\title{Optimal two-phase experimental designs where Phase~1 experiment is arranged in completely randomised design}

\author{Kevin Chang}

\date{\today}

\maketitle

**\section**{Introduction}

**\label**{sec:intro}

This chapter discusses searching for optimal designs for two-phase proteomics experiments, where the Phase 1 experiment is arranged in a completely randomised design (CRD) with multiplexing technique such as iTRAQ in the Phase~2 experiment. The first phase (Phase 1) experiment involves the organisms that are to be perturbed by the experimental conditions of interest. Since the protein abundances cannot be measured directly from the organisms, the second phase (Phase~2) experiment involves the MudPIT-iTRAQ\textsuperscript{\texttrademark} experiments for measuring the protein abundances of the experimental materials from the Phase~1 experiment.

The CRD of Phase~1 experiment consists of $v$ treatments and $r\_b$ biological replicates. Hence, $v r\_b$ equals the total number of animals, denoted by $n\_A$. To estimate the measurement errors coming from the Phase~2 experiment, $r\_t$ technical replicates are used. The Phase~2 experiment is arranged in a randomised block design (RBD) where the numbers of MudPIT runs, $n\_R$, and iTRAQ tags, $n\_\gamma$, correspond to the numbers of blocks and block size, respectively. The total number of observations, denoted by $n$, equals to both $v r\_b r\_t$ and $n\_R n\_\gamma$. The numbers of treatments, biological replicates, technical replicates, runs and tags are collectively known as the \emph{design parameters}.

There are many ways to allocate the samples from the Phase~1 experiment to the Phase~2 experiment. The number of ways that these samples can be assigned to each run is

\[

\binom {n} {n\_\gamma}.

\]

For example, if $n=10$, then there are $210$ ways to assign these samples to a run in a four-plex experiment. An optimal design can be obtained from a specific allocation of these samples to each run, based on various optimality criteria. Some optimality criteria have been described in \citep{John1987}. This chapter focuses only on the MS- and A-optimality criteria which are further discussed in Section~\ref{sec:MSandAoptimal}.

The method of finding the optimal designs for different classes of design such as block, row-column and $\alpha\_n$ designs has been previously discussed \citep{whitaker1990, Williams1996, John2002}. These methods aim to find the designs with the most treatment information in the most bottom stratum. As for finding the optimal two-phase designs, the allocation of the block factors from the Phase~1 experiment to block factors of the Phase~2 experiment also needs to be monitored. This is because if there is confounding between the block effects of Phase~1 and 2 experiments, the design may not have a valid test for the treatment effects. Hence, the goal is not only to find two-phase optimal designs that has the most treatment information in the most bottom stratum, but also to find two-phase optimal designs where a formal test for the treatment effects can be conducted in that stratum. A suitable method of finding a such two-phase optimal designs has yet to be described; the main reason is that the optimality criterion has not been defined.

This chapter describes the methods of generating two-phase optimal designs by optimising the objective function using the \emph{simulated annealing algorithm} (SA). The optimality criterion is defined in the \emph{objective function}, which is a mathematical expression describing the relationship between the variables. SA is a well-known heuristic method for finding the values of variables that result in a maximum or minimum of an objective function \citep{Kirkpatrick1983}. In the case of finding optimal designs, the variables of the objective function correspond to the candidate designs. SA plays the role in continuously generating the new candidate design and comparing the values generated from the objective function between the candidate designs. The optimality criterion and the objective function are to be discussed in detail in Section~\ref{sec:objFun}. In addition, this chapter presents an improved version of SA which shown to speed up the search for the optimal designs in Section~\ref{sec:sa}.