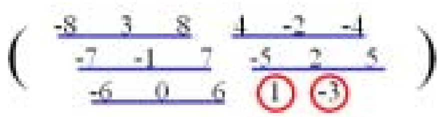
(3)$12|n-3$ : Fill in the element in the last one of Sequence 1 and Sequence 3, whose value plus the third last one of self sequence must be equally. Then filling in 0 in the second last element of Sequence 2 will yield a best run order.



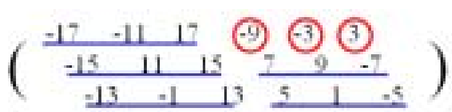

(4)$12|n-4$ : Consider the last one of Sequence 2 and Sequence 3, let the last one of these two sequences be filled in so that it will make the sum with the third last element of self sequence equal. Then filling in 1 and -1 in the last one and the third last elements of Sequence 3 will get the minimum result.



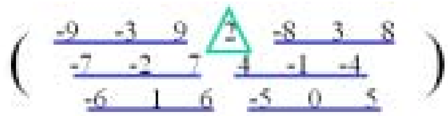

(5)$12|n-5$ : Fill in an unit in the last three members of Sequence 1 and Sequence 2. The elements in the third last of Sequence 1 and Sequence 2 should coordinate with the second last element of Sequence 3.



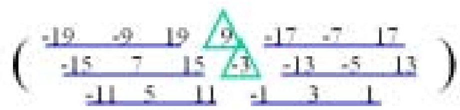

(6)$12|n-6$ : Fill in an unit in the last three members of Sequence 2 and Sequence 3. The elements in the third last of Sequence 2 and Sequence 3 should coordinate with the last three elements of Sequence 1.



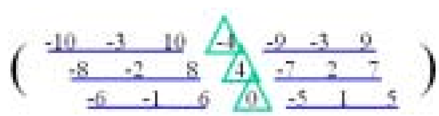
(7)$12|n-7$ : Fill in an unit in the last three members of Sequence 2 and Sequence 3. The elements in the third last of Sequence 2 and Sequence 3 should coordinate with the fourth last element of Sequence 1.



(8)$12|n-8$ : Fill in 1 and -1 in the last one and the third last elements of Sequence 3. The elements in the fourth last of Sequence 1 and Sequence 2 should coordinate with the product of the third last and the fourth last elements in Sequence 3.

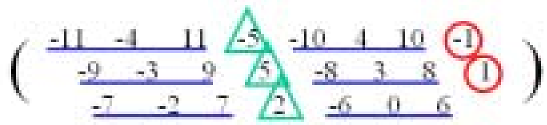


(9)$12|n-9$ : We arrange the elements in the three sequences by units and links will yield a best run order.

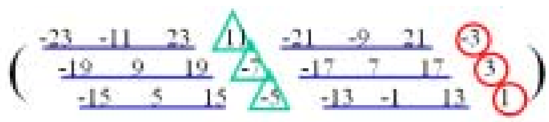


(10)$12|n-10$ : Fill in 1 in the last one of Sequence 1 and revise the value of the links will yield a best run order.

(11)$12|n-11$ : The elements in the last one of Sequence 1 and Sequence 2 should coordinate with the elements of the links.




(12)12$|n$ : Fill in -3, 3,1 in the last one of three sequences in turn, its value after computing is 1. We can not make the summation of the elements which between the units be 0, so we make this summation value be -2.




The methods we used in this section are similar to that when $t = 2$ in the subset MA($k$) process. We arrange the last few members first and do some adjustments on the elements between units. In each group we use similar method to arrange a run order, however there are still minor details. Actually the main difference is how many groups we need to divide, which is dependent on the order of the process. If we want to analyze the subset MA($k$) process as $t = 1, ..., n-2$, the observations of size $n$ will be divided into $4 \times t$ groups. When $t = 1$, $min|\sum z_i z_{i+1}| = 0$ if $n$ is odd and $min|\sum z_i z_{i+1}| = 1$ if $n$ is even. When $t = 2$, whatever $n$ is odd or even, $min|\sum z_i z_{i+2}| = 0$. When $t = 3$, this value is 0 if $n$ is odd and 1 if $n$ is even, the situation is the same as $t = 1$. The key is in the number of the summation on $z_i z_{i+t}$ for $i + t \leq n$. If $n$ is even, $z_i$ and $z_i z_{i+t}$ are both odd, when $t$ is also odd, the number of the summation on $z_i z_{i+t}$ is odd, then $min|\sum z_i z_{i+t}| = 1$; when $t$ is even it will make the number of the summation on $z_i z_{i+t}$ be even, then we can get $min|\sum z_i z_{i+t}| = 0$. If $n$ is odd, $y_1, ..., y_n$ are serial integers from $-(n-1)/2$ to $(n-1)/2$, we can get $min|\sum z_i z_{i+t}| = 0$. According to this reason, when $n$ is even and $t$ is odd in the subset MA($k$) process, the $min|\sum z_i z_{i+t}| = 1$, and the others have the $min|\sum z_i z_{i+t}| = 0$ for $k$ is a positive integer and $t = 1, 2, ..., k$.

# 4 Discussions and conclusions

While in the process of investigating the pattern of a run order for the MA(1) process, we are interested in the coverage constructed by $CVF$. Most of our work are focus on how to get a best run order, but we do not know that what will be of the coverage in the different run orders.

According to the numerical analysis by the computer, we general 1000 random run orders for any fixed size $n$, then plot some charts of coverage in Appendix C. For a given $\rho$, whatever n is, the value of the $CVF$ may be uniformly distributed from -1.5 to 1.5, if $CVF < 0$, the coverage will be larger than the nominal level, and if $CVF > 0$, the coverage will be smaller than the nominal level, but minimized the $|CVF|$ will make the coverage as possible as closely to the nominal level.

In this work a method is proposed for finding the best run order of a given design in a simple linear regression model with the MA(1) errors. Using this method, we can find a best run order which minimizes the $|CVF(\xi, P)|$ exactly. It can give us more accurate inferences for regression parameters without estimating the correlation matrix $P$ for the errors. This avoids the difficult problem of finding a reliable estimate of $P$ from regression residual.

In the MA($k$) process there are $k$ parameters, if one of the parameters is more significant, we may consider a subset MA($k$) process. The best run order can be found with the subset MA($k$) errors by using the similar methods.

After discussing about the MA(1) and the subset MA($k$) process in the simple linear regression model, now we are interested in finding the best run order for the general MA(2) process, such that it minimizes the $|CVF|$. It means that both of the two MA(2) parameters are important. According to the numerical analysis of some examples by the computer, there are apparently some rules of run orders for MA(2) process. In the future, we will study these rules and try to find a general method to solve the experimental design problems in a simple linear regression with general MA($k$) errors.

## References


[1] Angelis, L. , Bora-Senta, E. , and Moyssiadis, C. (2001). Optimal exact experimental designs with correlated errors through a simulated annealing algorithm. *Computational Statistics and Data Analysis.* **37(3)**, 275-296.

[2] Bickel, P. J. and Herberg A. M. (1979). Robust of designs against autocorrelation in Time 1: Asymptotic theorey. optimality for location and linear regression. *The Annals of Statistics.* **7**, 77-95.

[3] Bischoff, W. (1992). On exact D-optimal designs for linear models under correlated observations. *The Annals of the Instinae of Statistical Mathematics.* **44**, 229-238.

[4] Constantine, G. M. (1989). Robust designs for seriallr correlated observations. *Biometrika.* **76**, 245-251.

[5] Haines, L. M. (1987). The application of the annealing algorithm to the construction of the exact optimal designs for linear-regression models. *Technometrics.* **29**, 439-447.

[6] Hughes-Oliver, J. M. (1998). Optimal designs for nonlinear models with correlated errors. *New Developments and Applications in Experimental Design (IMS Lecture Notes Monograph Series.)* **34**, 163-174.

[7] Sacks, J. and Ylvisaker, D. (1966). Designs for regression problems with correlated errors. *The Annals of Statistics.* **37**, 66-89.

[8] Sacks, J. and Ylvisaker, D. (1968). Designs for regression problems with correlated errors; many parameters. *The Annals of Statistics.* **39**, 49-69.

[9] Sacks, J. and Ylvisaker, D. (1970). Designs for regression problems with correlated errors, III. *The Annals of Statistics.* **41**, 2057-2074.

[10] Su, Y. and Cambanis, S. (1994). Sampling designs for regression coefficient estimation with correlated errors. *Annals of the Institute of Statistical Mathematics.* **46**, 707-722.

[11] Wiens, D. P. and Zhou, J. (1996). Minimax regression designs for approximately linear models with autocorrelated errors. *Journal of Statistical planning and inference.* **55**, 95-106.

[12] Wu, C. F. (1981). On the robustness and efficiencey of some randomized designs. *The Annals of Statistics.* **9**, 1168-1177.

[13] Zergaw, G. (1988). Searching optimal designs in the presence of serially correlated errors. *Journal of Mathematical Methods in Biosciences.* **30**, 615-625.

[14] Zhou, J. (2001). A robust criterion for experimental designs for serially correlated observations. *Technometrics.* **43**, 462-467.

# Appendix

## A.1. The simplification of the change of variance function for MA(1)

$$
\begin{aligned}
CVF_a(\xi,P) &= \frac{a^T[V(P)-V(I)]a}{a^T V(I)a} \\
&= \frac{a^T\sigma^2[(X^TX)^{-1}(X^TPX)(X^TX)^{-1}-(X^TX)^{-1}]a}{a^T\sigma^2(X^TX)^{-1}a} \\
&= \frac{a^T\sigma^2[(X^TX)^{-1}X^T(P-I)X(X^TX)^{-1}]a}{a^T\sigma^2(X^TX)^{-1}a} \\
&= \frac{1}{n^2\cdot\sum_{i=1}^{n}x_i^2}\cdot(0\ n)\begin{pmatrix} (2n-2)\rho & \rho(2\sum_{i=1}^{n}x_i-x_1-x_n) \\ \rho(2\sum_{i=1}^{n}x_i-x_1-x_n) & 2\rho\sum_{i=1}^{n-1}x_ix_{i+1}\end{pmatrix}\begin{pmatrix}0\\ n\end{pmatrix} \\
&= \frac{2\rho\cdot\sum_{i=1}^{n-1}x_ix_{i+1}}{\sum_{i=1}^{n}x_i^2}
\end{aligned}
$$

where $(x_1, \ldots, x_n)$ is equally spaced on [-1,1] with $-1 = x_1 < \ldots < x_n = 1$

and we choose $a^T = (0, 1)$, therefore we have

$$
X^TX = \begin{pmatrix} n & \sum_{i=1}^{n}x_i \\ \sum_{i=1}^{n}x_i & \sum_{i=1}^{n}x_i^2 \end{pmatrix} = \begin{pmatrix} n & 0 \\ 0 & \sum_{i=1}^{n}x_i^2 \end{pmatrix}
$$

$$
(X^TX)^{-1} = \frac{1}{n\cdot\sum_{i=1}^{n}x_i^2}\begin{pmatrix} \sum_{i=1}^{n}x_i^2 & 0 \\ 0 & n \end{pmatrix}
$$

hence

$$
a^T[(X^TX)^{-1}X^T(P-I)X(X^TX)^{-1}]a = 2\rho\cdot\sum_{i=1}^{n-1}x_ix_{i+1}
$$

$$
a^T(X^TX)^{-1}a = \sum_{i=1}^{n}x_i^2 \ .
$$

**A.2. The correlation matrix and change of variance function for MA($k$) and the subset MA($k$)**

MA(1) $\begin{pmatrix} 1 & \rho & 0 & \cdots & 0 \\ \rho & 1 & \rho & 0 & \vdots \\ 0 & \rho & 1 & \rho & 0 \\ \vdots & 0 & \rho & 1 & \rho \\ 0 & \cdots & 0 & \rho & 1 \end{pmatrix}$ Subset MA(1) $\begin{pmatrix} 1 & \rho & 0 & \cdots & 0 \\ \rho & 1 & \rho & 0 & \vdots \\ 0 & \rho & 1 & \rho & 0 \\ \vdots & 0 & \rho & 1 & \rho \\ 0 & \cdots & 0 & \rho & 1 \end{pmatrix}$

MA(2) $\begin{pmatrix} 1 & \rho_1 & \rho_2 & 0 & \cdots & 0 \\ \rho_1 & 1 & \rho_1 & \rho_2 & 0 & \vdots \\ \rho_2 & \rho_1 & 1 & \rho_1 & \rho_2 & 0 \\ 0 & \rho_2 & \rho_1 & 1 & \rho_1 & \rho_2 \\ \vdots & 0 & \rho_2 & \rho_1 & 1 & \rho_1 \\ 0 & \cdots & 0 & \rho_2 & \rho_1 & 1 \end{pmatrix}$ Subset MA(2) $\begin{pmatrix} 1 & 0 & \rho_2 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \rho_2 & 0 & \vdots \\ \rho_2 & 0 & 1 & 0 & \rho_2 & 0 \\ 0 & \rho_2 & 0 & 1 & 0 & \rho_2 \\ \vdots & 0 & \rho_2 & 0 & 1 & 0 \\ 0 & \cdots & 0 & \rho_2 & 0 & 1 \end{pmatrix}$

MA(3) $\begin{pmatrix} 1 & \rho_1 & \rho_2 & \rho_3 & 0 & \cdots & 0 \\ \rho_1 & 1 & \rho_1 & \rho_2 & \rho_3 & 0 & \vdots \\ \rho_2 & \rho_1 & 1 & \rho_1 & \rho_2 & \rho_3 & 0 \\ \rho_3 & \rho_2 & \rho_1 & 1 & \rho_1 & \rho_2 & \rho_3 \\ 0 & \rho_3 & \rho_2 & \rho_1 & 1 & \rho_1 & \rho_2 \\ \vdots & 0 & \rho_3 & \rho_2 & \rho_1 & 1 & \rho_1 \\ 0 & \cdots & 0 & \rho_3 & \rho_2 & \rho_1 & 1 \end{pmatrix}$ Subset MA(3) $\begin{pmatrix} 1 & 0 & 0 & \rho_3 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & \rho_3 & 0 & \vdots \\ 0 & 0 & 1 & 0 & 0 & \rho_3 & 0 \\ \rho_3 & 0 & 0 & 1 & 0 & 0 & \rho_3 \\ 0 & \rho_3 & 0 & 0 & 1 & 0 & 0 \\ \vdots & 0 & \rho_3 & 0 & 0 & 1 & 0 \\ 0 & \cdots & 0 & \rho_3 & 0 & 0 & 1 \end{pmatrix}$

**Table 1.** $CVF$ for MA($k$) and subset MA($k$) with some $k$

| $k$ | $CVF$ for MA($k$) | for Subset MA($k$) |
|---|---|---|
| 1 | $2\rho_1 \cdot \frac{\sum_{i=1}^{n-1} x_i x_{i+1}}{\sum_{i=1}^{n} x_i^2}$ | $2\rho_1 \cdot \frac{\sum_{i=1}^{n-1} x_i x_{i+1}}{\sum_{i=1}^{n} x_i^2}$ |
| 2 | $2\rho_1 \cdot \frac{\sum_{i=1}^{n-1} x_i x_{i+1}}{\sum_{i=1}^{n} x_i^2} + 2\rho_2 \cdot \frac{\sum_{i=1}^{n-2} x_i x_{i+2}}{\sum_{i=1}^{n} x_i^2}$ | $2\rho_2 \cdot \frac{\sum_{i=1}^{n-2} x_i x_{i+2}}{\sum_{i=1}^{n} x_i^2}$ |
| 3 | $2\rho_1 \cdot \frac{\sum_{i=1}^{n-1} x_i x_{i+1}}{\sum_{i=1}^{n} x_i^2} + 2\rho_2 \cdot \frac{\sum_{i=1}^{n-2} x_i x_{i+2}}{\sum_{i=1}^{n} x_i^2} + 2\rho_3 \cdot \frac{\sum_{i=1}^{n-3} x_i x_{i+3}}{\sum_{i=1}^{n} x_i^2}$ | $2\rho_3 \cdot \frac{\sum_{i=1}^{n-3} x_i x_{i+3}}{\sum_{i=1}^{n} x_i^2}$ |

**B. The best run order**

**Table 2.** The best run orders in MA(1) process

| n | Run order | $\lvert\sum$ [illegible] $\rvert$ |
|---|---|---|
| 5 | ( -2, -1, 2 ,0, 1 ) | 0 |
| 6 | ( -5, -1 , 5, 1, 3, -3 ) | 1 |
| 7 | ( -3, -2, 3, 0, -1, 2, 1 ) | 0 |
| 8 | ( -7, 5, 7, 3, -5, 1, -3, -1 ) | 1 |
| 9 | ( -4, -3, 4, 3, -2, 2, -1, 0, 1 ) | 0 |
| 10 | ( -9, -7, 9, 7, -5, 5, -1, -3, 1, 3 ) | 1 |
| 11 | ( -5, -4, 5, 4, -3, 2, 3, -2, 1, 0, -1 ) | 0 |
| 12 | ( -11, -9, 11, 9, -7, 1, 7, -5, -3, 3, -1, 5 ) | 1 |
| 13 | ( -6, -5, 6, 5, -4, 3, 4, -3, -2, 1, 0, -1, 2 ) | 0 |
| 14 | ( -13, -11, 13, 11, -9, 7, 9, -7, -5, 1, 5, -3, -1, 3 ) | 1 |
| 15 | ( -7, -6, 7, 6, -5, 4, 5, -4, -3, 2, 3, -2, -1, 0, 1 ) | 0 |
| 16 | ( -15, -13, 15, 13, -11, 9, 11, -9, -7, 5, 7, -5, -3, -1, 3, 1 ) | 1 |
| 17 | ( -8, -7, 8, 7, -6, 5, 6, -5, -4, 3, 4, -3, -2, -1, 2, 1, 0 ) | 0 |
| 18 | ( -17,-15,17,15,-13,11,13,-11,-9, 7, 9, -7,-5,-3, 5, 3,-1, 1 ) | 1 |
| 19 | ( -9, -8, 9, 8, -7, 6, 7, -6, -5, 4, 5, -4, -3, -2, 3, 2, -1, 0, 1 ) | 0 |
| 20 | ( -19,-17,19,17,-15,13,15,-13,-11,9,11,-9,-7,3,7,5,-3,1,-1,-5) | 1 |

**Table 3.** The best run orders in the subset MA($k$) process with $t = 2$

| n | Run order | $\left\lvert \sum z_i z_{i+2} \right\rvert$ |
|---|---|---|
| 5 | ( -2, 1, -1 ,0, 2 ) | 0 |
| 6 | ( -5, -3 , -1, 1, 5, 3 ) | 0 |
| 7 | ( -3, -2, -1, 1, 3, 2, 0 ) | 0 |
| 8 | ( -7, -5, -3, 5, 7, 1, 3, -1 ) | 0 |
| 9 | ( -4, -3, 1, 0, 4, 3, -2, 2, -1 ) | 0 |
| 10 | ( -9, -7, -5, -3, 9, 7, -1, 1, 3, 5 ) | 0 |
| 11 | ( -5, -3, -4, 4, 5, 3, -1, 1, 0, 2, -2 ) | 0 |
| 12 | ( -11, -9, 7, 1, 11, 9, 5, -3, -7, -5, -1, 3 ) | 0 |
| 13 | ( -6, -4, 3, 2, 6, 4, -3, 1, -5, -1, -2, 0, 5 ) | 0 |
| 14 | ( -13, -9, -3, -1, 13, 9, -5, 5, -11, -7, 3, 1, 11, 7 ) | 0 |
| 15 | ( -7, --5, -2, -1, 7, 5, -3, 3, -6, -4, 2, 1, 6, 4, 0 ) | 0 |
| 16 | ( -15, 11, -7, -5, 15, 11, 5, -3, -13, -9, 7, 3, 13, 9, -1, 1 ) | 0 |
| 17 | ( -8, -6, 2, -1, 8, 6, -3, 3, -7, -5, 2, 1, 7, 5, -4, 4, -2 ) | 0 |
| 18 | ( -17, -13, -9, -1, 17, 13, -3, 3, -15, -11, -5, 1, 15, 11, -7, 7, 5, 9 ) | 0 |
| 19 | ( -9, -7, -4, -3, 9, 7, -5, 5, -8, -6, 4, 3, 8, 6, -1, 1, 0, 2, -2 ) | 0 |
| 20 | ( -19, -15, -11, -3, 19, 15, -7, 5, -17, -13, -5, 3, 17, 13, -9, 9, 7, 11, -1, 1 ) | 0 |

**Table 4.** The best run orders in the subset MA($k$) process with $t = 3$

| n | Run order | $\left\|\sum z_t z_{t+3}\right\|$ |
|---|---|---|
| 9 | ( -4, -3, -2, -1, 1, 0, 4, 3, 2 ) | 0 |
| 10 | ( -9, -7, 3, -5, -3, -1, 9, 7, 5, 1 ) | 1 |
| 11 | ( -5, -4, -2, 3, 1, -1, 5, 4, 0, 2, -3 ) | 0 |
| 12 | ( -11, -9, -7, -5, 5, -1, 11, 9, 7, -3, 3, 1 ) | 1 |
| 13 | ( -6, -5, -4, -2, 2, 1, 6, 5, 4, -3, 3, 0, -1 ) | 0 |
| 14 | ( -13, -11, -1, 7, 5, -3, 13, 11, 1, -5, 3, 9, -9, -7 ) | 1 |
| 15 | ( -7, -6, -5, 4, 2, -1, 7, 6, 5, -3, 0, 3, -4, 1, -2 ) | 0 |
| 16 | ( -15, -13, -11, 7, -3, 3, 15, 13, 11, -1, 9, -5, 5, -9, -7, 1 ) | 1 |
| 17 | ( -8, -7, -6,-3, -1, 0, 8, 7, 6, 4, -5, 1, -2, 2, -3, -4, 5 ) | 0 |
| 18 | ( -17, -15, -13, -11, 11, -1, 17, 15, 13, -9, 7, 5, -3, 9, 1, 3, -7, -5 ) | 1 |
| 19 | ( -9, -7, -6, -3, -2, 1, 9, 7, 6, 2, 4, -5, -8, -1, 0, 3, -4, 5, 8 ) | 0 |
| 20 | ( -19, -15, -11, -9, 7, 5, 19, 15, 11, 9, -3, -1, -17, -13, 3, -7, -5, 1, 17, 13 ) | 1 |
| 21 | ( -10, -8, -6, -3, -2, -1, 10, 8, 6, -4, 4, 0, -9, -7, -5, 3, 2, 1, 9, 7, 5 ) | 0 |
| 22 | ( -21,-17,-13, -9, -7, 5, 21, 17, 13, -5, -3, -1,-19,-15,-11, 9, 7, 3, 19, 15, 11, 1 ) | 1 |
| 23 | ( -11, -9, -7, -4, -3, -2, 11, 9, 7, -5, 5, 2, -10, -8, -6, 4, 3, 0, 10, 8, 6, -1, 1 ) | 0 |
| 24 | ( -23,-19,-15,-11, 9, 5, 23,19,15,11,-7,-5,-21,-17,-13,-9, 7,-1,21,17,13,-3, 3, 1 ) | 1 |

## C. The corresponding values of the *CVF* with respect to size $n$ in randomly selected run order under MA(1) error

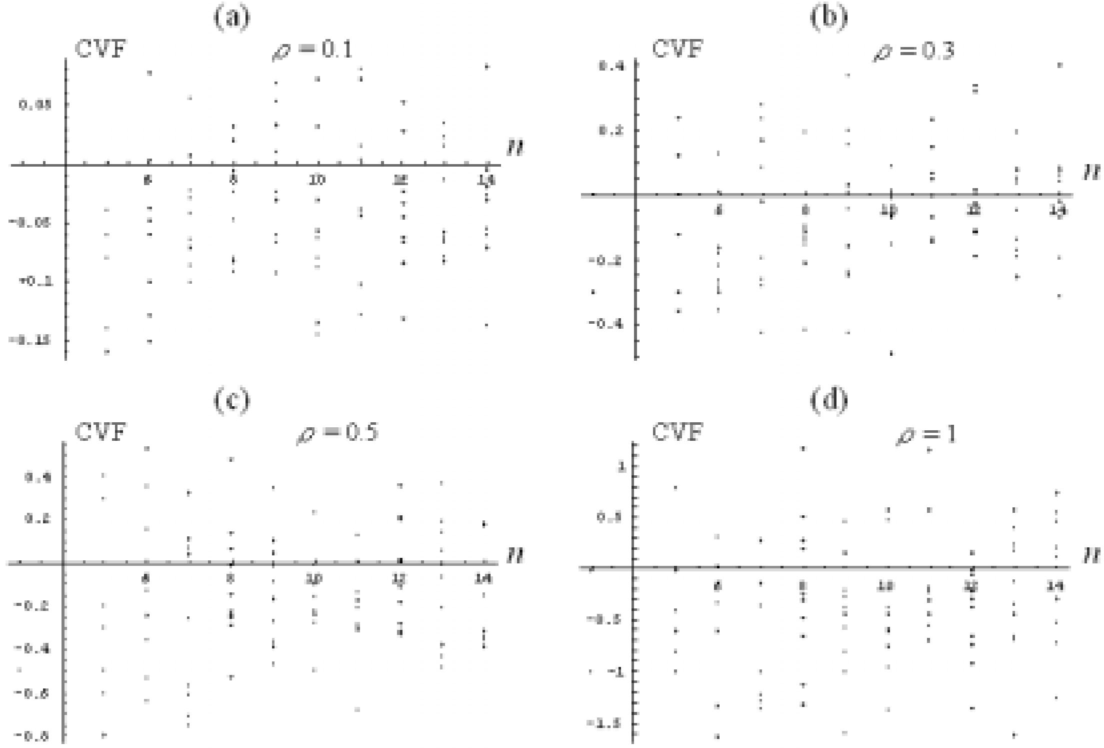



**Figure C.1 The corresponding values of the *CVF***

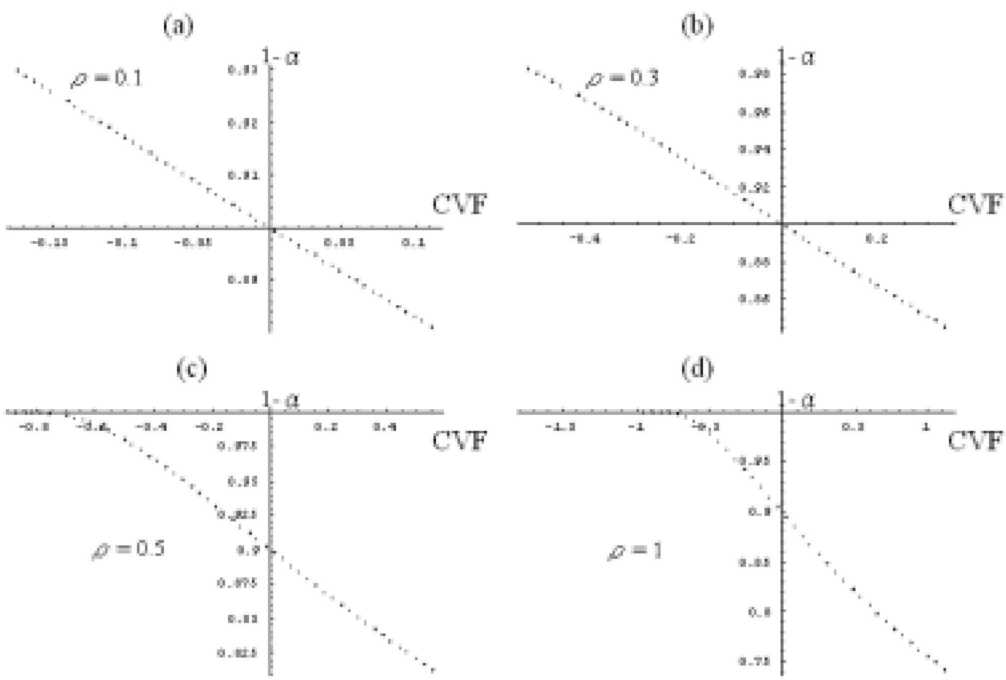



**Figure C.2 The coverage probability 90% for some given $\rho$**