Designs for Sensory Multi-Session Trials with Preparation Constraints

Carola Deppe

submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

the degree is awarded by DE MONTFORT UNIVERSITY

Supervised by Dr. Nicholas Longford and Prof. Byron Jones

Sponsored by Unilever R & D, Colworth

March 2004

Acknowledgements

I like to thank Roland Carpenter and Graham Cleaver from Unilever R & D, Colworth for their support and commitment to the project and my supervisors Prof. Byron Jones and Dr. Nicholas Longford for all their advice and encouragement, Byron especially for continuing the supervision after leaving DMU and Nick for taking over the supervision. Additionally, I am very grateful to Ian Wakeling for providing me with a copy of Design Express. Finally, I'd like to thank Owen Williams for his technical support and patience with all my computer questions.

Design of sensory multi-session trials with preparation constraints

Designs for sensory studies must satisfy several requirements. Usually a given number of products are to be evaluated and there is an upper limit to the number of assessors available. Due to variation in sensory perception, inter-assessor product comparisons are preferred. For large product numbers, trials are split into sessions to avoid sensory fatigue and the sequential presentation of products can cause order and carry-over effects. Thus, resolvable row-column or cross-over designs are required, which ensure that each assessor tastes all products the same number of times.

In this thesis a three-step procedure is proposed to generate designs for trials where the number of products prepared for or served in each session is limited. First, an incomplete block design with a special column structure, the preparation design, is created, assigning products to sessions. Secondly, a cross-over design is constructed, assigning the columns of the preparation design to assessors. In the third step the two designs are combined by identifying the column-order of the preparation design that results in the highest average efficiency of the complete cross-over design. Search algorithms for incomplete block and cross-over designs are modified to produce preparation and panel designs with a special structure to guarantee resolvability of the complete sensory design.

This procedure has been enhanced to produce designs for trials involving a control and several test products, in which control-test comparisons are estimated with higher precision than test-test comparisons. Two distinct construction methods have been developed for this case. By using factorial preparation designs the three-step procedure can also be adapted for creating factorial multi-session designs with or without a control product.

Contents

List of Tables

Chapter 1

Introduction

Sensory science is defined as the study of sensory responses of humans or animals to products or services (Chambers & Wolf, eds, 1996). Its scientific aim is to understand the basic chemical and physical characteristics that cause our perception of quality. Food producers use sensory methods to develop and maintain products of high quality. The statistical analysis related to food studies is called sensometrics and its main task is to provide analytical models to extract relevant information from multi-dimensional data collected in sensory studies (Martens and Martens, 1989). This thesis is concerned with finding efficient designs, so that the maximum amount of information can be obtained from a sensory trial with the resources available. Using efficient designs means that a panel can be run at minimum cost, which can accumulate substantial savings over time.

Sensory methods to analyse food products can be divided into analytical and affective methods. At the analytical stage the interest is in detecting differences and specifying their size. This analysis is conducted by a trained panel of assessors. Affective methods, on the other hand, are carried out by a random sample of expected future customers, whose preference and acceptance of a new product is explored.

In analytical experiments difference tests are used to detect or confirm suspected small differences in product characteristics or product quality, if differences between products are small. If the differences between products can be detected more easily and an estimate of the size of the differences is also of interest, ranking and scoring procedures can be used. The main part of this thesis is concerned with designs for sensory profiling trials, i.e. scoring procedures, but may also be useful for small consumer trials, difference testing and ranking procedures.

Sensory profiling is used to measure the type and intensity of characteristic attributes of a product. While there are specific methods used for profiling, for example, the Flavour Profile Method or Quantitative Descriptive Analysis, many sensory laboratories have their own sensory profiling techniques and their own statistical models to analyse the data. In the usual form of sensory profiling experiments assessors are presented sequentially with the products to be compared and rate them on the attributes of the vocabulary for this set of products. Profiling experiments are too complex for consumers, who are not familiar with the sensory language used in these experiments.

The data obtained in sensory profiling is highly multi-dimensional. Assessors rate several products repeatedly on a large number of attributes. Consequently both univariate and multivariate methods are used to describe the differences between products. Univariate procedures are mainly used for describing the size of the differences between products on an attribute, while multivariate procedures highlight more complex relationships between products and attributes. Both methods complement each other. Before adequate designs for a study can be constructed, it is important to know how the data will be analysed. Design plans are only optimal for the specific analysis model they are created for. Thus, in Chapter 2 univariate models for sensory profiling data are presented.

Due to large assessor differences, product comparisons made by the same assessor are more similar than scores from different assessors. Thus, usually repeated measurement designs are used for profiling experiments. Due to the sequential serving of the food products two other factors need to be accounted for in adequate designs and analysis models. These are order and carry-over effects. Order effects are present when the perceived intensity of a product varies with the position it is served in. Carry-over effects refer to the effects a product of the previous serving position has on the perceived intensity in the following serving position.

The function of the sensory design is to allocate product samples to assessors and to determine the serving order for every assessor. In the simplest setting all products can be assessed by an assessor in a single session. An overview of complete block designs, which are the relevant designs for this setting, is given in Chapter 3.

Often there are too many products to be tasted successively without the onset of sensory fatigue. If many products are consumed, the palate becomes less sensitive to the stimuli of the products. Then the trial has to be split into sessions, in which a subset of the products is tested. Over all sessions each assessor still tastes all the products. Chapter 4 discusses resolvable designs, which can be used for multi-session trials, and gives an overview of construction methods for such designs.

If the products to be compared involve extensive preparation before serving, it is often not possible to have every product available for every session. Even if all products could be available, it is sometimes advantageous to reduce the number of products available in one session. This way mistakes in the preparation process and with allocating products to assessors are more likely to be avoided. On the other hand, as many products as possible should be assessed in each session to minimise context effects. Context effect is a general term that describes the dependence of the attribute scores of a product on the other products assessed along with it. A product might be assessed differently, if it is tasted alongside very similar or very different products. For multi-session trials with preparation constraints, the designs introduced in Chapter 4 cannot be used. Thus, in Chapter 5 a new procedure is proposed that takes the preparation constraint into consideration. It consists of three steps. In the first step, a preparation design is constructed that assigns the product subsets to sessions. In a second step, a panel design is constructed which assigns the columns of the preparation design to assessors. In the third step, the two designs are combined, while optimising the column order of the preparation design. For the procedure to work both, the preparation and the panel design, require a special structure, which is not catered for by existing design algorithms. Different construction techniques for preparation designs (incomplete block designs with a special structure) and panel designs (cross-over designs with a special structure) are discussed and compared, and the influence of the preparation constraint is explored.

Sometimes designs are necessary, in which a control product is assessed alongside with other test products. In such a trial comparisons between the test products and the control are of primary importance and comparisons not involving the control product are of secondary or no interest. A control has a special status within all test products. It can be the leading product on the market, the cheapest product or any other special competitor product or a standard product always included in studies concerning the product category at hand. Designs that reflect the priority of control-test comparisons, so called treatment-control designs, are discussed in Chapter 6. A review is given about the relevant theory on treatment-control designs and a derivation of the three-step procedure for such designs is developed. This procedure can be used to construct treatment-control designs for multi-session trials for a chosen control replication. The two methods introduced can be used for treatment-control designs with different control replication. The first method creates designs with a control in every block and the second method creates designs with fewer control replications.

The products in sensory trials often have a factorial structure, for example, in experiments in which a product is changed systematically to find a cheaper, healthier or otherwise more attractive formula for the product. In such studies, the interest is not in finding differences between products, but between factors. Thus, designs are required that are optimised for the specific contrasts of interest rather than for all pairwise comparisons. In Chapter 7 it is shown how the three-step procedure can be modified to create factorial designs for multi-session trials with preparation constraints.

Chapter 2

Univariate models for sensory profiling experiments

In the analysis of food products sensory profiling methods are used to assess the intensity of certain attributes of a set of products. In contrast to difference testing which is used to detect slight differences between products, profiling is used when it is known that differences between products exist. The aim of profiling is to identify which sensory attributes differ between products and to measure and describe their differences. The measuring instrument in sensory profiling is a panel of trained and appropriately instructed assessors. The number of assessors in a panel is fixed, usually 8–20. They have been trained in food tasting and are familiar with the products and the attributes.

The panel develops a sensory language for each product type it is studying, or modifies an existing one, to describe all of the sensory properties of that type of food. During a trial, assessors rate several products of a product type, one after the other, on the attributes of the relevant sensory language. The set of scores an assessor assigns to all attributes of the products in all replications is called a profile. Various different scales are used in practise for sensory scoring procedures, e.g. point scales with or without labelled categories or line scales with or without marks or labels on the lines.

In this chapter, several univariate models for estimating product differences for a single attribute are discussed. The basic univariate model for sensory profiling experiments is a linear two-way analysis of variance model with additive fixed product and assessor effects, assuming normality for the scores (Næs, 1990):

$$M_{N0}: \qquad y_{atr} = \mu + \alpha_a + \tau_t + \varepsilon_{atr} \qquad \text{with } \operatorname{var}(\varepsilon_{atr}) = \sigma^2$$

where
y_{atr} is the assessor score of assessor a on product t in replication r,
α_a the assessor location effect of assessor a,
τ_t the product effect of product t,
ε_{atr} the residual of assessor a on product t in replication r.

A term for the assessor-by-product interaction could be included in the model.

$$M_{N1}: \qquad y_{atr} = \mu + \alpha_a + \tau_t + (\alpha\tau)_{at} + \varepsilon_{atr} \qquad \text{with } \operatorname{var}(\varepsilon_{atr}) = \sigma^2$$

where $(\alpha\tau)_{at}$ refers to the assessor-by-product interaction of assessor a and product t.

Order, session and carry-over effects and their pairwise interaction terms can be added to this model. Although these effects tend to be an order of magnitude smaller than assessor, treatment and assessor-treatment interaction effects, they cannot be ignored. Given the usually extremely large variation in sensory profiling data, it it is useful to include as many relevant sources of variation as necessary to reduce bias of the product estimators.

The simple additive linear model has two major drawbacks. First, differences in scoring behaviour between the assessors tend to be large in spite of intensive training: assessors often use different parts or different amounts of the scale and differ in precision. Second, there are scaling and distribution problems with the attribute scores. Even when the products are scored on a continuous scale, the linearity of the scale is not ensured, i.e. differences on the upper part of the scale have not necessarily the same meaning as differences of the same magnitude on the lower part of the scale. And for a considerable number of attributes the product scores are skewed toward the left ("no/hardly any sensation felt").

These problems can be handled in two ways: by more extensive training of the assessors and by incorporating anticipated effects in the model. When searching for a model two characteristics have to be taken into account, bias and precision. The more parameters are included in the model the "better" is the fit of the model, which means there is less bias in the inferences, but sampling variation is greater. For more parsimonious models the sample variance is smaller, but there is more bias. The task of the statistician is to find a balance between these extremes. The following sections discuss a number of models for sensory trials that have been suggested in the literature. To make the comparisons easier, a unified notation different from most of the notation in the publications is used throughout this chapter. An overview of the notation is given in the appendix, Section A.1. In this chapter, the focus is mainly on univariate models using normal theory, since these are most frequently used in practise. A short summary of generalised linear models suggested for sensory trials is given in Section 2.5.

A "good" assessor scores product attributes consistently, that is, with small variation over replications of the same product. Assuming an underlying order of intensity of the products for a given attribute, assessors are supposed to recognise this order and detect existing differences between products. They should also score accurately any reference products on their agreed value, when they are presented "blindly" in the trial. Defined control products are the only way to assess the proximity of the assessor's scores to an actual (true) value, as true product scores cannot be established in sensory profiling. It is preferable if assessors use a large part of the scale, because it makes it easier to specify existing differences between products. The products to be compared should therefore span as wide a range of the scale as possible. This may not always be feasible for all attributes of the sensory language simultaneously.

The members of the panel are trained to acquire these abilities and their performance is checked regularly to ensure the quality of the panel. Despite extensive training differences between assessors in a panel are unavoidable, since large differences between individuals exist in the constitution of their sensory organs, resulting in different perceived intensities of stimuli and different time courses of perceptions. In profiling, assessors are asked to taste or smell a product, judge the differences between the perceived stimuli and score the product attributes accordingly on a specified scale (Dijksterhuis, 1997).

During training, the panel is exposed to a range of the stimuli and it is aligned by reference products, since it is important that every assessor relates stimulus and attributes in the same way. Example products, that demonstrate the attributes, are used to ensure this. Additionally, one or more well chosen control products are often used, such that their values for the attributes are either agreed by the whole panel or set by the panel leader. All other products can then be compared and scored in relation to these products. These measures are taken to align assessors in their basic levels of assessment. Due to the complexity of the task and in spite of intensive training, between-assessor differences will most likely still exist. The inconsistency that has not been eliminated by extensive training can be identified and appropriate terms included in the model.

The between-assessor differences in location can be accounted for in the design of the trial by using assessors as blocks. If every assessor tastes every product at least once, product scores can be compared for each assessor. The assessor term in a linear model (α_a in model M_{N0}) can be associated with the location differences in scoring behaviour of assessors.

If a product is tasted more than once (by the same assessor), a measure of assessor consistency can also be estimated. This is important, since assessors vary in their consistency over replications, which is referred to as heteroscedasticity and is discussed in Section 2.1. The simplest way to compensate for differences in repeatability is by separate standardisation of each assessor's scores (Næs, 1990) before an analysis. Pritchett-Mangan (1992) and Wilkinson and Yuksel (1997) suggest using a weighted analysis instead, where weights are taken relative to the assessor's variation. The idea in both cases is to give greater weight to more consistent assessors.

Another problem in sensory profiling is that assessors differ also in the range of the scale they are using, which means that they are separating products by a different number of units, and this inflates the assessor-treatment interaction. This is called the "Rubber-Yardstick-Problem" (Gay and Mead, 1992). Brockhoff and Skovgaard (1994), Brockhoff (1997) and Mead and Gay (1995) tackle the problem by using a multiplicative term for assessor expansion in the model equation, discussed in Section 2.2. The Brockhoff model additionally allows for different individual variances to reflect the assessor differences in consistency.

Another question for the analysis of sensory studies is whether assessors should be associated with random or fixed effects, depending on the viewpoint of the analysis. This controversy is illustrated in Section 2.3. Further influence factors relevant to sensory trials are discussed in Section 2.4. Scale problems and resulting nonlinearities can be handled by generalised linear models. Suggested strategies are outlined in Section 2.5.

2.1 Data transformation and weighted ANOVA models

Assessor scores often seem to differ in precision, which is caused by two effects: assessors use different ranges of the scale and they differ in consistency. Næs (1990) and Næs and Solheim (1991) suggest to transform the data before fitting the linear model M_{N1}. The aim of their suggested transformations is to assimilate the individual profiles to have similar ranges and similar means. The suggested general form of the transformation is

$$\hat{G}_a Y_a + \hat{T}_a = \begin{bmatrix} \hat{g}_{a1} & & 0 \\ & \ddots & \\ 0 & & \hat{g}_{aI} \end{bmatrix} \begin{bmatrix} y_{a11} & \cdots & y_{a1t} & \cdots & y_{a1T} \\ \vdots & & \vdots & & \vdots \\ y_{ai1} & \cdots & y_{ait} & \cdots & y_{aiT} \\ \vdots & & \vdots & & \vdots \\ y_{aI1} & \cdots & y_{aIt} & \cdots & y_{aIT} \end{bmatrix} + \begin{bmatrix} \hat{t}_{a1} & \cdots & \hat{t}_{a1} \\ \vdots & & \vdots \\ \hat{t}_{ai} & \cdots & \hat{t}_{ai} \\ \vdots & & \vdots \\ \hat{t}_{aI} & \cdots & \hat{t}_{aI} \end{bmatrix}$$

where

y_{ait} is the score of assessor a on product t for attribute i
$\hat{g}_{ai}$ is a shrinking/stretching factor for assessor a and attribute i, where $\hat{g}_{ai}$ can either be assumed to be the same ($\hat{g}_{ai} = \hat{g}_a \forall i$) or different for the attributes under consideration
$\hat{t}_{ai}$ is a constant added to achieve common centre points for all assessors.

Two choices of $\hat{G}_a$ and $\hat{T}_a$ are suggested:

- Standardise data to have zero mean and standard variance.

- Standardise data to achieve minimum distance between profiles, where T_a is chosen to centre the scores and G_a as the solution to the minimisation of

$$\text{trace} \sum_{a<a'} (G_a Y_a - G_{a'} Y_{a'})(G_a Y_a - G_{a'} Y_{a'})^t$$

Næs (1990) found it preferable to transform each attribute i separately, instead of using the same transformation for all attributes. This changes the distribution of the scores though, since mean and standard deviation are random variables and thus the scores are divided by the square root of a Chi-square distributed variable.

In the literature, this procedure has been criticised because variation caused by using different ranges of the scale is treated in the same way as variation caused through inconsistent scoring of products. Assessors who use a large part of the scale are as much down-weighted as assessors with high imprecision.

Pritchett-Mangan (1992) suggests to include assessor discrepancies directly into the model using a weighted linear model, with different weights assigned to each assessor and every product. The idea is to weigh each observation by the estimated individual assessor variance as a measure of precision. But this adjustment only makes sense in relation to the actual length of the scale that has been used by the assessor, therefore the following weights are defined:

$$\hat{w}_{at} = \frac{V_{\text{max}}}{V^{at}_{\text{actual}}}$$

where

$$V_{\text{max}} = \frac{1}{(r-1)}\left(m\,\text{Range}^2 - \frac{(m\,\text{Range})^2}{r}\right)$$

is the maximum variance any of the products could reach given the observed range. This transformation of the scale range is used to keep the weights dimension-free.

V^{at}_{actual} is the actual observed sample variance of assessor a for product t (replication variance)

r is the number of times a product is scored by an assessor,

m is the integer part of $\frac{r}{2}$ and

Range is the observed range of the scores over the samples/products.

To achieve a more symmetric distribution of the weights, a logarithmic transformation is used for the weighted analysis of variance (ANOVA). The specific form of the ANOVA model is not specified by Pritchett-Mangan, but the use of random assessor effects is suggested.

The advantage of this approach over transformation is that the adjustment is part of the model and that the weights are adjusted for range differences.

Wilkinson and Yuksel (1997) also model the variation dependent on the assessor who scores the product. They fit separate models for the location and the dispersion. The simplest form of their more general model, which is discussed in Section 2.5, assumes normality for the product scores. The location model is a weighted version of model M_{N1}. To account for assessor specific variances, the model is fitted using estimates for weights that reflect the precision of the assessor

$$w_a = \frac{1}{\sigma_a^2}$$

which are estimated from the dispersion model. The dispersion model takes the logarithm of the squared residuals from the location model as the response variable and regresses it against an overall mean effect and an assessor effect. The models are fitted alternately, using the most recent fitted value as the input value in the other model. Initial values are estimated by using weights of 1 in the first run of the location model.

2.2 Assessor expansiveness

Gay and Mead (1992), Mead and Gay (1995), Brockhoff and Skovgaard (1994) and Brockhoff (1997) model the heterogeneous scale ranges between assessors directly in the model equation. They emphasise, that range differences contribute to the assessor-by-treatment interaction. The remainder of that interaction is caused by nonlinearities in scoring, for example different rank orders for products. Hence, they split the interaction term into a multiplicative term of "assessor expansiveness" and the remainder for "nonlinearities" in scoring. In this way, the presence of assessor expansiveness and nonlinearities effects can be tested.

The difference between the models by Mead/Gay and Brockhoff/Skovgaard is in the emphasis on session-dependent assessor location effects in the former model and assessor dependent variances in the latter model.

The model equation for the general interaction model suggested by Brockhoff and Skovgaard (1994) differs from M_{N1} only by allowing for assessor-specific variances.

$$M_{B0}: \qquad y_{atr} = \mu + \alpha_a + \tau_t + (\alpha\tau)_{at} + \varepsilon_{atr} \qquad \text{with var}(\varepsilon_{atr}) = \sigma_a^2$$

A typical sensory trial is made up of several sessions. Assessors can taste only a certain number of products, usually three to six, without sensory fatigue. During an experiment, efforts are made to keep all conditions constant. Nonetheless, if products have to be prepared for every session, the products might differ slightly from one session to the next. Mead and Gay therefore include a session effect ζ in their model. Instead of a common session effect and additive assessor location effects, they combine them to allow for session effects specific to the assessors, $(\alpha\zeta)_{as}$, which they call the blocking effect. A block is defined here as the set of scores from an assessor within one session.

$$M_{MG0}: \qquad y_{ast} = \mu + (\alpha\zeta)_{as} + \tau_t + (\alpha\tau)_{at} + \varepsilon_{ast} \qquad \text{with var}(\varepsilon_{ast}) = \sigma^2$$

where

$(\alpha\zeta)_{as}$	is a block effect
y_{ast}	is the observation and
ε_{ast}	is the residual of assessor a in session s on product t,

The product-by-assessor interaction term in both models can then be decomposed into a term for assessor expansiveness and one for the remaining interactions caused by nonlinearities in the scoring behaviour. A model without any such interaction can be written in the notation of Mead and Gay as

$$M_{MG1}: \qquad y_{ast} = \mu + (\alpha\zeta)_{as} + \beta_a\tau_t + \varepsilon_{ast}, \qquad \text{with var}(\varepsilon_{ast}) = \sigma^2$$

where β_a is the stretching factor of assessor a and the other terms are defined as in model M_{MG0}. M_{MG0} and M_{MG1} are fitted by maximum likelihood methods and then summarised in an ANOVA table.

The corresponding notation for the Brockhoff model without nonlinearities is:

$$M_{B1}: \qquad y_{atr} = \alpha_a + \beta_a \tau_t + \varepsilon_{atr} \qquad \text{var}\left(\varepsilon_{atr}\right) = \sigma_a^2$$

In this model, three parameters are fitted for each assessor. The location parameters α_a allow for assessor-specific basic levels. The variance, resulting from the assessors using different ranges of the scale, is explained through the stretching factor β_a and the within-assessor variation is modelled through the assessor dependent variance σ_a^2. A measure of assessor precision is then defined by

$$\text{Sensitivity}_a = \frac{\beta_a^2}{\sigma_a^2}$$

which is the squared 'signal-to-noise' ratio.

As mentioned before, the size of the individual assessor variance is affected not only by inconsistent scoring, but also by the range used. This explains why assessor sensitivity is measured as the ratio between the squared stretching factor and the individual variance.

The second criterion for assessor performance are the nonlinearity factors

$$\text{Nonlinearities}_a = \sqrt{\sigma_a^2(M_{B1}) - \sigma_a^2(M_{B0})}$$

which quantify the portion of the variance that is added to the error term in model M_{B1}. The interaction term in model M_{B0} consists of two parts, individual range usage and nonlinearities. The latter is dropped in model M_{B1}. Hence the difference between both error terms is an estimate of the variance due to nonlinearity. The square root is taken, so that it can be interpreted on the units of the scale. The smaller the nonlinearity factor the better is the performance of the assessor. Models

M_{B0} and M_{B1} are fitted by an iterative algorithm for maximum likelihood, which is available as a SAS-macro.

The Brockhoff and the Mead and Gay models are the most elaborate among the proposed linear models with normality assumptions. The advantage over Pritchet-Mangan's approach is that individual variances and differences in scale usage are separated in the model and each can be tested for significance. Therefore it allows more insight into the structure of the data. On the other hand, estimates are less precise and depend heavily on the assessor dependent variances, which are estimated with only modest precision. This disadvantage also holds for approaches by Næs (1990), Pritchett-Mangan (1992) and Wilkinson and Yuksel (1997).

To gain insight into assessor performance, Brockhoff and Skovgaard (1994) and Brockhoff (1997) suggest to find the most parsimonious model that adequately fits the data from a profiling experiment. They suggest to compare five hierarchically related models, of which the first two are M_{B0} and M_{B1}. Three further models with different assessor structures are introduced.

If all stretching factors β_a are the same, model M_{B1} simplifies to

$$M_{B2}: \qquad y_{atr} = \mu + \alpha_a + \tau_t + \varepsilon_{atr} \qquad \text{with } \mathrm{var}(\varepsilon_{atr}) = \sigma_a^2$$

while model $M_{B2.2}$ reduces to model M_{B1} with the additional parameter constraint of constant sensitivities for all assessors $\left(\frac{\beta_1^2}{\sigma_1^2} = \ldots = \frac{\beta_A^2}{\sigma_A^2} = c\right)$.

$$M_{B2.2}: \qquad y_{atr} = \mu + \alpha_a + \beta_a\tau_t + \varepsilon_{atr} \qquad \mathrm{var}\left(\varepsilon_{atr}\right) = \frac{\beta^2}{c^2}$$

Model M_{B3} assumes no product differences, resulting in a model with assessor location effects only, still allowing for heterogeneous assessor variances.

$$M_{B3}: \qquad y_{atr} = \mu + \alpha_a + \varepsilon_{atr} \qquad \text{with } \mathrm{var}(\varepsilon_{atr}) = \sigma_a^2$$

In all the models, the errors ε_{atr} are independently normally distributed.

The models are compared by testing the following hypotheses:

$$\begin{array}{llcll}
H_0^I: & \sigma_1^2 = \ldots = \sigma_A^2 & \text{vs.} & H_1^I: & M_{B0} \\
H_0^{II}: & \quad M_{B1} & \text{vs.} & H_1^{II}: & M_{B0} \\
H_0^{III}: & \beta_1 = \ldots = \beta_A & \text{vs.} & H_1^{III}: & M_{B1} \\
H_0^{IV}: & \frac{\beta_1^2}{\sigma_1^2} = \ldots = \frac{\beta_A^2}{\sigma_A^2} & \text{vs.} & H_1^{II}: & M_{B1}
\end{array}$$

A likelihood-ratio test statistic is defined as minus twice the log-likelihood, and it follows asymptotically a χ^2-distribution. The Bartlett test is a bias-corrected modification of the likelihood-ratio test. For hypotheses II to IV, the likelihood-ratio tests are used, as each of the models specified under H_0 is nested within the corresponding model under H_1. Hypothesis I is tested with the Bartlett test.

The appropriate models should be expanded to all sub-models with equal variances, because hypotheses H_0^{II} to H_0^{IV} are applicable only if the first hypothesis has been rejected. If this selection process for the best fitting analysis model is used for every trial it might not only result in different models for different attributes, but also in different models for the same attribute in different trials. Thus, it will be difficult to explain results and compare them between models. Also, such a data-based decision process changes the error probabilities when testing for product differences in the selected model.

2.3 Fixed or random assessor effects

In consumer studies, the subjects (consumers) are assumed to be random since a sample of consumers testing a set of products is supposed to represent the consumers in general. Assessors, on the other hand, are highly trained individuals whose taste abilities are far more precise than those of average consumers. O'Mahony (1998) sees the panel as an analytical instrument. Averaging over assessors is used to stabilise the results. For him the process of selection and training make the panelists non-representative of any population. Differences found with such a panel do not imply that consumers would find the same differences.

The results from sensory profiling about product differences are useful only if any other panel that has been trained in the same way would yield similar results and Lundahl and McDaniel (1988) believe that anyone chosen to take part in a similar training would attain similar abilities. In that sense, the panel is representative of experts in the particular field. Therefore assessors are still representatives of such a population, which makes them trained consumers. When they are chosen for some special abilities, they represent that subgroup.

There is a general agreement that due to their training assessors do not represent the average consumer. Thus, differences detected by a trained panel would not necessarily mean that consumers would also identify that difference. But, since the panel is more perceptive, it will be able to detect smaller differences and when no differences are detected by the panel it can generally be concluded that the consumer would not detect any differences either.

The interpretation of analysis results is slightly different if assessors are taken as fixed or random. Identifying assessors with random effects puts the emphasis on the fact, that product difference are assessor dependent. Assessors in the panel are representative of some distinct population of qualified experts and inference from a mixed model with random assessor effects is therefore representative for this population and it is assumed that similar results would be obtained with different panels, while for a fixed model inference is limited to the panel at hand.

2.4 Further factors and factorial structure

Further influence factors in sensory profiling experiments are session, order and carry-over effects, batch-to-batch variation and temporal dependencies. Time-related effects describe variation over time of the assessor's scores or changing product characteristics over time. A discussion of session effects is given in Chapter 4, while order, carry-over and batch effects as well as time dependence are discussed in the following paragraphs.

In sensory trials, two order effects are regarded as important, the first-sample effect and a reduction in perceived intensity over time. Hunter (1996) describes the first-sample or first-period effect, where the first product in a session is scored relatively higher for a positive factor (lower for a negative factor) than in any of the following serving positions. To prevent a first period-effect the inclusion of a pre-period has been suggested. An arbitrary product, such as a control product, is tasted first, but not scored. Alternatively, it is scored by the assessors, but the scores are not included in the analysis. The other reported order effect is caused by sensory fatigue, resulting in reduction of perceived intensities in later servings. The session length of a sensory trial should therefore be set carefully, according to the product categories in the trial. Avery and Masters (1999) report trial fatigue for long studies, in which assessors lose concentration in later sessions. In contrast, O'Mahony (2001) mentions a warm-up effect, i.e. that assessors become more reliable after the first few tastings, when they have adjusted to the trial situation. If a first-sample order effect or an order effect caused by sensory fatigue is suspected in spite of a careful set up of the trial, the serving position of each product should be included in the analysis as a blocking factor. The estimates of product comparisons can then be corrected for order effects. In the design plan, the presentation order of the products should be varied for every assessor, ensuring that each product appears approximately the same number of times in every position. For multi-session trials there is an issue of whether an order effect occurs only within a session, or also across sessions, depending on the experiment, especially the kind of product tested and the time period between sessions. Order effects have commonly been reported in sensory experiments, for example, by MacFie et al. (1989), Muir and Hunter (1991/92), Hunter (1996) and Ball (1997).

Carry-over effects are expected mostly for products that are bitter or have a strong aftertaste. Different kinds of carry-over effects have been reported for sensory trials. First, there is the so-called first-order carry-over effect, arising when the product tasted has an influence on the score of the following product. The size of the influence is assumed to depend on the previous product only. A special case of this is reported by Kempton et al. (2001), where the first-order carry-over effect is proportional to the size of the direct product effect. Second, a product with a strong or otherwise distinctive quality may influence all subsequent tastings. Finally, the influence of the previous product can vary with the product that is affected by the carry-over. This is fairly complicated to model. If carry-over effects are assumed in sensory studies, simple first-order carry-over effects are modelled as a priority. First-order carry-over effects are the only case of carry-over discussed in this thesis. For sensory studies carry-over effects are generally smaller and rarer than order effects (Durier et al. , 1997) and might only be relevant for certain types of food (Wakeling and Buck, 2001). The problem of possible carry-over effects arising by successively tasting products is discussed, for example by MacFie et al. (1989), Muir and Hunter (1991/92), Schlich (1993), Wakeling and MacFie (1995), Hunter (1996), Ball (1997), Durier et al. (1997) and Kunert (1998).

For all models discussed so far inferential interest is assumed to be in a general treatment effect. When products have a special structure, as when products are monitored over time or when the products have a factorial structure, interest might be in specific contrasts or treatment factors and their interaction. Models are needed that reflect the structure of the products. The special case of models and designs for a factorial product structure in sensory trials are discussed in Chapter 7. An outline of the issues for experiments in which products change over time or differ between batches is given in the remainder of this section.

The product from a single product type or brand may often vary over time. Products can depend on the time and place where they have grown, bred or produced. This is the case, for example, for vegetables. Food samples of one product might not be homogeneous. They might come from different batches, that comprise homogeneous samples, while the batches themselves are heterogeneous.

If in a trial, samples of a product are used from only one batch, all assessors taste comparable samples of a products. Product differences found in such an experiment are equivalent to brand differences, but no batch-to-batch variation for each product can be estimated. Therefore generalisations of product differences are not warranted: inferences apply only to the batch included in the trial. If more general inferences are sought, products from several batches have to be represented in the trial, and each assessor has to taste product samples from more than one batch; otherwise batch differences and assessor differences in scoring are confounded.

Special care should be exercised in constructing the design when between-batch variation dominates between-product variation. Steinsholt (1998) gives examples of experiments in which batch-to-batch variation is present, where he is mainly concerned about how to adjust the simple ANOVA model for the special structure of products.

In the following chapters it will be assumed that all samples of a product are homogeneous.

2.5 Scale problems: Ordinal data and skewness

Products in sensory profiling experiments are scored on a scale from "no sensation felt" to "maximal sensation felt". Reference products are provided either as a general example of the attribute or as an anchor for the magnitude of a specific sensation. But not even extensive training can ensure that assessors use the scale identically. A linear interpretation of scores over the whole scale can always be questioned. The differences on the lower part of the scale are not necessarily comparable to those on the upper part of the scale and they can also be of different interest for the scientist.

If a categorical scale is used, attributes are often scored only in a small number of classes and highly skewed or long tailed distributions are common. With a lot of products some of the attributes may be relevant only for a subset of the products. For the other products no intensity is felt for these attributes and almost all scores will be equal or close to zero. Hence, assuming normality might lead to biased results. Skewed distributions occur also for data scored on continuous scales. Whenever normal theory methods are used the model assumptions should be checked, e.g. by inspecting the residuals.

Another possibility, especially for data from ordinal scales, is to use models specific to ordinal data, such as ordinal logistic regression. The ordinal fixed-effect model can be expanded to incorporate random assessor effects and the dispersion can also be modelled.

In logistic regression for ordinal variables, the cumulative response probabilities $P(Y_t < k)$ are modelled instead of the category probabilities $P(Y_t = k)$. Avery and Masters (1999), McCullagh and Nelder (1989) and Durier et al. (1997) use the logistic link for sensory data, which is the natural link for the binomial distribution. The simplest model of this form is the proportional odds model, which assumes parallel regression lines for the cumulative categories.

Avery and Masters (1999) propose to use the mean of scores from one assessor for a product instead of the single scores. This is in contrast to most of the other scientists who believe that information is lost by such averaging, because session and assessor differences can no longer be modelled. McCullagh and Nelder (1989) use the raw scores, acknowledging that they might be correlated for answers from one assessor. Therefore they take the estimated model probabilities as upper limits, as conclusions from a model with independent observations are conservative. Jones and Wang (2000) fit an ordinal logistic regression model to the raw scores, adjusting for the repeated scoring, and therefore possible correlations, by using generalised estimation equations (GEE).The exchangeable correlation structure was found to fit their data best; in this structure, every pair of scores is assumed to have the same correlation independent of their time-difference. Fahrmeier and Tutz (1994) use a fixed-effects cumulative logistic model with a random intercept for each assessor to adjust for the clustering.

Wilkinson and Yuksel (1997) model location and dispersion parameters jointly by alternately fitting a location and a dispersion model. This procedure is defined within the quasi-likelihood extension of the GLM framework. In such models not only the expectation of an observation, but also its variance depends linearly on some external explanatory variables.

A normality assumption for the Wilkinson and Yuksel model, as described in Section 2.1, implies variances independent of the mean score, no skewness of the scores and linear dependence of the scores from the co-variates. These assumptions are unlikely to be satisfied in the sensory context. Hence, their location model is extended for non-normal data by using a logistic link function. It ensures that the estimated values all lie within the range of the scale used for scoring. It is specified as follows:

$$M^{l}_{WY}: \qquad \log\left(\frac{\mathrm{E}[y_{atr}]}{\max - \mathrm{E}[y_{atr}]}\right) = \mu + \alpha_a + \tau_t + (\alpha\tau)_{at}$$

The variance of an observation for the corresponding binomial distribution is given by the variance function, $V(\pi) = \pi(1-\pi)$, with $\pi = \mathrm{E}\left[y_{atr}\right]/\mathrm{Max}$;

$$\mathrm{var}[y_{atr}] = \phi V\left(\mathrm{E}[y_{atr}]\right) = \frac{\mathrm{E}[y_{atr}]\left(\mathrm{Max} - \mathrm{E}[y_{atr}]\right)}{\mathrm{Max}}$$

where Max denotes the maximum possible value of the scale and ϕ the dispersion parameter, which is 1 for a Bernoulli distribution. This variance function relates to the assumption that the variance varies according to the proximity of the score to the extremes of the scale (decreasing toward the extremes).

For the dispersion model, the variance function associated with the Gamma distribution is used. The variance of $d_{(y)a}$ with $d_a^{(y)} = (y_{atr} - \mathrm{E}[y_{atr}])^2$ can then be expressed as

$$\mathrm{var}[d_{(y)a}] = \phi V(d_{(y)a}) = \frac{1}{\nu}\left(E(d_{(y)a})\right)^2$$

with ν being the precision parameter of the Gamma distribution $\Gamma(\mathrm{E}(d_{(y)a}), \nu)$.

The linear predictor for the dispersion model is chosen to reflect the dependence on the assessor

$$M_{WY}^{d}: \qquad \log\left(d_{(y)a}\right) = \gamma + \delta_a$$

The log function is used as the link function rather than the natural link of the gamma distribution, because it ensures positive estimate of the dispersion parameter.

For fitting one model, the parameters of the other are required. Therefore, an iterative see-saw algorithm between location and dispersion model is used, as described in Section 2.1 for the Wilkinson and Yuksel model assuming normality.

Erichsen (1998) expands on the idea of modelling location and dispersion and on the use of ordinal logistic regression. He proposes an ordinal threshold model with random effects in both the location and the dispersion models.

The underlying normal model has the following general form

$$M_E^l : \qquad y_{at} = \phi_a \left(x'_{at}\theta + z'_{at}u_a + \varepsilon_{at} \right)$$

with $\mathrm{var}(y_{at}) = \phi_a \left(1 + z'\Sigma z\right)$, where

y_{at} is the unobserved normal random variable underlying the observed values r_{at}
θ is the vector of fixed effects,
$u_a \sim N(0, \Sigma)$ is the vector of random effects for assessor a,
x_{at}, z_{at} are vectors of known explanatory variables corresponding to the fixed and random effects, and
$\varepsilon_{at} \sim N(0, 1)$ is a standardised residual error.

With the structure of model M_{N1} Erichsen's location model can be expressed as

$$M_E^l : \qquad y_{at} = \phi_a \left(\mu + \tau_t + \alpha_a + (\alpha\tau)_{at} \right)$$

where the assessor effect and the assessor-by-treatment interaction terms are random. The variances of the two random effects are denoted $\mathrm{var}(\alpha_a) = \sigma_\alpha^2$ and $\mathrm{var}\left((\alpha\tau)_{at}\right) = \sigma_{\alpha\tau}^2$. The scale parameters are assumed to be random and dependent on the assessor through the dispersion model.

$$M_E^d : \qquad \log(\phi_a^2) = \gamma + \delta_a$$

where
γ is a mean parameter and
$\delta_a \sim N(0, \sigma^2)$ is a normal variate varying over assessors.

In contrast to Wilkinson and Yuksel's model, the dispersion model is not estimated by fitting the deviance residuals, but by maximising the h-likelihood (hierarchical likelihood, see Lee and Nelder (1996)), as a function of γ, δ_a and σ^2.

The ordinal scores are assumed to represent classes of the underlying normal variable y_{at}:

$$r_{at} = k \qquad \Leftrightarrow \qquad \xi_{k-1} < y_{at} \leq \xi_k, \quad k = 1, \ldots, K$$

where

r_{at}	is the discrete observation of assessor a and product t with possible values between 1 and K,
k	the observed value of r_{at},
$\xi_1, \ldots, \xi_{K-1}$	are the unknown cut-points satisfying $\xi_0 = -\infty \leq \xi_1 \leq \cdots \leq \xi_K = \infty$.

Without loss of generality, ξ_1 and γ can be fixed at 0. The remaining (unknown) cut-points can be collected in the vector $\xi = (\xi_2, \ldots, \xi_{K-1})^t$.

2.6 Conclusions

This chapter discussed several univariate models that have been proposed for sensory data. These models attempt to describe the assessor impact on the data and yield unbiased estimators of product differences. Fixed- and mixed-effects models that assume normality are still most frequently used, due to their ease of interpretation and their speed in calculation, which is an important factor in the everyday use of the model. The list of attributes for which a separate model is fitted is usually fairly long, so the computing time for each model has to be reasonably short. With increasing computing power, generalised mixed models will become more attractive for everyday use if they can provide more insight into the product structure.

While some of the sophisticated models seem to explain the underlying structure more clearly, they can be firmly established only if their estimators reflect product differences more accurately and precisely than simpler models. It is tempting to use a model that includes all conceivable parameters, to reduce the bias of the estimates of the product differences. This is not useful if it is accompanied by a substantial increase in the sampling variation especially when assessor-specific variances are included. A more complex model does therefore not necessarily yield "better" results, especially for studies with few replications.

Chapter 3

Designs for sensory one-session trials

Statistical design of experiments is concerned with finding a design plan in which all the treatment comparisons of interest are estimated with maximum precision with a given minimum of resources. In this chapter the basic properties required for single-session sensory designs are explored and the appropriate statistical background on linear models and experimental design is reviewed. In the following chapters these results are extended to more complicated designs.

In sensory testing, the **treatments** are the different consumer **products** whose attributes are to be compared in the study. The number of observations in an experiment is determined by the number of products that are compared, the size of the panel and the product replication, which is the number of times each assessor tastes each product. Product replication is mostly determined by the available resources (time, financial and the like). This means that for sensory designs the design size is fixed and interest is in design plans with maximum precision of the product comparisons for a given product range and panel size and a few other constraints which will be explained in the following paragraphs.

One of the difficulties that can arise when designing experiments is heterogeneity of the experimental units on which the treatments are measured. In the sensory context, the **experimental unit** is an **assessor** tasting a number of samples. For homogeneous experimental units, treatments are usually assigned randomly to each unit, the simplest sort of a design plan. If some experimental units are more alike than others and variation between experimental units is expected to be as large as or even larger than treatment differences, similar units can be grouped together into **blocks** of homogeneous units. Then block designs are needed. Whilst in such a setting it is possible to control the heterogeneity between units, it complicates the construction of the design plan, since comparisons within blocks are now made with higher precision than comparisons between blocks. Thus, the assignment of treatments to blocks has an influence on the precision of the treatment comparisons.

Designs used in sensory profiling are usually set up as **repeated measurement designs** in which a number of different treatments are given to each experimental unit successively. Thus, each **experimental unit** forms a block of **observational units**. The observational unit in sensory experiments is **an assessor tasting a product sample**. Repeated measurement designs are used because assessors in a panel can be very heterogeneous in their perception and their scoring in spite of intensive training. If every assessor tastes every product at least once, product differences can be assessed through within-assessor comparisons, which are usually less variable than between-assessor comparisons.

If the block size in a block design is the same as the number of treatments, designs are called **complete block designs**. For such designs, all treatments occur in each block, so that each assessor tastes every product. Complete block designs are introduced in Section 3.1. A prerequisite for the use of such designs in the sensory context is that the number of products to be compared is small enough so that sensory fatigue is no concern. If sensory fatigue occurs an identical stimulus feels less strong on the palate than it feels earlier in the trial. To prevent sensory fatigue the experiment has to be subdivided into shorter sessions. Designs for experiments with more than one session are discussed from Chapter 4 onward.

If financial or time limitations prohibit assessors from tasting all products under consideration, a subset of the products can be given to each assessor instead. This means that the block size is smaller than the number of products, and therefore **incomplete block designs** have to be used (described in Section 3.2). For incomplete block designs, the number of times products are directly compared within a block depends on the chosen design plan. Product differences are then partly estimated through within-assessor comparisons and partly through between-assessor comparisons, where the latter has a larger variance than the former. If all treatment differences are of equal interest, it seems advantageous to have each product occur the same number of times within a block with all other products apart from itself, which would result in equal precision of the product comparisons. Within-block comparisons of the same treatment, on the other hand, do not give any information about treatment differences. These are useful only if interest is in the consistency of the attribute scores rather than in product differences.

Usually all assessors in a panel taste the same number of products. For all sensory design plans that we consider from now on, all **blocks** are **equi-sized** and all **products** are **equi-replicated**. The only exception are the designs discussed in Chapter 6, where a control product is allowed to have higher replication than all other products. In this and the following chapter, it is further assumed that there is no restriction on the availability of products, i.e., each product could be served to any number of assessors at any serving position. Designs for multi-session trials with a limit on the number of different products that can be served at any serving position are introduced in Chapter 5.

In sensory experiments there is often more than one blocking factor to consider. The most important blocking factor is the assessor, which will account for the largest part of the heterogeneity in the attribute scores. Since assessors are presented sequentially with the products in the trial, two other effects are often regarded to be important: **order** and **carry-over effects**.

Interest in sensory designs is mostly in all pairwise product comparisons for each attribute. Designs for which these contrasts are estimated with equal or almost equal precision are discussed in Chapters 3 to 5. More complicated contrasts are discussed in Chapters 6 and 7.

As mentioned before, sensory designs are usually equi-replicated, each assessor tastes all products, either once or a fixed number of times, which is the same for each assessor. Section 3.1 covers mainly designs in which each assessor tastes every product exactly once. Introduced are randomised complete block designs and row-complete row-column and cross-over designs. In Section 3.2 the incomplete block version of these three classes is reviewed, and efficiency bounds for them are given in Section 3.3. Construction methods are discussed in Section 3.4 for cyclic designs and in Section 3.5 for non-cyclic designs.

3.1 Complete block designs

In **complete block designs** (CBD) all treatments appear once within each block. Thus the block size equals the number of treatments. For sensory designs this means that an assessor tastes all products within a session, with assessors representing the blocks. They are also known as randomised complete block designs, since treatments are assigned randomly to each unit in the block.

The parameters for complete block designs are

a = number of **assessors** taking part in the trial = number of **blocks** = number of times a product is tested

p = number of **products** compared in the trial (number of **treatments**) = **block size**

The aim in using a design plan is to minimise the variance for the treatment comparisons of interest. The choice of a "good" design depends on the chosen **treatment contrasts** and on the appropriate **linear model** that describes the data. Suitable treatment contrasts and linear models for sensory trials are given in the next two sections and some characteristics for optimal designs are derived for these cases.

3.1.1 Three alternative linear models

Sensory trials generate high dimensional data. All assessors in a panel score each product on a list of attributes, which generates the sensory profile of a product. For each attribute a separate univariate linear model has to be fitted. Since each attribute is created from the same process, it is generally assumed that the same model can be fitted for each of the attributes. Due to this argument, one optimal design can be found for all attributes in a trial simultaneously. The parameter values of the model are, of course, assumed to differ between the attributes. All following models refer to a single attribute.

In Chapter 2 elaborate models for the univariate analysis of sensory trials have been introduced. Since optimal designs are model dependent, for the purpose of deriving and assessing designs in this thesis it has been decided to use simple fixed effects models. This has been done for two reasons. Firstly, these models are commonly used in experimental design theory and secondly, the more advanced models suggested in Chapter 2 are rarely used in practise. Three basic univariate models seem appropriate for univariate analysis of sensory profiling experiments. For all the following models the errors are assumed to be independently normally distributed with mean 0 and variance σ^2 ($\varepsilon \sim N(0, \sigma^2 I)$). In unreplicated CBD's, the number of observations for each model is *ap*.

For all sensory designs, **treatments** are equivalent to the different **products** that are tasted and for complete block designs, **blocks** represent **assessors**. The simplest model for CBD includes fixed treatment and block effects, or in sensory terms product and assessor effects.

$$Y_{jk} = \mu + \alpha_j + \tau_k + \epsilon_{jk} \quad 1 \le j \le a, \quad 1 \le k \le p \qquad (3.1)$$

with assessor effect α_j, and product effect τ_k. In matrix notation this model can be expressed as

$$Y = X\theta + \varepsilon = [1_{ap} \;\; X_A \;\; X_P] \begin{bmatrix} \mu \\ \alpha \\ \tau \end{bmatrix} + \varepsilon$$

where X_A is a binary design matrix referring to assessor effects and X_P the design matrix referring to product effects. Each column of the design matrix corresponds to a dummy variable from one of the effects and each row of the matrix belongs to an observation.

To illustrate the model structure, model equation (3.1) is shown for an example design for four products tasted by four assessors. Since each assessor tastes every product once, there are 16 observations. The model matrix includes one variable for the overall mean, four dummy variables for the assessor effect (one for each of the assessors) and four dummy variables for the product effect (one for each of the products).

Example 3.1 Model equation (3.1) for a randomised complete block design of four products tasted by four assessors.

The following design is a randomised complete block design. Each assessor is served the four products in a different (random) order. Properties of randomised block design are discussed in Section 3.1.3.

	Trial			
	Serving 1	*Serving 2*	*Serving 3*	*Serving 4*
Assessor 1	*2*	*3*	*1*	*4*
Assessor 2	*4*	*1*	*2*	*3*
Assessor 3	*2*	*3*	*1*	*4*
Assessor 4	*4*	*1*	*3*	*2*

Model equation (3.1) for the above design is:

$$\begin{bmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{21} \\ y_{22} \\ y_{23} \\ y_{24} \\ y_{31} \\ y_{32} \\ y_{33} \\ y_{34} \\ y_{41} \\ y_{42} \\ y_{43} \\ y_{44} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mu \\ \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{14} \\ \varepsilon_{21} \\ \varepsilon_{22} \\ \varepsilon_{23} \\ \varepsilon_{24} \\ \varepsilon_{31} \\ \varepsilon_{32} \\ \varepsilon_{33} \\ \varepsilon_{34} \\ \varepsilon_{41} \\ \varepsilon_{42} \\ \varepsilon_{43} \\ \varepsilon_{44} \end{bmatrix}$$

Apart from product and assessor effects, some sort of serving order effect is often assumed. If a general effect for serving order is included in the model, model (3.1) changes into model (3.2):

$$Y_{ijk} = \mu + \pi_i + \alpha_j + \tau_k + \epsilon_{ijk} \quad 1 \leq j \leq a, \quad 1 \leq i, k \leq p \tag{3.2}$$

with the notation as above and an additional period or serving order effect π_i. This means fitting one variable for each serving and it allows for differences in location for each serving position. If more is known about the type of order effect, a decreasing linear effect or any other function could be fitted instead. Often though there is not enough information about such a special structure of the order effect to fit a more specific function. The order effect included in any of the following models will be of the general kind as used in this model.

When order effects are expected, row-column designs should be used for an experiment. These are introduced in Section 3.1.4.

In matrix notation equation (3.2) is expressed as

$$Y = X\theta + \varepsilon = [1_{ap} \;\; X_T \;\; X_A \;\; X_P] \begin{bmatrix} \mu \\ \pi \\ \alpha \\ \tau \end{bmatrix} + \varepsilon,$$

where X_T is the design matrix referring to the order effects.

The model matrix for model (3.2) now includes additional columns, which identify the serving position a product is served in.

Example 3.2 Model matrix (3.2) for a row-column design for four assessors and four products.

In the following row-column design each assessor is served each of the four products in random order, with the constraint that each product is served in each serving position.

	Trial			
	Serving 1	Serving 2	Serving 3	Serving 4
Assessor 1	1	2	4	3
Assessor 2	3	1	2	4
Assessor 3	4	3	1	2
Assessor 4	2	4	3	1

Model matrix (3.2) for the above design is:

$$\begin{bmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{21} \\ y_{22} \\ y_{23} \\ y_{24} \\ y_{31} \\ y_{32} \\ y_{33} \\ y_{34} \\ y_{41} \\ y_{42} \\ y_{43} \\ y_{44} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mu \\ \\ \pi_1 \\ \pi_2 \\ \pi_3 \\ \pi_4 \\ \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{14} \\ \varepsilon_{21} \\ \varepsilon_{22} \\ \varepsilon_{23} \\ \varepsilon_{24} \\ \varepsilon_{31} \\ \varepsilon_{32} \\ \varepsilon_{33} \\ \varepsilon_{34} \\ \varepsilon_{41} \\ \varepsilon_{42} \\ \varepsilon_{43} \\ \varepsilon_{44} \end{bmatrix}$$

If additionally a first-order carry-over effect is included in the model, the model equation changes to model (3.3),

$$Y_{d(ij)} = \mu + \pi_i + \alpha_j + \tau_{d(i,j)} + \rho_{d(i-1,j)} + \epsilon_{ij} \qquad 1 \leq i \leq p, \quad 1 \leq j \leq a \qquad (3.3)$$

where d denotes a repeated measures design with $d(i,j)$ being the product tested in the ith serving position by assessor j and $\tau_{d(i,j)}$ denotes the product effect for the product that is tasted by assessor j in serving position i, and $\rho_{d(i-1,j)}$ stands for the residual effect of the product that is tasted by assessor j in serving position $i-1$. Carry-over effects are not defined for the first product that each assessor tastes. This would of course be different if assessors were given a product to taste in a pre-period that functions as a warm-up. For all following models it will be assumed that no pre-period is used.

For experiments in which carry-over effects are expected cross-over designs are used, which take account of the assumed order and carry-over effects. These are introduced in Section 3.1.5.

Model equation (3.3) is equivalent to the following matrix equation:

$$Y = X\theta + \varepsilon = [1_{ap} \;\; X_T \;\; X_A \;\; X_P \;\; X_R] \begin{bmatrix} \mu \\ \pi \\ \alpha \\ \tau \\ \rho \end{bmatrix} + \varepsilon$$

where X_R represents the design matrix of the carry-over effects, which are defined by the products served in the serving position before and are equal to zero for observations in the first serving position.

Example 3.3 Model matrix (3.3) for a cross-over design for four products tasted by four assessors.

The following design serves the four products to each assessor in a different order, each product is served in each serving position once and each product is served after each of the other product once.

	Trial			
	Serving 1	Serving 2	Serving 3	Serving 4
Assessor 1	1	4	2	3
Assessor 2	2	1	3	4
Assessor 3	3	2	4	1
Assessor 4	4	3	1	2

Model matrix (3.3) for the above design is:

$$\begin{bmatrix} y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{21} \\ y_{22} \\ y_{23} \\ y_{24} \\ y_{31} \\ y_{32} \\ y_{33} \\ y_{34} \\ y_{41} \\ y_{42} \\ y_{43} \\ y_{44} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mu \\ \pi_1 \\ \pi_2 \\ \pi_3 \\ \pi_4 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \\ \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \end{bmatrix} + \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{12} \\ \varepsilon_{13} \\ \varepsilon_{14} \\ \varepsilon_{21} \\ \varepsilon_{22} \\ \varepsilon_{23} \\ \varepsilon_{24} \\ \varepsilon_{31} \\ \varepsilon_{32} \\ \varepsilon_{33} \\ \varepsilon_{34} \\ \varepsilon_{41} \\ \varepsilon_{42} \\ \varepsilon_{43} \\ \varepsilon_{44} \end{bmatrix}$$

The estimator for product differences differs according to the assumptions made about existing nuisance effects, i.e. it depends on the model. For a set of products the most parsimonious model should be used that fits the data adequately. The smaller the number of parameters which has to be estimated the smaller is the estimated standard error for each parameter estimator. On the other hand, all existing nuisance effects should be included in the model, since the omission of relevant factors will give biased parameter estimators. For different sets of products the assumed model might vary. Some products are more likely to show carry-over effects than others and the same applies to order effects. The model equation should therefore always be chosen specifically for the set of products that are assessed in the trial.

In the next section it will be shown how the estimator for product differences and its variance differs for these three models (3.1)–(3.3).

3.1.2 Estimating product contrasts

The aim of a descriptive analysis in sensory testing is the comparison of certain attributes for a set of products. The interest is in finding out on which attributes the products differ and for which they are similar. For each attribute, products can then be divided into groups of similar intensities. Translated into design terminology, this means that for a selected attribute the interest is often in a **set of product contrasts** $L\tau$, where $L = [c_1 \dots c_n]'$ is a set of n different contrast vectors c_i, where $c_i = [c_{i1}, \dots, c_{ip}]$ with $\sum_{j=1}^{p} c_{ij} = 0$. A contrast that compares two products is called an **elementary contrast** or a **paired or pairwise comparison**. The elements of such a contrast vector c are all zero apart from one 1 and one -1. In sensory testing, the contrasts of interest are generally all $p(p-1)/2$ pairwise product comparisons.

During this and the following two chapters it will be assumed that the aim of the sensory experiment is to efficiently estimate all pairwise comparisons. In Chapters 6 and 7, other product contrasts are considered and methods given to find adequate designs for cases where some contrasts are more important than others.

Example 3.4 Six pairwise comparisons for four products.

$$L\tau = \begin{bmatrix} c_1' \\ c_2' \\ c_3' \\ c_4' \\ c_5' \\ c_6' \end{bmatrix} \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{pmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{pmatrix} = \begin{pmatrix} \tau_1 - \tau_2 \\ \tau_1 - \tau_3 \\ \tau_1 - \tau_4 \\ \tau_2 - \tau_3 \\ \tau_2 - \tau_4 \\ \tau_3 - \tau_4 \end{pmatrix}$$

For a given design, the variance of a paired product comparison $c'\tau$ can differ for the three linear models. The variance of each product contrast depends on the **model information matrix**, which will be introduced in the following section.

The parameters of interest in sensory testing are usually all product differences. While assessor differences in scoring are assumed to be present, their values are of minor interest.[1] The same holds for serving order and carry-over effects. These are the blocking effects in the model, the so-called nuisance effects. They are included in the model only to make the estimators of product differences unbiased. One method to find such unbiased estimators is by solving the reduced normal equations, which are introduced in the following section.

Reduced normal equations for product effects

The **normal equations** $(X'X)\hat{\theta} = X'Y$ of a linear model of the form $Y = X\theta + \varepsilon$ with $\varepsilon \sim N(0, \sigma^2 I)$ with more than one dependent variable included in the parameter vector θ can, in general, be expressed by partitioning the design matrix X into two sub-matrices, $X = [X_1 \;\; X_2]$, where X_1 is the part of the design matrix related to nuisance or blocking effects θ_1 and X_2 the part of X concerned with the effects of interest θ_2, e.g. the product effects.

$$\begin{array}{ccc} (X'X) & \hat{\theta} & = & X'Y \\ \begin{pmatrix} X_1'X_1 & X_1'X_2 \\ X_2'X_1 & X_2'X_2 \end{pmatrix} & \begin{pmatrix} \hat{\theta}_1 \\ \hat{\theta}_2 \end{pmatrix} & = & \begin{pmatrix} X_1'Y \\ X_2'Y \end{pmatrix} \end{array} \qquad (3.4)$$

[1] In contrast, in assessor evaluation studies assessor differences are of primary interest.

Example 3.5 Normal equations for model (3.1) with four products and four assessors.

$$\begin{bmatrix} 16 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \\ 4 & 4 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 4 & 0 & 4 & 0 & 0 & 1 & 1 & 1 & 1 \\ 4 & 0 & 0 & 4 & 0 & 1 & 1 & 1 & 1 \\ 4 & 0 & 0 & 0 & 4 & 1 & 1 & 1 & 1 \\ 4 & 1 & 1 & 1 & 1 & 4 & 0 & 0 & 0 \\ 4 & 1 & 1 & 1 & 1 & 0 & 4 & 0 & 0 \\ 4 & 1 & 1 & 1 & 1 & 0 & 0 & 4 & 0 \\ 4 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 4 \end{bmatrix} \begin{pmatrix} \hat{\mu} \\ \hat{\alpha}_1 \\ \hat{\alpha}_2 \\ \hat{\alpha}_3 \\ \hat{\alpha}_4 \\ \hat{\tau}_1 \\ \hat{\tau}_2 \\ \hat{\tau}_3 \\ \hat{\tau}_4 \end{pmatrix} = \begin{pmatrix} y_{++} \\ y_{1+} \\ y_{2+} \\ y_{3+} \\ y_{4+} \end{pmatrix}$$

where the subscript + indicates summation over the factor (e.g. $y_{++} = \sum_i \sum_j y_{ij}$ *and* $y_{i+} = \sum_j y_{ij}$*).*

The reduced normal equations can then be found by solving the top equation in (3.4) for $\hat{\theta}_1$. As $X'X$ is not of full rank, normally neither $X_1'X_1$ nor $X_2'X_2$ are necessarily of full rank. Thus, there is no single solution, but the following range of solutions for the parameter vector θ_1,

$$\tilde{\theta}_1 = (X_1'X_1)^- \left(X_1'Y - X_1'X_2\hat{\theta}_2\right) + \left[(X_1'X_1)^- (X_1'X_1) - I\right] z$$

where z is an arbitrary vector of appropriate size (for a derivation see Searle, 1971) and A^- denotes the generalised inverse of a matrix A (see B.1 in the appendix for the properties of a generalised inverse).

Example 3.6 Range of solutions for $\theta_1 = \begin{bmatrix} \mu \\ \alpha \end{bmatrix}$ for model (3.1) with four assessors and four products.

$$\begin{pmatrix} \tilde{\mu} \\ \tilde{\alpha}_1 \\ \tilde{\alpha}_2 \\ \tilde{\alpha}_3 \\ \tilde{\alpha}_4 \end{pmatrix} = \begin{pmatrix} & & 4\hat{\tau}_+ & - & z_1 \\ \frac{1}{4}y_{1+} & + & \hat{\tau}_+ & + & z_2 \\ \frac{1}{4}y_{2+} & + & \hat{\tau}_+ & + & z_3 \\ \frac{1}{4}y_{3+} & + & \hat{\tau}_+ & + & z_4 \\ \frac{1}{4}y_{4+} & + & \hat{\tau}_+ & + & z_5 \end{pmatrix}$$

Substituting the expression for $\tilde{\theta}_1$ into the second equation results in the **reduced normal equations**, which are independent of the special solution for $\tilde{\theta}_1$:

$$\begin{aligned}
& (X_2'X_1)\left((X_1'X_1)^{-}\left(X_1'Y - X_1'X_2\hat{\theta}_2\right)\right. \\
& \quad \left. + \left[(X_1'X_1)^{-}(X_1'X_1) - I\right] z\right) + X_2'X_2\hat{\theta}_2 &=&\; X_2'Y \\
\Leftrightarrow\; & (X_2'X_1)(X_1'X_1)^{-}\left(X_1'Y - X_1'X_2\hat{\theta}_2\right) + X_2'X_2\hat{\theta}_2 &=&\; X_2'Y \\
\Leftrightarrow\; & X_2'X_2\hat{\theta}_2 - X_2'X_1'(X_1'X_1)^{-} X_1'X_2\hat{\theta}_2 &=&\; X_2'Y - (X_2'X_1)(X_1'X_1)^{-} X_1'Y \\
\Leftrightarrow\; & X_2'\left(I - X_1(X_1'X_1)^{-} X_1'\right) X_2\hat{\theta}_2 &=&\; X_2'\left(I - X_1(X_1'X_1)^{-} X_1'\right) Y \\
\Leftrightarrow\; & X_2'P_1X_2\hat{\theta}_2 &=&\; X_2'P_1Y \\
\Leftrightarrow\; & C\hat{\theta}_2 &=&\; Q
\end{aligned} \tag{3.5}$$

where $P_1 = I - X_1(X_1'X_1)^{-} X_1'$ denotes the projection matrix. See Section B.2 in the appendix for the characteristics of a projection matrix.

Example 3.7 Reduced normal equations for product effects for model (3.1) with four assessors and four products.

$$\left(\begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} - \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}\right) \begin{pmatrix} \hat{\tau}_1 \\ \hat{\tau}_2 \\ \hat{\tau}_3 \\ \hat{\tau}_4 \end{pmatrix} = \begin{pmatrix} y_{+1} - \frac{1}{4}y_{++} \\ y_{+2} - \frac{1}{4}y_{++} \\ y_{+3} - \frac{1}{4}y_{++} \\ y_{+1} - \frac{1}{4}y_{++} \end{pmatrix}$$

The matrix $C = X_2'P_1X_2$ of the reduced normal equations (3.5) is called the information matrix. Its form depends on the linear model and on the specific design it is calculated from. To demonstrate its dependence on the chosen design, denoted with d, it is often written as C_d.

Definition 3.1 (Information matrix) *Let* $Y = X\theta + \varepsilon = [X_1 \;\; X_2]\,[\theta_1 \;\; \theta_2]'$ *with* $\varepsilon \sim N(0, \sigma^2 I)$, *where* X_2 *is the part of the design matrix referring to the parameters of interest,* θ_2. *Let* P_1 *denote the projection matrix* $P_1 = I - X_1(X_1'X_1)^{-}X_1'$. *The* $p \times p$ *coefficient matrix* C_d *of the reduced normal equations for* θ_2,

$$C_d = X_2'P_1X_2$$

is called the information matrix for the above linear model.

In over-parameterised linear models, such as models (3.1)–(3.3), the vector of product effects τ itself is not estimable. A solution of the normal equations will depend on the specific choice of the generalised inverse of the information matrix, C_d^-. But certain functions of $c'\tau$ are estimable.

Definition 3.2 (Estimable functions) *A parametric function $c'\theta$ is called an estimable function under a given model $Y = X\theta + \varepsilon$ if there exists a vector a such that $E(a'Y) = c'\theta$.*

It can be shown that the function $c'\theta$ is estimable if $c'(X'X)^- X'X = c'$. Estimators obtained for estimable functions are independent of the specific solution of the normal equations.

With the help of the reduced normal equations the general form of an unbiased estimator of an estimable function of product contrasts $c'\tau$ is

$$\widehat{c'\tau} = c'C_d^- Q_d \tag{3.6}$$

where τ refers to the product effect parameters and c denotes a contrast that satisfies the estimability constraints. Apart from the estimator itself, interest lies also in the variance of the estimator calculated under the given linear model and a design d.

$$\mathrm{var}(\widehat{c'\tau}) = \sigma^2\, c'C_d^- c \tag{3.7}$$

The specific form of C_d^- is often quite complicated, but for certain designs simple expressions are obtained. To describe these cases adequately, some terminology is introduced first.

Definition 3.3 (Orthogonal block designs) *A block design is called orthogonal if for each treatment i the proportion of units in a block that receives treatment i is the same for every block: $n_{ij} = \frac{N_{i.}N_{.j}}{N_{..}}$ for all combinations i, j, where n_{ij} denotes the number of units for each block-treatment combination (cell), $N_{i.}$ the number of units in each block, $N_{.j}$ the number of units for each treatment, and $N_{..}$ the number of units in the whole trial.*

Under an orthogonal design the estimators for the pairwise product contrasts have the same form as under a completely randomised design ($\widehat{c'\tau} = \sum_i \widehat{c_i\tau_i} = \sum_i c_i\bar{Y}_i$ for both cases, where $\bar{Y}_i$ is the mean of all scores for product i.).

A slightly less stringent characteristic for designs than orthogonality is variance balance, which characterises a design in which all elementary contrasts have equal variances.

Definition 3.4 (Variance balanced designs) *A block design d is variance balanced if $var_d(\widehat{\tau_i - \tau_j})$ does not depend on the choice of i and j, $i, j = 1, \ldots p$.*

With the general form of the reduced normal equations and the experimental design terms defined above, the information matrix can now be derived for models (3.1)–(3.3).

3.1.3 Randomised complete block designs

An important matrix in design of experiments is the incidence matrix of a design.

Definition 3.5 (Incidence matrix) *$N_A = X'_P X_A$ is called the block incidence matrix for a given block design d with design matrix X, where X_P denotes the part of the design matrix X referring to treatment effects and X_A the part of the design matrix referring to the blocking factor.*

The block incidence matrix N_A is a $p \times b$ matrix with a row for each product and a column for each block. Its element n_{ij} represents the number of times the ith treatment occurs in the jth block. Its importance in characterising and deriving good designs will be more obvious for incomplete block designs, which are discussed in Section 3.2. Since in complete block designs each treatment occurs exactly once in each block, all entries of the incidence matrix N_A are equal to one, ($N_A = J_{p,a} = 1_p 1'_a$, where J represents a $p \times a$ matrix and 1_p a p-vector, whose elements are all 1's).

The information matrix C_d for product effects under model (3.1) is

$$\begin{aligned} C_d &= X_2 P_1 X_2 = X_P' \left(I - X_A X_A'\right) X_P \\ &= X_P' X_P - X_P' X_A X_A' X_P = aI - N_A N_A' \end{aligned} \tag{3.8}$$

with the projection matrix $P_1 = I - X_A X_A'$ (see Section B.4 in the appendix for a derivation).

The information matrix C_d for CBD for model (3.1) is therefore

$$C_d = aI - \frac{a}{p} J_{p,p} = a(I - \frac{1}{p} J_{p,p}) \tag{3.9}$$

For this special matrix structure the generalised inverse has a simple form (see Solution (B.3) in the appendix).

$$C_d^- = \frac{1}{a} I \tag{3.10}$$

Thanks to the simple form of the generalised inverse of the information matrix the estimate for a treatment contrast for model (3.1) can also be expressed explicitly.

$$\widehat{c'\tau} = c' C_d^- Q_d = c'(\bar{Y}_P - \bar{Y}_{++} 1_p) \tag{3.11}$$

where $\bar{Y}_P = \left(\bar{Y}_1, \ldots, \bar{Y}_p\right)$ is the vector of product means and $\bar{Y}_{++}$ the overall mean of all observations.

The variance for the contrast's estimate $\widehat{c'\tau}$ is

$$\text{var}(\widehat{c'\tau}) = \sigma^2 c' C_d^- c = \frac{1}{a}\, \sigma^2\, c'c \tag{3.12}$$

The estimator of the contrast $c'\tau$ and its variance are the same as for a completely randomised design without blocking. That means that treatment effects in CBD are orthogonal to block effects. In other words, all CBD are orthogonal.

The variance of a pairwise comparison from a CBD is

$$\text{var}_d(\widehat{\tau_i - \tau_j}) = \frac{2}{a}\sigma^2, \qquad i, j = 1, \ldots, p$$

which is the same for all treatment pairs. Thus, all CBD are variance balanced.

No matter in what order the products are served to each assessor (order of products within blocks), all product contrasts will have the minimum variance. A design plan with complete blocks for model (3.1), created by randomising products within each block separately, is called a **randomised complete block design**. Randomisation is done at three different stages. First, a randomised complete block design of the appropriate size is created by assigning an independently drawn random list of all p product numbers to each of the a blocks. Second, product names are randomly assigned to the product numbers of the design plan. Third, assessors are randomly allocated to the blocks of the design plan.

3.1.4 Row-column designs

For model (3.2), which includes the additional factor of serving order, two blocking systems exist. Assessors represent the row-structure whilst serving position represents the column structure. With these two coordinates, the position of a product in a design for model (3.2) is completely determined. Changing the order of products within blocks in this model influences the column structure and therefore the period incidence matrix $N_T = X_P'X_T$. In the sensory context, periods are equivalent to serving positions.

The information matrix C_d for a row-column design for model (3.2) with complete rows has the following form

$$C_d = aI_p - \frac{1}{a}N_T N_T' \tag{3.13}$$

Thus assessor effects are orthogonal to product effects.

For the derivation of the information matrix (3.13), see Section B.4 in the appendix. Its precise form depends on the period incidence matrix $N_T = X_P'X_T$. Each element n_{ij}, $i = 1 \ldots a$, $j = 1 \ldots p$, of this matrix denotes the number of times product i occurs in serving position j. Until now, only complete block designs have been covered, where the block size of the rows equals p, the number of treatments. The following definition is for blocks of any size, where the block size is denoted by t. Its extension to other block sizes will be used in later sections.

Definition 3.6 (Uniform designs) *A row-column design d for p treatments in a blocks of size t is said to be*

uniform on the units *if* $n_{ij} = \frac{t}{p}$ *for all* $1 \leq i \leq p, 1 \leq j \leq a$, *where* n_{ij} *is the (i,j)'s element of* N_T

uniform on the periods *if* $n_{ik} = \frac{a}{p}$ *for all* $1 \leq i \leq p, 1 \leq k \leq p$, *where* n_{ik} *is the (i,k)'s element of* N_A

uniform *if it is uniform on units and periods.*

Sensory designs are generally uniform on units; each assessor tastes all products. They can also be uniform on periods (serving positions) if the number of assessors a is a multiple of the number of products p. To be uniform on periods, each product has to occur the same number of times in each period. If model (3.1) is assumed, i.e. no serving order effect, uniformity on periods does not improve the design. When periods are included in the model equation, as they are in models (3.2) and (3.3), uniformity on periods becomes a useful characteristic.

Uniform complete block designs

A prerequisite for the existence of uniform complete block designs is that the number of assessors is a multiple of the number of products $a = pr_a$, where r_a is an integer. All elements of the period incidence matrix N_T then have the value r_a, i.e. $N_T = r_a J_{p,p}$. For uniform complete block designs all products are tasted by every assessor once and in every serving position r_a times. For uniform complete block designs, the information matrix (3.13) simplifies to

$$\begin{aligned} C_d &= aI - \frac{1}{a}N_T N_T' = aI - \frac{r_a r_a}{a} J_{p,p} J_{p,p} \\ &= aI - r_a J_{p,p} = a\left(I_p - \frac{1}{p} J_{p,p}\right) \end{aligned} \tag{3.14}$$

which is the same as information matrix (3.9). Thus, the estimator and its variance for a uniform design under model (3.2) are also the same as for a design uniform on assessors under model (3.1). For designs that are not uniform on periods the variance for product contrast will differ for the two models and depend on the specific form of the period incident matrix N_T.

Thus, all uniform complete block designs are variance balanced and product effects are orthogonal to both blocking factors. Design plans for uniform complete block designs, where the number of assessors equals the number of products, i.e. $r_a = 1$, are called Latin squares.

Definition 3.7 (Latin squares) *A Latin square is an arrangement of m symbols in a $p \times p$ array such that each symbol occurs once in each row and once in each column.*

Two squares are **pairwise orthogonal** *if, when one square is superimposed on the other, each symbol of one square occurs once with each other label of the other square.*

Three or more squares are **mutually orthogonal** *if any two of them are pairwise orthogonal. For some values of p there is a* **complete set** *of $p-1$ mutually orthogonal Latin squares.*

Selected Latin square design plans for 3 to 12 products are given in Cochran and Cox (1957). Mutually orthogonal Latin squares are given in Abel et al. (1996) for $p \leq 10$ and construction methods are given for squares with $p \leq 56$. Wakeling and

MacFie (1995) have written a macro based on the SAS® procedure PROC FACTEX to create a set of mutually orthogonal Latin squares. They suggest Latin squares for consumer studies, where the number of tasters is often far larger than in sensory profiling with a panel of trained assessors.

The randomisation procedure for creating Latin square design plans is more difficult than for randomised complete block designs. First a Latin square of the adequate size has to be chosen randomly from the list of possible Latin squares. Then assessors are assigned randomly to blocks and products to product numbers.

Latin squares can be used only when the number of assessors is equal to the number of products or is a multiple of the number of products. In the first case one Latin square is used, in the second case multiple Latin squares are joined so that the number of rows equals the number of assessors. If a design is needed for a number of assessors that is not a multiple of the number of treatments, one possibility is the use of parts of Latin squares. This will most likely result in the loss of uniformity on periods. The exception are the cases in which Youden squares can be used.

Definition 3.8 (Youden square) *Row-column designs in which the number of columns is equal to the number of treatments, and in which the treatments are balanced for their occurrence in columns, are called Youden squares.*

Let x be the number of assessors a reduced modulo p ($x = a$ modulo p). If a balanced incomplete block design for p products in p blocks of size x exists, Youden squares can be created and combined with Latin squares to create orthogonal designs for a assessors and p products. For the definition of balanced incomplete block designs see Definition 3.12 in Section 3.2. Youden squares are row-orthogonal designs, in which the column structure consists of a complete block design and the row structure of an incomplete block design.

In the standard notation for block designs, blocks are equivalent to columns. For sensory designs, it is more natural to write assessors as rows and serving positions as columns. For this reason, transposed Youden squares have to be used to add the necessary blocks to the Latin square.

Example 3.8 CBD created from a Latin square and a Youden square.

A row-column design for seven products and ten assessors

Assessor 1	1	2	3	4	5	6	7	Latin square
Assessor 2	2	3	4	5	6	7	1	
Assessor 3	3	4	5	6	7	1	2	
Assessor 4	4	5	6	7	1	2	3	
Assessor 5	5	6	7	1	2	3	4	
Assessor 6	6	7	1	2	3	4	5	
Assessor 7	7	1	2	3	4	5	6	
Assessor 8	7	1	2	3	4	5	6	Youden square
Assessor 9	1	2	3	4	5	6	7	
Assessor 10	3	4	5	6	7	1	2	

The assessor concurrence matrices for the Latin square, the Youden square and the combined design are $N_{A;LS}N'_{A;LS} = 7J_{7,7}$, $N_{A;YS}N'_{A;YS} = 3J_{7,7}$ and $N_AN'_A = 10J_{7,7}$ respectively, since all rows are complete blocks. The serving order concurrence matrices are $N_{T;LS}N'_{T;LS} = 7J_{7,7}$ for the Latin square, where the column is also a complete block and $N_{T;YS}N'_{T;YS} = 2I + J_{7,7}$ for the Youden square, where the columns represent a balanced incomplete block design. This results in $N_TN'_T = 2I + 14J_{7,7}$ for the complete design.

Youden squares are not squares contrary to their name. Their number of rows is not equal to the number of columns, but they are incomplete Latin square designs. An extensive list of Youden squares can be found in Cochran and Cox (1957).

If no Youden square exists, non-orthogonal row-column designs can be constructed which are as nearly balanced as possible for products in rows and columns (partial balance) with the help of computer algorithms (see Section 3.5).

3.1.5 Cross-over designs

In model (3.3), additionally to product, assessor and serving-order effects, first-order residual or carry-over effects are assumed to have an influence on the assessment of the products. This is the effect that the product tasted in the serving position before exerts on the score of the following product. The projection matrix P_1, which is part of the information matrix C_d, is the same for the cross-over model (3.3) and for the row-column model (3.2). But the parameters of interest in the cross-over model incorporate a vector of direct treatment effects and a vector of carry-over treatment effects. The reduced normal equations can therefore be split again, into equations for direct product effects and equations for carry-over effects.

The reduced normal equations for direct product and carry-over effects

If the parameter vector θ_2 consists of a vector τ for direct treatment effects and a vector ρ for carry-over effects $\left(\theta_2 = \binom{\tau}{\rho}\right)$ and $X_2 = [X_P X_R]$ denotes the design matrix for direct treatment and carry-over effects, then the reduced normal equations for the direct and carry-over effects can be estimated from the normal equations according to the following equation, where $\hat{\theta}_2$ denotes the estimator for θ_2:

$$\begin{aligned}
& & C\hat{\theta}_2 &= Q \\
&\Leftrightarrow & X_2'P_1X_2 \begin{bmatrix} \hat{\tau} \\ \hat{\rho} \end{bmatrix} &= Q \\
&\Leftrightarrow & [X_P X_R]' P_1 [X_P X_R] \begin{bmatrix} \hat{\tau} \\ \hat{\rho} \end{bmatrix} &= [X_P X_R]' P_1 Y \\
&\Leftrightarrow & \begin{bmatrix} X_P'P_1X_P & X_P'P_1X_R \\ X_R'P_1X_P & X_R'P_1X_R \end{bmatrix} \begin{bmatrix} \hat{\tau} \\ \hat{\rho} \end{bmatrix} &= \begin{bmatrix} X_P'P_1Y \\ X_R'P_1Y \end{bmatrix} \\
&\Leftrightarrow & \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \hat{\tau} \\ \hat{\rho} \end{bmatrix} &= \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}
\end{aligned}$$

with

$$\begin{aligned}
C_{11} &= X_P'P_1X_P \\
C_{12} &= X_P'P_1X_R = C_{21}' \\
C_{22} &= X_R'P_1X_R \\
Q_1 &= X_P'P_1Y \\
Q_2 &= X_R'P_1Y
\end{aligned}$$

The reduced normal equations for the direct treatment and for the carry-over effects can then be expressed as

$$\begin{aligned} C_D\hat{\tau} &= Q_D \\ C_R\hat{\rho} &= Q_R \end{aligned} \qquad (3.15)$$

where

$$\begin{aligned}
C_D &= C_{11} - C_{12}C_{22}^{-}C_{21} &=& X'_R P_1 \left[I - X_P \left(X'_P P_1 X_P\right)^{-} X'_P\right] P_1 X_R \\
C_R &= C_{22} - C_{21}C_{11}^{-}C_{12} &=& X'_P P_1 \left[I - X_R \left(X'_R P_1 X_R\right)^{-} X'_R\right] P_1 X_P \\
Q_D &= Q_1 - C_{12}C_{22}^{-}Q_2 &=& X'_R P_1 \left[I - X_P \left(X'_P P_1 X_P\right)^{-} X'_P\right] P_1 Y \\
Q_R &= Q_2 - C_{21}C_{11}^{-}Q_1 &=& X'_P P_1 \left[I - X_R \left(X'_R P_1 X_R\right)^{-} X'_R\right] P_1 Y
\end{aligned}$$

Reduced normal equations for complete block cross-over designs

Since assessors represent complete blocks, all elements of the assessor incidence matrix N_A are equal to one (i.e. $N_A = J_{a,a}$). For the complete block cross-over design for model (3.3), the projection matrix P_1 is the same as for model (3.2). That means that the matrix C_{11} is the same as the information matrix for product effects for model (3.2), information matrix (3.13).

$$C_{11} = aI_p - \frac{1}{a}N_T N'_T$$

For complete block cross-over designs, for which $N_A = J_{p,a}$, the matrices C_{12} and C_{22} have the following form:

$$\begin{aligned}
C_{12} &= M - \frac{1}{a}N_T\tilde{N}'_T - \frac{1}{p}N_A\tilde{N}'_A + \frac{1}{p}1_p\boldsymbol{r}'_R \\
&= M - \frac{1}{a}N_T\tilde{N}'_T - \frac{1}{p}J_{p,a}X_T X'_R + \frac{1}{p}1_p\boldsymbol{r}'_R \\
&= M - \frac{1}{a}N_T\tilde{N}'_T - \frac{1}{p}J_{p,ap}X'_R + \frac{1}{p}1_p\boldsymbol{r}'_R \\
&= M - \frac{1}{a}N_T\tilde{N}'_T - \frac{1}{p}1_p\boldsymbol{r}'_R + \frac{1}{p}1_p\boldsymbol{r}'_R \\
&= M - \frac{1}{a}N_T\tilde{N}'_T
\end{aligned}$$

$$C_{22} = \boldsymbol{r}_R^{\delta} - \frac{1}{a}\tilde{N}_T\tilde{N}_T' - \frac{1}{p}\tilde{N}_A\tilde{N}_A' + \frac{1}{ap}\boldsymbol{r}_R\boldsymbol{r}_R'$$

where $M = X_P'X_R$, $\tilde{N}_T = X_R X_T'$, $\tilde{N}_A = X_R X_A'$, $\boldsymbol{r}_R = X_R' 1_p$ and $\boldsymbol{r}_R^{\delta}$ the diagonal matrix with the vector $\boldsymbol{r}_R$ on the diagonal.

The information matrices for direct treatment and residual effects C_D and C_R have only a closed form if the respective matrices C_{22} and C_{11} do.

A generalised inverse C_{11}has a simple form when the design is also uniform on periods (i.e. $N_T = \frac{a}{p}J_{p,p} = r_a J_{p,p}$). For **uniform complete block cross-over design** the matrices C_{11}, C_{12} and C_R simplify to:

$$\begin{aligned} C_{11} &= aI_p - \frac{a}{p}J_{p,p} \\ C_{12} &= M - \frac{1}{a}N_T\tilde{N}_T' \\ &= M - \frac{1}{p}J_{p,p}\tilde{N}_T' \\ C_R &= C_{22} - \frac{1}{p}C_{21}C_{12} \end{aligned}$$

The exact form of the generalised inverse of C_{22}, which is necessary to calculate the information matrix C_D for direct product effects, depends on the residual incidence matrices $\tilde{N}_T$ and $\tilde{N}_A$ and on $\boldsymbol{r}_R$, the replication vector of the carry-over effects.

Definition 3.9 (Balanced repeated measurement design) *A repeated measurement design d is said to be* **balanced** *if the treatment-residual incidence matrix $M = X_P'X_R$ is of the form*

$$M = \frac{a(t-1)}{p(p-1)}(J_{p,p} - I_p) = r_a\frac{t-1}{p-1}\left(J_{p,p} - I_p\right)$$

It is said to be strongly balanced if

$$M = \frac{a(t-1)}{p^2}J_{p,p} = r_a\frac{t-1}{p}J_{p,p}$$

In strongly balanced repeated measure designs, each treatment is preceded by every treatment the same number of times, whilst in balanced repeated measurement designs this is relaxed to every treatment preceding every other treatment the same number of times apart from itself.

For complete block designs with $t = p$ the matrix M simplifies to

$$M = \frac{a}{p}(J_{p,p} - I_p)$$

The replication vector for the carry-over effects $\boldsymbol{r}_R$ and the residual period incidence matrix $\tilde{N}_T$ also simplify for balanced repeated measurement designs t:

$$\begin{aligned} \boldsymbol{r}_R &= X_R' 1_{ap} = (a-1)1_p \\ \tilde{N}_T &= \begin{bmatrix} p & 0_{1,p} \\ 0_{p,1} & J_{p-1,p-1} \end{bmatrix} \end{aligned}$$

A balanced repeated measurement design that is used frequently for sensory experiments is the Williams Latin square designs (Williams, 1949). **Williams Latin square designs** are Latin square designs in which every treatment occurs the same number of times after every other treatment excluding itself, i.e. they are balanced repeated measurement designs with $t = p$. For the residual-treatment incidence matrix for complete block designs, this means that $M = \frac{a}{p}(J_{p,p} - I_p)$, i.e. every product is followed $r_a = \frac{a}{p}$ times by every other product. For a single square this means every product is followed exactly once by every other product, since $a = p$. The construction of Williams Latin squares is explained in Section 3.4.2. These designs have been recommended for the use in sensory profiling for example by Hunter (1996), Durier et al. (1997) and Schlich (1993). A bibliography for cross-over designs in the sensory context is given by Jones and Deppe (2001).

Williams designs can be used to generate complete block cross-over designs if the number of assessors is equal to or a multiple of the number of products. If the number of assessors is not a multiple of the number of products, a subset of blocks from a Williams designs can be chosen. These designs are no longer balanced for carry-over nor uniform on periods, but they are nearly balanced for order and carry-over effects. Schlich (1993) suggests to choose blocks randomly from the square, Russell (1991) suggests to select the rows that maximise the average efficiency of the reduced Williams designs (see Definition 3.16 for the definition of average efficiency of block designs).

3.1.6 Replicated complete block designs

In experimental design literature, replication denotes usually the number of times each treatment occurs in the design. For sensory designs this means the number of times a product is tasted over the whole trial. For complete block designs this number equals a, the number of assessors, since each assessor tastes each product once. Since for sensory designs it is generally demanded that each assessor tastes all products, replication in sensory design literature often refers to the number of times each product is tasted by a single assessor or, in other words, the number of times the whole trial is replicated. Therefore different notation is introduced for these two kinds of replication.

r = number of **replicates**,
the number of times each assessor tastes all p products, where one replicate of the experiment refers to all assessors tasting every product once.

r_p = **product replication**,
the number of times each product is tasted in the trial, ar.

Since they are the experimental units, assessors are not regarded as replicating the experiments. Having more assessors will improve the precision of the product means, but replication of the experiment will give information on the consistency of the product scores given by the assessors. Hunter (1996) recommends $r = 3$ as sufficient to gain information on the assessor-replication interaction.

The important parameters for replicated complete block designs are

a = number of **assessors** taking part in the trial

= number of **blocks**

= number of times a product is tested

p = number of **products** compared in the trial (number of **treatments**)

= **block size**

r = number of **replicates.**

For complete block designs the number of replicates is 1 (i.e. $r = 1$) and each product is replicated a times (i.e. $r_p = a$).

If sensory fatigue is not an issue it might be possible to have all rp products tasted within one session. If the rp tastings are too many for one session, the most likely split would be to have each replication in a single session, so that the whole trial is made out of $s = r$ sessions. Design plans for both strategies are described in the following section.

Replication within one session

If products are compared more than once by each assessor and it is not necessary to do this over multiple sessions, replication can be done in single-session designs. Here again, a block is represented by an assessor, but the block size is rp. If the number of assessors equals the number of products, Latin rectangles can be used. Latin rectangles are designs with rp columns and p rows (Hinkelmann and Kempthorne, 1994), which are generated from a combination of r independently randomised Latin squares. [2] The advantage of such a construction is that each replication could be analysed separately, in case something goes wrong in one of the replications or the trial is stopped prematurely. It ensures that after p tastings, each product has

[2] Some authors use the name Latin rectangle for Youden squares, which are different to the designs needed for replicated complete block designs.

been compared by every assessor. The disadvantage of such a construction for the cross-over model is that carry-over effects between replications are not taken into consideration and the resulting design is therefore not necessarily balanced for residual effects, even if the single complete block designs are balanced for residual effects.

The model equations for the replicated form within a session are essentially the same as for the unreplicated form for models (3.1)–(3.3), only the indices vary. There are now arp observations and each of the assessors tastes rp products.

Replication over multiple sessions

A more likely scenario is that each replication is done in a separate session, where each replication consists of one complete block design. In such a model, session and replication describe the same effects. The model equation for such a sensory trial is increased by the session term. It can be derived as a special case of the resolvable designs described in Chapter 4.

Carry-over and order effects are assumed to exist only within a session. Balance for residual effects will therefore hold for the replicated design, when it holds for every session separately.

3.2 Incomplete block designs

We argued in Section 3.1 that it is advantageous in sensory experiments to let every assessor taste every product. Sometimes the number of products is so large or take so long to assess that it is impossible for one assessor to assess all products, so that each assessor is left to taste only a subset of the products. Limitations of product resources and time can also create this undesirable situation. Designs for this case are called **incomplete block designs** (IBD). In these designs the block size (t, the number of servings per assessor) is smaller than the number of products. i.e. $t < p$. Mead and Gay (1995) and Cochran and Cox (1957) give examples of incomplete block designs in sensory studies. Hunter (1996) discourages their use in sensory studies. When incomplete block designs are used for profiling the number of assessors has to be fairly large. For consumer studies on the other hand, incomplete block designs can be very useful, since consumers can only taste a small number of products without sensory fatigue. In these studies, a large number of consumers will be asked to assess a small number of products. Even though incomplete block designs are rarely used in sensory studies, they are the basic constructs for some classes of larger designs. Resolvable block designs, introduced in Chapter 4, are used when the trial is split into sessions and each session represents an incomplete block design, but over the whole trial an assessor still tastes all products. In Chapter 5, resolvable designs with additional constraints are constructed by combining incomplete block designs with incomplete cross-over designs. For the construction of these two groups of designs, knowledge about incomplete block designs is necessary. The properties of incomplete block designs are studied in this section.

The important parameters for equi-replicated incomplete block designs, in which all blocks have the same size, are

a = number of **assessors** taking part in the trial = number of **blocks**

p = number of **products** compared in the trial (number of **treatments**)

t = number of **products** compared by one assessor = **block size**

r_p = number of times each product is tested in the trial

Since every assessor tastes only a subset of all p products, the replication term r, introduced for complete block designs, does not make sense. Replication in incomplete block designs is the number of times each product is tasted, and it is equal to the number of assessors that taste the product. It is $r_p = at/p$, and it has to be an integer for equi-replicated designs, which are generally used for sensory and consumer studies. For incomplete block designs the block size is smaller than the number of products, $t < p$, and it is assumed that products do not occur twice in the same block; they occur either once in a block or not at all. IBD of this sort are called binary designs.

Definition 3.10 (Binary block designs) *An IBD is called binary if the block incidence matrix N_A is binary, i.e. all its elements are either zero or one.*

An element n_{ij} of a binary assessor incident matrix N_A is equal to one if product i is tasted by assessor j and zero otherwise. For equi-replicated binary designs with equal block sizes, $\sum_i n_{ij} = r_i = r_p$ and $\sum_j n_{ij} = t_i = t$. For the symmetric concurrence matrix $\Lambda_A = N_A N_A'$ it follows that $\sum_i \lambda_{ij} = \sum_i \sum_k n_{ik} n_{kj} = \sum_k n_{kj} \sum_i n_{ik} = t \sum_k n_{kj} = r_p t$.

Whilst for complete block designs all product differences are always estimable since every product occurs in every block, this is not necessarily the case for incomplete block designs.

Definition 3.11 (Connected and disconnected designs) *A design d is* **disconnected** *if the treatments can be split into groups in such a way that no treatment from one group occurs in any block with any treatment from another group. A design which is not disconnected is said to be* **connected**.

An essential feature of any incomplete block design is therefore that it is connected. Only for connected designs are all treatment comparisons estimable. The rank of the information matrix for connected designs is rank$(C) = p - 1$, while for disconnected designs it is smaller (John and Williams, 1995).

3.2.1 Randomised incomplete block designs

The model equation for a randomised incomplete block design is the same as for a randomised complete block design for model (3.1).

$$Y_{jk} = \mu + \alpha_j + \tau_j + \epsilon_{jk} \quad 1 \leq j \leq a, \quad 1 \leq k \leq p \tag{3.16}$$

The difference in the information matrix is due to the block incidence matrix N_A.

$$C_d = r_p I_p - \frac{1}{t} N_A N'_A = r_p I_p - \frac{1}{t} \Lambda_A \tag{3.17}$$

While for complete block designs $N_A = J_{a,p}$, for incomplete block designs this is a binary matrix. The matrix element n_{ij} of N_A is equal to one if assessor j tastes product i and zero otherwise. Whilst in CBD each product is tasted a times, once by each assessor, in IBD each product is tasted $r_p = \frac{at}{p}$ times. This is also reflected in the information matrix.

When all treatments are of equal interest an attractive characteristic is **balance**, which means that each pair of treatments occurs together in a block the same number of times.

Definition 3.12 (Balanced incomplete block design (BIB design)) *A balanced incomplete block design is an equi-replicated binary IBD for which all blocks have the same size and in which every pair of products occurs together in exactly λ blocks.*

For BIB designs all pairwise comparisons are made with the same accuracy, which means they are variance balanced. The parameters of a BIB design satisfy the following two relationships:

$$\begin{aligned} at &= pr_p \\ r_p(t-1) &= \lambda(p-1) \end{aligned} \tag{3.18}$$

A BIB design can only exist if equations (3.18) are satisfied and all parameters are integers. Even when both equations are satisfied, a suitable BIB design may not exist. It is therefore not always possible to find a BIB design of the required size for a specific sensory experiment. All complete block designs are balanced by definition, since every product occurs in every block. Lists of BIB designs are given, for example, in Cochran and Cox (1957), Ragharavao (1971) and Mathon and Rosa (1996).

Definition 3.13 (Complement) *The complement of an equi-replicated and equi-sized IBD for p treatments with a blocks of size t is an IBD for p treatments and a blocks of size $p - t$, in which each block of the complementary design includes the treatments that are not included in the block of the base design.*

Let N_A be the incidence matrix of an incomplete block design $d(p, a, t)$, a design for p products in a blocks of size t. The incidence matrix of the complementary design of d, $d_c(p, a, p - t)$ is $N_A^C = J_{p,a} - N_A$, since each of the a blocks is a complement of the blocks in d. The complementary design of a balanced incomplete design is therefore also an IBD (John and Williams, 1995).

Definition 3.14 (Symmetric block designs) *An equi-replicated and equi-sized IBD is called symmetric if the number of treatments equals the number of blocks.*

For sensory designs symmetry means that $a = p$ and $t = r_p$.

For model (3.16), the information matrix for product effects depends on the block incidence matrix N_A or rather on the matrix $\Lambda_A = N_A N_A'$. This is the so called **concurrence matrix**. The diagonal elements of the concurrence matrix λ_{ii} give the number of times product i occurs in the design. Its elements on the off-diagonal $\lambda_{ij}, i \neq j$, give the number of blocks in which products i and j occur together. For BIB designs the concurrence matrix has the special form

$$N_A N_A' = \lambda J_{p,p} + (r_p - \lambda) I_p, \text{ with } \lambda = \frac{r_p(t-1)}{p-1}$$

Every product is tasted λ times with every other product, apart from itself, by an assessor and every product is replicated r_p times. Thus, the information matrix for a balanced design simplifies to

$$C_d = \frac{\lambda p}{t}\left(I_p - \frac{1}{p} J_{p,p}\right)$$

and one of its generalised inverses is

$$C_d^- = \frac{t}{\lambda p} I_p$$

Example 3.9 BIB design example for seven treatments in seven blocks of size three.

$$\begin{aligned} p &= a \\ r_p &= \frac{at}{p} = t = 3 \\ \lambda &= \frac{r_p(t-1)}{p-1} = \frac{3 \cdot 2}{6} = 1 \end{aligned}$$

The product replication for seven treatments in seven blocks of size three is three, hence this is a symmetric design. It is balanced since each product occurs once with every other product in a block.

A BIB design for 7 treatments in 7 blocks of size 3 with its block incidence matrix N_A and its concurrence matrix Λ_A is given by

$$d(7,7,3,3) = \begin{bmatrix} 1 & 2 & 4 \\ 2 & 3 & 5 \\ 3 & 4 & 6 \\ 4 & 5 & 7 \\ 5 & 6 & 1 \\ 6 & 7 & 2 \\ 7 & 1 & 3 \end{bmatrix} \qquad N_A = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

$$\begin{aligned} \Lambda_A &= N_A N_A' = J_{7,7} + (3-1)I_7 \\ C_d &= r_p I_p - \frac{1}{t} N_A N_A' = 3I_7 - \frac{1}{3}\left(J_{7,7} + 2I_7\right) = \frac{1}{3}\left(7I_7 - 2J_{7,7}\right) \\ C_d^- &= \frac{t}{\lambda p} I_p = \frac{3}{7} I_7 \end{aligned}$$

BIB designs exist only for a limited number of combinations of p, a and t. This will rarely fit the parameters of the sensory design that are required for an experiment. While BIB designs provide estimators for product differences which have the same precision for all product comparisons, it is rather more important to find a design for the required parameters than adjusting the trial to fit a BIB design. If all pairwise product comparisons are equally important, then the relative precision of treatment comparisons is determined primarily by the number of joint occurrences of the treatments within blocks. Therefore the pairwise occurrence of treatments should be as even as possible. These numbers are given through the elements of the concurrence matrix $\Lambda_A = N_A N'_A$. A design in which the concurrence matrix consists of two different elements is a partially balanced incomplete block design with two associate classes.

Definition 3.15 (Partially balanced incomplete block design (PBIBD)) *An equi-replicated binary incomplete block design for p treatments in a blocks of size t is said to be partially balanced with two associate classes if a relation of association can be established between any two treatments satisfying the following requirements.*

a) Two treatments are either first associates or second associates.

b) Each treatment has exactly n_i associates (i=1,2).

c) Given any two treatments that are ith associates, the number of treatments common to the jth associate of the first and the kth associate of the second is p^i_{jk} and is independent of the pair of treatments we start with. Also $p^i_{kj} = p^i_{jk}, (i, j, k = 1, 2)$.

Two treatments which are ith associates occur together in exactly λ_i blocks $(i = 1, 2)$.

Regular graph designs are a special class of partially balanced designs with two associate classes which satisfy the additional constraint $\lambda_2 = \lambda_1 + 1$.

Example 3.10 Regular graph design for eight treatments in blocks of size four.

$$d(8,8,4,4) = \begin{bmatrix} 1 & 2 & 3 & 5 \\ 2 & 3 & 4 & 6 \\ 3 & 4 & 5 & 7 \\ 4 & 5 & 6 & 8 \\ 5 & 6 & 7 & 1 \\ 6 & 7 & 8 & 2 \\ 7 & 8 & 1 & 3 \\ 8 & 1 & 2 & 4 \end{bmatrix}$$

$$N_A = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \Lambda_A = \begin{bmatrix} 4 & 2 & 2 & 1 & 2 & 1 & 2 & 2 \\ 2 & 4 & 2 & 2 & 1 & 2 & 1 & 2 \\ 2 & 2 & 4 & 2 & 2 & 1 & 2 & 1 \\ 1 & 2 & 2 & 4 & 2 & 2 & 1 & 2 \\ 2 & 1 & 2 & 2 & 4 & 2 & 2 & 1 \\ 1 & 2 & 1 & 2 & 2 & 4 & 2 & 2 \\ 2 & 1 & 2 & 1 & 2 & 2 & 4 & 2 \\ 2 & 2 & 1 & 2 & 1 & 2 & 2 & 4 \end{bmatrix}$$

Each product occurs in 4 blocks and together in one block either once or twice with any other product.

When searching for "good" designs it would be useful to be able to compare the quality of different designs of the same size. One possibility would be to use the variances for each contrast of interest to see which of them estimates the contrasts of interest with higher precision. Since in most cases more than one contrast is of interest though, a measure has to be defined that combines the variances from the multiple contrasts of the design under consideration to a single measure. This is the idea of **optimality criteria**. Instead of the variances themselves, these measures use the variance in relation to an orthogonal design.

Definition 3.16 (Efficiency) *Let $c'\tau$ denote an arbitrary contrast of a treatment comparison and $\widehat{c'\tau}$ its estimator. Let $var_d(\widehat{c'\tau})$ denote the estimator variance under design d and $var_o(\widehat{c'\tau})$ the variance under an orthogonal design with the same treatment replication and the same error variance σ^2.*

The ratio of the variances

$$E_{\widehat{c'\tau}} = \frac{var_o(\widehat{c'\tau})}{var_d(\widehat{c'\tau})}$$

is called the **efficiency factor** *of contrast* $\widehat{c'\tau}$.

For the special case of pairwise comparisons, $\widehat{c'\tau} = (\widehat{\tau_i - \tau_j})$, *with treatment* i *replicated* r_i *times and treatment* j r_j *times, the efficiency factor*

$$E_{(\widehat{\tau_i - \tau_j})} = \frac{var_o(\widehat{\tau_i - \tau_j})}{var_d(\widehat{\tau_i - \tau_j})}$$

is called a **pairwise efficiency factor**. *We have* $var_o(\widehat{\tau_i - \tau_j}) = \left(\frac{1}{r_i} + \frac{1}{r_j}\right)\sigma^2$.

Let C_d *be the information matrix of an equi-replicated, equi-sized incomplete block design and let* e_i *be the i-th largest eigenvalue of the matrix* $\frac{1}{r_p}C_d$. *The* $p-1$ *eigenvalues* e_i *of* $\frac{1}{r_p}C_d$ *are called the* **canonical efficiency factors**.

If interest is in more than one pairwise comparison, the **average efficiency** *can be calculated. It is*

$$E_A = \frac{1}{\sum_{i=1}^{p-1}\sum_{j=i+1}^{p}\frac{1}{E_{(\widehat{\tau_i - \tau_j})}}} = \frac{\sum_{i=1}^{p-1}\sum_{j=i+1}^{p} var_o(\widehat{\tau_i - \tau_j})}{\sum_{i=1}^{p-1}\sum_{j=i+1}^{p} var_d(\widehat{\tau_i - \tau_j})} = \frac{p-1}{\sum_{i=1}^{p-1} e_i^{-1}}$$

A design d is called **efficiency balanced** *if all its pairwise efficiency factors* $E_{(\widehat{\tau_i - \tau_j})}$ *are the same.*

All orthogonal designs have efficiency 100%, e.g. complete block and uniform row-column designs.

The average efficiency factor is a lower limit of the efficiency of an IBD compared to a randomised complete block design. The loss occurs when all experimental units are as homogeneous as the observational units within a block.

When designs are not equally replicated, but have replication vector $\boldsymbol{r} = [r_1, \ldots, r_p]'$, efficiency can be calculated in two ways. Let $\boldsymbol{r}^\delta$ denote the diagonal matrix with $\boldsymbol{r}$ on the diagonal. Using formula

$$E_{A;\boldsymbol{r}} = \frac{p-1}{\sum_{i=1}^{p-1} \frac{1}{e_i^*}} \tag{3.19}$$

where e_i^* is the ith largest eigenvalue of the matrix

$$C_d^* = \boldsymbol{r}^{-\frac{\delta}{2}} C_d \boldsymbol{r}^{-\frac{\delta}{2}}$$

$E_{A;\boldsymbol{r}}$ compares the design to an orthogonal design with the same replication. This measure is most appropriate if the replication vector $\boldsymbol{r}$ is fixed. If $n = \sum_{i=1}^{p} r_i$ is fixed rather then $\boldsymbol{r}$ itself, the design can be compared to an orthogonal design with replication $\bar{r} = \frac{n}{p}$. Efficiency can then be calculated as

$$E_{A;n} = \frac{p-1}{\sum_{i=1}^{p-1} \frac{\bar{r}}{\tilde{e}_i}} \tag{3.20}$$

where $\tilde{e}_i$ is the ith largest eigenvalue of the information matrix C_d.

Definition 3.17 (A-optimality) *A-optimal designs are designs in which the sum of variances of the pairwise comparisons of the treatments is at a minimum. For equi-replicated designs it is equal to the maximal value of E_A,*

$$O_A = \frac{2}{p(p-1)} \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} \frac{1}{\sigma^2} var_d \left(\widehat{\tau_i - \tau_j}\right)$$

If l_{ij} denotes the row vector of the elementary contrast for elements i and j and L the matrix of all essentially different elementary contrasts with $i < j$, E_A can be calculated as follows:

$$O_A = \frac{2}{p(p-1)} \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} \frac{1}{\sigma^2} \mathrm{var}_d \left(\widehat{\tau_i - \tau_j}\right)$$

$$\begin{aligned} &= \frac{2}{p(p-1)} \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} l_{ij} C_d^- l_{ij}' \\ &= \frac{1}{p(p-1)} \text{trace}(L C_d^- L') \end{aligned}$$

Another optimality criterion, also useful when all pairwise comparisons are of equal interest, is (M,S)-optimality. It is a two-step criterion, which was developed by Eccleston and Hedayat (1974).

Definition 3.18 ((M,S)-optimality) *In the first step, the mean of the efficiency factors is maximised (M-step).*

$$\sum_{i}^{p-1} e_i = trace\left(\frac{1}{r_p} C_d\right)$$

Within the class of M-optimal designs, the spread of the efficiency factors is minimised (S-step).

$$\sum_{i}^{p-1} {e_i}^2 = trace\left(\left(\frac{1}{r_p} C_d\right)^2\right)$$

For all binary block designs, trace$(NN') = \sum_i^p \lambda_{ii} = \sum_i^p r_i$, where r_i is the replication of product i. Thus, trace$\left(\frac{1}{r_p} C_d\right)$ is fixed. For equi-replicated designs, $\lambda_{ii} = r_p$, so $\sum_i^{p-1} e_i = p\left(1 - \frac{1}{t}\right)$.

The S-step is equivalent to minimising the sum of squares of the elements of the concurrence matrix, $\sum_i^p \sum_j^p \lambda_{ij}^2$. For equally replicated designs John and Williams (1995) show that A-optimal designs are all within the class of (M,S)-optimal designs.

The model equations for row-column and cross-over IBD are the same as for the respective complete block designs. The number of observations at is smaller, since the block size is smaller than the number of products $(t < p)$. Thus, the actual form of the projection matrix changes and with it the form of the information matrix.

3.2.2 Row-incomplete row-column design

The modified model equation for incomplete row-column designs $D_{\text{RC}}(p, a, t)$ is

$$Y_{ijk} = \mu + \pi_i + \alpha_j + \tau_k + \epsilon_{ijk} \quad 1 \le i \le t, \quad 1 \le j \le a, \quad 1 \le k \le p \tag{3.21}$$

with information matrix

$$C_d = r_p \left(I_p + \frac{1}{p} J_{p,p} \right) - \frac{1}{a} N_T N_T' - \frac{1}{t} N_A N_A'$$

which depends on the assessor incidence matrix, as discussed in the previous section and on the serving position incidence matrix with t servings. Row-column designs are orthogonal when $N_T = \frac{a}{p} J_{p,t}$ and $N_A = \frac{t}{p} J_{p,a}$. A row-column design is connected if $\text{rank}(C_d) = p - 1$.

For uniform row-column designs, in which the number of blocks equals the number of treatments, Youden squares can be used if they exist (see Definition 3.8).

A more general construction method is used for cyclic designs which are always uniform on experimental units. Alternatively, there are several computer algorithms given in the literature that search for optimal row-column designs. Both methods will be discussed further in Sections 3.4 and 3.5.

3.2.3 Row-incomplete cross-over designs

For incomplete block cross-over designs $D_{\text{CO}}(p, a, t)$, model (3.3) is modified to

$$Y_{d(i,j)} = \mu + \pi_i + \alpha_j + \tau_{d(i,j)} + \rho_{d(i-1,j)} + \epsilon_{ij} \qquad 1 \le i \le t, \quad 1 \le j \le a \tag{3.22}$$

The information matrices C_D and C_R for treatments and residuals as defined in (3.15) consist of the following parts:

$$\begin{aligned} C_{11} &= X'_P P_1 X_P \\ &= r_p I_p - \frac{1}{a} N_T N'_T - \frac{1}{t} N_A N'_A + \frac{1}{p} J_{p,p} \qquad (3.23) \\ C_{12} &= X'_P P_1 X_R \\ &= M - \frac{1}{a} N_T \tilde{N}'_T - \frac{1}{t} N_A \tilde{N}'_A + \frac{r_p}{ap} 1_p \boldsymbol{r}'_R \qquad (3.24) \\ &= C'_{21} \\ C_{22} &= X'_R P_1 X_R \\ &= \boldsymbol{r}^{\delta}_R - \frac{1}{a} \tilde{N}_T \tilde{N}'_T - \frac{1}{t} \tilde{N}_A \tilde{N}'_A + \frac{1}{ap} \boldsymbol{r}_R \boldsymbol{r}'_R \qquad (3.25) \end{aligned}$$

Balanced cross-over designs, introduced in Definition 3.9, exist only for a limited number of parameter combinations. If balanced repeated measurement designs do not exist, partially balanced repeated measurement design can be used instead.

Definition 3.19 (Partially balanced repeated measurement designs) *A partially balanced cross-over design based on a partially balanced incomplete block design is an arrangement of p treatments in a rows and t columns such that*

a) Every treatment occurs at most once in a row.

b) Every treatment occurs p times in each column.

c) Every pair of treatments (i, j) occurs together in μ_k columns if i and j are k-th associates.

d) Deleting the last column of the design, every pair of treatments (i, j) occurs together in ν_k rows if i and j are k-th associates.

e) Every ordered pair of treatments (i, j) occurs together in successive periods in λ_i rows if i and j are k-th associates.

f) *For every pair of treatments (i, j), the number of columns in which j occurs when i is in the last column is the same as the number of columns in which i occurs when j is in the last column.*

Computer algorithms for the creation of efficient cross-over designs are described in Section 3.5.

3.3 Efficiency bounds

Computer algorithms search for designs according to a chosen optimality criterion. For these, it is useful to know efficiency bounds, so the algorithm will stop when a design is found that is sufficiently close to the bound. The bounds that can be used as stopping criteria will depend on the chosen optimality criterion. In this section, bounds will be given for algorithms using (M,S)-optimality and A-optimality criteria. These will be given for the three design types; incomplete block designs, row-column designs and cross-over designs. While efficiency is defined in relation to orthogonal designs, which often do not exist for the design specification of interest, efficiency bounds can give a more realistic picture of how "good" a design is than efficiency alone.

3.3.1 Incomplete block designs

The value of the optimality criterion of a design d depends on its information matrix, which for equi-replicated incomplete block designs is $C_d = r_p I_p - \frac{1}{t} N_A N_A'$ (see equation (3.17)), and therefore depends on the actual form of the block concurrence matrix $\Lambda_A = N_A N_A'$. To find a minimal value of the optimality criteria, some characteristics of the incidence matrix N_A of **equi-replicated binary designs** with equal block sizes and its concurrence matrix $N_A N_A'$ are needed. These are listed below.

Let n_{ij} and λ_{ij} be the respective elements (i, j) of N_A and Λ_A.

$$\begin{aligned}
n_{ij} &= \begin{cases} 1 & \text{if treatment } i \text{ occurs in block } j \\ 0 & \text{otherwise} \end{cases} \\
\sum_{j=1}^{a} n_{ij} &= r_p \quad \text{column/block sum of } N_A \\
\sum_{i=1}^{p} n_{ij} &= t \quad \text{row/treatment sum of } N_A \\
\lambda_{ij} &= \sum_{k=1}^{a} n_{ik} n_{jk} \\
\sum_{j}^{p} \lambda_{ij} &= \sum_{j=1}^{p} \sum_{k=1}^{a} n_{ik} n_{jk} = \sum_{k=1}^{a} n_{ik} \sum_{j}^{p} n_{jk} = \sum_{k=1}^{a} (n_{ik} r_p) = t r_p \\
\lambda_{ii} &= \sum_{k=1}^{a} n_{ik}^2 = r_p \quad \text{if } N_A \text{ is binary} \\
\sum_{j}^{p} \lambda_{ii} &= p r_p = \text{trace}(\Lambda_A) = \text{trace}(N_A N_A') \\
\sum_{j=1, j \neq i}^{a} \lambda_{ij} &= \left(\sum_{j}^{a} \lambda_{ij} \right) - \lambda_{ii} = (t-1) r_p \\
\text{trace}(\Lambda_A^2) &= \sum_{i=1}^{p} \sum_{k=1}^{p} \lambda_{ij}^2
\end{aligned}$$

Let

$$\bar{\lambda} = \frac{1}{p(1-p)} \sum_{i=1}^{p} \sum_{j=1, i \neq j}^{p} \lambda_{ij} = \frac{(t-1) r_p}{(p-1)} = \lambda_1 + \lambda_R$$

where

$$\begin{aligned}
\lambda_1 &= \lfloor \bar{\lambda} \rfloor \qquad \text{is the integer part of } \bar{\lambda} \text{ and} \\
\lambda_R &= \bar{\lambda} - \lambda_1 \text{ is the fractional part of } \bar{\lambda}
\end{aligned} \tag{3.26}$$

Let

$$\eta = (p-1)\lambda_R \tag{3.27}$$

When a balanced design does not exist the minimum value of $\text{trace}(\Lambda_A^2)$ will be achieved for a regular graph design, in which each row and column has η elements of λ_1 and $p-1-\eta$ elements of λ_2 (Russell et al., 1981). For the off-diagonal of the matrix Λ_A this means that λ_1 and λ_2 occur with the following frequencies

$$\begin{aligned}\mathrm{freq}_{\Lambda_A}(\lambda_1) &= p\eta \\ \mathrm{freq}_{\Lambda_A}(\lambda_2) &= p(p-1-\eta)\end{aligned}$$

These values are used as stopping rules, in the (M,S)-optimality search algorithm of Russell et al. (1981), which is described in detail in Section 3.5.2 (algorithm REK1).

Other bounds that can be used for (M,S)-optimality search algorithms are lower bounds for the corrected second moment of the $p-1$ canonical efficiency factors.

The first three moments of the canonical efficiency factors are

$$\begin{aligned}
\bar{e} &= \frac{1}{p-1}\sum_{i=1}^{p-1} e_i = \frac{1}{p-1}\mathrm{trace}\left(\frac{1}{r_p}C_d\right) \\
&= \frac{1}{p-1}\left(\mathrm{trace}(I_p) - \frac{1}{r_pt}\mathrm{trace}(\Lambda_A)\right) = \frac{1}{(p-1)}\left(p - \frac{pr_p}{r_pt}\right) \\
&= \frac{p(t-1)}{t(p-1)} \\
S_2' &= \sum_{i=1}^{p-1} e_i^2 = \mathrm{trace}\left(\frac{1}{r_p^2}C_d^2\right) = \mathrm{trace}\left(\left(I_p - \frac{1}{r_pt}\Lambda_A\right)^2\right) \\
&= \mathrm{trace}(I_p) - \frac{2\mathrm{trace}(\Lambda_A)}{r_pt} + \frac{\mathrm{trace}(\Lambda_A^2)}{(r_pt)^2} \\
&= p - \frac{2p}{t} + \frac{1}{(r_pt)^2}\left(\sum_i^p\sum_j^p \lambda_{ij}^2\right) \\
&= \frac{p(t-1)^2}{t^2} + \frac{1}{(r_pt)^2}\left(\sum_i\sum_{j;i\neq j}\lambda_{ij}^2\right) \\
S_2 &= \sum_{i=1}^{p-1}(e_i-\bar{e})^2 = \frac{1}{(r_pt)^2}\sum_{i=1}^{p}\sum_{j=1,j\neq i}^{p}(\lambda_{ij}-\bar{\lambda})^2 \\
&= S_2' - (p-1)\bar{e}^2 \\
S_3 &= \sum_{i=1}^{p-1}(e_i-\bar{e})^3
\end{aligned}$$

Note that the second and third corrected moments S_2 and S_3 vanish for efficiency balanced designs (i.e. $e_i = \bar{e},\ i = 1, \ldots, p-1$).

For regular graph designs holds

$$S_2 \leq S_{2L} = \frac{p(p-1)\eta(1-\eta)}{(r_p t)^2}$$

This can be used as a lower bound for (M,S)-optimal designs, when balanced designs do not exist.

A lower bound for S_3 is given by Jarrett (1977):

$$S_3 \leq S_{3L} = \frac{p(p-1)\eta}{(r+pt)^3}$$

$$\text{where} \quad z = \begin{cases} \eta\left((p+1)\eta - 3\right) & \text{if} \quad \eta < \frac{p}{2(p-1)} \\ (1-\eta)\left(p-(p+1)\eta\right) & \text{if} \quad \eta \geq \frac{p}{2(p-1)} \end{cases}$$

If $\lambda_1 = 0$ then another bound for S_3 is given by Paterson (1983)

$$S_3 \leq S_{3L'} = \frac{p(p-1)\eta\left((p+1)\eta^2 - 3\eta - t + 2\right)}{(r_p t)^3}$$

Upper bounds for A-efficiency can also be expressed in terms of the moments S_2 and S_3. For binary designs, all canonical efficiency factors are positive, hence

$$E_A = \frac{p-1}{\sum_{i=1}^{p-1} e_i^{-1}} \leq U_0 = \bar{e}$$

This upper bound will only be reached for efficiency balanced IBD. For cases, where efficiency balanced designs do not exist, Jarrett (1977) gives the following tighter bound:

$$U_1 = \bar{e} - \frac{(p-1)S^2}{\bar{e} + (p-3)S} \quad \text{with} \quad S^2 = \frac{1}{(p-1)(p-2)} S_2$$

Alternatively, Tjur's (1990) bound can be used:

$$U_2 = \bar{e} - \frac{(1-\bar{e})S_2}{(p-1)(1-\bar{e}) - S_2}$$

A tighter bound that makes use of the third moment has been given by Jarrett (1983):

$$U_3 = \bar{e} - \frac{S_2^2}{(p-1)(S_3 + \bar{e}S_2)}$$

In general, S_2 and S_3 are unknown, thus, S_2 has to be replaced by S_{2L} in U_1 and U_2. An estimate of U_3 can be constructed by substituting S_{2L} for S_2 and S_3 with either S_{3L} or $S_{3L'}$.

3.3.2 Row-column designs

Upper bounds for the average efficiency in row-column designs can be found in similar fashion as for IBD. The information matrix of a binary equi-replicated row-column design is

$$C_d = r_p(I_p - \frac{1}{p}J_{p,p}) - \frac{1}{a}N_T N_T' - \frac{1}{t}N_A N_A',$$

which depends on the concurrence matrices of the row- and the column-component $\Lambda_T = N_T N_T'$ and $\Lambda_A = N_A N_A'$. The first moment of the $p-1$ canonical efficiency factors of the information matrix of a row-column design is

$$\bar{e} = \frac{ta - t - a + r_p}{r(p-1)}$$

which can be used as an upper bound,

$$E_A = E_{RC} \leq U_0 = \bar{e}$$

When at least one component of the design is non-binary an improved bound has been given by Park and Dean (1990):

$$E_A \le U_1 = 1 - \frac{a'(p-a')}{a^2(p-1)} - \frac{t'(p-t')}{t^2(p-1)}$$

where $a' = a$ modulo p and $t' = t$ modulo p.

Eccleston and McGilchrist (1985) give the following upper bound

$$E_A \le U_2 = \left(\frac{1}{U_a} + \frac{1}{U_t} - 1\right)^{-1} = \frac{U_1 U_2}{U_1 + U_2 - U_1 U_2}$$

where U_a and U_t are upper bounds for the average efficiency from the row and the column component of the design. The inequality holds when the row-column design is an adjusted orthogonal row-column design, so that $N_A' N_T = r_p J$, and the upper bounds for the component designs are replaced by their actual A-efficiency.

3.3.3 Cross-over designs

For partially balanced cross-over designs, Raghavarao and Blaisdell (1985) give the following bound for the A-efficiency of the direct treatment effect:

$$E_d \le U_1 = \frac{(t-1)p\left((t-1)tp - (t+p)\right)}{t^2(p-1)\left(p(t-1)-1\right)}$$

The upper bound is attained if the cross-over design is balanced.

Most of the bounds for incomplete block, row column and cross-over designs use estimates of the moments of the canonical efficiency factors, since the actual values are not known. The quality of the bounds depends therefore on the quality of the estimate. This has to be kept in mind when a comparison between the efficiency of a design and its estimated bound is made.

In the following two sections, construction techniques for the introduced design classes are discussed, concentrating on methods for settings for which balanced designs do not exist.

3.4 Cyclic designs

Cyclic designs for p products in p blocks of size t are defined through the first block of the design, called the **initial row**. From this block, all other $p-1$ blocks are created through cyclic development. Each element is advanced by one reduced modulo p, the number of products in the trial.

Example 3.11 Cyclic IBD for $p = 12$ products and assessors with block size $k = 6$.

1	2	5	7	9	10	⇐ initial row
2	3	6	8	10	11	⇐ initial row +1 (modulo 12)
3	4	7	9	11	12	⇐ initial row +2 (modulo 12)
4	5	8	10	12	1	·
5	6	9	11	1	2	·
6	7	10	12	2	3	·
7	8	11	1	3	4	
8	9	12	2	4	5	
9	10	1	3	5	6	·
10	11	2	4	6	7	·
11	12	3	5	7	8	·
12	1	4	6	8	9	⇐ initial row +11 (modulo 12)

From the initial row a difference matrix $\Delta = [d_{i,j}]_{t,t}$ can be derived, which reveals the properties of the concurrence matrix $N_A N_A'$ for the whole cyclic design. The difference matrix of size $t \times t$ is computed by calculating the difference between the t elements of the initial block b modulo p:

$$d_{i,j} = (b_i - b_j) \text{ reduced modulo } p \quad i, j = 1, \ldots, t$$

where b_i is the ith element of the initial block b. The diagonal of the difference matrix consists of structural zeros. This difference matrix can then be summarised by a vector δ which lists the number of times each difference occurs.

Since the concurrence matrix is circular, it is identified fully by its first row, which can also be calculated from the difference matrix as explained above. Its faster construction compared to that of the concurrence matrix $N_A N_A'$ calculated from the incidence matrix is especially useful in search algorithms for cyclic designs. (M,S)-optimal incomplete block designs can easily be found by searching for the initial row that minimises the sum of the squared entries of the summary vector.

Example 3.12 Relationship between the difference matrix and the concurrence matrix in a cyclic design.

Concurrence matrix $N_A N_A'$

Difference matrix Δ

$b_i - b_j$	*1*	*2*	*5*	*7*	*9*	*10*
1	*0*	*11*	*8*	*6*	*4*	*3*
2	*1*	*0*	*9*	*7*	*5*	*4*
5	*4*	*3*	*0*	*10*	*8*	*7*
7	*6*	*5*	*2*	*0*	*10*	*9*
9	*8*	*7*	*4*	*2*	*0*	*11*
10	*9*	*8*	*5*	*3*	*1*	*0*

Difference summary vector δ - frequency distribution of the elements of Δ:

Δ	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*
δ	*6*	*2*	*2*	*3*	*4*	*3*	*2*	*3*	*4*	*3*	*2*	*2*

$$N_A N_A' = \begin{bmatrix} \mathbf{6} & \mathbf{2} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{3} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{3} & \mathbf{2} & \mathbf{2} \\ 2 & 6 & 2 & 2 & 3 & 4 & 3 & 2 & 3 & 4 & 3 & 2 \\ 2 & 2 & 6 & 2 & 2 & 3 & 4 & 3 & 2 & 3 & 4 & 3 \\ 3 & 2 & 2 & 6 & 2 & 2 & 3 & 4 & 3 & 2 & 3 & 4 \\ 4 & 3 & 2 & 2 & 6 & 2 & 2 & 3 & 4 & 3 & 2 & 3 \\ 3 & 4 & 3 & 2 & 2 & 6 & 2 & 2 & 3 & 4 & 3 & 2 \\ 2 & 3 & 4 & 3 & 2 & 2 & 6 & 2 & 2 & 3 & 4 & 3 \\ 3 & 2 & 3 & 4 & 3 & 2 & 2 & 6 & 2 & 2 & 3 & 4 \\ 4 & 3 & 2 & 3 & 4 & 3 & 2 & 2 & 6 & 2 & 2 & 3 \\ 3 & 4 & 3 & 2 & 3 & 4 & 3 & 2 & 2 & 6 & 2 & 2 \\ 2 & 3 & 4 & 3 & 2 & 3 & 4 & 3 & 2 & 2 & 6 & 2 \\ 2 & 2 & 3 & 4 & 3 & 2 & 3 & 4 & 3 & 2 & 2 & 6 \end{bmatrix}$$

Tables of efficient cyclic designs have been extensively published in the literature, see for example John, Wolock and David (1972), John (1981) or Lamacraft and Hall (1982). Alternatively, computer algorithms can be used to search for efficient cyclic designs.

3.4.1 Search algorithms

Search algorithms for efficient cyclic designs have, for example, been suggested by John et al. (1993) and Nguyen (1994). The algorithm by John et al. (1993) uses integer programming to find A-optimal cyclic designs. Unfortunately it was not possible to obtain the publication by Nguyen (1994). To create cyclic designs for this dissertation, a simple exhaustive algorithm was sufficient, though.

The amount of computation for an exhaustive search algorithm can be reduced substantially when the search is done only within non-isomorphic designs. Isomorphic designs are designs derived from each other by re-labelling or permuting treatments, which are the products in the sensory context. For the algorithm used here, the initial rows removed from the set of candidate designs are the ones that can be derived by cyclic development or through permutation of the t entries of the initial rows.

If the number of products is fairly large the candidate list for non-isomorphic initial rows is extensive and the algorithm requires increasing amounts of memory and time to inspect all these designs. Since calculating differences is less time intensive than calculating generalised inverses, the search algorithm for cyclic designs used in this dissertation searches for A-optimal designs within the class of (M,S)-optimal designs. It consists of the following four steps:

Algorithm CYC1:

Step 1: Create a candidate list of initial rows $\{C_i\}$

Step 2: Reduce a candidate list to non-isomorphic initial rows $\{C_i^{iso}\}$

Step 3: Calculate the difference matrix $\Delta = [d_{ij}]_{t,t}$ and the summary vector λ. Calculate the sum of squares of λ. Find the initial rows that minimise the sum of squares $\{C_i^{SS}\}$

Step 4: Calculate the average efficiency for designs from $\{C_i^{SS}\}$ and find designs with maximum average efficiency $\{C_i^{A}\}$

This algorithm can be used for all combinations of (p, t) with $p > t$ of reasonable size. For p much larger than 30, the number of non-isomorphic initial rows might be too large to process.

By construction, all cyclic designs are uniform on periods, and so symmetric cyclic designs used in sensory studies are always balanced for order. Therefore no special algorithm is necessary for the construction of cyclic row-column designs. This balance for order is lost when the number of assessors is not a multiple of the number of products. When the number of assessors is larger than the number of products, cyclic designs can be constructed by using more than one initial row. When the number of assessors is not a multiple of the number of products, only some of the blocks from a cyclic set are used. Such sensory designs will then be only nearly balanced for order.

3.4.2 Cyclic cross-over designs

When first-order carry-over effects are included in the model, as in model (3.22), and interest is mainly in the direct product effects, optimality criteria depend on the information matrix C_D for the direct product effects with C_{11}, C_{12}, C_{21} and C_{22} defined as in Equations (3.23)–(3.25). That means that an ideal combination of the matrices $N_A, N_T, M, \tilde{N}_A$ and $\tilde{N}_T$ has to be found.

Complete block cross-over designs

For complete block designs, MacFie et al. (1989), Schlich (1993), Durier et al. (1997) and Hunter (1996) suggest **Williams Latin square designs** (introduced in Section 3.1.5). Williams Latin squares are cyclic Latin squares balanced for first-order carry-over effects. The initial row of a Williams Latin square, for which p, the number of products is even, has the following form:

$$\begin{bmatrix} p & 1 & p-1 & 2 & p-2 & 3 & p-3 & \dots & \frac{p}{2} \end{bmatrix}$$

For an odd number of products two Latin squares are needed to reach balance, which can be constructed with the following initial rows:

$$\begin{bmatrix} p & 1 & p-1 & 2 & p-2 & \dots & \frac{p-1}{2} & \frac{p+1}{2} \end{bmatrix}$$

and its reverse

$$\begin{bmatrix} \frac{p+1}{2} & \frac{p-1}{2} & \dots & p-2 & 2 & p-1 & 1 & p \end{bmatrix}$$

Wakeling et al. (2001) show that Williams Latin squares are only a subclass of cyclic designs balanced for first-order carry-over effects.

If the number of assessors a is smaller than the number of products p, a subset from a Williams square can be taken. Russell (1991) suggests an algorithm comparing all non-isomorphic connected subsets of a rows according to their A-optimality criterion. These non-isomorphic subsets are equivalent to the non-isomorphic initial rows for cyclic designs.

Incomplete block cross-over designs

For cyclic cross-over designs in incomplete blocks, Ball (1997) describes an algorithm that starts with an initial row whose elements are permuted and the respective designs evaluated according to a chosen optimality criterion. The optimality function for the cyclic designs d uses the precedence matrix M of the design:

$$c(d) = \text{range}(M) + \text{std}(M)$$

For small designs all permutations can be evaluated but for larger designs a non-exhaustive search algorithm is necessary. Ball (1997) suggests selecting n_1 initial rows randomly from a given set of initial rows and performing n_2 random permutations on each of these. The resulting cyclic designs are then evaluated according to the above criterion.

Cyclic designs are easily constructed and an exhaustive search is possible for designs of reasonable size. Their disadvantage is that they are limited to cyclic patterns. More efficient designs may be found in the class of all designs $d(p, a, t)$. Thus, non-cyclic computer search algorithms are an important alternative to cyclic construction techniques.

3.5 Non-cyclic construction methods

For creating incomplete block designs, two kinds of algorithms are used, **block-exchange** and **treatment-interchange** algorithms. Both are usually non-exhaustive algorithms that start with a random or partially random starting design. In block-exchange algorithms, blocks are exchanged between the design and a list of candidate blocks. At each step, the pair of blocks that causes the largest improvement in the optimality criterion is swapped between the design and candidate list. In interchange algorithms, single treatments are exchanged between blocks. At each step in the algorithm, treatment-block pairs are chosen that cause the largest improvement of the chosen optimality criterion. Both kinds of algorithms stop when a design is found for which the optimality criterion is close enough to a given upper bound (stopping rule) or when no swap can be found that would improve the optimality criterion any further.

Optimality criteria often involve time-intensive calculation methods, including the inversion of matrices. For some methods, updating formulae make it unnecessary to perform these calculations at every step in the algorithm, increasing the speed for finding optimal designs.

The disadvantage of exchange and interchange procedures is that searches frequently result in sub-optimal designs when the algorithm gets trapped in a local optimum of the objective function. One way to overcome this problem is to repeat the search with different starting designs. Unfortunately this does not guarantee that the global optimum will be found. To increase the chances of finding an optimal or nearly optimal design, **tabu search** or **simulated annealing methods** have been suggested. Tabu search methods prevent the algorithm from inserting elements that have been taken out in previous steps. A list of forbidden swaps is created. In simulated annealing methods randomly chosen swaps are always made when they improve the optimality criterion and they are made with a certain probability p, that decreases over the course of the search, even when they make the optimality criterion worse. Both methods can be used in combination with block-exchange and treatment-interchange algorithms.

In the following sections, selected construction algorithms are discussed in more detail. All of them are simple exchange or interchange algorithms. Since they generally find sufficiently efficient designs, it is not necessary to modify them into tabu-search or simulated annealing algorithms. Most of the algorithms introduced in this chapter will be extended in later chapters to construct more complicated designs, for which incomplete block and cross-over designs are the building blocks.

3.5.1 Block-exchange algorithms

Nguyen and Dey (1990) describe a block-exchange algorithm using the (M,S)-optimality criterion. A set of candidate blocks is created, from which a starting design is chosen at random. From this, complete blocks are swapped out and in, using the *minimum-maximum* principle, i.e., *first a block in the design is found whose* deletion causes the smallest loss in the optimality criterion. For the reduced design a block from the candidate list is then found that causes the largest increase in the optimality criterion. In this algorithm the following results for trace(C_d^2) are used:

If a block with incidence vector n is taken from design d, resulting in design d_- or added to design d, resulting in design d_+, an update formula for the optimality criterion of the new designs is

$$\begin{aligned} \text{trace}(C_{d_-}^2) &= \text{trace}(C_d^2) + t - 1 - 2S(n) \\ \text{trace}(C_{d_+}^2) &= \text{trace}(C_d^2) + t - 1 + 2S(n) \end{aligned}$$

where

$$S(n) = \alpha r'n + \beta n'NN'n$$

with $\alpha = 1 - \frac{2}{k}$, $\beta = \frac{1}{k^2}$, and $r = N1_a$. If block incident vector n_d in the design is replaced with block incident vector n_c to create design d',

$$\text{trace}(C_{d'}^2) = \text{trace}\left(C_d^2\right) + 2\left(t - 1 + 2S\left(n_c\right) - S'\left(n_d\right)\right)$$

where

$$S'(n_c, n_d) = S(n_c) + \alpha n_d' n_c + \beta (n_d' n_c)^2$$

The four steps of the algorithm are as follows:

Algorithm ND1: (Nguyen and Dey, 1990)

Step 1: From a randomly chosen b-block starting design, its incidence matrix N and the concurrence matrix NN' are calculated. For each candidate block not in the design, $S(n_c)$ is calculated.

Step 2: Among all candidate blocks, the block with incident vector $n_{c'}$ is identified for which $S(n_c)$ is minimised. For each block in the design, $S(n_d)$ is calculated.

Step 3: With these values, a block within the current design with incidence vector $n_{d'}$ is found for which $S'(n_{c'})$ is maximised.

Step 4: If $t - 1 + S(n_{d'}) - S'(n_{c'})$ is greater than a chosen negative small number, -10^{-5} say, then stop. Otherwise exchange $n_{d'}$ for $n_{c'}$, update NN' and each $S(n_c)$ and return to Step 2.

This algorithm is useful only for small incomplete block designs. For larger designs the following problems occur:

a) It gets frequently trapped in local optima of the objective function, yielding sub-optimal designs.

b) For designs with a candidate block list of more than 15000, the computer algorithm requires an enormous amount of memory, which my computer was not able to provide.

It will therefore be used only for the construction of the panel designs, discussed in Section 5.4, which are sufficiently small.

Interchange algorithms are an alternative to block-exchange algorithms. For these, a list of candidate blocks is not needed and so the requirements on computer memory are modest.

3.5.2 Interchange algorithms

In this section, selected interchange algorithms for incomplete block designs and cross-over designs are introduced, as these algorithms are needed in Chapter 5.

Incomplete block designs

Several distinct search algorithms for incomplete block designs have been given in the literature. Examples are the publications by Whitaker and Triggs (1990), who use non-linear 0-1 programming, and Whitaker (1995), who introduces a simulated annealing algorithm. The two interchange algorithms, that have been been chosen for use in this dissertation are introduced by Russell et al. (1981) and Jones and Eccleston (1980).

Russell et al. (1981) use the characteristics of a PBIB design with two associate classes λ_1 and $\lambda_2 = \lambda_1 + 1$, also referred to as a regular graph design. These designs are (M,S)-optimal for design sizes for which balanced designs do not exist. Whilst their complete algorithm creates efficient row-column designs, in the first step of their algorithm an (M,S)-optimal randomised IBD is created. Only this part of the algorithm will be described in detail.

The authors use the following relationships to find the stopping rules for the algorithm:

$$\begin{aligned}
r &= \alpha t + \beta, && \alpha, \beta \in N \\
\lambda_{ii} &= t\alpha^2 + 2\alpha\beta + \beta && \\
\sum_{i}^{p} \lambda_{ij} &= ar && \\
\sum_{j,i\neq j}^{p} \lambda_{ij} &= ar - \lambda_{ii} && \\
&= ar - t\alpha^2 + 2\alpha\beta + \beta && \\
&= \lambda_1(p-1) + \eta, && \lambda_1, \eta \in N_0, 0 \leq \eta < (p-1) \\
\sum_{j,i\neq j}^{p} \lambda_{ij}^2 &= (p-1)\lambda_1^2 + 2\lambda_1\eta + \eta && \text{for (M,S)-optimal designs}
\end{aligned}$$

Let λ_1 and η be defined according to equations (3.26) and (3.27). With this notation the steps of the algorithm can be summarised as follows:

Algorithm REK1 (Russell et al., 1981):

Step 1: Create the incidence matrix N_A of an M-optimal incomplete block design as a starting design according to the following steps:

Step 1.1: Find integers α and β such that $r = \alpha t + \beta$. (For symmetric designs with $r = t$, $\alpha = 1$ and $\beta = 0$).

Step 1.2: Generate a $p \times t$ matrix of α's.

Step 1.3: Numbering the columns of N_A as $0, 1, \ldots, (t-1)$, add 1 to the cells in columns $(i-1)\beta, (i-1)\beta + 1, \ldots (i-1)\beta + (\beta - 1)$ of row i, where each $(i-1)\beta + j$ is reduced modulo $t, j = 0, \ldots, (\beta - 1)$.

Step 2: Modify matrix N_A of step 1 according to the following steps so that it becomes the incidence matrix of an (M,S)-optimal IBD :

Step 2.1: Calculate $\Lambda_A = N_A N_A'$, $\sum_{i,i \neq j}^{p} \lambda_{ij}$, $\sum_{i,i \neq j}^{p} \lambda_{ij}^2$ and find λ_1, η. and the optimality criterion $s = \text{trace}\left((NN')^2\right) = \sum_i^p \sum_j^p \lambda_{ij}^2$.

Step 2.2: Create the vector v_{row} with the p elements $\sum_{i,i \neq j}^{p} \lambda_{ij}^2, j = 1, \ldots, p$ and rank them in increasing order, setting all elements that equal $\sum_{i,i \neq j}^{p} \lambda_{ij}^{2\ (\text{M,S})\text{-opt}}$ to rank 0.

Step 2.2.1: Select row i belonging to the next element of v_{row} if its rank is positive.

Step 2.2.2: Find the next element $\lambda_{ij}, i \neq j$ in row i that exceeds $\lambda_2 = \lambda_1 + 1$. If there is no such j go to Step 2.2.6.

Step 2.2.3: Find column pair c_1 and c_2 and row r_1 such that

$$\begin{aligned} N(i, c_1) &= N(j, c_1) = N(r_1, c_2) = \alpha + 1 \\ &\text{and} \\ N(i, c_2) &= N(j, c_2) = N(r_1, c_1) = \alpha. \end{aligned}$$

If such a combination cannot be found return to Step 2.2.1.

Step 2.2.4: Create N_{new} by changing $N(i, c_1)$ and $N(r_1, c_2)$ to α and $N(i, c_2)$ and $N(r_1, c_1)$ to $\alpha + 1$.

Step 2.2.5: Calculate $\Lambda_{\text{new}} = N_{\text{new}} N_{\text{new}}'$ and $s_{\text{new}} = \text{trace}(\Lambda_{\text{new}})$. If $s_{\text{new}} < s$ then replace N with N_{new} and return to Step 2.1. Otherwise do not update N, and go to Step 2.2.3 to choose another (c_1, c_2, r_1) combination.

Step 2.2.6: Find the first element $\lambda_{ij}, i \neq j$ in row i, which is less than λ_1. If there is no such j go to Step 2.2.10.

Step 2.2.7: Find column pair c_1 and c_2 and row r_1 such that

$$\begin{aligned} N(i,c_1) &= N(j,c_2) = N(r_1,c_2) = \alpha+1 \\ &\text{and} \\ N(i,c_2) &= N(j,c_1) = N(r_1,c_1) = \alpha. \end{aligned}$$

If such a combination cannot be found go back to Step 2.2.6.

Step 2.2.8: Create N_{new} by changing $N(i,c_1)$ and $N(r_1,c_2)$ to α and $N(i,c_2)$ and $N(r_1,c_1)$ to $\alpha+1$.

Step 2.2.9: Calculate $\Lambda_{\text{new}} = N_{\text{new}}N'_{\text{new}}$ and $s_{\text{new}} = \text{trace}(\Lambda_{\text{new}})$. If $s_{\text{new}} < s$ then replace n with N_{new} and return to Step 2.1. Otherwise do not update and go to Step 2.2.7 to choose another (c_1, c_2, r_1) combination.

Step 2.2.10: Change the value of i to the number of the row with the next highest rank. If this rank is nonzero go to Step 2.2.2. If no new value exists or if its rank is zero, then the current N is a (M,S)-optimal incidence matrix and the algorithm terminates.

As an alternative to using the (M,S)-optimality criterion, Jones and Eccleston (1980) use an update formula of the A-optimality criterion for their interchange algorithm.

The exchange-interchange procedure suggested by Jones and Eccleston (1980) is described here for equi-sized block designs only. In its original form it can produce designs with differing block sizes. It uses the weighted A-optimality criterion and an update formula for the information matrix.

Some notation is necessary to describe the steps of the algorithm: Let d_n be an n-element design with treatment-block incidence matrix N_n for p products in a blocks of size t. Let e_i be the elementary vector, whose elements are zero except element i, which is one. Also, let $N[,i] = Ne_i$ denote the ith column of matrix N which represents the incident vector of block i.

The information matrix of a randomised equi-sized block design with n entries is

$$C_n = \boldsymbol{r}_n^\delta - \frac{1}{t} N_n N_n',$$

where $\boldsymbol{r}_n$ is the product replication vector. If an equi-replicated design is constructed, $\boldsymbol{r}_n^\delta = r_p I_p$, but this algorithm can also be used when products have unequal replication. This is the case for some panel designs introduced in Chapter 5 and the treatment-control designs in Chapter 6.

Let d_{n-1} be design d_n with product p_1 in block b_0 removed, and $d_{n'}$ be design d_n with product p_1 substituted by p_2 in block b_0. Let N_{n-1} and $N_{n'}$ be the respective incidence matrices, C_{n-1} and $C_{n'}$ the information matrices.

Jones and Eccleston (1980) show that for connected designs the inverse of the information matrix for designs d_{n-1} and $d_{n'}$ can be calculated by the following updating formulae.

$$\begin{array}{rcrcll}
C_{n-1}^- & = & C_n^- & + & \frac{C_n^- c_1 c_1' C_n^-}{1 - c_1' C_n^- c_1} & \text{where } c_1 = \sqrt{\frac{1}{t(t-1)}} \left(-N_n\left[,b_0\right] + k e_{p1}\right) \\
C_{n'}^- & = & C_{n-1}^- & - & \frac{C_{n-1}^- c_2 c_2' C_{n-1}^-}{1 + c_2' C_{n-1}^- c_2} & \text{where } c_2 = \sqrt{\frac{1}{t(t-1)}} \left(-N_n\left[,b_0\right] + e_{p1} + (t-1)\, e_{p2}\right)
\end{array}$$

With this notation the steps of exchange-interchange algorithm for IBD can be described as follows:

Algorithm JE1: (Jones and Eccleston, 1980)

Step 1: Define contrast matrix L for the contrasts of interest, generally all elementary contrasts. Provide the weights matrix $W = w^\delta$, with weights for each contrast in L. When all elementary contrasts are of equal interest, $W = I$.

Step 2: Create a connected starting design d_n according the to algorithm by Jones (1976). Calculate its incidence matrix N_n, the information matrix C_n, its generalised inverse C_n^- and the optimality criterion for weighted A-optimality, $E_n = \text{trace}(LWL'C_n^-)$.

Step 3: Exchange algorithm iteration:

Ex 1: Create a list of differences $\Delta_o = E_{n-1} - E_n$ for all elements in the design. Sort this list in reversed order (smallest Δ_o first) and work through this list until an element is found for which $E_{n'} < E_n$.

Ex 1.1: Create d_{n-1}, c_1 and C_{n-1}^- by taking the next block-product combination (b_0, p_1) from the list, out of design d_n.

Ex 1.2: For all products apart from p_1, calculate $c_2, C_{n'}^-$ and $\Delta_i = E_{n-1} - E_{n'}$.

Ex 1.3: Find the product p_2 for which Δ_i is maximum.

Ex 1.4: If $E_n > E_{n'}$ for product p_2 then update the old design d_n with the new design $d_{n'}$, in which product p_2 is substituted for p_1 in block b_0 and update N_n, C_n, C_n^- and E_n.

Ex 2: Repeat from Step Ex 1 until no further improvement is found, i.e. the optimal replication scheme has been found.

Step 4: Interchange algorithm iteration

Int 1: Create a list of differences $\Delta_o = E_{n-1} - E_n$ for all elements in the design. Sort this list in reverse order (smallest Δ_o first) and work through this list until an element is found for which $E_{n''} < E_n$.

Int 1.1: Create d_{n-1}, c_1 and C_{n-1}^- by taking the next block-product combination (b_1, p_1) from the list out of design d_n.

Int 1.2: Create a list of differences $\Delta_b = E_{n-1} - E_{n'}$ for all products which are not in block b_1, sort them by size and work through this list until an element is found for which $E_{n''} < E_n$.

Int 1.2.1: Calculate $c_2, C_{n'}^-$ and take $\Delta_i = E_{n-1} - E_{n'}$ from the next product p_2 from list Δ_b.

Int 1.2.2: For each block b_2 not equal to block b_1, swap p_2 out of block b_2 in design $d_{n'}$ and create $d_{n'-1}$, c_1' and $C_{n'-1}^-$. Swap p_1 into block b_2 in design $d_{n'-1}$ and create $d_{n''}$, c_2' and $C_{n''}^-$. Calculate $E_{n''}$ and continue until a block b_2 is found for which $E_{n''} < E_n$. If none is found, go back to step Int 1.2.1. If no more elements are in the list go back to step Int 1.1.

Int 2: If $E_{n''} < E_n$, update the old design d_n with the new design $d_{n''}$, in which product p_1 in block b_1 is swapped with product p_2 from block b_2. Update N_n, C_n, C_n^- and E_n accordingly.

Int 3: Repeat from Step Ex 1 until no further improvement can be made.

Row-column algorithms

Whilst there is a large number of search algorithms for row-column designs suggested in the literature, these will not be discussed in this dissertation since they are not needed in the remainder of this dissertation. A review of row-column algorithms can be found in John and Williams (1995).

Cross-over designs

Several algorithms have been suggested in the literature for the construction of balanced or partially balanced cross-over designs, which are summarised e.g. in Jones and Kenward (1990). Unfortunately, many of these designs need too many blocks to be useful for sensory designs. Computer algorithms on the other hand, can be used to construct cross-over designs of any specified size. Algorithms for cross-over designs are given for example by Donev (1997), Ball (1997) and Eccleston and Whitaker (1999). The program Design Express (2001) also provides efficient cross-over designs.

It was intended to construct cross-over designs in Chapter 5, using a modified version of the multi-objective simulated annealing algorithm by Eccleston and Whitaker (1999). But despite considerable effort, it was not possible to run the obtained C-program successfully. While the program did not provide any error messages, the output designs produced where clearly not efficient. Attempts to program the described algorithm from scratch were also not successful. For these reasons the simulated annealing approach was not pursued any further. Instead, the simpler algorithm by Ball (1997) is used to create cross-over designs.

The algorithm by Ball (1997) uses a weighted optimality criterion of the mean square of the serving order concurrence matrix $N_T N_T'$ and the mean square of the precedence matrix $M = X_P' X_R$. Let $MS(X) = \frac{1}{n_1 n_2} \sum_i^{n_1} \sum_j^{n_2} x_{i,j}^2 = \frac{1}{n_1 n_2} \text{trace}(X'X)$, where $x_{i,j}$ denotes the elements of matrix X. With this, the actual objective function can be expressed:

$$c(D) = w_{\text{order}} MS(N_T N_T') + w_{\text{carry-over}} MS(M)$$

where w_{order} and $w_{\text{carry-over}}$ are a pair of chosen weights. The algorithm comprises of the following steps, with $w_{\text{order}}, w_{\text{carry-over}}, n_1, n_2$ and n_3 chosen in advance:

Algorithm Ball1:

Step 1: Create an efficient incomplete block designs as a starting design *d*.

Step 2: Choose a row in *d* at random.

Step 3: Perform a random swap in this row. Update the design if the swap improves the optimality criterion.

Step 4: Repeat Step 3 n_1 times.

Step 5: Repeat Steps 2 to 4 n_2 times.

Step 6: Increase n_1 and n_2 if no improvement is found in n_3 iterations. Set n_1 and n_2 back to their initial setting if an improved design has been found.

3.6 Conclusions

This chapter introduced the characteristics of efficient sensory single-session designs in which all product comparisons are of equal interest. Complete block designs are useful when the number of products is small enough, so that all products can be tested within a session without the onset of sensory fatigue. Due to substantial heterogeneity of assessors, repeated measurement designs are used, with assessors as blocks. In such a setup, order and carry-over effects are often experienced and have to be accommodated for in the design. Thus, interest for sensory trials is mainly in row-column or cross-over designs, rather than in simple randomised block designs. The relevant linear models and complete block as well as incomplete block designs have been introduced and their respective information matrices are given. Whilst incomplete block designs are generally not recommended for use in sensory experiments, they are still described here since there are the building blocks of the designs introduced in the following chapters. For these, efficiency bounds and selected computer search algorithms have been introduced. The algorithms described here in detail are modified in later chapters for the construction of more complex sensory designs.

Chapter 4

Sensory designs as multi-session trials

In the previous chapter complete block designs were discussed for single-session trials, in which each assessor tastes every product. The number of products that is compared in a sensory profiling experiment is often too large to be assessed in a single session without causing sensory fatigue, and so the whole trial needs to be split into sessions. This chapter deals with multi-session designs, which ensure that each assessor tastes each product. It is shown how the split of a trial into sessions influences the properties of the estimates of product differences based on the linear models introduced in Chapter 3. Several models and design algorithms are discussed.

The advisable length of a session can differ for different product categories. The stronger a flavour, the less products can be assessed directly after one other without sensory fatigue. If the products require a very complicated preparation procedure before they can be served, a split into sessions might be necessary due to organisational constraints. If products deteriorate very easily and, for instance, loose their optimal taste within a very short time, sessions have to be fairly short, to keep the products comparable. All products have to be of equal quality for each serving position. The number of servings in a session, t, also called the **assessor constraint,** is primarily determined by the number of products an assessor can taste without the onset of sensory fatigue.

If the trial is split into several sessions it is common to include a session effect in the model. A strict cooking regime for preparing products and a standard routine for distributing the products to assessors should prevent any differences between sessions, but may do so imperfectly. The inclusion of a session term in the model caters for the eventuality that differences between products or serving conditions between the sessions occur in spite of all precautions.

The time between sessions can vary in practise from minutes to several days and not all periods between sessions are necessarily of the same length. For the analysis of the data from a multi-session trial it is generally assumed that the length of the time span between sessions has no influence on a session effect. If effects of serving order and carry-over are included in the model, such effects are expected to occur within each session separately, not over the whole trial, and these effects are the same for each session. For carry-over effects, the last product from one session is not expected to have an effect on the first product in the next session. This seems a reasonable assumption since there is an interval between the sessions, intended to prevent sensory fatigue.

For complete block or incomplete block designs in sensory single-session experiments assessors represent the blocks. For multi-session trials, assessors also form the blocks, now called long-blocks, but each block is subdivided into sub-blocks, representing an assessor within a session. One idea for constructing designs for multi-session experiments would be to ignore the large blocks and use an incomplete block design for the as sub-blocks. Their drawback is that IBD do not guarantee that each assessor tastes every product. As an alternative, one could use complete block designs and split these into sessions. The disadvantage of CBD is that they do not take into account that comparisons within a sub-block are more precise than between sub-blocks. The assignment of products to sub-blocks in split complete block designs can therefore be sub-optimal. Thus, designs are needed for multi-session trials that assign all products to each assessor and that assign products to assessors within sessions in an efficient way. A class of designs that satisfy this constraint are **resolvable incomplete block designs**.

In resolvable incomplete block designs blocks can be grouped together, so that each treatment is replicated exactly once in each group. Thus, each long-block constitutes a **replicate**.

The parameters of a multi-session resolvable sensory design $D_R(p, a, s, t)$ are

- p, the number of products, which represents the number of treatments
- a, the number of assessors in the panel, which represents the number of long-blocks (replicates),
- s, the number of sessions, which represents the number of sub-blocks within each long-block, and
- t, the number of servings per session, which represents the block size of the sub-blocks.

The number of sessions, s, depends on the amount of replications required, which is often related to the budget for the trial and time constraints. If $r = 1$ *then* $s = \frac{p}{t}$, and the number of products p needs to be a multiple of the session length t.

Example 4.1 $D_R(12, 12, 3, 4)$: Resolvable design for 12 products in 3 sessions with 4 servings each for 12 assessors.

	Session 1				Session 2				Session 3			
Serving	1	2	3	4	1	2	3	4	1	2	3	4
Assessor 1	7	1	12	8	2	11	4	10	6	3	9	5
Assessor 2	4	11	2	10	9	5	3	6	1	8	7	12
Assessor 3	3	5	6	9	12	8	1	7	11	4	2	10
Assessor 4	8	10	11	5	4	12	9	1	3	2	6	7
Assessor 5	9	12	4	1	7	2	6	3	10	11	5	8
Assessor 6	6	2	3	7	11	10	8	5	12	9	4	1
Assessor 7	5	4	8	3	6	1	7	11	2	12	10	9
Assessor 8	2	9	10	12	3	4	5	8	7	1	11	6
Assessor 9	1	6	7	11	10	9	2	12	8	5	3	4
Assessor 10	11	8	9	6	1	3	10	4	5	7	12	3
Assessor 11	10	3	1	4	5	7	12	2	9	6	8	11
Assessor 12	12	7	5	2	8	6	11	9	4	10	1	3

This design has been generated with Design Express (2001), a design program by Ian Wakeling, which specialises in sensory designs. It is a resolvable cross-over design and will be used throughout this chapter as an illustration of several linear models. At this stage, it is used to illustrate resolvability and the general layout of a multi-session design.

Subsequent discussions revealed that this design is not only resolvable for assessors, but also resolvable within sessions: in each session the blocks of three consecutive assessors comprise all 12 products. This additional restriction on the design is not required and can result in a loss of efficiency.

If the trial has to be conducted in multiple sessions, a session effect is included in the model equation of the univariate analysis. Adequate design plans for such a trial differ from those for single-session experiments. The three single-session models for incomplete randomised designs, row-column designs and cross-over designs (models (3.16), (3.21) and (3.22)) can be extended for the multi-session case. The resulting changes of models, information matrices and optimal designs are explained in the following three sections and selected algorithms for resolvable designs are discussed.

4.1 Resolvable incomplete block designs

The easiest way to motivate a resolvable design is to regard it as a randomised incomplete block design with as blocks of size t that could be analysed according to model (3.16), in which the assessor effect is replaced by an assessor-by-session effect:

$$Y_{jkl} = \mu + (\alpha\beta)_{jk} + \tau_l + \varepsilon_{jkl} \qquad 1 \leq j \leq a,\ 1 \leq k \leq s,\ 1 \leq l \leq p \qquad (4.1)$$

where $(\alpha\beta)_{jk}$ denotes an assessor-by-session effect. Its information matrix is

$$C_{4.1} = arI_p - \frac{1}{t}N_{AS}N'_{AS}$$

The model usually assumed for resolvable designs includes, apart from the treatment-effect, a replicate and a block-within-replicate effect. For a sensory design, this translates into the following model:

$$Y_{jkl} = \mu + \alpha_j + \beta_k(\alpha_j) + \tau_l + \varepsilon_{jkl} \qquad 1 \leq j \leq a,\ 1 \leq k \leq s,\ 1 \leq l \leq p \qquad (4.2)$$

where the assessor effect α_j represents the replicate effects and $\beta_k(\alpha_j)$, the block-within-replicate effect.

For sensory trials a model, in which the session effect is crossed with assessors rather than nested within assessors, seems more appropriate. Such a model is:

$$Y_{jkl} = \mu + \alpha_j + \beta_k + \tau_l + \varepsilon_{jkl} \qquad 1 \leq j \leq a,\ 1 \leq k \leq s,\ 1 \leq l \leq p \qquad (4.3)$$

In single replicate multi-session designs every assessor tastes every product once, thus $N_A = J_{p,a}$, i.e. assessor effects are orthogonal to product effects, as in complete block designs. The information matrix for model (4.3) is therefore

$$C_{4.3} = arI_p - \frac{1}{at}N_SN'_S$$

so it depends solely on the assignment of products to sessions. Designs for this model are balanced if all products occur equally often in every session. A derivation of information matrices in this chapter is given in the appendix.

Additionally, an assessor-by-session interaction could be assumed, leading to model

$$Y_{jkl} = \mu + \alpha_j + \beta_k + (\alpha\beta)_{jk} + \tau_l + \varepsilon_{jkl} \qquad 1 \le j \le a,\ 1 \le k \le s,\ 1 \le l \le p \quad (4.4)$$

where $(\alpha\beta)_{jk}$ represents the interaction effect of assessor j in session k. This is simply a re-parameterisation of model (4.1) and its information matrix is equal to $C_{4.1}$.

Example 4.2 Characteristics of design $D_{\mathbb{R}}(12, 12, 3, 4)$ in Example 4.1. *Design $D_{\mathbb{R}}(12, 12, 3, 4)$ in Example 4.1 is balanced for assessor, session and serving order effects: each product occurs four times in each session, three times in each serving position and is served once to every assessor.*

The assessor-by-session concurrence of the design is:

$\lambda_{N_{AS}N'_{AS}}$	*0*	*3*	*6*
count	*24*	*72*	*36*

Since assessor and session effects are orthogonal to product effects its efficiency is 100% according to model (4.3). For model (4.1) the efficiency is 79.26%. These efficiencies are listed in Table 4.1 again for a more extended comparison between models and designs.

A special class of resolvable designs, called **square lattice designs**, was introduced by Yates (1936). They are resolvable incomplete block designs with $p = s^2$, where s is the number of sub-blocks and $t = s$ the block size of the sub-blocks. Thus each replicate consists of s^2 products in s blocks of size s. Square lattice designs are constructed with the help of orthogonal Latin squares (see Definition 3.7). **Rectangular lattice designs**, introduced by Harshbarger (1949), are resolvable designs for

$v = s(s-1)$, constructed according to a similar procedure as that for square lattice designs. Patterson and Williams (1976) introduced a more general class of resolvable designs, called **α-designs**. An α-design is constructed from a $t \times r$ **generating array**, a generalised cyclic design with cyclic development in steps of t modulo p. The search for an efficient α-design is therefore equivalent to searching for an appropriate generating array. As for cyclic designs, for which the concurrence matrix can already be inferred from the initial row, the concurrence matrix for α-designs can be inferred from the generating array. With $\lambda = [\lambda_1, \lambda_2, \ldots]$ being the summary vector of the concurrence matrix, α-designs are referred to as $\alpha(\lambda_1, \lambda_2, \ldots)$-designs.

Example 4.3 Resolvable design $D_R(12, 6, 3, 4)$ created as an (0,2)-α-design. *The design has been created with the ALPHA algorithm of the ALPHA+ design generation package with 100 tries (i.e. the search for an optimal design is repeated with 100 starting designs and the best designs from these 100 attempts is chosen).*

generating array

0	1	0	0
1	2	2	0
0	0	1	1
0	0	2	0
2	1	2	1
1	0	0	2

resolvable design

5	1	7	10
2	8	6	11
9	4	3	12
12	8	1	5
10	2	9	6
4	3	7	11
9	2	12	5
11	1	8	4
6	10	7	3
9	1	4	10
7	2	5	11
12	3	6	8
10	4	8	2
7	12	1	6
3	5	11	9
9	6	1	11
5	10	8	3
4	7	2	12

For a sensory trial it can be presented in the following more familiar form:

	Session 1				Session 2				Session 3			
Assessor 1	5	1	7	10	2	8	6	11	9	4	3	12
Assessor 2	12	8	1	5	10	2	9	6	4	3	7	11
Assessor 3	9	2	12	5	11	1	8	4	6	10	7	3
Assessor 4	9	1	4	10	7	2	5	11	12	3	6	8
Assessor 5	10	4	8	2	7	12	1	6	3	5	11	9
Assessor 6	9	6	1	11	5	10	8	3	4	7	2	12

With a balanced assessor-concurrence matrix, the information matrix for model (4.3) depends solely on the session incidence matrix N_S, while for model (4.4) it depends on the assessor-by-session concurrence matrix. Due to the construction principle, the assessor-by-session concurrence matrix can never be balanced for α-designs and they are not optimised for session effects.

$$N_{AS}N'_{AS} = \begin{bmatrix} 6&0&0&2&2&2&2&2&2&2&2&2 \\ 0&6&0&2&2&2&2&2&2&2&2&2 \\ 0&0&6&2&2&2&2&2&2&2&2&2 \\ 2&2&2&6&0&0&2&2&2&2&2&2 \\ 2&2&2&0&6&0&2&2&2&2&2&2 \\ 2&2&2&0&0&6&2&2&2&2&2&2 \\ 2&2&2&2&2&2&6&0&0&2&2&2 \\ 2&2&2&2&2&2&0&6&0&2&2&2 \\ 2&2&2&2&2&2&0&0&6&2&2&2 \\ 2&2&2&2&2&2&2&2&2&6&0&0 \\ 2&2&2&2&2&2&2&2&2&0&6&0 \\ 2&2&2&2&2&2&2&2&2&0&0&6 \end{bmatrix} \qquad N_S = \begin{bmatrix} 4&2&0 \\ 2&3&1 \\ 0&1&5 \\ 2&1&3 \\ 3&2&1 \\ 1&3&2 \\ 1&2&3 \\ 2&3&1 \\ 3&1&2 \\ 3&2&1 \\ 1&3&2 \\ 2&1&3 \end{bmatrix}$$

The efficiencies of Example 4.3 according to models (4.3) and (4.4) are 96.85% and 80.49%. The lower efficiency from model (4.4) compared to model (4.3) can be explained with the small block size of the sub-blocks (assessor within session, $t = 4$) compared to the number of units within a session.

Note that the efficiency values from Example 4.1 cannot be compared to Example 4.3, since the α-design has only half the number of blocks. An α-design for 12 assessors could not be created with ALPHA+ and a resolvable design for 6 assessors could not be created with Design Express.

A summary of the exact construction procedure for square and rectangular lattices and α-designs can be found in John and Williams (1995). The program used to construct the α-design for Example 4.3, ALPHA+ (Williams and Talbot, 1993), applies the algorithms by Nguyen (1993). It is an interchange algorithm in which a random pair of products is swapped between blocks and compared according to the (M,S)-criterion, optimising each replication sequentially.

While the efficiency factors for incomplete block designs are also applicable to resolvable incomplete block designs, some tighter bounds are known according to model (4.2), with assessors within blocks as a blocking factor. These bounds can be found, for example, in John and Williams (1995).

Subsequent discussions have revealed that α-designs are not suitable for sensory trials. The method of construction will always give 0 concurrences for some pairs of products. The consequence is that they provide efficient designs for low numbers of replication (2,3,4), but not generally for more than 4 replicates. It is recommended that the number of assessors in sensory trials should be at least 12. The use of a design for 6 assessors, as shown in Example 4.3, is therefore not advisable.

4.2 Resolvable row-column designs

As discussed in Chapter 3, in sensory trials an order effect is often expected, and thus models (4.3) and (4.4) can be extended with the inclusion of a serving order effect, resulting in the following two models:

Extension of model (4.3):

$$Y_{ijkl} = \mu + \pi_i + \alpha_j + \beta_k + \tau_l + \varepsilon_{ijkl} \qquad (4.5)$$

Extension of model (4.4):

$$Y_{ijkl} = \mu + \pi_i + \alpha_j + \beta_k + (\alpha\beta)_{jk} + \tau_l + \varepsilon_{ijkl} \qquad (4.6)$$

with

$$1 \le i \le t,\ 1 \le j \le a,\ 1 \le k \le s,\ 1 \le l \le p$$

The effect of serving order in these models refers to servings within sessions only. The information matrices for models (4.5) and (4.6) are

$$\begin{aligned} C_{(4.5)} &= arI_p - \frac{1}{as}N_T N_T' - \frac{1}{at}N_S N_S' \\ C_{(4.6)} &= arI_p - \frac{1}{as}N_T N_T' - \frac{1}{t}N_{AS} N_{AS} \end{aligned}$$

The analysis of variance model generally assumed for a resolvable row-column design includes, apart from the treatment effects (products, τ_l,) also terms for replicates (assessors, α_j), rows within replicates (session within assessors, $\beta_k(\alpha_j)$), columns within replicates (servings within assessors, $\pi_i(\alpha_j)$) and the row-column interaction within replicates term (session-by-serving interaction within assessors, $(\beta_k\pi_i)(\alpha_j)$):

$$Y_{ijkl} = \mu + \alpha_j + \beta_k(\alpha_j) + \pi_i(\alpha_j) + (\beta_k\pi_i)(\alpha_j) + \tau_l + \varepsilon_{ijkl} \qquad (4.7)$$

with $1 \le j \le a,\ 1 \le i \le t,\ 1 \le k \le s,\ 1 \le l \le p$

For sensory models, session and order effects are generally not assumed to be nested within assessors and models (4.5) and (4.6) are more appropriate than model (4.7).

Example 4.4 Characteristics of design $D_R(12, 12, 3, 4)$ in Example 4.1 as a resolvable row-column design.

The design in Example 4.1 is balanced for serving order ($N_T = 12J_{12,4}$), i.e. the serving order effect is orthogonal to the product effect and the efficiencies for models (4.5) and (4.6) are the same as for the models without order effects (4.3) and (4.4), *which are listed in Table 4.1.*

Algorithms for resolvable row-column designs are given, for example, by John and Eccleston (1986), who gives a construction mechanism for row-column α-design, by John and Whitaker (1993), who use a simulated annealing approach, and by Nguyen and Williams (1993), who use an efficient RIBD with blocks as columns as a starting design and an updating formula for the (M,S)-optimality criterion as the objective function. This is an extension of the Nguyen (1993) algorithm. These algorithms could not be obtained and their designs could therefore not be included in the comparison.

4.3 Resolvable cross-over designs

If a carry-over effect is assumed to exist the previous models can be extended by including a first-order carry-over effect.

Model (4.5) changes to

$$Y_{d(ijk)} = \mu + \pi_i + \alpha_j + \beta_k + \tau_{d(i,j,k)} + \rho_{d(i-1,j,k)} + \varepsilon_{ijk} \tag{4.8}$$

and Model (4.6) changes to

$$Y_{d(ijk)} = \mu + \pi_i + \alpha_j + \beta_k + (\alpha\beta)_{jk} + \tau_{d(i,j,k)} + \rho_{d(i-1,j,k)} + \varepsilon_{ijk} \tag{4.9}$$

with

$$1 \leq i \leq p,\ 1 \leq j \leq n,\ 1 \leq k \leq s$$

The relevant components of the information matrices (3.15) for direct product and first-order residual effects are as follows:

for model (4.8)

$$
\begin{aligned}
C_{11;4.8} &= arI_p - \frac{1}{at}N_S N_S' - \frac{1}{as}N_T N_T' \\
C_{12;4.8} &= M - \frac{1}{at}N_S \tilde{N}_S' - \frac{1}{st}N_A \tilde{N}_A' - \frac{1}{as}N_T \tilde{N}_T' + \frac{1}{p}1_p \boldsymbol{r}'_R \\
C_{22;4.8} &= \boldsymbol{r}_R^{\delta} - \frac{1}{at}\tilde{N}_S \tilde{N}_S' - \frac{1}{st}\tilde{N}_A \tilde{N}_A' - \frac{1}{as}\tilde{N}_T \tilde{N}_T' + \frac{1}{ast}\boldsymbol{r}_R \boldsymbol{r}'_R
\end{aligned}
$$

for model (4.9)

$$
\begin{aligned}
C_{11;4.9} &= arI_p - \frac{1}{as}N_T N_T' - \frac{1}{t}N_{AS} N_{AS} \\
C_{12;4.9} &= M - \frac{1}{as}N_T \tilde{N}_T' - \frac{1}{t}N_{AS} \tilde{N}_{AS} + \frac{1}{p}1_p \boldsymbol{r}'_R \\
C_{22;4.9} &= \boldsymbol{r}_R^{\delta} - \frac{1}{as}\tilde{N}_T \tilde{N}_T' - \frac{1}{t}\tilde{N}_{AS} \tilde{N}_{AS} + \frac{1}{ast}\boldsymbol{r}_R \boldsymbol{r}'_R
\end{aligned}
$$

Example 4.5 Characteristics of design $D_R(12, 12, 3, 4)$ in Example 4.1 as a resolvable cross-over design.

As mentioned before, $D_R(12, 12, 3, 4)$ has been created as an efficient resolvable cross-over design. For cross-over designs the precedence matrix M gives information about the carry-over effects. For design $D_R(12, 12, 3, 4)$ in Example 4.1 the summary vector for M is

λ_M	0	1	2	3
count	63	55	25	1

which means that 55 pairs of products occur once one after the other, 25 pairs of products occur twice in the design, one pair occurs once and 63 pairs do not occur at all. This is a fairly tight distribution of λ values. The efficiencies are listed in Table 4.1.

To accommodate order and carry-over effects, Muir and Hunter (1991/92) propose Williams Latin squares with each design split into sessions. This construction method is used in Example 4.6.

Example 4.6 $D_R(12,12,3,4)$: Splitting a Williams Latin square.

	Session 1				Session 2				Session 3			
Serving	1	2	3	4	1	2	3	4	1	2	3	4
Assessor 1	1	12	2	11	3	10	4	9	5	8	6	7
Assessor 2	2	1	3	12	4	11	5	10	6	9	7	8
Assessor 3	3	2	4	1	5	12	6	11	7	10	8	9
Assessor 4	4	3	5	2	6	1	7	12	8	11	9	10
Assessor 5	5	4	6	3	7	2	8	1	9	12	10	11
Assessor 6	6	5	7	4	8	3	9	2	10	1	11	12
Assessor 7	7	6	8	5	9	4	10	3	11	2	12	1
Assessor 8	8	7	9	6	10	5	11	4	12	3	1	2
Assessor 9	9	8	10	7	11	6	12	5	1	4	2	3
Assessor 10	10	9	11	8	12	7	1	6	2	5	3	4
Assessor 11	11	10	12	9	1	8	2	7	3	6	4	5
Assessor 12	12	11	1	10	2	9	3	8	4	7	5	6

This design is created from a Williams square for 12 products and split into 3 sessions with 4 servings each. Every assessor tastes every product once, every product occurs four times in each session and three times in each serving position, so that the design is balanced for assessor, session and order effects.

Due to the split into sessions, the design can be balanced for first-order carry-over effects only partially; 108 product-pair sequences occur once and 36 sequences not at all. The latter are the sequences of the products in serving position 4 of the previous session followed by the product in serving position 1 in the following session, where the time gap between sessions eliminates the carry-over effects.

While the distributions of the entries of M are tighter than in Example 4.1, the distribution of the λ values for the assessor-by-session concurrence matrix N_{AS} is much wider.

$\lambda_{N_{AS}N'_{AS}}$	0	2	4	8
count	24	48	36	24

This is reflected in the efficiencies for the split Williams design, which are listed in Table 4.1. One reason for the low efficiency of the Williams Square for model

Table 4.1: Efficiencies of Examples 4.1 and 4.6
Efficiencies (in percent) are compared for models 4.3, 4.5, 4.8, 4.8, 4.4, 4.6, 4.9 and 4.8.

Example 4.1						Example 4.6	
incomplete block design		row-column design		cross-over design		cross-over design	
model	efficiency	model	efficiency	model	efficiency	model	efficiency
4.3	100.00	4.5	100.00	4.8	92.05	4.8	97.49
4.4	79.26	4.6	79.26	4.9	66.85	4.9	55.87

(4.9) is, that Williams designs are not created with the split into sessions in mind. Their assignment of products to blocks (assessors within sessions) is therefore not optimised. It is optimal for assignments to sessions, servings, and carry-over though.

4.4 Conclusions

In this chapter, models for multi-session trials have been introduced. The examples have shown how appropriate designs depend on the assumed model. While the class of resolvable designs assures that each assessor tastes each product, they do not take the assignment of products to sessions into account. Thus, RIBD are efficient if the assessor-by-session interaction is included in the model. If no interaction is assumed, then the assignment of products to sessions is of sole interest, not the assignment to sub-blocks. In that case Williams squares are efficient for the cross-over model, since they are balanced for order effects and consequently also for session effects.

Interest in this thesis is mostly in a special kind of multi-session trials, trials with preparation constraints. In the following chapter a construction procedure is developed for this special class of designs and it will be shown that this algorithm can also be used for the more general case of designs without preparation constraints.

Chapter 5

Multi-session trials with preparation constraints

In Chapter 4 resolvable designs for multi-session trials have been discussed, denoted by $D_{\mathrm{R}}(p, a, s, t)$. These designs are useful for experiments in which all p products can be prepared for all s sessions. In sensory experiments there is often a limit on the number of different products that can be prepared for each session. This is referred to as the **preparation constraint** and denoted by k. The constraint is common when assessing food products, especially those requiring carefully controlled cooking procedures or other preparation prior to assessment. The number of products that can be prepared for a session may depend upon the available kitchen facilities – for example the number of hobs or cooking utensils, or upon the problems of achieving tight control of temperatures, volumes and especially of timings, when too many products are being processed simultaneously. Thus, when product numbers are very large, it might be advisable to use only a subset of all the products at each session to minimise handling mistakes, even if some efficiency is sacrificed in the process.

For multi-session experiments with a preparation constraint, design plans are required that allow for this extra condition. They are denoted as $D_{\mathrm{R}}(p, a, s, t; k)$. Such plans are more difficult to construct than the regular resolvable designs discussed in

Chapter 4 and no commercially available or otherwise published algorithms exist.

Often the construction of designs with preparation constraints is handled by limiting the parameter k more than necessary and setting it equal to the session length t. Hence designs of the kind $D_{\mathrm{R}}(p, a, s, t; t)$ are constructed rather than $D_{\mathrm{R}}(p, a, s, t; k)$ with $k > t$. This limitation is used as a matter of convenience, because construction of designs with $k = t$ is fairly easy, as will be explained in the following paragraph. Thus, even when more than t products could be prepared at one time in the kitchen, the number of different products that is **prepared for a session** is reduced to the number of products that can be **assessed within a session**, the so called assessor constraint. Unfortunately the number of servings in a session, t, is often fairly small, creating a much harsher constraint than a limitation to k products. This limits severely the number of possible direct product comparisons within a session. In such a design the products that can be compared within a session are the same for all assessors, only their presentation order differs between assessors. The influence of the size of the preparation constraint on the efficiency of a design is examined in Section 5.6 and its impact on the efficiency of the design is assessed.

When the preparation constraint is chosen to be equal to the assessor constraint ($k = t$), designs $D_{\mathrm{R}}(p, a, s, t; t)$ can be created by choosing t products for each of the s sessions according to an incomplete block design $D(p, s, t)$. This design is called the **preparation design**, since it sets out which products are prepared for a session. For each session, the available t products are then assigned to the a assessors in a different order. This order can, for example, be determined according to one or more Williams Latin squares of size t, which are balanced for order and carry-over effects. A design plan for assigning the order of the available products to the assessors is called the **panel design**.

Example 5.1 $D_R(12, 6, 6, 4; 4)$: A resolvable design with the preparation constraint equal to the assessor constraint ($k = t$).

The preparation design

6	9	11	12
2	4	7	9
4	5	8	11
1	2	3	5
3	6	7	10
1	8	10	12

Assigning products to sessions
created with Algorithm JE2

The panel design

3	4	2	1
4	1	3	2
1	2	4	3
2	3	1	4
2	1	3	4
1	4	2	3

Assigning serving order to assessors
created with Design Express

Combining the preparation design with the panel design results in the following resolvable design:

	Session 1				Session 2				Session 3			
Serving	1	2	3	4	1	2	3	4	1	2	3	4
Assessor 1	11	12	9	6	7	9	4	2	8	11	5	4
Assessor 2	12	6	11	9	9	2	7	4	11	4	8	5
Assessor 3	6	9	12	11	2	4	9	7	4	5	11	8
Assessor 4	9	11	6	12	4	7	2	9	5	8	4	11
Assessor 5	9	6	11	12	4	2	7	9	5	4	8	11
Assessor 6	6	12	9	11	2	9	4	7	4	11	5	8

	Session 4				Session 5				Session 6			
Serving	1	2	3	4	1	2	3	4	1	2	3	4
Assessor 1	3	5	2	1	7	10	6	3	10	12	8	1
Assessor 2	5	1	3	2	10	3	7	6	12	1	10	8
Assessor 3	1	2	5	3	3	6	10	7	1	8	12	10
Assessor 4	2	3	1	5	6	7	3	10	8	10	1	12
Assessor 5	2	1	3	5	6	3	7	10	8	1	10	12
Assessor 6	1	5	2	3	3	10	6	7	1	12	8	10

Note, this example shall only demonstrate the construction principle, a design for only 6 assessors is not recommended for use in practice.

This simple procedure for multi-session designs with a product constraint $k = t$ can be extended and modified to cope with a preparation constraint larger than the assessor constraint, i.e. $k > t$. This modified procedure is called the **three-step procedure** and is described in Deppe et al. (2001). The initial idea was suggested by Ian Wakeling and Roland Carpenter. The contribution made in this dissertation is a formalisation of this idea. In addition, the procedure is closely examined and algorithms for a computerised version have been created. In Chapters 6 and 7 extensions of the procedure are derived for treatment-control and factorial designs.

The linear models for sensory experiments with preparation constraints are the same as described in Chapter 4. The existence of the preparation constraint has an influence only on the actual structure of the information matrix. Due to the extra limitations on the number of different products assigned to blocks (assessors within sessions) an optimal design within the class of sensory designs with preparation constraints, $D_{\mathrm{R}}(p, a, s, t; k)$, is not necessarily optimal in the class of unconstrained designs, $D_{\mathrm{R}}(p, a, s, t)$. All designs in this chapter are optimised for model (4.8), while efficiencies according to model (4.9) are also given for selected designs.

An overview of the three-step procedure is given in Section 5.1. The scope of the three-step procedure is outlined in Section 5.2, which includes an introduction to the parameters that determine the design size and an explanation of the constraints on these parameters. A range of sensory designs are defined and examined in this chapter with different algorithms. The algorithms that are used for the construction of the preparation design, the panel design and for combining them are explained in Sections 5.3 to 5.5 respectively. In Section 5.6, multi-session sensory designs with a preparation constraint are compared for a range of preparation constraints k. This is an examination of the effect of the preparation constraint on the efficiency of the complete sensory design. In Section 5.7, an extension of the three-step procedure is derived.

5.1 The three-stage procedure

Multi-session trials with a preparation constraint are constructed from two smaller designs which combined make up the complete sensory design. In the first step of the three-step procedure, the preparation design, k products are assigned to each of the s sessions. This is done with a randomised incomplete block design for p products in s blocks of size k, denoted by $D^q(p, s, k)$. The meaning of the index q will be explained in Section 5.2.1. It is related to an additional limit on the contents of each column: to ensure that each assessor tastes every product r times, each column of the preparation design has to contain a certain subset of all products. Construction algorithms that create preparation designs with the required structure are introduced in Section 5.3.

Example 5.2 The preparation design.

$D^2(18, 9, 8)$ – *An incomplete block design for 18 products in 9 sessions with a preparation constraint of 8 products, equal to the block size.*

	Chosen products for each session							
Session 1	10	11	16	14	5	1	2	3
Session 2	11	12	17	15	6	2	3	4
Session 3	12	13	18	16	7	3	4	5
Session 4	13	14	10	17	8	4	5	6
Session 5	14	15	11	18	9	5	6	7
Session 6	15	16	12	10	1	6	7	8
Session 7	16	17	13	11	2	7	8	9
Session 8	17	18	14	12	3	8	9	1
Session 9	18	10	15	13	4	9	1	2
	product set 1				product set 2			
Column no.	1	2	3	4	5	6	7	8

Each row/block defines the products that will be prepared for that session. For example, the products available for session 1 according to the preparation design are products 1, 2, 3, 5, 10, 11, 14 and 16. Which subset of these products is given to which assessor and in what order is determined by the panel design. Each column of *this preparation design includes either products 1 to 9 or products 10 to 18, the two subsets of the 18 products. By choosing an arbitrary column from each of the two sets such two columns form a partition. The column numbers printed in boldface at the bottom of the table are the columns assigned to assessor 1 according to the first row of the panel design in Example 5.3.*

In the second step, the panel design, the k columns of the preparation design are assigned to the t serving positions for each of the a assessors. This is done using a cross-over design, denoted by $D^q_{\mathrm{CO}}(k, a, t)$. For a panel design, there is an additional constraint on the contents of each row, which has to refer to an equal number of columns from each subset of products. This is indicated by q in the notation. The exact structure of the panel design, necessary for the resolvability of the complete sensory design, is explained in Section 5.2.2. Algorithms for panel designs are discussed in Section 5.4.

Example 5.3 The panel design.

$D^2_{CO}(8, 12, 4)$ - *A cross-over design with entries 1 to 8, which refer to the 8 columns of the preparation design, listed in* $a = 12$ *blocks of size* $t = 4$.

	Columns chosen from the preparation design			
Assessor 1	**3**	**5**	**7**	**1**
Assessor 2	1	8	5	4
Assessor 3	4	7	8	1
Assessor 4	8	2	3	7
Assessor 5	8	6	4	3
Assessor 6	5	3	2	8
Assessor 7	2	4	7	6
Assessor 8	5	4	6	3
Assessor 9	2	1	6	5
Assessor 10	7	3	1	6
Assessor 11	7	5	4	2
Assessor 12	6	1	2	8

Two of the entries in each row/block refer to columns 1 to 4 of the preparation design (subset 1) and two entries to columns 5 to 8 (subset 2).

Writing out the assignment of columns from the preparation design to assessors yields the complete sensory design. The preparation design from Example 5.2 combined with the panel design from Example 5.3 generates the complete sensory design shown in Example 5.5. To show this procedure more clearly the design plan for assessor 1 is listed in Example 5.4. It is created from the first row of panel design 5.3 and the appropriate columns of preparation design 5.2.

Example 5.4 Product assignment for assessor 1.

$D_{CO}(18, 9, 4)$ - *A cross-over design for 18 products in 9 blocks (sessions) of size 4 (servings per session).*

Design plan for assessor 1

	Serving			
	1	2	3	4
Session 1	16	5	2	10
Session 2	17	6	3	11
Session 3	18	7	4	12
Session 4	10	8	5	13
Session 5	11	9	6	14
Session 6	12	1	7	15
Session 7	13	2	8	16
Session 8	14	3	9	17
Session 9	15	4	1	18

columns

3 5 7 1

of the preparation design

This design is defined from the first row of the panel design, ([3 5 7 1] *printed in boldface in Example 5.3) and refers to the columns of the preparation design (in boldface in the bottom row of Example 5.2). These four columns are listed above in the order given by this row of the panel design, which is the order they are served to assessor 1. Over the 9 sessions, assessor 1 is assigned each product exactly twice.*

Example 5.5 The complete sensory design.

$D_R(18,12,9,4;8)$ *A resolvable cross-over design for 18 products assigned to 12 assessors in 9 sessions with 4 servings each and a preparation constraint of 8, i.e. for each session 8 different products are prepared. The design is resolvable: over the whole trial every assessor tastes every product twice.*

	Session 1				Session 2				Session 3				Session 4				Session 5				Session 6				Session 7				Session 8				Session 9			
Assessor 1	16	5	2	10	17	6	3	11	18	7	4	12	10	8	5	13	11	9	6	14	12	1	7	15	13	2	8	16	14	3	9	17	15	4	1	18
Assessor 2	10	3	5	14	11	4	6	15	12	5	7	16	13	6	8	17	14	7	9	18	15	8	1	10	16	9	2	11	17	1	3	12	18	2	4	13
Assessor 3	14	2	3	10	15	3	4	11	16	4	5	12	17	5	6	13	18	6	7	14	10	7	8	15	11	8	9	16	12	9	1	17	13	1	2	18
Assessor 4	3	11	16	2	4	12	17	3	5	13	18	4	6	14	10	5	7	15	11	6	8	16	12	7	9	17	13	8	1	18	14	9	2	10	15	1
Assessor 5	3	1	14	16	4	2	15	17	5	3	16	18	6	4	17	10	7	5	18	11	8	6	10	12	9	7	11	13	1	8	12	14	2	9	13	15
Assessor 6	5	16	11	3	6	17	12	4	7	18	13	5	8	10	14	6	9	11	15	7	1	12	16	8	2	13	17	9	3	14	18	1	4	15	10	2
Assessor 7	11	14	2	1	12	15	3	2	13	16	4	3	14	17	5	4	15	18	6	5	16	10	7	6	17	11	8	7	18	12	9	8	10	13	1	9
Assessor 8	5	14	1	16	6	15	2	17	7	16	3	18	8	17	4	10	9	18	5	11	1	10	6	12	2	11	7	13	3	12	8	14	4	13	9	15
Assessor 9	11	10	1	5	12	11	2	6	13	12	3	7	14	13	4	8	15	14	5	9	16	15	6	1	17	16	7	2	18	17	8	3	10	18	9	4
Assessor 10	2	16	10	1	3	17	11	2	4	18	12	3	5	10	13	4	6	11	14	5	7	12	15	6	8	13	16	7	9	14	17	8	1	15	18	9
Assessor 11	2	5	14	11	3	6	15	12	4	7	16	13	5	8	17	14	6	9	18	15	7	1	10	16	8	2	11	17	9	3	12	18	1	4	13	10
Assessor 12	1	10	11	3	2	11	12	4	3	12	13	5	4	13	14	6	5	14	15	7	6	15	16	8	7	16	17	9	8	17	18	1	9	18	10	2

When the preparation design and the panel design are not completely balanced the complete design can be optimised by permuting the row-order of the preparation design and choosing the order that provides the highest efficiency for the complete design. Thus the third step of the procedure consists of optimising the column order of the preparation design to maximise the efficiency of the complete sensory design, a resolvable cross-over design. The preparation design in Example 5.2 is already listed with its optimal column order. The column optimisation of the preparation design is illustrated in Section 5.5.

5.2 Parameter constraints

As explained in Section 5.1, the three stage procedure is used to construct designs for the special case when only a subset of products (k) can be prepared for each session. It is assumed that the number of products (p) is larger than the preparation constraint (k), which in turn is larger than the assessor constraint (t, the number of servings per session). In other words, this procedure is adequate for designs for which $p > k > t$.

Additional to the parameters p, a, s, and t, as defined in Chapter 4, resolvable designs $D_{\mathrm{R}}(p, a, s, t; k)$ have the preparation constraint k as an additional parameter. From these five defining parameters, two others can be derived:

- the number of times each assessor tastes each product $r = st/p$ and
- the number of sessions each product is prepared for $r_s = sk/p$.

When a design is requested by a sensory scientist, p, a, k and t are often fixed by the circumstances. Either s, the number of sessions, or r, the number of times each assessor tastes each product, has to be chosen. When there is a financial or time limit given for the trial, this might provide an upper limit for s, the number of sessions. More often though, the minimum number of replications may be requested, or there may be a convention in a sensory laboratory or a company to use a certain number of replications for all trials which will define the parameter r. A compromise may have to be found between high replication (r) and short trial length (s).

The choice of parameters for the sensory design defines the size of the preparation and the panel designs. The relationships between the parameters and the constraints that exist for their choice are explained in the following two sections. A range of parameters is chosen for which designs are created as part of this dissertation with different algorithms.

5.2.1 The preparation design

The parameters in the complete sensory design that are relevant for the preparation design $D(p, s, k)$, are:

- the number of products (p),
- the number of sessions (s), equal to the number of blocks in the preparation design, and
- the number of products that can be prepared for one session (the preparation constraint, k), which determines the size of each block.

These three parameters define the product replication in the preparation design $r_s = sk/p$, which represents the number of sessions each product will be prepared for. For Example 5.2, $r_s = \frac{9 \cdot 8}{18} = 4$, which means that every product occurs in 4 of the 9 sessions.

To ensure that every assessor tastes every product the same number of times, the products are divided into q sets. The number of sets gives the number of distinct columns that have to include all p products and is determined by the ratio of the number of products and the number of sessions: $q = \frac{p}{s}$. According to the definition of r_s, this is equal to the ratio of the preparation constraint and the number of sessions each product is served in.

$$q = \frac{p}{s} = \frac{k}{r_s} \tag{5.1}$$

Therefore, p and s as well as k and r_s have to be chosen so that q is an integer and equation (5.1) holds. To stress the limitations on the parameters introduced by q, preparation designs are denoted as $D^q(p, s, k)$.

Example 5.6 Division of products into sets.

In Example 5.2, the preparation design with $p = 18$ *products in* $s = 9$ *sessions, products are partitioned into* $q = \frac{p}{s} = \frac{18}{9} = 2$ *sets:*

Product set 1	1	2	3	4	5	6	7	8	9
Product set 2	10	11	12	13	14	15	16	17	18

Each column in this design consists of products either from product set 1 (products 1–9 in columns 1–4) or product set 2 (products 10–18 in columns 5–8).

Thus the preparation design is an incomplete block design $D^q(p, s, k)$ for which two extra conditions hold:

- Condition C1: Each column consists of all products from one set.
- Condition C2: The k columns consist of r_s columns from each set.

When the panel design is described in Section 5.2.2, it will become apparent that these two extra conditions ensure the resolvability of the complete sensory design. Each assessor is served r replicates of each product, an essential condition for a sensory design.

The column order of the preparation design at this stage is arbitrary. It is important to know only which column belongs to which set before the construction of the panel design, because the panel design has to ensure that an equal number of columns from each set is chosen. We always assume that the columns are arranged by sets as shown in Example 5.7. The columns belonging to one set are called a column-block. The order of the sets and the order within each set are still arbitrary, i.e. the same panel design can be used for any column order within each set and any order of the sets of the preparation design.

Example 5.7 The structure of the preparation design

column *1* ... r_s	*column* $r_s + 1$ ... $2r_s$		*column* $(q-1)r_s + 1$ ... s
set 1 ... *set 1*	*set 2* ... *set 2*	...	*set q* ... *set q*
1 ... r_s *column-block 1*	*1* ... r_s *column-block 2*		*1* ... r_s *column-block q*

We consider a selected list of preparation and panel designs which resemble those used in industry. With the inequality $p > k > t$ in mind, values for t are chosen as 4 and 6, k is chosen to vary between 4 and 20 and p between 6 and 30. Larger designs can also be constructed with this procedure but the chosen designs are used as examples to illustrate the procedure and to compare different construction algorithms. Table 5.1 gives the full range of preparation designs that have been constructed, representing the first step of the three-step procedure.

The designs in these tables have been classified by the number of sets, which reveals the pattern of possible preparation designs quite clearly. As q and r_s have to be integers, only certain combinations of p, s and k are possible. The parameters p and k have to be multiples of q, which in combination with q define the values of s and r_s (see equation (5.1)). Designs with $q = 1$ involve preparation and panel designs without constraints. As these can be constructed with any generally available algorithm, they have not been included in the examples. Every algorithm for designs with the required constraints that is developed in this chapter is suitable also for

Table 5.1: Construction range for the preparation designs

$q = 2$: $s = \frac{p}{2}, r_s = \frac{k}{2}$

$q = 2$	k						
p	2	4	6	8	10	12	14
4	D	L					
6	D	U	L				
8	D	X	U	L			
10	D	X	C_4	U	L		
12	D	X	X	C_4	U	L	
14	D	X	X	C_6	C_4	U	L
16	D	X	X	X	C_6	C_4	U
18	D	X	X	X	C_8	C_6	C_4
20	D	X	X	X	X	C_8	C_6
22	D	X	X	X	X	C_{10}	C_8
24	D	X	X	X	X	X	C_{10}
26	D	X	X	X	X	X	C_{12}
28	D	X	X	X	X	X	X
30	D	X	X	X	X	X	X

$q = 3$: $s = \frac{p}{3}, r_s = \frac{k}{3}$

$q = 3$	k				
p	3	6	9	12	15
6	D	L			
9	D	U	L		
12	D	X	U	L	
15	D	X	C_6	U	L
18	D	X	X	C_6	U
21	D	X	X	C_9	C_6
24	D	X	X	X	C_9
27	D	X	X	X	C_{12}
30	D	X	X	X	X

$q = 4$: $s = \frac{p}{4}, r_s = \frac{k}{4}$

$q = 4$	k		
p	4	8	12
8	D	L	
12	D	U	L
16	D	X	U
20	D	X	C_8
24	D	X	X
28	D	X	X

X	these designs are constructed
C_x	complements of constructed designs with $k = x$
U	complements of disconnected designs
L	complete block designs
D	disconnected designs

constructing unconstrained designs.

If every product is prepared for one session only ($r_s = 1$), the preparation design is disconnected and therefore the whole sensory design will be disconnected because products are compared only in one session. Thus, session effects will be confounded with product differences. For this reason, disconnected preparation designs, marked D in Table 5.1, will not be considered.

Only designs are created in which the number of products is larger than the preparation constraint ($p > k$). All preparation designs $D^q(p, s, k)$ can therefore be constructed as binary designs. Some of these can be constructed as complements of other designs. More specifically, designs $D^q(p, s, k)$ and $D^q(p, s, p - k)$ are one another's complements. For the designs listed in Table 5.1, we decided to construct the smaller designs with $k \leq \frac{p}{2}$. The larger designs, that can be constructed from these smaller ones, are marked C_k, where k refers to the design with block size k, from which the design can be constructed.

For all designs in which the product replication is one less than the number of products divided by the number of sets ($r_s = \frac{p}{q} - 1$), the design is a so-called trivial matrix. In a trivial matrix, only one of s products from every set is missing in each block and so, without loss of generality, as long as a different product is omitted from every block, all such designs are isomorphic to each other. These designs, marked U in Table 5.1, are complements of disconnected designs. These are not constructed, since no search algorithm is needed to construct them. Designs marked L are complete block designs, which are not constructed either. For $p = k$ there is no preparation constraint and the algorithms for regular resolvable designs introduced in Chapter 4, can be used to construct these complete sensory designs, i.e. no preparation design is needed. Thus, only the remaining designs, marked X, are constructed for discussion in this chapter.

While in general we assume that $t \leq k < p$ with t either 4 or 6, the preparation designs with $k < 4$ are included for two reasons. First, they can be used to calculate their complementary designs and second, they are required in algorithm REK2 in Section 5.4 for the construction of larger kitchen designs with $q > 1$.

5.2.2 The panel design

The **panel design** $D^q_{\text{CO}}(p, a, t)$ is determined by the following parameters

- the number of assessors (number of blocks, a),
- the number of servings (block size, t),
- the kitchen constraint (k).

To ensure the resolvability of the complete sensory design, the structure of the panel design is also characterised by q, the number of sets the products are partitioned into. The preparation design consists of r_s columns from each of the q sets, thus for each block in the panel design an equal number of entries from each set is required, i.e. $r = \frac{t}{q}$ entries in each block refer to one set. This will be referred to as **condition C3**. The parameter r is also the number of replications of the complete sensory design, defined as the number of times each assessor tastes all products. Together with equation (5.1) from the preparation design this gives the additional constraints on the parameters of the complete sensory designs:

$$q = \frac{p}{s} = \frac{k}{r_s} = \frac{t}{r} \tag{5.2}$$

If the number of plots of the panel design, at, is a multiple of k then the panel design is equi-replicated – all columns of the preparation design are used the same number of times in the complete sensory design. Equal replication is not a necessary condition for panel designs though, since the complete design will be equi-replicated, namely $r_p = ar$, independently of the number of assessors taking part in the trial, as the design plan for every assessor is equi-replicated with a replication of r. The three-step procedure can therefore be used for any number of assessors, i.e. the number of assessors is not constrained by equation (5.2) in any way.

Like preparation designs, panel designs can be ordered according to the number of sets to illustrate their underlying pattern. The number of blocks of the panel design is defined as the number of assessors (a). Since order and carry-over effects are important for panel designs, designs cannot be constructed from their complements. Designs with $k = t$ are complete block designs and can be formed from Latin squares or Latin rectangles or fractions of these, depending on the number of assessors. They are marked L in Table 5.2.

Each of the designs in Table 5.2 has to be created for the exact number of assessors in the trial. For the panel designs created in this chapter, a panel comprising $a = 12$ assessors is assumed.

In this dissertation, only designs $D_{\mathrm{R}}(p, a, s, t; k)$ with $k > t$ are considered. Therefore panel designs have to be constructed only for $k > t$. Such panel designs are binary. Whilst designs could be constructed in which fewer products are available for every session than there are servings, these designs are not binary - an assessor tastes a product more than once in the same session. In such a setting, it is more likely that the session size would be reduced to the preparation constraint.

The sensory designs constructed from the panel designs with $q = 4$ (Table 5.2 d) have a product replication $r = 1$, which means that every assessor tastes every product only once. In such a setup, assessor inconsistency cannot be tested. Hunter (1996) suggests to use at least $r = 3$. For designs with $q = 3$ the assessor constraint would need to be at least $t = 9$ and for $q = 4$ at least $t = 12$ to satisfy the constraint $r \geq 3$, which is fairly high and might not be feasible for a lot of products. An alternative to increasing the assessor constraint or accepting the low product replication, is to

Table 5.2: Panel range

b) $q = 2$

	t	
k	4	6
4	L	
6	X	L
8	X	X
10	X	X
12	X	X
14	X	X
16	X	X
18	X	X
20	X	X
$2r_s$	2	3
	r	

c) $q = 3$

k	$t = 6$
6	L
9	X
12	X
15	X
18	X
$3r_s$	$r = 2$

d) $q = 4$

k	$t = 4$
8	X
12	X
16	X
20	X
$4r_s$	$r = 1$

X	Incomplete block cross-over designs
L	Complete block cross-over designs

increase the number of sessions. The trial would then be longer, and so a smaller q can be chosen. See also Section 5.7 for alternative designs in which q is not an integer.

5.2.3 The combined sensory design

With the range of preparation and panel designs defined in Tables 5.1 and 5.2, a large number of sensory designs can be constructed. A preparation design $D^q(p, s, k)$ and a panel design $D^q(k, a, t)$ can be combined when they agree on the preparation constraint k and the number of sets q $(q = \frac{p}{s} = \frac{t}{r} = \frac{k}{r_s})$. The parameter q specifies the table in Table 5.1, while k specifies the column in that table. All preparation designs from this column can be combined with the panel designs of the row identified by k in the table in Table 5.2 identified by q.

Example 5.8 Sensory designs constructable from preparation and panel designs with preparation constraint $k = 8$ and $q = 2$ sets.

Given a preparation constraint of 8 in 2 sets, any preparation design with $k = 8$ for which the number of products and sessions has a ratio of 2 can be used, i.e. all designs $D^2(p, \frac{p}{2}, 8)$. For the preparation design in Table 5.1 these are the pairs of s and $p = 2s$ with $s = 6, 7, \ldots, 15$.

These preparation designs can be combined with the panel designs in Table 5.2 b, row 5. Possible pairs of t and $r = \frac{t}{q}$ in this table are panel designs $D_{co,2}(8, a, 4)$ and $D_{co,2}(8, a, 6)$ with any panel size a.

Preparation	*Panel designs*	
designs	$D^2_{CO}(8, a, 4)$	$D^2_{CO}(8, a, 6)$
$D^2(12, 6, 8)$	$D_R(12, a, 6, 4; 8)$	$D_R(12, a, 6, 6; 8)$
$D^2(14, 7, 8)$	$D_R(14, a, 7, 4; 8)$	$D_R(14, a, 7, 6; 8)$
$D^2(16, 8, 8)$	$D_R(16, a, 8, 4; 8)$	$D_R(16, a, 8, 6; 8)$
$D^2(18, 9, 8)$	$D_R(18, a, 9, 4; 8)$	$D_R(18, a, 9, 6; 8)$
$D^2(20, 10, 8)$	$D_R(20, a, 10, 4; 8)$	$D_R(20, a, 10, 6; 8)$
$D^2(22, 11, 8)$	$D_R(22, a, 11, 4; 8)$	$D_R(22, a, 11, 6; 8)$
$D^2(24, 12, 8)$	$D_R(24, a, 12, 4; 8)$	$D_R(24, a, 12, 6; 8)$
$D^2(26, 13, 8)$	$D_R(26, a, 13, 4; 8)$	$D_R(26, a, 13, 6; 8)$
$D^2(28, 14, 8)$	$D_R(28, a, 14, 4; 8)$	$D_R(28, a, 14, 6; 8)$
$D^2(30, 15, 8)$	$D_R(30, a, 15, 4; 8)$	$D_R(30, a, 15, 6; 8)$

For a sensory laboratory with constant panel size, a large number of sensory designs can be generated from just a few panel designs. It is then easy to create a catalogue of designs, that covers most design requests for sensory multi-session trials with preparation constraints, when all pairwise comparisons are equally important. When the panel size varies, panel designs $D^q_{\rm CO}(k, a, t)$ have to be created for several values of a. As many designs are then required it is useful to have a fast algorithm for the construction of sensory multi-session designs with preparation constraints. In the following sections, computer algorithms for each step of the three-step procedure are discussed.

5.3 Creating the preparation design

At this stage, it is useful to summarise the properties of the preparation designs introduced in Section 5.2.1. The preparation design of the three-step procedure selects the products that will be prepared for each session. It is a special kind of row-column design, in which the rows represent the products prepared for each session. The row structure of the design is an incomplete block design and computer search algorithms can be used to construct it. The column structure, on the other hand, is fixed. It is defined through q, the number of sets the products are divided into. The entries in a single column are a permutation of the numbers of one of the sets, where each set is represented in r_s columns.

Due to the special column structure, available computer algorithms for efficient incomplete block designs have to be modified. The optimal preparation design is found in the subset of incomplete block designs that observe the restriction C2 of having an equal number of products from each set in every block (see page 117). The only exception is the case of a single set. For $q = 1$, the preparation design is a simple incomplete block design with no restrictions, and the usual search algorithms are applicable.

A design that complies with condition C2 can always be transformed into a row-column design that satisfies also condition C1, i.e., a design in which each column represents all products from one set. This can be done, for example, with the Algorithm J1 of Jones (1980), described in Section 5.3.2. This is not necessary for cyclic preparation designs, introduced in Section 5.3.1, since they automatically satisfy constraint C1.

In the following two sections, the necessary modifications are discussed for the incomplete block design algorithms, introduced in Chapter 3. In Section 5.3.1, cyclic construction techniques are described, while in Section 5.3.2 algorithms REK1 and

JE1 from Chapter 3 are modified for the construction of preparation designs that satisfy conditions C1 and C2.

5.3.1 Cyclic preparation designs

Cyclic designs seem to be a straightforward solution for the construction of preparation designs as the cyclic construction ensures that each column includes successive product numbers. A cyclic design is fully determined by its initial block. All other blocks are generated from the initial block, adding one to each element and reducing each entry modulo p. With a careful choice of the initial row, this should provide the structure necessary for an efficient incomplete block design with an equal number of entries from each set of products.

A careful choice is necessary since, in general, simple cyclic designs do not satisfy constraints C1 and C2 for $q > 1$. When the products are split into q sets, a preparation design has to be composed of q independent incomplete block designs, with s blocks of size $r_s = \frac{k}{q}$, referred to as sub-designs. Each sub-design though can be constructed as a cyclic design. So, from all cyclic designs, only those initial rows can be chosen that satisfy these two conditions.

Each sub-block represents a subset of the $s = \frac{p}{q}$ distinct products that build the set. That means that one block of the complete preparation design is formed from q of these sub-blocks. To create an efficient preparation design, an optimal combination of q sub-designs has to be found. These sub-designs do not have to be optimal.

Alternatively the preparation design can be viewed as a special kind of 2-cyclic design. For 2-cyclic designs, each product label is represented by a set of two numbers and the cyclic development is carried out for each number in turn.

For preparation designs, the first number identifies the set and the second the product within this set; the products within a set are always assigned numbers between 1 and s. In contrast to regular 2-cyclic designs, only the second number is developed cyclically, while the first remains fixed.

Example 5.9 Cyclic preparation design $D^2(18, 9, 8)$.

The designs in this example are the same as the preparation design in Example 5.2. The first design is shown in numerically increasing column order. Each sub-design is cyclically developed within each set, and the first sub-design is reduced modulo 9. The second sub-design is also reduced modulo 9 but 9 is then added to each reduced entry.

sub-design 1				sub-design 2				
1	2	3	5	10	11	14	16	← initial row
2	3	4	6	11	12	15	17	
3	4	5	7	12	13	16	18	
4	5	6	8	13	14	17	10	
5	6	7	9	14	15	18	11	
6	7	8	1	15	16	10	12	
7	8	9	2	16	17	11	13	
8	9	1	3	17	18	12	14	
9	1	2	4	18	10	13	15	
product set 1–9				product set 10–18				

In the second design in this example the 18 products are numbered according to a 2-cyclic design. Each subsequent row is created from the initial row by cyclical development according to the second digit, calculated modulo 9.

sub-design 1				sub-design 2				
11	12	13	15	21	22	25	27	← initial row
12	13	14	16	22	23	26	28	
13	14	15	17	23	24	27	29	
14	15	16	18	24	25	28	21	
15	16	17	19	25	26	29	22	
16	17	18	11	26	27	21	23	
17	18	19	12	27	28	22	24	
18	19	11	13	28	29	23	25	
19	11	12	14	29	21	24	26	
product set 11–19				product set 21–29				

For regular cyclic designs, the initial row defines a difference matrix that reveals the properties of the block concurrence matrix NN' for the whole cyclic design due to the circular structure of the concurrence matrix for cyclic designs. For these special 2-cyclic designs, differences can also be calculated. Differences within a set (the same sub-design) are called **pure differences**, differences between sets (different sub-designs) are called **mixed differences**. If the p products are denoted 1 to p, all differences are calculated modulo s, not modulo p as it would be done for a regular cyclic design. In the notation as for a two-cyclic design, differences for the cyclic preparation design are calculated on the second number only, also modulo s. The difference matrix is divided into q^2 sub-matrices representing the q pure difference matrices and $q(q-1)$ mixed difference matrices. The diagonal of the difference matrix consists of structural zeros. The number of zeros for the pure differences will therefore always be r_s, the number of columns in each set. For the mixed differences there are no structural zeros.

Example 5.10 Difference matrix for the 2-cyclic design $D^2(18,9,8)$.
The top left and the bottom right sub-matrices of the difference matrix of the cyclic design in Example 5.9 are the matrices of the pure differences and the top right and the bottom left matrices are the mixed differences. All differences are calculated modulo 9.

		11	12	13	15	21	22	25	27
		1	2	3	5	10	11	14	16
11	1	0	1	2	4	0	1	4	6
12	2	8	0	1	3	8	0	3	5
13	3	7	8	0	2	7	8	2	4
15	5	5	6	7	0	5	6	0	2
		pure(1,1)				mixed(1,2)			
		mixed(2,1)				pure(2,2)			
21	10	0	1	2	4	0	1	4	6
22	11	8	0	1	3	8	0	3	5
25	14	5	6	7	0	5	6	0	2
27	16	3	4	5	7	3	4	7	0

The concurrence matrix of a 2-cyclic design, as well as its difference matrix, consists of q^2 different sub-matrices. Each sub-matrix, rather than the complete concurrence matrix, has the circular structure of a cyclic design.

Example 5.11 Concurrence matrix for the 2-cyclic design $D_2(18, 9, 8)$. *The concurrence matrix for the cyclic design in Example 5.9 consists of four circular sub-matrices.*

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	4	2	2	1	1	1	1	2	2	3	1	2	1	2	2	2	1	2
2	2	4	2	2	1	1	1	1	2	2	3	1	2	1	2	2	2	1
3	2	2	4	2	2	1	1	1	1	1	2	3	1	2	1	2	2	2
4	1	2	2	4	2	2	1	1	1	2	1	2	3	1	2	1	2	2
5	1	1	2	2	4	2	2	1	1	2	2	1	2	3	1	2	1	2
6	1	1	1	2	2	4	2	2	1	2	2	2	1	2	3	1	2	1
7	1	1	1	1	2	2	4	2	2	1	2	2	2	1	2	3	1	2
8	2	1	1	1	1	2	2	4	2	2	1	2	2	2	1	2	3	1
9	2	2	1	1	1	1	2	2	4	1	2	1	2	2	2	1	2	3
10	3	2	1	2	2	2	1	2	1	4	1	1	2	2	2	2	1	1
11	1	3	2	1	2	2	2	1	2	1	4	1	1	2	2	2	2	1
12	2	1	3	2	1	2	2	2	1	1	1	4	1	1	2	2	2	2
13	1	2	1	3	2	1	2	2	2	2	1	1	4	1	1	2	2	2
14	2	1	2	1	3	2	1	2	2	2	2	1	1	4	1	1	2	2
15	2	2	1	2	1	3	2	1	2	2	2	2	1	1	4	1	1	2
16	2	2	2	1	2	1	3	2	1	2	2	2	2	1	1	4	1	1
17	1	2	2	2	1	2	1	3	2	1	2	2	2	2	1	1	4	1
18	2	1	2	2	2	1	2	1	3	1	1	2	2	2	2	1	1	4

For general cyclic designs, the difference matrix can be summarised by the summary vector δ, which is equivalent to the first line of the concurrence matrix. For the cyclic preparation design with $q > 1$, a separate summary vector δ_{ij} is defined for each sub-matrix of the difference matrix, where δ_{ij} refers to the differences between set i and set j. Generally, δ_{ij} and δ_{ji} will be different. Each summary vector is equivalent to the first row of the matching sub-matrix of the concurrence matrix. Calculating the sum of the squared entries of the concurrence matrix is therefore equivalent to calculating the sum of the squared entries of the concatenated q^2 summary vectors. This combined vector is denoted δ.

Example 5.12 Summary vector for the difference matrix in Example 5.10. *The four summary vectors for the sub-matrices of the complete difference matrix are associated with each of the four sub-matrices of the concurrence matrix.*

pure(1,1)

δ_{11}	4	2	2	1	1	1	1	2	2

mixed(1,2)

δ_{12}	3	1	2	1	2	2	2	1	2

δ_{21}	3	2	1	2	2	2	1	2	1

mixed(2,1)

δ_{22}	4	1	1	2	2	2	2	1	1

pure(2,2)

The four summary vectors in this example δ_{11}, δ_{12}, δ_{21} *and* δ_{22} *make up the combined summary vector* δ.

$$\delta = [4\;2\;2\;1\;1\;1\;1\;2\;2\quad 3\;1\;2\;1\;2\;2\;2\;1\;2\quad 3\;2\;1\;2\;2\;2\;1\;2\;1\quad 4\;1\;1\;2\;2\;2\;2\;1\;1]$$

For finding (M,S)-optimal incomplete block designs, it is sufficient to minimise the squared entries of the summary vector of the differences. The same holds for the special 2-cyclic case: (M,S)-optimal cyclic preparation designs can be found by minimising $\delta'\delta$.

The algorithm to find A-optimal cyclic preparation designs is a generalisation of algorithm CYC1, described in Chapter 3. The steps of the modified algorithm are summarised below:

Algorithm CYC2:

Step 1: Create a candidate list of initial rows for one set, $\{C_1\}$.

Step 2: Reduce the candidate list to non-isomorphic initial rows, $\{C_1^{iso}\}$.

Step 3: Create all combinations from the reduced set ($\{C_1^{iso}\}$) for all q sets, $\{C_q\}$.

Step 4: Calculate the difference matrix $D = [d_{ij}]_{k,k}$ for all pure and mixed differences and calculate $\delta'\delta$. Find the initial rows with minimum $\delta'\delta$, $\{C_q^{SS}\}$.

Step 5: Calculate the average efficiency for designs from $\{C_q^{SS}\}$ and identify the designs with maximum average efficiency, $\{C_q^A\}$.

An optimal preparation design is not necessarily built from optimal cyclic designs for one set, due to the consideration of the mixed differences. Thus, the candidate list in Step 2 cannot be reduced any further by finding optimal designs for one set first and then creating all combinations.

For the range of preparation designs with $q > 1$, defined in Table 5.1, all designs are created with algorithm CYC2. For several designs there is more than one optimal initial row. For each design an initial row is selected at random from the optimal rows and listed in Tables 5.3 and 5.4. The number of optimal rows are listed in Table 5.5.

The average efficiencies of the corresponding designs are listed in Table 5.6. For these tables, the efficiency is calculated in comparison to orthogonal designs. Alternatively, average efficiencies could be calculated by comparison to any of the efficiency bounds for randomised incomplete block designs that have been introduced in Chapter 3.

In Section 5.3.3, the cyclic preparation designs are compared to computer generated designs, described in the next section.

5.3.2 Computer generated preparation designs

The second method considered for the construction of preparation designs is by an extension of available computer algorithms. Existing algorithms for incomplete block designs are modified to include an extra option that limits searching within the group of designs that include an equal number of treatments from each set in each block (condition C2). The chosen algorithms are the two interchange algorithms REK1 and JE1, introduced in Chapter 3. Their respective changes are explained in

Table 5.3: Optimal initial rows for $q = 2$

$k = 4$ to $k = 10$

p	$k = 4$				$k = 6$						$k = 8$								$k = 10$									
8	1	2	5	7																								
10	1	2	6	8																								
12	1	2	7	9	1	2	3	7	8	11																		
14	1	3	8	11	1	2	4	8	9	13																		
16	1	2	9	12	1	2	5	9	10	15	1	2	4	5	9	10	12	14										
18	1	4	10	14	1	2	8	10	12	16	1	2	4	5	10	11	13	15										
20	1	3	11	14	1	2	5	11	13	18	1	2	4	5	11	12	17	19	1	2	3	6	9	11	12	13	18	19
22	1	5	12	17	1	2	10	12	14	18	1	2	5	6	12	14	17	19	1	2	3	7	10	12	13	14	18	21
24	1	3	13	16	1	2	10	13	15	18	1	2	6	11	13	14	20	22	1	2	3	6	9	13	14	15	20	23
26	1	5	14	20	1	2	5	14	16	21	1	2	5	7	14	15	18	20	1	2	3	9	11	14	15	17	18	22
28	1	6	15	21	1	2	12	15	17	22	1	2	7	12	15	16	23	27	1	2	4	7	8	15	16	21	25	27
30	1	3	16	23	1	2	5	13	18	25	1	2	6	14	16	17	25	27	1	2	8	11	13	16	17	24	26	28

$k = 12$ and $k = 14$

p	$k = 12$												$k = 14$													
24	1	2	3	7	8	11	13	14	16	17	19	21														
26	1	2	3	4	8	12	14	15	17	21	22	25														
28	1	2	3	6	10	12	15	16	18	19	21	23	1	2	3	4	7	12	13	15	16	17	20	22	23	27
30	1	2	3	4	8	14	16	17	19	23	26	27	1	2	3	8	10	13	14	16	17	18	23	25	28	29

Table 5.4: Optimal initial rows for $q = 3$ and $q = 4$

Optimal initial rows for $q = 3$

p	$k = 6$	$k = 9$	$k = 12$	$k = 15$
12	1 2 5 6 9 11			
15	1 2 6 8 11 13			
18	1 2 7 9 13 16	1 2 3 7 8 11 13 14 17		
21	1 2 8 10 15 18	1 2 6 8 9 13 15 17 19		
24	1 2 9 11 17 20	1 2 4 9 10 13 17 18 23	1 2 3 6 9 10 11 15 17 19 20 23	
27	1 2 10 12 19 23	1 2 7 10 11 17 19 21 23	1 2 3 8 10 11 13 16 19 20 24 26	
30	1 2 11 14 21 25	1 2 3 11 13 18 21 24 27	1 2 3 9 11 12 15 17 21 22 26 29	1 2 3 4 8 11 12 13 18 19 21 22 24 26 28

Optimal initial rows for $q = 4$

p	$k = 8$	$k = 12$
16	1 2 5 6 9 10 13 15	
20	1 2 6 7 11 13 16 18	
24	1 2 7 8 13 15 19 22	1 2 3 7 8 10 13 14 16 19 20 23
28	1 2 8 10 15 18 22 25	1 2 3 8 9 11 15 16 19 22 24 26

Table 5.5: Number of A-efficient initial rows for the cyclic designs in Table 5.3 and Table 5.4

$q=2$

p	$k=4$	$k=6$	$k=8$	$k=10$	$k=12$	$k=14$
8	1					
10	1					
12	1	2				
14	3	3				
16	1	4	8			
18	3	12	6			
20	2	8	12	20		
22	5	20	35	13		
24	1	8	20	32	48	
26	3	4	10	63	21	
28	3	12	36	96	108	4
30	2	4	5	64	168	84

$q=3$

p	$k=6$	$k=9$	$k=12$	$k=15$
12	1			
15	2			
18	1	3		
21	1	5		
24	1	8	8	
27	1	21	45	
30	2	20	25	136

$q=4$

p	$k=8$	$k=12$
16	1	
20	1	
24	1	4
28	3	7

Table 5.6: Efficiencies for the cyclic designs in Table 5.3 and Table 5.4

$q = 2$

p	$k = 4$	$k = 6$	$k = 8$	$k = 10$	$k = 12$	$k = 14$
8	80.77					
10	78.95					
12	75.00	89.47				
14	71.91	88.35				
16	71.43	86.87	92.66			
18	68.79	85.90	91.98			
20	67.86	85.20	91.20	94.35		
22	66.56	84.75	90.65	93.90		
24	65.80	84.02	90.27	93.41	95.40	
26	64.77	84.42	90.08	93.05	95.08	
28	63.78	83.19	89.58	92.74	94.75	96.12
30	63.00	83.33	89.25	92.51	94.49	95.89

$q = 3$

p	$k = 6$	$k = 9$	$k = 12$	$k = 15$
12	88.00			
15	84.62			
18	82.93	93.06		
21	82.35	92.11		
24	80.70	91.18	95.15	
27	79.59	90.56	94.57	
30	78.06	89.86	94.16	96.25

$q = 4$

p	$k = 8$	$k = 12$
16	90.73	
20	88.79	
24	86.97	94.75
28	85.77	94.03

this section.

The modified algorithms, REK2 and JE2, generate all preparation designs for the range defined in Table 5.1. All the efficiencies are calculated after 100 runs of each algorithm. For some designs, the number of runs may be slightly higher because the algorithm had to be restarted after some runs. (The designs were mostly run overnight and the ones that had not completed by the morning were stopped and run again the next night.)

The easiest way to ensure the necessary block structure is to create a set of candidate blocks that allows for constraint C2. From these blocks a random starting design is selected and the remaining blocks are swapped in and out according to a chosen criterion until no more improvement can be achieved. Such an algorithm would have to be adjusted for the three-step procedure only by reducing the candidate blocks to blocks with the same number of products from each set. The adequate candidate list is equivalent to the list of initial rows for the cyclic design before reducing for rows that are isomorphic through cyclic development.

Some of the preparation designs are simply too large to be constructed with a block-exchange algorithm due to the large number of candidate blocks. Thus, an interchange procedure may be better suited for constructing preparation designs.

Interchange procedures start with an equally replicated starting design. To create the special structure for preparation designs, two changes to the original interchange algorithms are necessary:

a) A starting design with the correct structure has to be created.

b) Interchanges are only allowed between blocks within the same set.

One such interchange procedure is algorithm REK1, which has been modified to allow for the partition of products into sets. The easiest way to adapt this algorithm to satisfy the constraints is to create the designs in a sequencc. The design for the first sub-design is created as before, the following sub-designs are created sequentially, so that interchanges are only allowed in the newly added sub-design.

The resulting efficiency from the algorithm REK2 after 100 runs is given in Table 5.7 for all designs in the range defined in Table 5.1.

The other algorithm that was modified is the interchange part of the JE1 algorithm. The exchange part of this algorithm is not required since, for designs in which all elementary contrasts are of equal importance, an equally replicated design is optimal. The starting design can therefore already be created as an equally replicated design. Thus, only interchanges are needed for optimising the design. To create an appropriate starting design with this algorithm, a starting sub-design is created separately for each set as a connected equally replicated design. These q starting designs are then combined to build the complete starting design.

To ensure that interchanges are only made within each group, the interchange part of the JE1 algorithm needs to be modified (Step 4). While the weakest entry of the design is determined over the whole design as before (Step 4 Int 1), the sub-design that contains the weakest entry is then identified. To find the best product pair for an interchange, only treatments in that sub-design are considered. That means that treatment swaps are allowed only within each sub-design.

The resulting efficiency of algorithm JE2 after 100 runs of the algorithm are given in Table 5.8 for all designs specified in Table 5.1.

The designs created with either of the two algorithms do not have the required column structure as defined in condition C1. The algorithm to combine the two component designs to form a row-column design by Jones (1980) can be used to achieve such a structure.

Table 5.7: Efficiencies for the designs created with algorithm REK2

$q = 2$

p	$k = 4$	$k = 6$	$k = 8$	$k = 10$	$k = 12$	$k = 14$
8	80.77					
10	78.95					
12	75.00	89.47				
14	72.18	88.35				
16	70.31	86.88	92.66			
18	68.96	85.87	91.89			
20	67.36	85.20	91.12	94.32		
22	65.59	84.53	90.59	93.82		
24	64.84	84.01	90.06	93.36	95.39	
26	63.93	83.24	89.69	92.98	95.00	
28	63.48	82.64	89.27	92.62	94.69	96.08
30	61.65	81.95	89.03	92.32	94.44	95.82

$q = 3$

p	k=6	k=9	k=12	k=15
12	88.00			
15	84.62			
18	82.93	93.06		
21	82.35	92.11		
24	80.25	91.17	95.15	
27	79.43	90.56	94.56	
30	78.43	90.01	94.07	96.24

$q = 4$

p	k=8	k=12
16	90.73	
20	88.79	
24	86.97	94.82
28	85.77	94.03

Table 5.8: Average efficiencies for designs created with algorithm JE2

$q = 2$

p	$k=4$	$k=6$	$k=8$	$k=10$	$k=12$	$k=14$
8	80.77					
10	78.95					
12	75.00	89.47				
14	72.18	88.35				
16	71.43	86.88	92.66			
18	68.96	85.91	91.98			
20	67.86	85.20	91.21	94.35		
22	66.56	84.75	90.65	93.90		
24	65.80	84.51	90.27	93.41	95.40	
26	64.77	84.42	89.98	93.05	95.07	
28	63.92	83.81	89.59	92.78	94.75	96.12
30	63.30	83.31	89.24	92.51	94.49	95.87

$q = 3$

p	$k=6$	$k=9$	$k=12$	$k=15$
12	88.00			
15	84.62			
18	82.93	93.06		
21	82.35	92.11		
24	80.70	91.18	95.15	
27	79.59	90.56	94.57	
30	78.53	90.20	94.16	96.24

$q = 4$

p	$k=8$	$k=12$
16	90.73	
20	88.79	
24	86.97	94.82
28	85.77	94.03

Algorithm J1 (Jones, 1980):

Step 1: A preparation design according to condition C2 and an auxiliary design specifying condition C1 are required as input designs. From these, their respective incidence matrices are calculated.

Step 2: For each plot in the design, create a candidate list of eligible treatments and a vector of zeros in which assigned treatments will be recorded.

Step 3: Assign a treatment to the plots successively according to the following steps:

Step 3.1: Find the next treatment from the list that is still eligible for the next plot.

Step 3.2Y: If one is found, add it to the vector of assigned treatments, take it out from both incidence matrices, invalidate it from the candidate list and repeat Step 3.1. for the next treatment.

Step 3.2N: If none is found undo the last step and invalidate the chosen treatment from the previous step. Return to Step 3.1 for the previous treatment.

Step 3.3: Stop the algorithm either when all treatments are assigned to plots or when the algorithm has traced back all its steps to the first treatment. In the latter case, the two designs cannot be combined to form a row-column design.

Step 4: If all treatments can be assigned to plots, generate the row-column design from the assignment list.

Example 5.13 Transforming the incomplete block design $D^2(18,9,4)$ to satisfy condition C1.

$D^2(18,9,4)$

9	1	13	18
2	4	13	10
6	8	16	14
8	7	18	11
5	9	11	17
2	7	16	12
4	6	17	15
1	3	15	12
3	5	10	14

A

auxiliary design

1	1	10	10
2	2	11	11
3	3	12	12
4	4	13	13
5	5	14	14
6	6	15	15
7	7	16	16
8	8	17	17
9	9	18	18

B

$D^2(18,9,4)$ with condition C1

1	9	13	18
2	4	10	13
6	8	16	14
8	7	18	11
9	5	11	17
7	2	12	16
4	6	17	15
3	1	15	12
5	3	14	10

C

The incomplete block design in panel A has been created with algorithm JE2. Algorithm J1 requires the auxiliary design in panel B to provide a design with structure C1. The resulting preparation design $D^2(18,9,4)$ with the required column structure is shown in panel C.

5.3.3 Comparison of the algorithms

To decide which of the three algorithms performs best, the efficiencies of the 64 different preparation designs listed in Tables 5.6, 5.7 and 5.8 are compared. For each of the designs, the maximum efficiency from its three alternatives is determined and the algorithm yielding the best result is identified. How frequently each algorithm performs best is listed in Table 5.9.

According to the second column in Table 5.9, all algorithms performed almost identically. This result does not seem to agree with the general impression from Tables 5.6, 5.7 and 5.8. A possible reason for this is that the size of the differences is not taken into account and that the efficiency value is analysed with all the digits

Table 5.9: Comparison of the efficiencies from algorithms CYC2, REK2 and JE2.

Algorithm with	Number of designs	
maximum efficiency	all digits	rounded
CYC2 only	16	10
REK2 only	17	0
JE2 only	19	13
CYC2 & REK2	1	0
CYC2 & JE2	8	18
REK2 & JE2	2	4
All equal	1	19

provided by SAS IML. Even when efficiencies differ, say, only in their 7th digit, one algorithm is classified as better than the other. Such small differences can be regarded as negligible. The analysis has therefore been repeated with efficiencies rounded to the fourth digit. These results are shown in the third column in Table 5.9.

The rounded efficiencies are in closer agreement with the visual inspection of Tables 5.6, 5.7 and 5.8. For 37 designs (57.8%), the cyclic algorithm performed as well as either of the two non-cyclic algorithms, for 19 (29.7%) of these all three have equal efficiencies. For 17 designs (26.6%), a computer algorithm outperforms the cyclic designs and for 10 designs (15.6%) the cyclic algorithm gives better results than either of the non-cyclic algorithms. Whilst for 23 designs algorithm REK2 performs as well as JE2, it is outperformed in 31 cases by algorithm JE2, ignoring the cases where CYC2 is superior. A comparison of the analysis using all digits compared to four digits shows that the designs for which REK2 performed best are better by only a very narrow margin.

To analyse the size of the differences, differences between the efficiencies of the three algorithms of a design of a certain size have been calculated as percentages from the

Table 5.10: Comparison of the efficiencies from algorithms CYC2, REK2 and JE2 (expressed as percentage differences).

Deviation from maximum	CYC2	REK2	JE2
Mean	0.0466	0.2222	0.0017
Standard deviation	0.1297	0.3887	0.0069

maximum efficiency of the three. Transformed in this way, the efficiency values for designs of different sizes can be compared. For this, the values of the efficiencies are used without rounding.

The resulting means and standard deviations, listed in Table 5.10, show that the algorithm JE2 performs most consistently and deviates the least of all three algorithms from the maximum efficiency. REK2 deviates the most from the maximum efficiency and also shows the largest variation. The cyclic designs are somewhere in-between these two extremes.

Since the cyclic designs are generated by an exhaustive search algorithm, cyclic designs perform worse than the non-cyclic designs when cyclic designs are too restricted. A computer algorithm performs worse than the cyclic algorithm when the algorithm gets trapped in a local optimum of the optimality criterion.

For the design range explored, algorithm JE2 performs generally better than algorithm REK2. One explanation for this is that designs with REK2 are constructed in a sequential manner and no interchanges are possible in its sub-designs, when the next sub-design is added. This is a strong limitation for the algorithm. The differences found for the preparation design algorithms REK2 and JE2 do not necessarily reflect performance differences of the unmodified algorithms REK1 and JE1.

5.4 Creating the panel design

The panel design is a cross-over design that allocates columns of the preparation design to the serving positions for assessors. The block size of the panel design is given by the number of servings per session (t) and the number of blocks is given by the number of assessors (a) attending the sensory trial. The t entries have to be chosen so that associated columns from the preparation design contain ra replications of the p products. For the panel design that means that each block has to consist of r entries referring to each of the q sets (condition C3 on page 121). In this section, cyclic and non-cyclic construction techniques for panel designs are discussed.

5.4.1 Cyclic panel designs

One possible method for creating panel designs is to use a similar cyclic structure as suggested for preparation designs. Since a cross-over design is required for panel designs, the designs should be as balanced as possible for both product and carry-over effects. Uniformity of order effects for cyclic designs is ensured by the cyclic structure.

Cyclic panel designs can be generated by a two-step algorithm. First, all A-optimal initial rows that satisfy condition C3, ignoring order and carry-over effects, can be identified. For each of these rows, all column permutations are created and the initial rows that maximise the A-efficiency for direct product effects according to model (3.22) are identified.

The simple cyclic method does not work for designs with $q > 1$. When products are split into sets, only the subset of cyclic preparation designs for which each block of the panel design refers to r columns from each set (condition C3) can be used. If the columns of the preparation design are sorted according to their set (i.e. columns referring to the same set are arranged in successive columns, so called column-blocks, as described in Example 5.7) the constraint is satisfied only if the sub-designs in each column-block are the same modulo r. This limits the number of possible initial rows substantially. The cyclic structure with the additional restriction results in a replication of the entries of the initial row after $r_s = k/q$ rows, but with a different order of the sets. Such a set of r_s consecutive rows in a cyclic design is known as a partial set.

Example 5.14 A cyclic panel design for $q = 2$: $D^2_{\text{co}}(6, 6, 4)$.
Two initial rows that satisfy condition C3 are [1 2 5 6] and [1 3 5 7]. Panel design A, constructed from initial row [1 2 5 6], consists of partial sets containing four rows, i.e. the design repeats itself after 4 rows, in a different order. For example, row 1 is [1 2 5 6], while row 5 is [5 6 1 2]. Modulo 4 the first sub-design (columns 1 and 2) is the same as the second sub-design (columns 3 and 4).

initial row
[1 2 5 6]

1	2	5	6
2	3	6	7
3	4	7	8
4	5	8	1
5	6	1	2
6	7	2	3
7	8	3	4
8	1	4	5

panel design A

initial row
[1 3 5 7]

1	3	5	7
2	4	6	8
3	5	7	1
4	6	8	2
5	7	1	3
6	8	2	4
7	1	3	5
8	2	4	6

panel design B

Panel design B, created from initial row [1 3 5 7], *already repeats itself after two rows, giving only two essentially different blocks. This design is disconnected and should therefore not be used.*

A slightly more complicated method can provide more efficient cyclic panel designs. Until now it was assumed that the columns in the preparation designs are ordered according to the set they belong to. If instead the columns are ordered alternately from each set, cyclic panel designs that do not have the repetitive behaviour shown in Example 5.14 can be easily found.

After a cyclic panel design has been found for the alternating preparation design, the columns of the preparation design can be reordered into column-blocks. Thus, the entries in the panel design have to be updated to refer to the column-block numbering. Whilst it was created as a cyclic panel design, the relabelled panel design no longer has a cyclic structure.

Example 5.15 Special cyclic panel design for $q = 2$: $D^2_{co}(8,8,4)$. *Preparation design 1 is listed in the usual column-block order as described in Example 5.7. The eight columns are permuted so that they alternate between the q sets, as shown in preparation design 2.*

Preparation design 1
$D^2(16,8,8)$
column-block order

1	**4**	6	**8**	9	10	**13**	**16**
6	**1**	2	**5**	11	14	**10**	**12**
2	**3**	5	**4**	13	11	**15**	**10**
4	**7**	1	**3**	16	12	**11**	**15**
3	**6**	8	**1**	12	15	**14**	**13**
8	**5**	3	**7**	10	16	**12**	**9**
5	**2**	7	**6**	14	13	**9**	**11**
7	**8**	4	**2**	15	9	**16**	**14**

Preparation design 2
$D^2(16,8,8)$
alternating order

1	9	**4**	10	6	**13**	**8**	**16**
6	11	**1**	14	2	**10**	**5**	**12**
2	13	**3**	11	5	**15**	**4**	**10**
4	16	**7**	12	1	**11**	**3**	**15**
3	12	**6**	15	8	**14**	**1**	**13**
8	10	**5**	16	3	**12**	**7**	**9**
5	14	**2**	13	7	**9**	**6**	**11**
7	15	**8**	9	4	**16**	**2**	**14**

For the case of $q = 2$, the necessary condition that equal numbers of each set occur in every block of the panel design translates into the condition that an equal number of odd and even entries occur in every block of the panel design. Thus, for this example, a cyclic panel design with two odd and two even numbers is required. Panel design A shows such a cyclic panel design, created with algorithm CYC3 (see following page). Its labelling is suitable for preparation design 2. To refer to preparation design 1, the entries of the cyclic panel design have to be relabelled according to the following assignment rules:

Cyclic design	1	2	3	4	5	6	7	8
Relabelled design	1	5	2	6	3	7	4	8

The resulting design is shown on the right (panel design B).

Panel design A (cyclic)

3	**6**	**8**	**7**
4	7	1	8
5	8	2	1
6	1	3	2
7	2	4	3
8	3	5	4
1	4	6	5
2	5	7	6

Panel design B (relabelled)

2	**7**	**8**	**4**
6	4	1	8
3	8	5	1
7	1	2	5
4	5	6	2
8	2	3	6
1	6	7	3
5	3	4	7

The assignment for assessor 1 can therefore be created either with preparation design 1 and panel design B or with preparation design 2 and panel design A. (The associated columns in both preparation designs and the first row in both panel designs related to assessor 1 are shown in boldfaced figures to highlight the construction process.)

	Assessor 1			
Session 1	4	13	8	16
Session 2	1	10	5	12
Session 3	3	15	4	10
Session 4	7	11	3	15
Session 5	6	14	1	13
Session 6	5	12	7	9
Session 7	2	9	6	11
Session 8	8	16	2	14

The initial rows for A-optimal panel designs in Table 5.12 are listed for the range defined in Table 5.2. The rows have been created with the following algorithm, based on Algorithm CYC1.

Algorithm CYC3:

Step 1: A list of all initial rows for a design $D(p,p,t)$ is compiled: $\{C_i\}$.

Step 2: Rows without the correct column structure are discarded (r_s entries from each of the q sets): $\{C_i^{c3}\}$.

Table 5.11: Efficiencies of the cyclic panel designs

a) $q = 2$

p	$t = 4$	$t = 6$
4	91.67	
6	84.18	96.67
8	81.34	92.74
10	78.53	90.53
12	76.83	89.18
14	75.62	88.03
16	73.37	87.35
18	73.67	86.92
20	73.06	86.44

b) $q = 3$

p	$t = 6$
6	96.67
9	91.40
12	89.18
15	87.65
18	86.92

c) $q = 4$

p	$t = 4$
8	81.34
12	76.83
16	74.56
20	73.06

Step 3: All the initial rows with minimum sum of squares of the concurrence matrix are identified: $\{C_i^{\text{SS}}\}$.

Step 4: From these, all A-optimal initial rows are identified according to model (3.16) (block effect only): $\{C_i^{\text{A}}\}$.

Step 5: For each initial row, the column orders that optimise the cross-over structure are identified, using the sums of the squared elements of M: $\{C_i^{\text{CO;SS}}\}$.

Step 6: From this list all designs A-optimal according to model (3.22) are identified (block, order and carry-over effects): $\{C_i^{\text{CO;A}}\}$.

The efficiencies of the cyclic panel designs generated with algorithm CYC3 are listed in Table 5.11. This algorithm is very similar to the one suggested by Ball (1997), introduced in Section 3.4.2. In algorithm CYC3, additionally to the (M,S) criterion, designs are selected according to the average efficiency of the direct product contrasts.

As usual for cyclic designs, all designs have p blocks. Their efficiencies are listed in Table 5.12. If the number of assessors a is not equal to the number of products

Table 5.12: Initial rows for A-optimal cyclic panel designs

a) $q = 2$

p	$t = 4$				$t = 6$					
4	1	4	2	3						
6	1	4	2	3	1	2	6	3	5	4
8	1	5	2	4	2	6	5	3	4	1
10	1	3	6	2	1	6	4	7	3	2
12	1	6	2	3	2	3	6	4	9	1
14	1	11	4	2	2	4	1	8	3	11
16	1	13	4	2	2	1	9	3	6	8
18	1	2	4	9	3	5	10	2	16	1
20	2	4	15	1	10	1	5	15	2	4

b) $q = 3$

p	$t = 6$					
6	1	2	6	3	5	4
9	1	6	3	5	4	2
12	2	3	6	4	8	1
15	1	4	8	6	2	9
18	1	2	4	8	3	12

c) $q = 4$

p	$t = 4$			
8	3	4	2	1
12	1	7	4	2
16	12	6	3	1
20	2	4	15	1

p, several cyclic designs have to be combined and subsets of cyclic designs have to be chosen to find designs with the required block size. Strategies for choosing such designs have been described in Chapter 3.

An alternative to cyclic designs are designs that are constructed for a specific number of assessors with the help of computer algorithms.

5.4.2 Computer generated panel designs

For generating panel designs using computer search algorithms a two-step procedure is applied. First, an IBD is constructed that complies with condition C3. This implies that each block of the panel design refers to the same number of columns from each set. Second, this IBD is optimised for order and carry-over effects.

Condition C3 for panel designs is equivalent to condition C2 for preparation designs. This means that the algorithms introduced for generating preparation designs can be used for panel designs. Algorithm JE2 can be used without changes, whereas REK2 has to be modified to handle unequal replication. As JE2 performs generally better than REK2 (see Section 5.3.2), REK2 has not been considered for generating panel designs. Instead, the block-exchange algorithm ND1 is used. To create valid panel designs with this algorithm only blocks are considered for the candidate list that comply with condition C3. This modified version with the reduced set of blocks is referred to as ND2. ND2 has not been used to create preparation designs, since its memory requirements are too large for larger preparation designs. The panel designs are much smaller and all designs in the suggested range could be constructed with the algorithm. The efficiencies of both algorithms based on a minimum of 100 runs are listed in Tables 5.13 and 5.14 for all designs defined in Table 5.2. For panel designs with unequal replication, efficiency is calculated according to formula (3.19).

Table 5.13: Efficiencies of the incomplete block panel design created with algorithm JE2

$q = 2$

k	$t = 4$	$t = 6$
4	100.00	
6	89.65	100.00
8	85.37	95.17
10	82.55	92.45
12	80.49	90.71
14	78.86	89.35
16	76.92	88.27
18	74.34	87.02
20	72.28	86.14

$q = 3$

k	$t = 6$
6	100.00
9	93.54
12	90.72
15	88.78
18	87.02

$q = 4$

k	$t = 4$
8	84.00
12	79.26
16	76.92
20	72.28

Table 5.14: Efficiencies of the incomplete block panel design created with algorithm ND2

$q = 2$

k	$t = 4$	$t = 6$
4	100.00	
6	89.65	100.00
8	85.37	95.17
10	82.55	92.45
12	79.90	90.40
14	78.61	89.12
16	75.22	88.03
18	73.86	86.10
20	71.10	-

$q = 3$

k	$t = 6$
6	100.00
9	93.54
12	90.39
15	88.37
18	86.19

$q = 4$

k	$t = 4$
8	83.67
12	79.21
16	74.88
20	70.98

Panel design $D_{\text{co}}(20, 12, 6)$ could not be generated due to insufficient memory of the computer.

To change these designs into cross-over designs the algorithm for non-cyclic designs described by Ball (1997) is used. Its steps have been listed as Algorithm B1 in Chapter 3. The transformation process of the panel design from an incomplete block design into a cross-over design is shown in Example 5.16. Since overall JE2 produced more efficient designs, the designs characterised in Table 5.13 are used as input designs for algorithm B1. Different weights have been explored and the designs with weights giving the highest efficiencies have been chosen, all based on a maximum of 100 runs. Their efficiencies according to model (3.22) using formula (3.19) with the chosen weights, are listed in Table 5.15. An example of the transformation of the panel design from an IBD to a cross-over design is shown in Example 5.16.

Example 5.16 Panel design $D_{\text{CO}}(8, 12, 4)$.

$D^2(8, 12, 4)$

1	3	5	7
1	4	5	8
1	4	7	8
2	3	7	8
3	4	6	8
2	3	5	8
2	4	6	7
3	4	5	6
1	2	5	6
1	3	6	7
2	4	5	7
1	2	6	8

Design A

$\Rightarrow$

$D^2_{\text{CO}}(8, 12, 4)$

3	5	7	1
1	8	5	4
4	7	8	1
8	2	3	7
8	6	4	3
5	3	2	8
2	4	7	6
5	4	6	3
2	1	6	5
7	3	1	6
7	5	4	2
6	1	2	8

Design B

Design A is an incomplete block design of the kind $D^2(8, 12, 4)$ with efficiency 85.37%. It is used as input design for algorithm B1 with weights $w_1 = 2$ for order effects and $w_2 = 1$ for carry-over effects. The resulting design, optimised for order and carry-over effects, is design B. Its efficiency according to model (3.22) is 81.96%.

All complete sensory designs created in this chapter are optimised for model (4.8), which includes assessor, order and carry-over effects. The three-step procedure can

Table 5.15: Efficiencies according to model (3.22) of the cross-over panel designs created with algorithm B1 The weights for the algorithm are given in brackets.

$q = 2$

k	$t = 4$	(w_1, w_2)	$t = 6$	(w_1, w_2)
4	91.67	(1,1)		
6	91.85	(1,2)	94.13	(1,1)
8	82.26	(1,2)	86.51	(1,1)
10	81.96	(1,2)	85.52	(1,1)
12	74.25	(5,1)	85.30	(1,1)
14	67.64	(1,2)	80.71	(1,1)
16	55.94	(1,1)	73.06	(1,2)
18	42.23	(1,1)	67.27	(1,1)
20	26.77	(1,1)	58.07	(1,1)

$q = 3$

k	$t = 6$	(w_1, w_2)
6	93.73	(1,1)
9	84.78	(1,1)
12	85.62	(1,1)
15	75.89	(1,1)
18	66.13	(1,1)

$q = 4$

k	$t = 4$	(w_1, w_2)
8	86.59	(1,2)
12	72.21	(1,2)
16	55.99	(1,2)
20	27.33	(1,1)

easily be changed to find optimal designs for cases in which carry-over effects are thought to be negligible. If carry-over effects are not expected, panel designs can be chosen as row-column designs rather than as cross-over designs.

5.5 The optimal column order for the preparation design

Since the preparation design is an incomplete block design, the order of the products within the block has no impact on its efficiency. The panel design attempts to balance for order and carry-over, but when it fails to achieve balance, the order of the columns of the preparation design can affect the precedence pattern of the complete design and its efficiency.

Whilst the column order of the preparation design has to keep the structure as described in Example 5.7, the order of the column-blocks and the column order within each column-block can be varied. The column order does not influence the efficiency of the preparation design, only the efficiency of the complete sensory design, since it influences the order and carry-over structure of the complete sensory design. To find the optimal column order for the preparation design, all permuted preparation designs are created and combined with the panel design, generating the different complete sensory designs. From the permuted preparation designs, the design that results in the highest efficiency for the complete sensory design is chosen. The procedure for generating the column permutations is illustrated on a small example.

Example 5.17 Permutations for a preparation design with 6 columns in 2 sets.

First, a list of all $r_s!$ permutations is compiled for each set. In this example, there are 2 sets with 3 columns each. The $r_s! = 6$ permutations of 3 columns are listed below.

Set 1			Set 2						
1	2	3	1	2	3		4	5	6
1	3	2	1	3	2		4	6	5
2	1	3	2	1	3	+ 3 =	5	4	6
2	3	1	2	3	1		5	6	4
3	1	2	3	1	2		6	4	5
3	2	1	3	2	1		6	5	4

These $q = 2$ lists are then combined, so that every permutation from each list occurs with every other row from the list, yielding $r_p!^q = 6^2 = 36$ rows, which represent all the within column-block permutations.

1	2	3	4	5	6	2	1	3	4	5	6	3	2	1	4	5	6
1	2	3	4	6	5	2	1	3	4	6	5	3	2	1	4	6	5
1	2	3	5	4	6	2	1	3	5	4	6	3	2	1	5	4	6
1	2	3	5	6	4	2	1	3	5	6	4	3	2	1	5	6	4
1	2	3	6	4	5	2	1	3	6	4	5	3	2	1	6	4	5
1	2	3	6	5	4	2	1	3	6	5	4	3	2	1	6	5	4
1	3	2	4	5	6	2	3	1	4	5	6	3	1	2	4	5	6
1	3	2	4	6	5	2	3	1	4	6	5	3	1	2	4	6	5
1	3	2	5	4	6	2	3	1	5	4	6	3	1	2	5	4	6
1	3	2	5	6	4	2	3	1	5	6	4	3	1	2	5	6	4
1	3	2	6	4	5	2	3	1	6	4	5	3	1	2	6	4	5
1	3	2	6	5	4	2	3	1	6	5	4	3	1	2	6	5	4

Additionally, the order of the sets can be permuted. The first permutation of this list $\boxed{1\quad 2\quad 3}\boxed{4\quad 5\quad 6}$ *could alternatively be presented with the second column-block first:* $\boxed{4\quad 5\quad 6}\boxed{1\quad 2\quad 3}$*, doubling the number of permutations for this example.*

The number of valid permutations is $n = (r_s!)^q\ q!$. Table 5.16 gives the number of possible column permutations for Step 3 of the three-step procedure for the preparation design.

For some of the preparation designs it is clearly not possible to check all possible permutations. A pragmatic solution is to choose a random subset from all permu-

Table 5.16: Number of valid column permutations

r_s	$r_s!$	$q = 2$	$q = 3$	$q = 4$
2	2	8	48	384
3	6	72	1296	31104
4	24	1152	82944	7962624
5	120	28800	10368000	4976640000
6	720	1036800	2239488000	6449725440000
7	5040	50803200	768144384000	15485790781440000

tations that has a manageable size. The best design from this subset can then be identified.

Example 5.18 Efficiencies for different column orders of the complete sensory design $D_{\text{R}}(18, 12, 9, 4; 8)$.

To find the most efficient column order, a permutation matrix with the 1152 different permutations has been generated. From each permutation, the corresponding preparation design is constructed from the original (Example 5.9) and combined with the panel design (Example 5.3). For each combined sensory design the efficiency is calculated according to model (4.8); the efficiencies vary between 85.13% and 88.89%. The maximum efficiency is obtained for permutation 1010 and results in the preparation design shown in Example 5.2. The minimum efficiency occurred for permutation 44, which corresponds to the initial row 1 2 3 5 10 11 14 15.

The assessor and session incidence matrices N_A and N_S are not affected by the column permutations. The session incidence matrix of the complete sensory design is directly related to the binary block-incidence matrix of the preparation design. Only 8 different products can be prepared for each session; so, if a product is available in a session, it is served to 6 assessors.

$$N_A = 2J_{12,12} \quad N_S = \begin{bmatrix} 6&0&0&0&0&6&0&6&6\\ 6&6&0&0&0&0&6&0&6\\ 6&6&6&0&0&0&0&6&0\\ 0&6&6&6&0&0&0&0&6\\ 6&0&6&6&6&0&0&0&0\\ 0&6&0&6&6&6&0&0&0\\ 0&0&6&0&6&6&6&0&0\\ 0&0&0&6&0&6&6&6&0\\ 0&0&0&0&6&0&6&6&6\\ 6&0&0&6&0&6&0&0&6\\ 6&6&0&0&6&0&6&0&0\\ 0&6&6&0&0&6&0&6&0\\ 0&0&6&6&0&0&6&0&6\\ 6&0&0&6&6&0&0&6&0\\ 0&6&0&0&6&6&0&0&6\\ 6&0&6&0&0&6&6&0&0\\ 0&6&0&6&0&0&6&6&0\\ 0&0&6&0&6&0&0&6&6 \end{bmatrix}$$

The column permutation affects the serving order incidence matrix N_T and the precedence matrix M, whose summary vectors are given below for the best and the worst column order.

best column permutation

λ_{N_T}	5	6	7
freq.	18	36	18

λ_M	0	1	2	3	4	5
freq.	54	198	45	0	9	0

worst column permutation

λ_{N_T}	5	6	7
freq.	18	36	18

λ_M	0	1	2	3	4	5
freq.	117	117	36	18	9	9

The difference between the two permutations is in the precedence structure. While for the chosen column permutation most pairwise product sequences occur once or twice and only a few not at all or four times, the precedence matrix for the worst column permutation has a much more dispersed distribution of entries and a larger number of product pairs that are never neighbours.

5.6 The preparation constraint and the efficiency of the sensory design

The preparation constraint k determines the number of products that can be compared within a session and therefore also the number of possible direct comparisons within each session. The following example illustrates how the preparation design influences the efficiency of the sensory design. Sensory designs $D^2_{\mathrm{R}}(18, 12, 9, 4; k)$ are created for $k \in \{4,\ 6,\ 8,\ 10,\ 12,\ 14,\ 16,\ 18\}$ and their efficiencies are determined.

The setting $k = p = 18$ is equivalent to a resolvable design without constraints and choosing $k = t = 4$ means that the preparation constraint is equal to the assessor constraint. For the three-step procedure, k has to be an even number since it has to be a multiple of $q = 2$.

The sensory design without constraints has been constructed with the program Design Express (2001). This program could create only a design for 27 assessors. To extract the required sub-design, 12 rows are chosen from the 27. The efficiency of the 12-assessor design varies across the subsets of rows. There are $\binom{27}{12} = 156454740$ possible subsets of 12 rows, too many to compare. We therefore draw a sample of 1000 random subsets of 12 rows and choose the most efficient of them. This yields the sensory design $D_{\mathrm{R}}(18, 12, 9, 4)$ in Example 5.19. The efficiency according to model (4.8) is 95.67%, while the least efficient design of the 1000 had an efficiency of 92.80%. Using the first 12 rows would have resulted in a slightly lower efficiency of 91.97%.

All designs $D_{\mathrm{R}}(18, 12, 9, 4; k)$ are created by the three-step procedure. Preparation designs $D^2(18, 9, 4)$, $D^2(18, 9, 6)$ and $D^2(18, 9, 8)$ are generated with algorithm JE2 and are part of the range whose efficiencies are listed in Table 5.8. Preparation designs $D^2(18, 9, 10)$, $D^2(18, 9, 12)$ and $D^2(18, 9, 14)$ are constructed as their respective complements. Preparation design $D^2(18, 9, 16)$ has been formed by taking out one

Example 5.19 **Design $D_R(18, 12, 9, 4)$ without a preparation constraint.**

	Session 1	Session 2	Session 3	Session 4	Session 5	Session 6	Session 7	Session 8	Session 9
Asscssor 1	3 15 10 12	17 1 13 9	9 6 7 11	5 2 18 1	7 2 8 3	14 11 16 5	6 14 4 8	16 10 13 4	17 12 18 15
Assessor 2	9 6 11 7	1 18 5 2	3 8 2 7	16 14 11 5	6 8 14 4	16 4 13 10	17 12 18 15	10 12 15 3	1 9 13 17
Asscssor 3	8 4 14 6	16 13 10 4	18 12 17 15	15 10 3 12	17 1 9 13	7 6 9 11	5 1 18 2	8 3 2 7	14 16 5 11
Assessor 4	17 9 1 13	6 11 7 9	1 5 18 2	2 3 8 7	5 11 16 14	6 4 14 8	4 16 10 13	15 17 18 12	10 15 12 3
Assessor 5	5 4 3 9	14 17 10 7	13 11 8 12	14 15 1 4	8 5 18 17	3 16 6 18	13 2 6 15	11 9 10 2	12 16 1 7
Assessor 6	7 16 12 1	9 3 4 5	17 14 7 10	11 8 12 13	1 4 14 15	18 8 17 5	16 6 3 18	2 13 15 6	9 11 2 10
Assessor 7	2 15 6 13	2 11 9 10	7 1 16 12	3 5 9 4	14 10 17 7	11 13 12 8	1 4 15 14	17 5 8 18	18 6 3 16
Assessor 8	1 14 4 15	17 8 18 5	16 18 3 6	2 13 6 15	9 11 10 2	1 16 7 12	4 3 5 9	14 7 10 17	13 8 11 12
Assessor 9	8 9 15 16	13 3 14 18	9 14 12 2	12 6 5 10	2 17 4 16	5 15 13 7	7 18 11 4	1 17 11 3	8 1 10 6
Assessor 10	3 13 18 14	9 14 2 12	5 12 10 6	16 4 2 17	13 5 7 15	11 7 18 4	3 11 17 1	8 6 1 10	16 15 9 8
Assessor 11	4 7 11 18	11 1 3 17	6 10 1 8	8 15 16 9	18 14 13 3	9 12 2 14	12 10 5 6	2 4 17 16	15 7 13 5
Assessor 12	6 12 5 10	4 2 16 17	15 7 5 13	18 4 7 11	11 17 3 1	8 6 10 1	15 8 9 16	18 14 3 13	2 14 9 12

product at random from each set from each block of a complete block design, with the constraint that overall all products are removed exactly once.

The efficiencies of the required panel designs $D^2_{\mathrm{CO}}(k, 12, 4)$ used for this example are given in Table 5.14. Preparation and panel designs are combined in the usual way giving sensory designs with the efficiencies listed in Example 5.20 according to models (4.8) and (4.9). The subset optimisation is used, with 1000 permutations selected at random, for all preparation designs for which more than one million permutations exist ($k \geq 12$).

Example 5.20 Efficiencies of the sensory designs $D_{\mathrm{R}}(18, 12, 9, 4; k)$ with $k \in \{4, 6, 8, 10, 12, 14, 16, 18\}$ (in percent).

	Exhaustive search				Random subset search				Design Express
k	4	6	8	10	12	14	16	18	18
Model (4.8)	63.95	81.98	88.41	91.22	93.60	95.17	96.14	96.80	95.67
Model (4.9)	63.57	68.60	69.67	69.69	68.79	68.49	69.02	68.91	68.68

The efficiencies according to the two models show two different patterns. For model (4.9), an increase can be seen from a preparation constraint of $k = 4$ (63.57%) to $k = 6$ (68.60%), all efficiencies for preparation contraints between 6 and 18 products are within the range of $69 \pm 0.7\%$. The slight decrease in efficiencies from $k = 10$ to $k = 12$ is most likely due to the subsets search.

The efficiencies computed according to model (4.8) are more strongly dependent on k and show the expected pattern. The number of direct comparisons increases with k and therefore so does the efficiency. The increase in efficiency is steepest between $k = 4$ and $k = 6$ but the increases are noticeable up to $k = 14$. For designs with preparation constraints between $k = 14$ and $k = 18$, the increases are only slight.

The information matrix for the direct product effects according to model (4.8) depends, among others, on the session concurrence matrix, which depends largely on k, the number of different products in a session. The information matrix for model (4.9), on the other hand, depends on the assessor-by-session concurrence matrix, which is much less dependent on k, since its block size is fairly small. This is also reflected in the size of the efficiencies; the efficiencies according to model (4.9) are far smaller than for model (4.8). Note also that the designs are optimised according to model (4.8) when choosing the column permutation.

This example shows the advantage of choosing a large k, at least as large as $t + 2$. It indicates that when k is chosen not too much smaller than p the loss in efficiency may not be too severe. The preparation constraint k can be chosen by a similar comparison of efficiencies for several values of k. This might help to balance the improvement due to the easier preparation procedure for a small k against the loss in efficiency.

In this example the design for $k = 18$ created with the three-step procedure is slightly more efficient than the design generated with Design Express. A possible reason is that the design made by Design Express was optimised for 27 assessors, not for 12. This result suggests that the three-step procedure might as well be used for generating sensory designs without preparation designs.

5.7 Extension of the three-step procedure

Up to now designs have been constructed with $q = \frac{p}{s} = \frac{k}{r_s} = \frac{t}{r}$ being an integer. The p products can then be divided into q distinct sets and each column of the preparation design consists of the products from one of the sets. If we relax this rule and allow $q = \frac{p}{s}$ to be a fraction, the three-step procedure needs to be slightly modified.

Let $q = \frac{q_1}{q_2}$, where $q_1 = \frac{p}{gcd(p,s)}$ and $q_2 = \frac{s}{gcd(p,s)}$, and $gcd(p, s)$ is the greatest common divisor of p and s. The products are divided first into q_1 distinct sets and q_2 of these sets are combined into super-sets, where all different $q_3 = \binom{q_1}{q_2}$ combinations are created. The sets are then arranged into a $q_2 \times q_3$ array, so that each column consists of the members of one super-set and each row of a set, so that each set occurs only once in each row (see the arrangement in Example 5.21).

Example 5.21 A design for 9 products in 6 sessions.

The 9 products are divided into 3 distinct sets.

set 1	set 2	set 3
1	4	7
2	5	8
3	6	9

These three sets can then be combined into 3 super-sets, which consist of two sets each. These are then arranged in a Latin rectangle with super-sets as columns.

super-set 1	super-set 2	super-set 3
set1	set2	set3
set2	set3	set1

Each super-set is replicated in $\frac{k}{q_3}$ columns, so that each product occurs in $r_s = \frac{q_2}{q_3}$ sessions. Therefore only preparation constraints that are multiples of $\binom{q_1}{q_2}$ can be used for the preparation design. Thus, this procedure is most useful for cases where $q_1 = q_2 + 1$, i.e. $q_3 = q_1$. In most other cases, $q_3 = \binom{q_1}{q_2}$ is too large to be useful. So, it will henceforth be assumed that $q_1 = q_2 + 1$. Fractions most likely to be useful are $q = \frac{3}{2}$ and $q = \frac{4}{3}$. Equation (5.2) changes for this extension of the procedure to

$$\frac{q_1}{q_1 - 1} = \frac{p}{s} = \frac{k}{r_s} = \frac{t}{r} \tag{5.3}$$

If $q_1 = q_2 + 1$ the preparation designs $D^{\frac{q_1}{q_1-1}}(p, s, k)$ can be constructed with Algorithm JE2 with the following design parameters: p products in s sessions of size

k with a constraint that $\frac{r_s}{(q_1-1)}$ entries in each block are from each of q_1 sets. The auxiliary design needs to be modified, so that it creates the Latin rectangle structure of the super-sets.

Example 5.22 A preparation design $D^{\frac{3}{2}}(9, 6, 6)$.

Design from JE2

2	3	6	4	9	7
2	1	6	5	7	8
1	3	5	4	9	8
1	2	6	4	9	8
3	2	4	5	7	8
1	3	6	5	7	9

Auxiliary design

1	1	4	4	7	7
2	2	5	5	8	8
3	3	6	6	9	9
4	4	7	7	1	1
5	5	8	8	2	2
6	6	9	9	3	3

Final design

2	3	4	6	7	9
1	2	6	5	8	7
3	1	5	4	9	8
4	6	8	9	1	2
5	4	7	8	2	3
6	5	9	7	3	1

The product replication in the preparation design is $r_s = 4$; each product will occur in 4 sessions of the complete sensory design. There are $\frac{r_s}{q_2} = 2$ columns that refer to each of the 3 super-sets.

The constraint on the panel design also changes from the regular procedure. To achieve r replications of all products in the whole design, $r_f = \frac{t}{q_1}$ columns of the preparation design have to be chosen from each of the q_1 super-sets. Therefore, the panel designs constructed with the previously introduced panel algorithms can be used. A panel design for a assessors with preparation constraint k and assessor constraint t will be necessary, where the t elements consist of r_f elements from each of the q_1 sets, i.e. $D_{\mathrm{CO}}^{\frac{q_1}{q_1-1}}(k, a, t) = D_{\mathrm{CO}}^{q_1}(k, a, t)$.

Example 5.23 A panel design $D_{CO}^{\frac{s}{2}}(6,12,3) = D_{CO}^{3}(6,12,3)$.

2	5	4
3	2	6
4	5	1
5	1	4
6	3	1
2	3	5
4	6	2
1	4	6
3	1	5
1	6	3
6	4	2
5	2	3

The preparation design and the panel design can be combined in the usual way, and $n = (r_s!)^{q_1} q_1!$ column permutations of the preparation design have to be considered.

Example 5.24 A sensory design $D_R^{\frac{s}{2}}(9,12,3,4;6)$.

	Session 1			Session 2			Session 3			Session 4			Session 5			Session 6		
Assessor 1	7	3	4	8	2	6	9	1	5	1	6	8	2	4	7	3	5	9
Assessor 2	6	7	2	5	8	1	4	9	3	9	1	4	8	2	5	7	3	6
Assessor 3	4	3	9	6	2	7	5	1	8	8	6	2	7	4	3	9	5	1
Assessor 4	3	9	4	2	7	6	1	8	5	6	2	8	4	3	7	5	1	9
Assessor 5	2	6	9	1	5	7	3	4	8	4	9	2	5	8	3	6	7	1
Assessor 6	7	6	3	8	5	2	9	4	1	1	9	6	2	8	4	3	7	5
Assessor 7	4	2	7	6	1	8	5	3	9	8	4	1	7	5	2	9	6	3
Assessor 8	9	4	2	7	6	1	8	5	3	2	8	4	3	7	5	1	9	6
Assessor 9	6	9	3	5	7	2	4	8	1	9	2	6	8	3	4	7	1	5
Assessor 10	9	2	6	7	1	5	8	3	4	2	4	9	3	5	8	1	6	7
Assessor 11	2	4	7	1	6	8	3	5	9	4	8	1	5	7	2	6	9	3
Assessor 12	3	7	6	2	8	5	1	9	4	6	1	9	4	2	8	5	3	7

According to this design each product is tasted twice by every assessor. Its efficiency according to model (4.8) is 87.11% and according to model (4.9) 57.42%.

With this extension of the three-step procedure the range of parameter combinations for which sensory designs can be generated is increased with only slight changes to the regular procedure. With this adaptation, the product replication can be handled more flexibly.

5.8 Conclusions

In many situations, multi-session designs are needed for which only a subset of all the products can be prepared for each session. General resolvable designs are not useful in these cases. In this chapter, a three-step procedure has been developed for generating efficient multi-session trials with preparation constraints. The procedure has been fully automated and generates the required designs within minutes. For the preparation and the panel design, existing algorithms have been modified to provide designs with the special structure and several techniques have been compared. It has been shown that algorithm JE2 can be used for the construction of efficient preparation designs and for generating the incomplete block structure of the panel design. The cross-over structure can then be induced with a regular algorithm. For this dissertation, algorithm B1 has been used.

We showed, that the three-step procedure is an improvement over the common strategy of forming multi-session designs for which only t products are prepared for each session. This procedure can be used for varies choices of the parameter constraint k and for a wide range of sensory designs. The only constraints on the parameter settings are given by equations (5.2) or (5.3).

An automated procedure enables us to generate designs for a range of parameter settings for a specific experiment and compare the resulting efficiencies. This is especially relevant for different numbers of assessors or different preparation constraints.

Chapter 6

Treatment-control designs for sensory trials

In the previous three chapters, designs were constructed for sensory trials in which all elementary contrasts are of equal interest. For such experiments, equally replicated designs are optimal, and all contrasts are estimated with equal precision. When a **control product** is included in the product range the degree of interest may vary between different elementary contrasts and designs can be constructed that reflect this change in priorities. To distinguish between the control and the other products, non-control products will be referred to in this chapter as **test products**.

In experiments that include a control product, there are contrasts of two kinds,

a) contrasts between the test products and the control product and

b) contrasts between two test products.

Comparisons of each test product with the control are of equal importance, as are the comparisons of each pair of test products. But the test-control comparisons are more important than the test-test comparisons. Then designs are needed in which control-test comparisons are made with higher precision than test-test comparisons. Such designs are called treatment-control designs. Higher precision of the test-control comparisons is achieved by replicating the control product more often than the test products.

The control in sensory studies refers to a reference product which at the planning stage already has a special status among the products. It can be the market leader, the cheapest product on the market or some other key product of a competitor. Interest is in finding the key sensory difference between the test products and the competitor product. Alternatively, a control might be the current commercial product and the test products its various modifications. The aim of such a trial is to determine which of the products will perform better or at least as well as the current product, seeking a decision to retain or replace the current product, for example, with one that has a better formulation or cheaper production. Interest is then solely in the comparisons between the control and the test products. This is often the case in the early stage of a long term trial, intended to compare the performance of the new products with the established control. The selected test products are then studied more intensively in the confirmatory stage of the experiment, in which both groups of contrasts are of interest, while not necessarily to the same degree.

In sensory designs without control products, two different situations have to be considered: single- and multi-session experiments. In single-session experiments each assessor tastes either all products (complete block designs) or only a subset of all products (incomplete block designs). For complete block designs, control-treatment designs are not relevant because block effects are orthogonal and all product comparisons are made with the same precision. Only when the block size is smaller than the number of products does the assignment of products to blocks change the precision of a specified contrast. In incomplete block designs, within-block (direct) comparisons are estimated with higher precision than between-block (indirect) comparisons and the use of treatment-control designs guarantees efficient estimates, when interest is mostly in test-control contrasts.

Whilst for sensory designs it is generally advisable that every assessor tastes all products the same number of times, this can be slightly relaxed for sensory treatment-control designs, so that every assessor tastes all v test products equally often, say r_t times, with the exception of the control product, which is replicated more often, say r_c times. In multi-session trials, the designs need to be resolvable for test treatments. The control, on the other hand, occurs in $r_c > r_t$ blocks, so that $r_t v + r_c = st$. Each set of s blocks that deliver one resolution of the $v + 1$ products makes up all the servings for a single assessor.

Single- and multi-session treatment-control designs have to be distinguished also by the assumed blocking factors in the analysis model. If the only blocking factor in the model is the assessor effect, then randomly assigned incomplete block treatment-control designs will be used. If order effects are also taken into account, row-column treatment-control designs are adequate and when first-order carry-over effects are also included in the model, cross-over treatment-control designs are chosen. This applies also for multi-session trials, but here blocks are represented by assessors within sessions and resolvable treatment-control designs are needed, in the sense of resolvable as described above.

In Section 6.1, an overview of the relevant literature on randomised, row-column and cross-over single-session treatment-control designs is given. More extensive reviews on the theory of treatment-control designs can be found in Hedayat, Jacroux and Majumdar (1988) and Majumdar (1996). In this chapter, only treatment-control designs with a single control product are considered, so that the p products of a sensory experiment consist of $v = p - 1$ test products and one control. The main purpose of this chapter is to develop construction procedures for sensory treatment-control designs for non-factorial multi-session trials with preparation constraints. Two procedures are introduced in Section 6.2, both an adaptation of the three-step procedure for treatment-control designs. The first procedure creates treatment-control designs, in which the control occurs in every block. This is discussed in

Section 6.2.1. The second procedure is for designs in which the control occurs only in some of the blocks, but in more blocks than either of the test products. This procedure is developed in Section 6.2.2.

6.1 Single-session designs

Before different classes of treatment-control designs can be described in detail some notation is needed. In single-session sensory trials there are $p = v + 1$ products, of which t are served to each of a assessors. This implies that all blocks in design d have equal block size t. As a convention in this text, product 1 will always be the control. Let $d \in D(v + 1, a, t)$ be a connected block design in the class of designs with $v + 1$ treatments arranged in a blocks of size t. Let N be the incidence matrix of d with a columns and $v + 1$ rows, where the elements of $N = [n_{ij}]_{(v+1,a)}$ represent the number of times each product occurs in a block and N_i denotes the ith row of N, i.e. the incidence vector for product i. Let $\lambda_{ij} = N_i N_j' = \sum_{l=1}^{a} n_{il} n_{jl}$ give the number of times product i and j occur together in the same block.

6.1.1 Randomised incomplete block treatment-control designs

The more frequent location of the control is not a consideration in the model choice, so the three linear models (3.16), (3.21) and (3.22) are also relevant for the analysis of data from treatment-control designs. The information matrices for the product effects are slightly more complex than their counterparts in equally replicated designs.

For model (3.16)

$$Y_{jk} = \mu + \alpha_j + \tau_k + \epsilon_{jk} \quad 1 \le j \le a, \quad 1 \le k \le p$$

the information matrix of a treatment-control design changes to

$$C = \boldsymbol{r}_P^\delta - \frac{1}{t}NN'$$

where $\boldsymbol{r}'_P = [r_1\, r_2\, \cdots\, r_{v+1}] = [r_c\, r_t\, \ldots\, r_t]$ is the replication vector for the $v+1$ products.

Treatment-control designs for model (3.16) are called balanced when all v test products occur the same number of times in a block with the control product and test products occur the same number of times in a block with any other product apart from itself. They are known under several names, such as **designs with supplemented balance** (Pearce, 1960), **balanced treatment block designs** (Jacroux, 1989) and for incomplete block designs they are called **balanced treatment incomplete block designs** introduced by Bechhofer and Tamhane (1981). The definition of Bechhofer and Tamhane will be used here since interest in sensory experiments is mostly in incomplete block designs.

Definition 6.1 (Balanced treatment incomplete block designs (BTIBD)) *A design $d \in D(v+1, a, t)$ with $t < v$ is a BTIB(v,a,t) if*

$$\lambda_{12} = \ldots = \lambda_{1v+1} = \lambda_c$$

and

$$\lambda_{23} = \lambda_{24} = \ldots = \lambda_{v-1,v} = \lambda$$

In other words, for a binary BTIB design the concurrence matrix $N_A N'_A$ has the pattern

$$N_A N'_A = \begin{bmatrix} r_c & \lambda_c \mathbf{1}'_{v+1} \\ \lambda_c \mathbf{1}_{v+1} & \lambda J_{v+1,v+1} + (r_t - \lambda) I_{v+1} \end{bmatrix}$$

Bechhofer and Tamhane (1981) show that for BTIB designs the linear estimates of the control-test contrasts have equal variances

$$\text{var}(\widehat{\tau_i - \tau_1}) = \frac{t(\lambda_c + \lambda)}{\lambda_c(\lambda_c + p\lambda)}\,\sigma^2$$

and equal correlation

$$\text{corr}(\widehat{\tau_i - \tau_1}, \widehat{\tau_{i'} - \tau_1}) = \frac{\lambda}{\lambda_c + \lambda}$$

for $i', i = 2, \ldots, v+1$ with $i \neq i'$.

A special subset of BTIB designs for which the number of control products in a block differs by at most one is defined by Stufken (1987):

Definition 6.2 *For integers $u \in \{0, 1, \ldots, t-1\}$, $s \in \{0, 1, \ldots, a-1\}$, design d with $v+1$ treatments in a blocks of size t is a BTIB$(v, a, t; u, s)$ if it is a BTIB(v,a,t) design with the additional property that*

$$n_{ij} \in 0, 1, \quad i = 2, 3, \ldots, v+1, \; j = 1, 2, \ldots, a$$

$$n_{12} = \ldots = n_{1s} = u + 1$$

$$n_{1,s+1} = \ldots = n_{1,a} = u$$

If $s = 0$, then the BTIB$(v, a, t; u, 0)$ is called a **rectangular (R) type design.** *When $s > 0$ then the BTIB$(v, a, t; u, s)$ is called a* **step (S) type design.**

This definition can be generalised to unbalanced treatment-control designs. If the control product occurs u times in every block of an incomplete block design, the design is called a **rectangular-type treatment-control design.** If the control product occurs in a subset of blocks u times and in the remaining blocks $u-1$ times, the design is called a **step-type treatment-control design.**

Example 6.1 Sub-classes of treatment-control designs.

Rectangular-type TC design

$u = 1,\ s = 0$

1	*2*	*3*	*5*
1	*3*	*4*	*6*
1	*4*	*5*	*7*
1	*5*	*6*	*8*
1	*6*	*7*	*2*
1	*7*	*8*	*3*
1	*8*	*2*	*4*

Step-type TC designs

$u = 0,\ s = 6$

2	*4*	*5*	*6*
1	*3*	*5*	*6*
1	*3*	*4*	*6*
1	*2*	*5*	*6*
1	*2*	*4*	*5*
1	*2*	*3*	*5*
1	*2*	*3*	*4*

$u = 1, s = 3$

1	*2*	*3*	*4*
1	*2*	*3*	*4*
1	*2*	*3*	*4*
1	*2*	*3*	*4*
1	*1*	*2*	*3*
1	*1*	*2*	*4*
1	*1*	*3*	*4*

Rectangular-type BTIB designs are also known as **balanced control incomplete block designs** (Spurrier and Edwards, 1986), **augmented designs** (e.g. Majumdar and Notz, 1983) and **reinforced block designs** (Das, 1958). For rectangular BTIB$(v, a, t; u, 0)$ designs the control concurrence is

$$\lambda_c = \frac{au(t-u)}{v}, \lambda = \frac{a(t-u)(t-u-1)}{v(v-1)}.$$

While BTIB designs exist only for designs of special size, Jacroux (1989) introduces two wider classes of designs, group divisible treatment designs and regular graph treatment designs. He shows that some designs of these two classes are A-optimal in the class of treatment-control designs, when BTIB designs of that size do not exist.

Definition 6.3 (Group divisible treatment designs (GDTD)) *A design $d \in D(v+1, a, t)$ with control replication r_c is called an GDTD with parameters m, n, λ_c, λ_1, and λ_2 if the products $1, \ldots, v+1$ can be partitioned into $m+1$ disjoint groups $V_c, V_1, \ldots, V_m$ of size $v_c, v_1, \ldots, v_m$, so that the following conditions are satisfied:*

a) $V_c = \{1\}$

b) $v_1 = \ldots = v_m = n$.

c) $\lambda_{12} = \lambda_c = (r_ct - R(r_c))/v$ *for* $l = 2, \ldots, v+1$ *and an appropriate constant* λ_c.

d) For $p, q \in V_i$ $(p \neq q; i = 1, \ldots, m)$, $\lambda_{pq} = \lambda_1$.

e) For $p \in V_i$, $q \in V_j$ $(i, j = 1, \ldots, m; i \neq j)$, $\lambda_{pq} = \lambda_2$.

where

$$R(r_c) = \left(r_c - a\left[\frac{r_c}{a}\right]\right)\left(\left[\frac{r_c}{a}\right]+1\right)^2 + \left(a - r_c + a\left[\frac{r_c}{a}\right]\right)\left[\frac{r_c}{a}\right]^2 \tag{6.1}$$

and $[x]$ denotes the integer part of a real number x.

This means that the off-diagonal elements of a concurrence matrix for GDTD comprise three distinct values, λ_c, λ_1 and λ_2 and a pair of test products occurs either λ_1 or λ_2 times, depending on whether they belong to the same group or not. If additionally $\lambda_1 = \lambda_2 + 1$, then a GDTD is called a regular graph treatment design.

Definition 6.4 (Regular graph treatment designs (RGTD)) *A design d in the class of $D(v + 1, a, t)$ with control replication r_c is called an RGTD if the following conditions are satisfied:*

a) $r_2 = \ldots = r_{v+1} = (at - r_c)/v = r_t$

b) $\lambda_{12} = \ldots = \lambda_{1v+1} = (r_c t - R(r_c))/v$

c) For all $i, j, p, q = 2, \ldots, v + 1, i \neq j, p \neq q$ and $|\lambda_{ij} - \lambda_{pq}| \leq 1$.

where $R(r_c)$ is defined by (6.1).

For both classes of designs, GDTD and RGTD, the control occurs in a block the same number of times with each test product. Compared to balanced designs this condition is relaxed only concerning the number of times test products occur with each other in a block.

6.1.2 Optimality criteria, efficiency and efficiency bounds

Optimality over all possible treatment-control designs of size (v, a, t) is not as simple to prove as in equally replicated designs. The main objective for treatment-control designs is to compare the control treatments with the test treatments. A meaningful criterion for design selection should therefore be based on the size of the variances of the control-treatment comparisons. Three criteria are most commonly used for assessing the quality of a treatment-control design:

a) A-optimality, also referred to as trace optimality, for which the average of the control-test contrast variances is minimised,

b) MV-optimality, for which the the maximum variance of the control-test contrasts is minimised,

c) Minimising the confidence region of the test-control contrasts (simultaneous confidence intervals).

In this thesis only the first of these three criteria is considered:

Definition 6.5 (A-optimality, trace optimality) *A design d is A-optimal in a given subclass D of $D(v+1,b,k)$ for estimating treatment differences involving the control treatment if it minimises*

$$\frac{1}{v}\sum_{i=1}^{v} var_d(\widehat{\tau_i - \tau_1})$$

over all designs in the class D.

Bechhofer and Tamhane (1981) show that for any $d \in D(v+1,b,k)$,

$$\frac{1}{v}\sum_{i=1}^{v}\text{var}(\widehat{\tau_i - \tau_1}) = \frac{1}{v}\text{trace}\left(C_{d11}^{-}\right) = \frac{1}{v}\text{trace}\left(L_c C_d^{-} L_c'\right) = \frac{1}{v}\sum_{i=1}^{v}\frac{1}{z_i}$$

where C_{d11} is the principal sub-matrix obtained from C_d after deleting row one and column one (associated with the control), $L_c = [1_v - I_v]$ is the matrix of test-control contrasts, and z_i the non-zero eigenvalues of C_{d11}^{-}.

In more recent publications, treatment-control designs are considered in which not only the precision of the control-test contrasts is optimised, but in which the precision of test-test is also considered, e.g. in Türe (1994) and Gupta et al. (1999). When both kinds of contrasts are of interest, the aim is to find designs, in which control-test contrasts are estimated with high precision and test-test contrasts with reasonably high precision. For a computer search of such designs the weighted A-optimality is useful.

Definition 6.6 (Weighted A-optimality) *With weights α and $\beta = 1 - \alpha$ a design d is called weighted A-optimal within a class $D(v+1, a, t)$ of designs, if it minimises*

$$\left[\beta \sum_{i=2}^{v+1} var(\hat{\tau}_1 - \hat{\tau}_i) + \alpha \sum_{i=1}^{v+1} \sum_{i'(\neq i)=1}^{v+1} var(\widehat{\tau_i - \tau_{i'}})\right]$$

For $\beta > \alpha$, test-control contrasts are more important than test-test contrasts. For $\alpha = 1, \beta = 0$ this definition reduces to trace-optimality, for $\alpha = \beta$ to the usual A-optimality, where all contrasts are equally important.

An efficiency measure for treatment-control designs has been defined by Stuffken (1988).

Definition 6.7 (Efficiency) *The efficiency E(d) of a design $d \in D(v+1, a, t)$ is defined as*

$$E^{d^*}_{A,c} = \frac{trace(L_c C^-_{d^*} L'_c)}{trace(L_c C^-_d L'_c)} = \frac{\sum_j var_{d^*}(\widehat{\tau_1 - \tau_j})}{\sum_j var_d(\widehat{\tau_1 - \tau_j})}$$

where d^* is a hypothetical A-optimal design in $D(v+1, a, t)$ for which $trace(L_c C^-_{d^*} L'_c)$ is minimum. $L_c C^-_{d^*} L'_c$ is the information matrix for the treatment-control contrasts.

Bounds for the average variance of control-test contrasts in treatment-control designs are known for balanced and regular graph treatment-control designs. They can either be given for designs with a fixed control replication r_c or for the optimal control replication r_o. The former is useful, when the control product supply is limited or the replication is fixed through financial or other external constraints. The latter is useful when the control replication can be chosen solely by statistical considerations.

Jacroux (1989) provides a bound on the A-efficiency for treatment-control designs with v test treatment in a blocks of size t and r_c fixed:

$$B_{r_c}(v,a,t) = \frac{vt}{r_c t - R(r_c)} + \frac{vt}{vat(t-1) - r_c(vt - v + t) + R(r_c)}$$

where the function $R(r)$ is defined as in Equation (6.1).

This bound is reached by a BTIB$(v+1,a,t)$ design with replication r_c, if such a design exists. For other settings certain classes of GDTD can be shown to be optimal. Details on tighter bounds for GDTD than the one above, which is only reached for BTIB designs, can be found in Jacroux (1989).

The advantage of rectangular designs is that they are easily constructed by augmenting BIB designs with the control in every block. While rectangular BTIB designs are A-optimal within the class of rectangular designs, they are not necessarily optimal in the class of all treatment-control designs of the same size. Stuffken (1987) shows that a BTIB$(v,b,k;t,0)$ is A-optimal for treatment-control contrasts whenever

$$(t-u-1)^2 + 1 \leq u^2 v \leq (t-u)^2$$

which simplifies to

$$(t-2)^2 \leq v \leq (t-1)^2 \text{ for } u = 1$$

The control replication for rectangular designs is always equal to the number of blocks, but it is not necessarily the optimal control replication for a design of the required size.

A lower bound for the average variance of control-test contrasts in treatment-control designs with optimal replication was derived by Majumdar and Notz (1983). They show that the lower bound B_{r_o} is the minimum of the function $g(r)$, defined as

$$g(r) = \frac{tv(v-1)^2}{avt(t-1) - r(tv - v + t) + h(r)} + \frac{tv}{tr - h(r)}$$

with

$$h(r) = a\left[\frac{r}{a}\right]^2 + \left(2\left[\frac{r}{a}\right] + 1\right)\left(r - a\left[\frac{r}{a}\right]\right)$$

for $r = 1, \ldots, a\left[\frac{t}{2}\right]$.

This bound is attained by a step-type BTIB$(v, b, k; u, s)$ design for which the values of s and u are defined from the optimal replication through the identity $r_o = bu + s$. Hedayat and Majumdar (1984) give the conditions under which such a balanced design exists:

a) A BTIB$(v, a, t; u, s)$ design (where $r_c = au + s$) exists if

$$\frac{a(t-1) - s}{v} = \frac{at - r_c}{v} = q_1$$

$$\frac{s(t-u-1)}{v} = q_2$$

$$\frac{q_2(t-u-2) + (q_1 - q_2)(t-u-1)}{v-1}$$

are integers; $r_c = au + s$.

b) For an R-type design it is necessary that $a \geq v$, while for an S-type design it is necessary that $a \geq v + 1$.

From this it can be seen that there have to be more blocks than products. For single-session incomplete block designs, this means that optimal sensory designs can only be found when the number of assessors is larger than the number of products.

Average efficiency as in Definition 6.7, where the average variance of the control-test contrasts is compared to one from an A-optimal treatment-control design, can be calculated only when lower bounds are known for the average variance of the control-test contrasts. Since for cross-over designs such a bound is not known, alternative measures are needed to compare different designs of the same size.

General average efficiency has been defined as:

$$\mathrm{E}^{o}_{A;\mathrm{r}} = \frac{\sum_{i,j}^{p} \mathrm{var}_o(\widehat{\tau_i - \tau_j})}{\sum_{i,j}^{p} \mathrm{var}_d(\widehat{\tau_i - \tau_j})}$$

by Definition 3.16, where $\mathrm{var}_o(\widehat{\tau_i - \tau_j})$ is the lower bound taken from the variance of the contrasts in a completely randomised design with the same number of treatments and the same replication. Efficiency defined in these terms measures the precision that is lost due to blocking, under the assumption that the error variance would be the same in the blocked design and the design without blocking, rather than giving a measure of how good the design is in comparison to the optimal design in the same class. For cross-over designs efficiency can be calculated for direct and residual treatment effects. The two advantages of this measure are that it can be calculated when lower bounds on the variance are unknown and that it can also be calculated for single contrasts, which is useful, for example, for factorial designs.

Average efficiency, as defined by Definition 3.16, is useful when all elementary contrasts are equally important. In treatment control designs the treatment-control contrasts are more important than the test-test contrasts. Average variance in this context refers to the average of all control-test contrast variances and average efficiency for treatment-control designs can, as an alternative to Definition 6.7, be defined as:

$$\mathrm{E}^{o}_{A,c} = \frac{\sum_{j=2}^{v} \mathrm{var}_o(\widehat{\tau_1 - \tau_j})}{\sum_{j=2}^{v} \mathrm{var}_d(\widehat{\tau_1 - \tau_j})}$$

(see also Definition 6.5 on A-optimality for treatment-control designs).

Often only the average efficiency of the control-test contrasts is given for treatment-control designs. When some interest is in test-test contrasts, information about the average efficiency of the test-test contrasts can also be useful to assess the quality of a design:

$$\mathrm{E}^{o}_{A,t} = \frac{\sum_{1<i<j}^{p} \mathrm{var}_o(\widehat{\tau_i - \tau_j})}{\sum_{1<i<j}^{p} \mathrm{var}_d(\widehat{\tau_i - \tau_j})}$$

Unfortunately there are no known lower bounds for the average variance of test-test contrasts in treatment-control designs.

The ratios $\mathrm{E}^{o}_{A,c}$ and $\mathrm{E}^{o}_{A,t}$ are used to compare designs of the same size and the same replication and assess the loss due to blocking. These two efficiencies cannot be compared directly to assess the gain in efficiency through the higher replication of the control. The average variance of the control-treatment contrasts is expected to be smaller than its test-test counterparts for both designs, the optimal and the design d under consideration. Rather than compare the efficiencies, the average variances can be compared directly. Thus, the ratio of the average variances of the test-test contrasts and the control-test contrasts provides a measure of the gain in precision for the control-test comparisons, which are of primary interest:

$$E_A^{\mathrm{ratio}} = \frac{2}{v+1} \frac{\sum_{1<i<j} \mathrm{var}_d(\widehat{\tau_i - \tau_j})}{\sum_{1<j} \mathrm{var}_d(\widehat{\tau_1 - \tau_j})}$$

This ratio has been suggested by Pigeon and Raghavarao (1987) and is a useful efficiency measure for all non-factorial treatment-control designs.

6.1.3 Row-column and cross-over treatment-control designs

As discussed in Chapter 3, in sensory experiments it is often useful to include order and carry-over effects in a model equation. Relevant designs for these cases differ from those with a single blocking effect. This is the same for treatment-control designs. Thus, row-column or cross-over treatment-control designs are required.

When assessor (α_j) and order (π_i) effects are assumed, model equation (3.21) can be used.

$$Y_{ijk} = \mu + \pi_i + \alpha_j + \tau_k + \epsilon_{ijk} \qquad 1 \leq j \leq a, \quad 1 \leq i, k \leq p$$

The information matrix in the treatment-control case reflects the unequal replication compared to designs with equal replication:

$$\begin{aligned} C_d &= \boldsymbol{r}_P^\delta - \frac{1}{a} N_T N_T' - \frac{1}{t} N_A N_A' + \frac{1}{at} \boldsymbol{r}_P \boldsymbol{r}_P' \\ &= \boldsymbol{r}_P^\delta - \frac{1}{a}\Gamma - \frac{1}{t}\Lambda + \frac{1}{at}\boldsymbol{r}_P \boldsymbol{r}_P' \\ &= \boldsymbol{r}_P^\delta - \Delta_P \end{aligned} \tag{6.2}$$

Balanced treatment-control designs in the row-column setting are known either as **balanced treatment row-column designs** (Türe, 1994) or as **balanced treatment vs. control row-column designs** (Majumdar and Tamhane, 1996).

Let $\lambda_{ij} = \sum n_{A;ik} n_{A;kj}$, $i, j = 1, \ldots, v+1$, be the matrix entries of the block concurrence matrix $N_A N_A'$, as before, and let $\gamma_{ij} = \sum n_{T;ik} n_{T;kj}$, $i, j = 1, \ldots, v+1$, be the matrix entries of the serving concurrence matrix $N_T N_T'$. Let $r_i = \sum_{j=1}^{a} n_{ij} = \sum_{k=1}^{t} n_i k$ be the replication of product i.

Definition 6.8 (Balanced Treatment Row-Column design (BTRCD)) *Let $\delta_{ii'}$ be the elements of Δ_P as defined in equation (6.2)*

$$\delta_{ii'} = \frac{\gamma_{ii'}}{a} + \frac{\lambda_{ii'}}{t} - \frac{r_i r_{i'}}{at}$$

A design d for $v+1$ treatments is a balanced treatment row-column design if

$$\delta_{12} = \ldots = \delta_{1v+1} = \delta_c$$

$$\delta_{23} = \ldots = \delta_{vv+1} = \delta_t$$

This implies that all treatment-control contrasts of a BTRCD have equal variance and equal correlation:

$$\mathrm{var}(\widehat{\tau_i - \tau_1}) = \frac{(\delta_c + \delta_t)}{\delta_c(\delta_c + p\delta_t)}\,\sigma^2$$

$$\mathrm{corr}(\widehat{\tau_i - \tau_1}, \widehat{\tau_{i'} - \tau_1}) = \frac{\delta_t}{\delta_c + \delta_t}$$

for $i', i = 2, \ldots, v+1$ with $i \neq i'$.

For row-column designs, Notz (1985) gave the upper bound for the average variance of the control-test contrasts as the minimum over $0 < r < at$ for the following function, which is attained for a BTRCD:

$$f(r) = \frac{v}{r + \frac{r^2}{at} - h^\star(r)} + \frac{v}{(v-1)\left((v-1)(at-r) - \frac{2r^2}{at} + h^\star(r)\right)}$$

where

$$h^\star = \frac{r + (2r-t)\left[\frac{r}{t}\right] - t\left[\frac{r}{t}\right]^2}{a} + \frac{\left(r + (2r-a)\left[\frac{r}{t}\right] - a\left[\frac{r}{t}\right]^2\right)}{t}.$$

When the selected model includes assessor, order and first-order carry-over effects $(\rho_{i-1,j})$, model equation (3.22) is required:

$$Y_{d(i,j)} = \mu + \pi_i + \alpha_j + \tau_{d(i,j)} + \rho_{d(i-1,j)} + \epsilon_{ij} \qquad 1 \leq i \leq t, \quad 1 \leq j \leq a,$$

The information matrix for direct product and residual product effects in the treatment-control case changes to

$$\begin{aligned} C_D &= C_{11} - C_{12}C_{22}^{-}C_{21} \\ C_R &= C_{22} - C_{21}C_{11}^{-}C_{12} \end{aligned}$$

where the matrices C_{11}, C_{12} have more complex forms due to the unequal product replication.

$$
\begin{aligned}
C_{11} &= \boldsymbol{r}_P^{\delta} - \frac{1}{a} N_T N_T' - \frac{1}{t} N_A N_A' + \frac{1}{at} \boldsymbol{r}_P \boldsymbol{r}_P' \\
C_{12} &= M - \frac{1}{a} N_T \tilde{N}_T' - \frac{1}{t} N_A \tilde{N}_A' + \frac{1}{at} \boldsymbol{r}_P \boldsymbol{r}_R'
\end{aligned}
$$

where $\boldsymbol{r}_R = X_P' X_R$ is the replication vector of the residual effects and $M = X_P' X_R$ the precedence matrix.

Pigeon and Raghavarao (1987) give the following definition for a balanced treatment-control design in the cross-over setting:

Definition 6.9 (Control balanced residual effects design) *An arrangement of $v+1$ products in t periods $(p \leq v+1)$ and a assessors such that every assessor receives a product in each period is said to be a* **control balanced residual effects design** *if*

a) no product is tasted more than once by an assessor;

b) the control product occurs ξ_c times in each period and each test product occurs ξ_t times in each period;

c) the control product occurs with each test product for λ_c assessors and each test product occurs with every other test product for λ assessors;

d) excluding the last period, the control product occurs with each test product for μ_c assessors and each test product occurs with every other test product for μ assessors;

e) the ordered product pairs $(1,i)$ and $(i,1)$ $(i = 1, \ldots, v+1)$ occur in successive periods for ν_c assessors and the ordered product pair (i,j) $(i \neq j; i,j = 2, \ldots, v+1)$ occurs in successive periods for ν assessors;

f) for every pair of distinct products θ and ϕ, the number of assessors for which θ occurs with ϕ in the last period is the same as the number of assessors for *which ϕ occurs with θ in the last period.*

The concurrence matrix $N_A N_A'$ of a control-balanced residual effects design has the same form as one from a BTIB design, while the period incidence matrix N_T and the precedence matrix M have the following form:

$$N_T = \begin{bmatrix} \xi_c & \cdots & \xi_c \\ \xi_t & \cdots & \xi_t \\ \vdots & & \vdots \\ \xi_t & \cdots & \xi_t \end{bmatrix}, \quad M = \begin{bmatrix} 0 & \nu_c & \cdots & \cdots & \cdots & \nu_c \\ \nu_c & 0 & \nu_t & & \cdots & \nu_t \\ \vdots & & \ddots & & & \\ \nu_c & \nu_t & & \cdots & 0 & \nu_t \\ \nu_c & \nu_t & & \cdots & \nu_t & 0 \end{bmatrix}$$

For cross-over designs upper bounds on the average variance of the test-control contrasts, similar to those of randomly assigned and row-column treatment-control designs are not known. Pigeon and Raghavarao (1987) propose a comparison of an elementary contrast between control-test products of a balanced treatment-control designs with t periods and v test products with that of an elementary contrast from a balanced cross-over design with t periods and $v+1$ treatments, where they also suggest some adjustment to counteract the different number of blocks needed for balance. This measure is introduced to provide information on the gain in precision for control-test contrasts, due to the use of control treatment-control design compared to equally replicated designs.

6.2 Resolvable treatment-control designs with preparation constraints

In sensory experiments involving multiple sessions, resolvable cross-over designs are used to accommodate assessor, session, serving order and first-order carry-over effects. In Chapter 5, the three-step procedure was introduced for the construction of multi-session designs, in which all elementary contrasts are of equal interest. Treatment-control designs will be discussed in more detail in this context. The three-step procedure can be modified to generate designs in which contrasts are estimated with unequal precisions. In the following two sections a construction procedure is developed for multi-session treatment-control designs with preparation constraints. This procedure can also be used to create simple multi-session designs as a special case.

Sensory designs are often limited by time and financial constraints as well as by product resources. Thus, here only designs are considered for which the replication of the control product for each assessor is limited by the number of blocks, i.e. $r_t < r_c \leq s$, so that the control occurs at most once in a block. The optimal replication might be higher, but such designs are in practise not feasible. Optimal replication is therefore of limited usefulness for sensory designs.

Increased replication of the control in the complete sensory design can be achieved either by increasing the replication of the control in the preparation design, in the panel design, or in both. The number of sessions a test product is served in, r_{st}, is smaller than the number of sessions the control product is served in, r_{sc}, which in turn does not exceed the number of sessions s. With block size k in the preparation design and t in the complete design, the number of design entries is

$$ks = vr_{st} + r_{sc} \tag{6.3}$$

for the preparation design and

$$ast = avr_t + ar_c$$

for the complete sensory design.

For binary treatment-control designs, the maximum replication of the control product is achieved when every assessor tastes the control product in every session together with $t-1$ test products (Design case A1); then $r_c = s$. In trials with a large number of assessors and/or sessions, this may result in a fairly high overall replication of the control product.

If such a high replication is undesirable or impossible because sufficient quantities of the control product are not available, the control product could be given only to some of the assessors, say y assessors, where $r_t a/s < y < a$. These y assessors would then taste the control in every session, while the remaining $a-y$ assessors would not taste the control product at all (Design case A2).

Alternatively, a reduction of the replication of the control could be achieved by reducing the number of sessions each assessor will taste the control in. For maximal control replication for this case, the control is still available in all sessions (Design case B1).

The control replication can still be reduced further, when the control is available only in a subset of all sessions. To ensure that the overall replication of the control is higher than for the test products, the number of sessions in which the control is available has to exceed the number of sessions the test products occur in (Design case B2).

To construct design cases A1 or A2, the preparation and the panel design have to be treatment-control designs. For designs of cases B1 and B2, only the preparation design has to be a treatment-control design. For designs A1, A2 and B1, the preparation design is a rectangular type treatment-control design, which ensures that the control occurs in every session. For design B2, the preparation design is a step-type design with $u = 0$, in which the control appears either once or not at all in a session. The structure of the design plans is summarised in Table 6.1, where 'No TCD' stands for no treatment-control design required (the panel designs are as equally replicated as possible for the required number of assessors).

Table 6.1: Types of treatment-control designs
Types are given for preparation and complete sensory designs for the four design cases.

	Preparation design	Panel design	Sensory design
A1	Rectangular	Rectangular	Rectangular
A2	Rectangular	Step-type	Step-type
B1	Rectangular	No TCD	Step-type
B2	Step-type	No TCD	Step-type

Table 6.2: Replication numbers for control and test products
Replication is listed for the three relevant design cases in the preparation and the complete sensory design.

	Complete design		Preparation design	
	Test	Control	Test	Control
A1	$r_t = \frac{s(t-1)}{v}$	$r_c = s$		
B1	$r_t = \frac{st(k-1)}{kv}$	$r_c = \frac{st}{k}$	$r_{st} = \frac{s(k-1)}{v}$	$r_{sc} = s$
B2	$r_t = \frac{st}{v+x}$ $2 < x < v/(k-1)$	$r_c = \frac{stx}{v+x}$ $v/(k-1)$	$r_{st} = \frac{sk}{v+x}$ $2 < x < v/(k-1)$	$r_{sc} = xr_{pc},$ $2 < x < v/(k-1)$

Designs of type A2 are not very useful for the sensory setting due to the large scoring differences between assessors. In this setup, $a - y$ assessors would not taste the control product at all. When interest is mainly in the test-control contrasts, it is preferable to rely on within-assessor contrasts of the test and control products. Hence, case A2 will not be pursued any further; it was introduced only for completeness.

The resulting replication of control and test products for the three relevant design cases A1, B1 and B2 are given in Table 6.2. The replication in the preparation design gives the number of sessions a products needs to be prepared for. When product resources are limited it is useful to check first, which of the three design cases can be used.

The construction method for case A1 is explained in Section 6.2.1. The sensory designs for this case are rectangular treatment-control designs, in which all assessors taste the control in every session. The complete sensory designs of case B1 and B2 are both step-type designs. Case B1 is essentially a special case of B2, where the preparation design is a rectangular treatment-control design. Therefore both can be constructed with the same procedure that will be described in Section 6.2.2.

6.2.1 Sensory designs with a control in every block

A rectangular sensory design for case A1 can be created by augmenting a balanced or nearly balanced sensory design. If a treatment-control design for $v+1$ products is needed, which are served in s sessions with t servings and a preparation constraint k, a sensory design with equal replication for v products in s sessions of $t-1$ servings with a product constraint of $k-1$ is used as the base design. Such a design can be constructed with the three-step procedure described in Chapter 5 and is then augmented with a control product in every block.

While the augmentation process takes care of the efficient assignment of products to blocks, the resulting design also needs to be optimised for its order and carry-over structure. A search algorithm is needed that transforms the augmented design into an efficient first-order cross-over design. A candidate for this is the algorithm by Ball (1997), which was applied in Chapter 5 to generate cross-over panel designs. The original form of the algorithm is described in detail in Section 3.5.2 (algorithm B1). Its adaptation for rectangular-type treatment-control designs requires two changes:

a) As most sensory designs are large, the number of possible product arrangements can be immense and a search for an optimal cross-over design may take a very long time. To simplify the process, the algorithm is modified to search only within a subgroup of all permutations.

b) The optimality criterion needs to be adapted for treatment-control designs, where treatment-control contrasts are of prime interest.

The first change limits the class of possible swaps: only permutations of the control product with a test product are considered for improvement of the objective function, no swaps between two different test products are made. In a randomly chosen block, a swap between the control product and a randomly chosen test product from that block is examined. If it improves the optimality criterion the swap is accepted; otherwise it is rejected and the next random swap is considered. This is continued until no more swaps are found that improve the criterion.

While these interchanges might not produce an optimal design due to the above limitations, they should still be sufficient to create an efficient cross-over design, since the base design without the control is already an efficient cross-over design, as shown in Chapter 5.

The second change modifies the original optimality criterion,

$$c(d) = w_{\text{order}}\mathrm{MS}(N_T) + w_{\text{carry-over}}\mathrm{MS}(M)$$

the weighted sum of the mean squares (MS) of the order incidence matrix N_T and the precedence matrix M.

The minimum for this criterion is achieved for a balanced cross-over design. For treatment-control cross-over designs, in which control products are replicated more often than test products, the values of these two matrices are expected to be higher for the entries related to the control product than either test product and are only expected to be the same within each of the two groups (see Definition 6.9).

N_T and M are therefore partitioned into entries related to the control and to the test products and the mean square of the entries of each block is calculated separately.

$$N_T = \begin{array}{|c|c} \multicolumn{1}{c}{1 \quad \cdots \quad t} & \\ \hline N_T(c) & 1 \\ \hline & 2 \\ N_T(t) & \vdots \\ & v+1 \\ \hline \end{array} \qquad M = \begin{array}{|c|c|c} \multicolumn{1}{c}{1} & \multicolumn{1}{c}{2 \quad \cdots \quad v+1} & \\ \hline 0 & M(c) & 1 \\ \hline & & 2 \\ M(c) & M(t) & \vdots \\ & & v+1 \\ \hline \end{array}$$

The order incidence matrix is split into the first row $N_T(c)$, which is related to the control $N_T(c)$ and the remaining rows $N_T(t)$ related to the test products. The part of the precedence matrix M related to the control comprises the elements m_{1i} and m_{i1}, for all $i = 2, \ldots, v+1$. The elements of M related to the test products are m_{ij} for all $i, j = 2, \ldots v+1,\ i \neq j$. The part of the optimality criterion related to each matrix is then the sum from both means squares, resulting in the following modified optimality criterion:

$$\begin{aligned} c(d) &= w_{\text{order}} \left(\text{MS}(N_{T,\text{control}}) + \text{MS}(N_{T,\text{test}}) \right) \\ &+ w_{\text{carry-over}} \left(\text{MS}(M_{\text{control}}) + \text{MS}(M_{\text{test}}) \right) \end{aligned}$$

The algorithm optimises the cross-over structure over the whole design.

Example 6.2 Rectangular-type treatment control design $D_{\text{R}}^{\text{TCR}}(17, 12, 9, 5; 9)$ with a control in every block.

The treatment-control design is constructed from the sensory multi-session design $D_R(16, 12, 9, 4; 8)$ and is augmented with a control in every block. The augmented design without the cross-over structure is shown in Table 6.3. The control replication in this example is $r_c = 108$ and the test products replication $r_t = 24$, so the control is replicated 4.5 times as often as the test products. The designs with the cross-over structure, induced with Algorithm B2, is shown in Table 6.4.

Table 6.3: The complete sensory design from Example 5.5, augmented with a control in every block.

	Session 1					Session 2					Session 3					Session 4					Session 5				
Assessor 1	**1**	17	6	3	11	**1**	18	7	4	12	**1**	19	8	5	13	**1**	11	9	6	14	**1**	12	10	7	15
Assessor 2	**1**	11	4	6	15	**1**	12	5	7	16	**1**	13	6	8	17	**1**	14	7	9	18	**1**	15	8	10	19
Assessor 3	**1**	15	3	4	11	**1**	16	4	5	12	**1**	17	5	6	13	**1**	18	6	7	14	**1**	19	7	8	15
Assessor 4	**1**	4	12	17	3	**1**	5	13	18	4	**1**	6	14	19	5	**1**	7	15	11	6	**1**	8	16	12	7
Assessor 5	**1**	4	2	15	17	**1**	5	3	16	18	**1**	6	4	17	19	**1**	7	5	18	11	**1**	8	6	19	12
Assessor 6	**1**	6	17	12	4	**1**	7	18	13	5	**1**	8	19	14	6	**1**	9	11	15	7	**1**	10	12	16	8
Assessor 7	**1**	12	15	3	2	**1**	13	16	4	3	**1**	14	17	5	4	**1**	15	18	6	5	**1**	16	19	7	6
Assessor 8	**1**	6	15	2	17	**1**	7	16	3	18	**1**	8	17	4	19	**1**	9	18	5	11	**1**	10	19	6	12
Assessor 9	**1**	12	11	2	6	**1**	13	12	3	7	**1**	14	13	4	8	**1**	15	14	5	9	**1**	16	15	6	10
Assessor 10	**1**	3	17	11	2	**1**	4	18	12	3	**1**	5	19	13	4	**1**	6	11	14	5	**1**	7	12	15	6
Assessor 11	**1**	3	6	15	12	**1**	4	7	16	13	**1**	5	8	17	14	**1**	6	9	18	15	**1**	7	10	19	16
Assessor 12	**1**	2	11	12	4	**1**	3	12	13	5	**1**	4	13	14	6	**1**	5	14	15	7	**1**	6	15	16	8

	Session 6					Session 7					Session 8					Session 9				
Assessor 1	**1**	13	2	8	16	**1**	14	3	9	17	**1**	15	4	10	18	**1**	16	5	2	19
Assessor 2	**1**	16	9	2	11	**1**	17	10	3	12	**1**	18	2	4	13	**1**	19	3	5	14
Assessor 3	**1**	11	8	9	16	**1**	12	9	10	17	**1**	13	10	2	18	**1**	14	2	3	19
Assessor 4	**1**	9	17	13	8	**1**	10	18	14	9	**1**	2	19	15	10	**1**	3	11	16	2
Assessor 5	**1**	9	7	11	13	**1**	10	8	12	14	**1**	2	9	13	15	**1**	3	10	14	16
Assessor 6	**1**	2	13	17	9	**1**	3	14	18	10	**1**	4	15	19	2	**1**	5	16	11	3
Assessor 7	**1**	17	11	8	7	**1**	18	12	9	8	**1**	19	13	10	9	**1**	11	14	2	10
Assessor 8	**1**	2	11	7	13	**1**	3	12	8	14	**1**	4	13	9	15	**1**	5	14	10	16
Assessor 9	**1**	17	16	7	2	**1**	18	17	8	3	**1**	19	18	9	4	**1**	11	19	10	5
Assessor 10	**1**	8	13	16	7	**1**	9	14	17	8	**1**	10	15	18	9	**1**	2	16	19	10
Assessor 11	**1**	8	2	11	17	**1**	9	3	12	18	**1**	10	4	13	19	**1**	2	5	14	11
Assessor 12	**1**	7	16	17	9	**1**	8	17	18	10	**1**	9	18	19	2	**1**	10	19	11	3

Table 6.4: Sensory cross-over design with a control product in every block

	Session 1					Session 2					Session 3					Session 4					Session 5				
Assessor 1	11	17	6	3	1	4	18	7	1	12	8	19	1	5	13	6	11	9	1	14	10	12	1	7	15
Assessor 2	11	1	4	6	15	16	12	5	7	1	6	13	8	1	17	18	14	1	9	7	1	15	8	10	19
Assessor 3	1	15	3	4	11	12	16	1	5	4	5	17	1	6	13	14	18	6	1	7	1	19	7	8	15
Assessor 4	1	3	12	17	4	4	5	13	18	1	6	1	14	19	5	6	7	15	1	11	16	8	1	12	7
Assessor 5	1	4	2	15	17	3	5	1	16	18	1	6	4	17	19	5	7	1	18	11	19	8	6	1	12
Assessor 6	4	6	17	1	12	18	7	1	13	5	8	1	19	14	6	1	9	11	15	7	8	10	12	16	1
Assessor 7	15	12	3	1	2	16	13	1	4	3	1	14	17	5	4	18	15	1	6	5	19	16	7	6	1
Assessor 8	15	6	17	2	1	1	7	16	3	18	1	8	17	4	19	18	1	9	5	11	12	10	19	1	6
Assessor 9	12	1	11	2	6	7	13	12	3	1	13	14	1	4	8	9	15	14	5	1	1	16	15	6	10
Assessor 10	17	3	2	11	1	3	4	18	12	1	19	1	5	13	4	14	6	11	5	1	6	7	12	15	1
Assessor 11	15	3	6	1	12	7	4	1	16	13	14	5	8	17	1	15	6	9	1	18	16	7	10	1	19
Assessor 12	2	1	11	12	4	1	3	12	13	5	4	1	13	14	6	15	1	14	5	7	15	1	6	16	8

	Session 6					Session 7					Session 8					Session 9				
Assessor 1	2	13	1	8	16	9	14	3	1	17	10	15	4	1	18	16	1	5	2	19
Assessor 2	16	1	9	2	11	12	1	10	3	17	13	18	2	4	1	5	19	3	1	14
Assessor 3	11	16	8	9	1	12	17	9	10	1	1	13	10	2	18	2	14	1	3	19
Assessor 4	17	9	13	1	8	9	1	18	14	10	19	2	1	15	10	3	2	11	16	1
Assessor 5	13	9	7	11	1	1	10	8	12	14	9	2	1	13	15	1	3	10	14	16
Assessor 6	2	1	13	17	9	10	3	14	18	1	4	1	15	19	2	1	5	3	11	16
Assessor 7	11	17	1	8	7	9	18	12	1	8	13	1	19	10	9	14	11	1	2	10
Assessor 8	11	1	2	7	13	1	3	12	8	14	1	4	13	9	15	10	5	14	1	16
Assessor 9	17	1	16	7	2	18	1	17	8	3	1	19	4	9	18	11	5	19	10	1
Assessor 10	16	8	13	1	7	1	9	14	17	8	1	10	15	18	9	16	2	19	1	10
Assessor 11	8	1	2	11	17	18	9	3	12	1	1	10	19	13	4	2	14	5	1	11
Assessor 12	16	7	1	17	9	17	8	10	18	1	2	9	18	19	1	19	10	1	11	3

Test-test contrasts in this multi-session treatment-control design are estimated on average with 1.64 times larger variance than control-test contrasts, which is the efficiency measure E_A^{ratio}. The order incidence matrix and the precedence matrix of this design are:

$$N_t = \begin{array}{|ccccc|}
\hline
22 & 21 & 21 & 21 & 23 \\
\hline
6 & 4 & 5 & 6 & 3 \\
3 & 7 & 6 & 5 & 3 \\
5 & 4 & 4 & 5 & 6 \\
3 & 6 & 4 & 7 & 4 \\
5 & 5 & 5 & 5 & 4 \\
2 & 7 & 4 & 4 & 7 \\
4 & 5 & 5 & 5 & 5 \\
5 & 6 & 5 & 4 & 4 \\
4 & 6 & 5 & 4 & 5 \\
6 & 2 & 5 & 5 & 6 \\
5 & 3 & 7 & 5 & 4 \\
4 & 5 & 6 & 5 & 4 \\
4 & 6 & 6 & 4 & 4 \\
6 & 5 & 4 & 4 & 5 \\
9 & 4 & 2 & 5 & 4 \\
4 & 4 & 5 & 6 & 5 \\
6 & 4 & 3 & 5 & 6 \\
5 & 4 & 6 & 3 & 6 \\
\hline
\end{array}
\qquad
M = \begin{array}{|c|cccccccccccccccccc|}
\hline
0 & 4 & 5 & 5 & 5 & 5 & 4 & 5 & 5 & 5 & 5 & 5 & 5 & 5 & 4 & 5 & 4 & 4 & 5 \\
\hline
5 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 4 & 0 & 1 & 2 & 1 & 0 & 0 & 1 & 2 \\
5 & 2 & 0 & 2 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 4 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\
5 & 1 & 1 & 0 & 1 & 2 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 2 & 1 \\
4 & 1 & 1 & 2 & 0 & 0 & 3 & 1 & 0 & 0 & 1 & 0 & 3 & 1 & 0 & 0 & 1 & 0 & 2 \\
5 & 0 & 1 & 1 & 1 & 0 & 2 & 0 & 1 & 1 & 2 & 0 & 2 & 0 & 1 & 1 & 2 & 0 & 0 \\
5 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 2 & 0 & 2 & 1 & 0 & 0 & 0 \\
4 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 3 & 0 & 1 & 1 & 1 & 1 & 1 & 2 & 0 & 1 \\
4 & 2 & 1 & 0 & 1 & 0 & 2 & 0 & 0 & 1 & 1 & 0 & 1 & 2 & 2 & 0 & 0 & 3 & 0 \\
4 & 1 & 2 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 2 & 0 & 1 & 2 & 0 & 0 & 1 & 3 \\
5 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 3 & 3 & 0 & 0 \\
6 & 0 & 2 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 0 & 0 \\
5 & 0 & 0 & 2 & 2 & 0 & 0 & 1 & 2 & 1 & 0 & 1 & 0 & 2 & 1 & 0 & 1 & 2 & 0 \\
4 & 0 & 1 & 0 & 4 & 3 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 2 & 2 & 1 \\
5 & 0 & 2 & 1 & 0 & 3 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\
5 & 1 & 1 & 0 & 0 & 0 & 4 & 4 & 0 & 0 & 0 & 1 & 2 & 0 & 1 & 0 & 0 & 1 & 0 \\
5 & 1 & 1 & 2 & 1 & 1 & 0 & 3 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
5 & 1 & 0 & 0 & 0 & 1 & 2 & 0 & 2 & 0 & 1 & 2 & 0 & 2 & 1 & 0 & 0 & 0 & 1 \\
5 & 2 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 3 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\
\hline
\end{array}$$

This augmentation procedure, which creates a treatment-control design with a control in every block, can be used with all equally replicated sensory designs. They include all the designs created with the three-step procedure as described in Chapter 5.

6.2.2 The treatment-control preparation design

In design cases B1 and B2 the availability of the control product is increased compared to a regular equally replicated design, but it is no longer necessary that every assessor tastes the control product in every session. Therefore only the preparation design needs the special treatment-control structure, not the panel design. The preparation design defines which subset of products is prepared for each session.

Should the control product be available for every session, it has to occur in every block in the preparation design; then the preparation design is constructed as a rectangular treatment-control design. For fewer replications of the control a step-type design is sufficient.

Rectangular designs for $v+1$ products in a blocks of size t can be constructed easily from efficient equally replicated IBD's for v products in a blocks of size $t-1$ by augmentation, i.e. by adding the control product to every block and then randomising the product order within the block. The construction of step-type designs is more difficult, since the control product occurs only in a subset of all blocks.

In Chapter 5 it has been shown that for equally replicated preparation designs, equation (5.2) $q = p/s = k/r_p$ has to hold. This ensures that the products can be divided into q distinct groups (or sets) to allow the panel design to arrange the products so that an assessor tastes all products the same number of times. For treatment control designs, equation (5.2) changes to

$$q = \frac{v+x}{s} = \frac{k}{r_t} \tag{6.4}$$

with integer x, $1 < x \leq v/(k-1)$. This identity is satisfied when the replication of the test products is $r_t = \frac{k}{q}$. In combination with equation (6.3) it implies that the control needs to be a multiple of the test product replication: $r_c = xr_t$. For binary designs, r_c is bounded by the number of sessions; $r_c \leq s = (v+x)/q$. The maximum value of x that still results in a binary design is $x = v/(k-1)$. Higher values of x result in designs with more than one control in a block and are not considered here. Rectangular preparation designs can only be used in the three-step procedure when $v/(k-1)$ is an integer, otherwise only step-type designs with fewer replication can be used for a preparation design of that size.

Example 6.3 Preparation designs with 12 blocks of size 6 with varying control replication $r_c = r_t + x$ for $x \in 2, 3, 4$.

These designs have been constructed using algorithm JE3.

$x = 2, p = 23$
step-type

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

$x = 3, p = 22$
step-type

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

$x = 4, p = 21$
rectangular

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

Several construction techniques for balanced and regular graph treatment-control designs have been introduced; see, for example, Majumdar (1996) for a review. Alternatively, computer algorithms can be used for constructing treatment-control designs that are more flexible with respect to the design size. Gupta et al. (1999), for example, suggest using the algorithm by Jones and Eccleston (1980), which is the approach adopted in this dissertation. It has been described in its basic form in Chapter 3 (algorithm JE1) and has been modified for the use in the three-step procedure in Chapter 5 (algorithm JE2). To use it for treatment-control designs it needs further adjustments, which will be described in this section. Jones and Eccleston (1980) use the weighted A-optimality criterion to search for optimal designs given a pair of weights for the two sets of contrasts. This algorithm can be used for step-type designs as well as for rectangular treatment-control designs.

In the first step of the algorithm, an optimal replication scheme for the products is created. This depends on the pair of weights provided by the user as an input for the procedure. For these weights, an exchange procedure substitutes products that increase the optimality criterion for other products in the design. After the optimal replication scheme has been found an interchange procedure is used to search for an efficient design by interchanges of treatments between blocks.

Let $\boldsymbol{w}' = [w_{12}, w_{13}, \ldots, w_{1v+1}, w_{23}, w_{24}, \ldots, w_{vv+1}]$ be the vector of weights for the $\frac{p(p+1)}{2}$ elementary contrasts. While the algorithm allows for different weights for all contrasts, for treatment-control designs only two different weights are necessary, one weight for the v control-test contrasts and one weight for the $\frac{v(v+1)}{2}$ test-test contrasts:

$$w_{12}, w_{13}, \ldots, w_{1v+1} = w_c \text{ and } w_{23}, w_{24}, \ldots, w_{vv+1} = w_t$$

These weights are then standardised, so that the weights add up to 1, i.e

$$v \cdot w_c + \frac{v(v+1)}{2} w_t = 1.$$

As discussed above, the three-step procedure requires a replication vector that has equal replication for all test products ($r' = [r_c, r_t, \ldots, r_t]$). The Jones and Eccleston algorithm does not necessarily find a design with equal replication for all test products for every chosen set of weights (w_c, w_t). Therefore, an appropriate set of weights has to be found, that does provide the required replication vector and that represents the weighting of the two groups of contrasts. Unfortunately, there is no easy relationship between the weights and the replication. Both are connected through the generalised inverse of the information matrix, which depends not only on the replication vector but also on the complex concurrence matrix. Therefore an iterative use of the exchange part of the algorithm is necessary to find suitable weights. Different weights are tried until a pair is found that represents the order of importance between the contrasts and that achieves equal replication for the test products and higher replication for the control. This can be quite a lengthy process.

As an alternative to the iterative search only the interchange part of the algorithm is used for the construction of treatment-control preparation designs. A starting design with the required replication structure is constructed and interchanges take place in that design, which will not change the given replication structure. The specified weights are then used only to calculate the weighted A-optimality criterion. This approach has been adopted for this dissertation.

While for $q = 1$ sets the algorithm by Jones and Eccleston (1980) can be used in its original form, for $q > 1$ the algorithm needs to be modified to create preparation designs with the structure explained in detail in Section 5.3.2. For equi-replicated preparation designs this has been shown in Chapter 5 with algorithm JE2. For treatment-control designs, this is not as straightforward. While the principle is the same, the replication of the control product xr_t can now be spread over the design in two ways; either all control products occur in one set (algorithm JE3) or the x replications are spread over several sets (algorithm JE4).

Example 6.4 Step-type designs with 12 blocks of size 6 for 22 products and a control replication of $r_c = 3\ r_t = 9$.

Design $P1$
constructed with JE3
control in set 1

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

Design $P2$
constructed with JE4
control in set 1 and set 2

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

Both designs have an average efficiency for the treatment control contrasts of 86.62% and, on average, test-test contrasts in this design are estimated with a 1.54 times larger variance than control-test contrasts.

Both assignment methods have been programmed and designs have been created for the specified range given below with 100 runs of each algorithm. To find out which version is superior the designs are compared by their average efficiencies for control-test contrasts ($E_{A;c}$), and their ratio of control-test and test-test contrasts ($E_{A;\text{ratio}}$). From 88 designs, only 28 (32%) differ in either of the efficiencies for the two algorithms. The details of these designs are listed in Table 6.5. For both measures, the design that has the larger of the two values is regarded as superior.

For eight designs (9%), Algorithm JE3 has performed better according to both efficiency measures and for eight designs Algorithms JE4 has performed better. For 12 cases (14%) the two measures give contradictary results. The differences are all fairly small, so that overall the two algorithms perform similarly. Thus, either of them can be used for the construction of treatment-control preparation designs and there is no advantage in the arrangement of control products in one or several sets.

Treatment-control preparation designs are created for the same product and preparation constraint range as the equally replicated preparation designs in Chapter 5. This is 6 to 30 products including the control product and preparation constraints between 4 and 20 products, which are constructed for designs with products divided into 2 to 4 sets. For treatment-control designs, the replication of the control can also be varied and has been chosen for a range of $r_c = 2r_v$ to $r_c = 4r_v$. Higher control replication is possible for a large number of the designs. These designs can also be constructed with the algorithm, but only some designs have been evaluated in this thesis. Due to limited resources, large control replication is less likely to be used in practise.

Table 6.5: Designs for which the efficiencies from algorithms JE3 and JE4 differ.

							$E_{A;c}$		$E_{A;\text{ratio}}$	
q	x	p	s	k	r_t	r_c	JE3	JE4	JE3	JE4
2	2	23	12	8	4	8	98.99	98.94	1.3545	1.3536
2	2	23	12	10	5	10	99.64	99.64	1.3561	1.3561
2	2	25	13	8	4	8	99.03	99.03	1.3538	1.3537
2	2	25	13	10	5	10	99.53	99.53	1.3532	1.3532
2	2	25	13	12	6	12	99.81	99.81	1.3538	1.3538
2	2	27	14	8	4	8	99.03	99.03	1.3542	1.3542
2	2	27	14	10	5	10	99.50	99.48	1.3520	1.3517
2	2	27	14	12	6	12	99.78	99.78	1.3522	1.3522
2	2	29	15	6	3	6	97.48	97.51	1.3591	1.3592
2	2	29	15	8	4	8	99.03	99.03	1.3544	1.3544
2	2	29	15	10	5	10	99.48	99.50	1.3511	1.3514
2	2	29	15	12	6	12	99.72	99.72	1.3505	1.3505
2	2	29	15	14	7	14	99.87	99.87	1.3506	1.3506
2	3	24	13	8	4	12	99.21	99.21	1.5380	1.5380
2	3	26	14	8	4	12	99.08	99.08	1.5351	1.5351
2	3	28	15	6	3	9	97.57	97.46	1.5383	1.5370
2	3	28	15	8	4	12	98.99	98.99	1.5329	1.5329
2	3	28	15	10	5	15	99.58	99.58	1.5317	1.5317
2	3	30	16	6	3	9	97.28	97.29	1.5360	1.5364
2	3	30	16	8	4	12	98.94	98.94	1.5310	1.5310
2	3	30	16	10	5	15	99.51	99.51	1.5293	1.5293
2	4	23	13	6	3	12	98.08	98.10	1.6532	1.6531
2	4	25	14	6	3	12	97.74	97.85	1.6484	1.6494
2	4	27	15	6	3	12	97.60	97.60	1.6474	1.6474
2	4	29	16	6	3	12	97.41	97.45	1.6452	1.6461
2	4	29	16	8	4	16	99.04	99.04	1.6406	1.6406
3	3	28	10	6	2	6	93.21	94.18	1.5446	1.5570
3	4	30	11	6	2	8	93.06	92.67	1.6478	1.6370

The two efficiency measures that have been chosen to characterise the constructed preparation designs, $E_{A;c}$ and $E_{A;\text{ratio}}$ are listed in Tables 6.6 and 6.7, respectively. The tables show the maximum efficiency value from Algorithms JE3 and JE4.

To create the necessary column structure for the preparation design, algorithm J1, as described in Chapter 5, can be used again. The auxiliary design for the column structure that is required by the algorithm as an input design has to be modified from the procedure described in Chapter 5. It has to allow for the increased occurrence of the control product, as demonstrated in Example 6.5.

Example 6.5 Transformation of the step-type designs from Example 6.4 into preparation designs with the required column structure.

To create preparation designs with the required column structure, the following auxiliary designs are needed for algorithm J1. The structure differs according to the number of sets the control appears in.

Design $A1$
auxiliary design

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

Design $A2$
auxiliary design

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

Using designs $A1$ and $A2$ to transform designs $P1$ and $P2$, respectively, results in the following preparation designs:

Table 6.6: Average efficiencies $E^{o}_{A,c}$ for test-control contrasts in the preparation designs

$q = 2$

$x = 2$

p	s	6	8	10	12	14
11	6	99.49				
13	7	98.83				
15	8	98.49	99.72			
17	9	98.34	99.47			
19	10	97.63	99.33	99.82		
21	11	97.04	99.17	99.70		
23	12	97.25	98.99	99.64	99.87	
25	13	97.55	99.03	99.53	99.81	
27	14	97.60	99.03	99.50	99.78	99.90
29	15	97.51	99.03	99.50	99.72	99.87

$x = 3$

p	s	6	8	10
16	9	98.65		
18	10	98.43		
20	11	98.15		
22	12	97.85	99.40	
24	13	97.79	99.21	
26	14	97.58	99.08	
28	15	97.57	98.99	99.58
30	16	97.29	98.94	99.51

$x = 4$

p	s	6	8
21	12	98.48	
23	13	98.10	
25	14	97.85	
27	15	97.60	
29	16	97.45	99.04

$q = 3$

p	s	6	9	12	15
11	4	98.26			
14	5	96.53			
17	6	95.03	99.41		
20	7	94.91	99.04		
23	8	94.57	98.78	99.70	
26	9	94.44	98.32	99.57	
29	10	93.49	98.35	99.45	99.82

p	s	6	9
16	6	97.30	
19	7	95.82	
22	8	95.03	
25	9	94.40	98.95
28	10	94.18	98.61

p	s	6
21	8	95.56
24	9	94.55
27	10	93.73
30	11	93.06

$q = 4$

p	s	8	12
15	4	98.52	
19	5	97.34	
23	6	96.50	99.49
27	7	95.60	99.21

p	s	8
22	6	97.14
26	7	96.37
30	8	95.72

p	s	8
29	8	96.70

Table 6.7: Average efficiencies E_A^{ratio} for test-control contrasts in the preparation designs

$x = 2$ $x = 3$ $x = 4$

$q = 2$

p	s	6	8	10	12	14
11	6	1.389				
13	7	1.379				
15	8	1.374	1.371			
17	9	1.371	1.366			
19	10	1.359	1.362	1.361		
21	11	1.349	1.358	1.358		
23	12	1.351	1.354	1.356	1.356	
25	13	1.354	1.354	1.353	1.354	
27	14	1.358	1.354	1.352	1.352	1.352
29	15	1.359	1.354	1.351	1.350	1.351

p	s	6	8	10
16	9	1.563		
18	10	1.554		
20	11	1.545		
22	12	1.540	1.542	
24	13	1.538	1.538	
26	14	1.537	1.535	
28	15	1.538	1.533	1.532
30	16	1.536	1.531	1.529

p	s	6	8
21	12	1.659	
23	13	1.653	
25	14	1.649	
27	15	1.647	
29	16	1.646	1.641

$q = 3$

p	s	6	9	12	15
11	4	1.399			
14	5	1.382			
17	6	1.362	1.368		
20	7	1.364	1.362		
23	8	1.372	1.359	1.357	
26	9	1.379	1.352	1.354	
29	10	1.373	1.352	1.352	1.351

p	s	6	9
16	6	1.571	
19	7	1.559	
22	8	1.555	
25	9	1.556	1.539
28	10	1.557	1.535

p	s	6
21	8	1.672
24	9	1.662
27	10	1.651
30	11	1.648

$q = 4$

p	s	8	12
15	4	1.380	
19	5	1.369	
23	6	1.366	1.359
27	7	1.358	1.355

p	s	8
22	6	1.555
26	7	1.544
30	8	1.534

p	s	8
29	8	1.651

Design $P'1$
constructed with JE3

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

Design $P'2$
constructed with JE4

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

An efficient cross-over panel design for k treatments in a blocks of size t is required for the treatment-control version of the three-step procedure. The usual panel designs described in Table 5.15 can be used for this purpose. For example, to construct a multi-session step-type treatment-control design $D_{\mathrm{R}}^{\mathrm{TCS}}(22, 12, 12, 4; 6)$ (that is a design for 22 products, including a control, to examine in 12 sessions of 4 servings with a preparation constraint of 6 products served to 12 assessors), a panel design $D_{CO}^{2}(6, 12, 4)$ is required.

Example 6.6 Panel design $D_{CO}^{2}(6, 12, 4)$.

5	4	3	2
1	6	2	4
3	5	6	1
6	4	2	3
5	2	6	1
3	6	5	2
4	1	3	5
1	3	4	6
2	5	1	4
4	3	1	6
6	2	5	3
2	1	4	5

In the last step, the panel design is combined with the treatment-control preparation design in the usual way to create the complete sensory design. In the optimisation

Table 6.8: $D_{\mathrm{R}}^{\mathrm{TCS}}(22, 12, 12, 4; 6)$

This is a step-type treatment-control design for 22 products in 12 sessions of 4 servings given to 12 assessors with a preparation constraint of 6 products. The control product is replicated three times as often as the test products.

	Session 1				Session 2				Session 3				Session 4				Session 5				Session 6			
Assessor 1	10	1	19	16	6	1	18	15	1	2	14	19	4	10	20	11	7	8	12	18	9	4	16	21
Assessor 2	22	7	16	1	19	4	15	1	20	9	19	2	13	1	11	10	14	1	18	8	12	5	21	4
Assessor 3	19	10	7	22	18	6	4	19	14	1	9	20	20	4	1	13	12	7	1	14	16	9	5	12
Assessor 4	7	1	16	19	4	1	15	18	9	2	19	14	1	10	11	20	1	8	18	12	5	4	21	16
Asscssor 5	10	16	7	22	6	15	4	19	1	19	9	20	4	11	1	13	7	18	1	14	9	21	5	12
Assessor 6	19	7	10	16	18	4	6	15	14	9	1	19	20	1	4	11	12	1	7	18	16	5	9	21
Assessor 7	1	22	19	10	1	19	18	6	2	20	14	1	10	13	20	4	8	14	12	7	4	12	16	9
Assessor 8	22	19	1	7	19	18	1	4	20	14	2	9	13	20	10	1	14	12	8	1	12	16	4	5
Asscssor 9	16	10	22	1	15	6	19	1	19	1	20	2	11	4	13	10	18	7	14	8	21	9	12	4
Assessor 10	1	19	22	7	1	18	19	4	2	14	20	9	10	20	13	1	8	12	14	1	4	16	12	5
Assessor 11	7	16	10	19	4	15	6	18	9	19	1	14	1	11	4	20	1	18	7	12	5	21	9	16
Assessor 12	16	22	1	10	15	19	1	6	19	20	2	1	11	13	10	4	18	14	8	7	21	12	4	9

	Session 7				Session 8				Session 9				Session 10				Session 11				Session 12			
Assessor 1	8	1	15	20	5	3	17	14	1	9	22	13	1	7	21	17	3	6	13	12	2	5	11	22
Assessor 2	21	3	20	1	18	10	14	3	17	8	13	9	11	6	17	7	16	2	12	6	15	1	22	5
Assessor 3	15	8	3	21	17	5	10	18	22	1	8	17	21	1	6	11	13	3	2	16	11	2	1	15
Assessor 4	3	1	20	15	10	3	14	17	8	9	13	22	6	7	17	21	2	6	12	13	1	5	22	11
Assessor 5	8	20	3	21	5	14	10	18	1	13	8	17	1	17	6	11	3	12	2	16	2	22	1	15
Assessor 6	15	3	8	20	17	10	5	14	22	8	1	13	21	6	1	17	13	2	3	12	11	1	2	22
Asscssor 7	1	21	15	8	3	18	17	5	9	17	22	1	7	11	21	1	6	16	13	3	5	15	11	2
Assessor 8	21	15	1	3	18	17	3	10	17	22	9	8	11	21	7	6	16	13	6	2	15	11	5	1
Assessor 9	20	8	21	1	14	5	18	3	13	1	17	9	17	1	11	7	12	3	16	6	22	2	15	5
Assessor 10	1	15	21	3	3	17	18	10	9	22	17	8	7	21	11	6	6	13	16	2	5	11	15	1
Assessor 11	3	20	8	15	10	14	5	17	8	13	1	22	6	17	1	21	2	12	3	13	1	22	2	11
Assessor 12	20	21	1	8	14	18	3	5	13	17	9	1	17	11	7	1	12	16	6	3	22	15	5	2

process for the column order of the preparation design in Example 6.7, the average variance of the test-control contrasts is minimised. Other measures could be used instead, for example, the average variance of the test-test contrast, the average of both sets of contrasts, their ratio, or a weighted average. The chosen column permutation has an influence on the efficiency of the final design, especially when the panel design is not an equally replicated design.

Example 6.7 Combining the treatment-control preparation design with the panel design.
To create the required step-type multi-session design $D_R^{TCS}(22, 12, 12, 4; 6)$, preparation design $P'1$ from Example 6.5 and the panel design shown in Example 6.6 are combined. The minimum average variance of the test-control contrasts (0.0667 σ^2) is reached for the column order [3 1 2 5 4 6]. The resulting complete design is shown in Table 6.8. The largest variance observed is 0.0669 σ^2. The efficiency ratio of test-test and test-control contrasts for the design in Table 6.8 is 1.54 and the average efficiency of all contrasts compared to an orthogonal designs with the same replication vector for this design is 81.29%.

6.3 Conclusions

In this chapter, the idea of treatment-control designs has been introduced and their theory relevant for sensory designs has been reviewed. Treatment-control designs are useful for sensory experiments when interest is mainly in comparisons between a control product and alternative products. Control products are frequently included in sensory trials, but their different standing within the product range is rarely considered when constructing sensory designs. One reason for this is most likely the lack of suitable readily available algorithms that satisfy the requirements for sensory designs. While much is known about single-session treatment-control designs, multi-session treatment-control designs are less well developed. In this chapter, two construction strategies have been developed for treatment-control designs for multi-session trials with preparation constraints. Both techniques are based on the three-step procedure developed in Chapter 5. The first algorithm constructs rectangular-type treatment control designs, in which the control occurs once in every block. With the second algorithm, step-type treatment-control designs are created, in which the control occurs once in a block or not at all, but in more blocks than each test product. These algorithms are flexible enough to create efficient treatment-control designs for a wide range of design sizes and preparation constraints.

Chapter 7

Factorial multi-session designs with preparation constraints

In Chapter 6 we presented examples of sensory designs, in which the product contrasts are of unequal importance and fully balanced designs are no longer the best choice. This has been discussed so far for the case in which individual products in the study have particular experimental or commercial relevance, e.g. an experimental reference or control products, or the market or brand leader.

In other cases, there may be specific structural relationships within the product set, such that some or all of the products constitute a full or fractional factorial design. This is the case, for example, when sensory profiling is used to explore formulation or process differences in product development or in optimisation experiments, or to investigate temperature, time, and packaging in storage or shelf-life testing. Examples of sensory trials with a factorial structure are given, e.g. by Durier et al. (1997) and Mead and Gay (1995).

Whilst for trials in which products have no factorial structure interest is almost always in pairwise product comparisons, for factorial designs interest is generally in the factors and their interactions rather than in the products themselves. Thus, contrasts related to these main effects and interactions have to be considered. These often differ in their importance for the analysis. Main effects are usually regarded as more important then lower-order interactions and these are regarded as more important than higher-order interactions. Some or all of the latter may be disregarded altogether. Therefore, trial-specific designs are needed that maximise the precision of the contrasts of interest at the expense of contrasts of lesser or no interest.

In incomplete block designs, some contrasts are completely or partially confounded with blocks. Only the contrasts of no or minor interest should be confounded. In sensory designs of the required size it is not always possible to confound solely contrasts of minor interest and a design has to be found, in which the contrasts of interest are estimated with maximum precision given such restrictions.

The linear models introduced in Chapter 4 are also relevant for factorial multi-session trials with preparation constraints. They differ only in their structure for the treatment effects. Durier et al (1997) and Hunter (1996) show how to partition the product sums of squares into its components due to main effects and interactions.

There are many different scenarios for factorial designs in the sensory setting. In the simplest case all products are part of the factorial structure. In other experiments there may be two groups of products: one set with a known factorial structure and one with the factor settings unknown. Interest is then on the one hand in the factorial comparisons and, on the other hand, in a comparison of the remaining products with either the factors or the single products. In other cases, control products may be included in a factorial experiment. The control product can either be part of the factorial structure or additional to it. The latter is a special case of the structure with two sets of products, one with factorial structure, one without. Here, the group of products without a factorial structure consist of a single control product. The specific contrasts of interest to the sensory scientist will differ from case to case, and will determine the actual form of the design.

As in the non-factorial case, efficient designs for multi-session trials with preparation constraints can be constructed with the three-step procedure. The factorial structure is easily incorporated into the three-step procedure at the preparation design stage. Thus, the focus in this chapter is, on the choice of adequate preparation designs for the factorial structure at hand. Construction methods for factorial preparation designs are discussed in Section 7.2.

The embedding of factorial designs in the three-step procedure is illustrated with two examples from the potentially vast range of sensory factorial designs. These will demonstrate how designs for a variety of different contrasts of interest can be created. Possible contrasts of interest for these two examples are discussed in Section 7.1. The modified algorithms of the three-step procedure for factorial designs are discussed in Section 7.2 and factorial designs with preparation constraints are constructed for Examples 7.1 and 7.2.

Example 7.1 A $2^4 = 2 \times 2 \times 2 \times 2$ factorial design.
The four factors in this example represent

A two formulations (usual and modified)

B two types of additives (additive X and additive Y)

C two storage times (short and long)

D two storage temperatures (low and high)

Suppose a multi-session design for 16 products served to 12 assessors in 8 sessions of 4 servings is required with a preparation constraint of 8 products: $D_R^{2^4}(16, 12, 8, 4; 8)$. *To optimise the factorial structure a factorial preparation design* $D^{2;2^4}(16, 8, 8)$ *is constructed.*

Example 7.2 A 3 × 4 factorial design.

In this example, the first factor (F) represents four fat types with levels F1, F2, F3 and F4 and the second factor (P) three preparation methods with levels P1, P2 and P3. Suppose a multi-session design for these 12 products served to 12 assessors in 6 sessions of 4 servings is required with a preparation constraint of 6 products: $D_R^{3\times4}(12, 12, 6, 4; 6)$. *Thus, the factorial structure has to be incorporated into preparation design* $D^{2;3\times4}(12, 6, 6)$.

7.1 Contrasts in factorial designs

A 2^n factorial is the simplest example of a factorial experiment. Each factor has two categories, usually labelled low and high. When each of the $2^4 = 16$ product combinations is available for the trial and a full factorial can be used, interest is most likely in identifying which of the two categories of each factor performs best and which two-factor interactions are relevant.

The factorial contrasts for Example 7.1 are listed in Table 7.1. The products are represented by the usual notation for factorial 2^n-designs and additionally with a binary four digit code, where each digit represents one of the four factors and the two categories of each factor are coded 0 or 1. E.g., product 8 with Factor A and D at their high levels and factors B and C at their low levels is represented as product *ad* according to the usual factorial notation and as product 1001 according to the binary notation.

Table 7.1 shows the main effect for factor A, for example, is represented by the contrast between all products made with formulation X compared to all products made with formulation Y. The 15 contrasts shown in Table 7.1 are independent and orthogonal and define one degree of freedom each in an ANOVA analysis. The product sum of squares can therefore be split into 15 terms referring to these contrasts and each contrast can be tested separately for significance.

Table 7.1: Contrasts for the 2^4 factorial in Example 7.1

Product number	Factorial setting	Factor coding	Main effects				Interaction										
							2-factor						3-factor				4-factor
			A	B	C	D	AB	AC	AD	BC	BD	CD	ABC	ABD	ACD	BCD	ABCD
1	1	0000	-1	-1	-1	-1	1	1	1	1	1	1	-1	-1	-1	-1	1
2	a	1000	1	-1	-1	-1	-1	-1	-1	1	1	1	1	1	1	-1	-1
3	b	0100	-1	1	-1	-1	-1	1	1	-1	-1	1	1	1	-1	1	-1
4	c	0010	-1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	1	-1
5	d	0001	-1	-1	-1	1	1	1	-1	1	-1	-1	-1	1	1	1	-1
6	ab	1100	1	1	-1	-1	1	-1	-1	-1	-1	1	-1	-1	1	1	1
7	ac	1010	1	-1	1	-1	-1	1	-1	-1	1	-1	-1	1	-1	1	1
8	ad	1001	1	-1	-1	1	-1	-1	1	1	-1	-1	1	-1	-1	1	1
9	bc	0110	-1	1	1	-1	-1	-1	1	1	-1	-1	-1	1	1	-1	1
10	bd	0101	-1	1	-1	1	-1	1	-1	-1	1	-1	1	-1	1	-1	1
11	cd	0011	-1	-1	1	1	1	-1	-1	-1	-1	1	1	1	-1	-1	1
12	abc	1110	1	1	1	-1	1	1	-1	1	-1	-1	1	-1	-1	-1	-1
13	abd	1101	1	1	-1	1	1	-1	1	-1	1	-1	-1	1	-1	-1	-1
14	acd	1011	1	-1	1	1	-1	1	1	-1	-1	1	-1	-1	1	-1	-1
15	bcd	0111	-1	1	1	1	-1	-1	-1	1	1	1	-1	-1	-1	1	-1
16	abcd	1111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

For Example 7.2, in which each factor has more than two levels, the contrasts of interest depend on the nature of the factor levels, which can either be qualitative or quantitative. Qualitative factors often represent different kinds of treatments and the different levels don't have a natural ordering. If both factors are qualitative, as suggested for this example, interest may be in the pairwise contrasts of the factor levels or in selected comparisons between the levels. These are so-called preplanned comparisons. They are determined by the intent of the experiment rather than by its outcome and the number of contrasts is generally small (Hinkelmann and Kempthorne, 1994). The comparisons of interest sometimes form a set of orthogonal contrasts, sometimes they are not orthogonal. The corresponding contrasts for the main effects for pairwise contrasts of the levels are listed in Table 7.2. They are not linearly independent nor are they orthogonal and there are more contrasts than degrees of freedom for each factor. While these contrasts cannot be used to subdivide the product sum of squares in an ANOVA analysis, they can still be estimated and tested and can be used in the design construction to identify a suitable design. The interest is often in a few of these contrasts rather than all of them.

The 18 contrasts for the interaction between the two main effects are

(F1–F2)(P1–P2) (F1–F2)(P2–P3) (F1–F2)(P1–P3)
(F1–F3)(P1–P2) (F1–F3)(P2–P3) (F1–F3)(P1–P3)
(F1–F4)(P1–P2) (F1–F4)(P2–P3) (F1–F4)(P1–P3)
(F2–F3)(P1–P2) (F2–F3)(P2–P3) (F2–F3)(P1–P3)
(F2–F4)(P1–P2) (F2–F4)(P2–P3) (F2–F4)(P1–P3)
(F3–F4)(P1–P2) (F3–F4)(P2–P3) (F3–F4)(P1–P3)

These can be constructed by multiplication of the main effect contrasts from each pair of factors listed in Table 7.2.

If a factor has more than two levels and it is quantitative, it is also of interest whether the levels are equi-distantly spaced. Non-linear relationships can be modelled only

Table 7.2: Contrasts of interest for Example 7.2 with qualitative levels
Pairwise contrasts between main effects are listed.

Product number	Factor setting	Main effect F						Main effect P		
		F1–F2	F1–F3	F1–F4	F2–F3	F2–F4	F3–F4	P1–P2	P1–P3	P2–P3
1	F1P1	1	1	1	0	0	0	1	1	0
2	F1P2	1	1	1	0	0	0	-1	0	1
3	F1P3	1	1	1	0	0	0	0	-1	-1
4	F2P1	-1	0	0	1	1	0	1	1	0
5	F2P2	-1	0	0	1	1	0	-1	0	1
6	F2P3	-1	0	0	1	1	0	0	-1	-1
7	F3P1	0	-1	0	-1	0	1	1	1	0
8	F3P2	0	-1	0	-1	0	1	-1	0	1
9	F3P3	0	-1	0	-1	0	1	0	-1	-1
10	F4P1	0	0	-1	0	-1	-1	1	1	0
11	F4P2	0	0	-1	0	-1	-1	-1	0	1
12	F4P3	0	0	-1	0	-1	-1	0	-1	-1

for factors with more than two levels; Mead and Gay (1995) recommend at most 4 levels. For quantitative levels that represent measurements from equally spaced intervals, interest is generally in the nature of the increase or decrease of the response with increasing factor levels, e.g. it is examined if there is a linear or quadratic relationship. Orthogonal and linearly independent contrasts can be created that divide the factors into related components. A table of such contrasts for factors with 3 to 12 levels can be found in Snedecor and Cochran (1980). The orthogonal contrasts for Example 7.2 are shown in Table 7.3.

The interaction between the two factors can also be divided into relevant contrasts by multiplying a contrast from each factor. The six degrees of freedom for the interaction term can be divided into the following contrasts:

$$F_LP_L \quad F_QP_L \quad F_CP_L$$
$$F_LP_Q \quad F_QP_Q \quad F_CP_Q$$

Table 7.3: Orthogonal contrasts for Example 7.2.

Product number	Factor setting	F_L	F_Q	F_C	P_L	P_Q
1	F1P1	-3	1	-1	-1	1
2	F1P2	-3	1	-1	0	-2
3	F1P3	-3	1	-1	1	1
4	F2P1	-1	-1	3	-1	1
5	F2P2	-1	-1	3	0	-2
6	F2P3	-1	-1	3	1	1
7	F3P1	1	-1	-3	-1	1
8	F3P2	1	-1	-3	0	-2
9	F3P3	1	-1	-3	1	1
10	F4P1	3	1	-1	-1	1
11	F4P2	3	1	-1	0	-2
12	F4P3	3	1	-1	1	1

Which set of contrasts is most appropriate depends solely on the research question of the trial. It is possible to construct contrasts that reflect unequally spaced levels. The optimal factorial design will differ for different sets of contrasts, so the relevant set of contrasts should be chosen that reflects the research objective closest.

7.2 Construction of factorial preparation designs

Preparation designs can be constructed "by hand" by deliberately confounding certain interactions with blocks (see e.g. John and Williams (1995) or Bailey (1977) for a description of the general technique) or by using designs from the vast literature on factorial designs (see e.g Cochran and Cox (1957)). An example of a preparation design constructed by partially confounding some of the higher-order interactions is given in Example 7.3. Designs obtained in this way do not necessarily provide the column structure that is needed for preparation designs (condition C2) and it may be difficult to include this structure intentionally. When choosing the interactions for confounding and relabelling the factorial products into numbers, preparation designs with the required structure can be found by trial and error.

The interchange part of the algorithm by Jones and Eccleston (1980) can also be used to create factorial preparation designs. In Chapter 5, we showed how this algorithm can be modified to generate designs that satisfy constraint C2 (algorithm JE2). For factorial designs, algorithm JE2 needs to be modified so that, instead of using all pairwise product contrasts, a matrix of factorial contrasts is specified. To accommodate all kinds of product structures, this contrast matrix is not created automatically within the algorithm, but is required as an input matrix.

Further, a weight is required for each contrast, specifying its relative importance in relation to the other contrasts. In the algorithm the input vector of weights is standardised, so that their sum is equal to one. The factorial version of algorithm JE2 is called JE5 and is used for the construction of the preparation designs in Examples 7.4 and 7.5.

For factorial designs satisfying condition C2, Algorithm J1 can be used, without any modifications, to create a preparation design with the required column structure (condition C1). The panel design can also be constructed in the usual way.

For step three of the three-step procedure the optimality criterion that identifies the best column structure needs to be modified, since in factorial designs not all contrasts are of equal importance and the average efficiency is not a meaningful criterion. Possible criteria for factorial designs are the sum of the contrasts of highest importance or a weighted sum of all contrast. A possible weighting is given by the vector of weights, required by algorithm JE5.

For factorial 2^n-experiments a lot of designs are tabulated and it is relatively easy to deliberately confound certain contrasts with blocks. Thus, if interest in Example 7.1 is mostly in main effects and two-factor interactions, some of the higher order interactions (ABCD, ABD, ABC, ACD in the example) can be confounded with blocks (sessions). First the confounding scheme for the blocks is devised and then to the entries in the resulting design are translated into product numbers. The equivalence relation between product numbers and factor settings is given in Table 7.1.

Example 7.3 Preparation design with partially confounded three- and four-factor interactions.

Factorial design								Relabelled design with constraint C2								Preparation design with constraints C1 and C2							
1	ab	ac	ad	bc	bd	cd	abcd	1	6	7	8	9	10	11	16	1	6	7	8	9	10	11	16
a	b	c	d	abc	abd	acd	bcd	2	3	4	5	12	13	14	15	2	3	4	5	12	13	14	15
a	b	d	ac	bc	cd	abd	abcd	2	3	5	7	9	11	13	16	3	5	2	7	11	9	16	13
1	c	ab	ad	bd	abc	acd	bcd	1	4	6	8	10	12	14	15	4	8	1	6	10	14	15	12
a	b	c	ad	bd	cd	abc	abcd	2	3	4	8	10	11	12	16	8	2	3	4	16	11	12	10
1	d	ab	ac	bc	abd	acd	bcd	1	5	6	7	9	13	14	15	5	6	7	1	13	15	9	14
a	c	d	ab	bc	bd	acd	abcd	2	4	5	6	9	10	14	16	6	4	5	2	14	16	10	9
1	b	ac	ad	cd	abc	abd	bcd	1	3	7	8	11	12	13	15	7	1	8	3	15	12	13	11

The blocks in the designs are chosen so that the ABCD interaction is confounded between the first and second block, i.e. all products with 1 in the ABCD contrasts are in block 1 and all products with -1 are in block 2. The ABD interaction is confounded between the third and fourth block, ABC between the fifth and sixth block and ACD between the seventh and eighth block. Due to the intentional confounding the efficiencies for the preparation design are 100% for the unconfounded contrasts (all main effects, two-factor interactions and the three-factor interaction BCD) and 75% for the partially confounded interactions (ABC, ABD, ACD and ABCD), see Table 7.4.

In this example, the division of the products into two sets is obvious. The relabelled design has already the C2 structure. The required column structure for a preparation design can be achieved by applying algorithm J1.

The factorial preparation design can then be combined with panel design, $D^2_{co}(8,12,4)$ to generate the complete sensory design with a factorial structure. The panel design shown in Example 5.3 can be used again for this example. The resulting complete sensory design is shown in Table 7.5. The column order of the preparation design has been optimised by using the permutation with the highest average efficiencies for all main effects. The different column permutations create sensory designs with average efficiencies of the main effects between 87.17% and 89.35%, and the highest efficiency is reached for column order [5 7 8 6 4 2 3 1].

The efficiencies of the factorial contrasts for the complete sensory design according to model (4.8) are listed in Table 7.4. For comparison, the efficiencies of the preparation design according to model (3.16) have also been included. It shows clearly that the factorial structure induced by the preparation design is transferred to the complete sensory design. All effects that are not confounded in the preparation design are also estimated with highest efficiency in the complete sensory designs, while the confounded contrasts in the preparation design have lower efficiencies for the complete sensory design.

The preparation design selects the products for each session. In a factorial preparation design, the direct comparisons within each block are chosen, so that the factorial contrasts are optimised. The panel design assigns a subset of these products to each block of the sensory design, where each subset is chosen nearly equally often. Although not all direct comparisons from the preparation design are made in each session, this setup still transfers the general factorial structure on to the final design.

Example 7.4 Factorial preparation designs for Example 7.2.

Interest in Example 7.2 with four fat types and three preparation methods is most likely in differences between the preparation methods and between formulations, as well as the resulting interactions, rather than in all pairwise comparisons between the products, disregarding the factors. The contrasts have been partitioned into two

Table 7.4: Efficiencies for the factorial contrasts for Example 7.1 (in %)

Contrast	Preparation design	Complete design
A	100	96.84
B	100	96.91
C	100	94.38
D	100	94.86
AB	100	97.75
AC	100	91.14
AD	100	98.48
BC	100	96.64
BD	100	96.13
CD	100	96.47
ABC	75	73.23
ABD	75	73.23
ACD	75	73.23
BCD	100	73.23
ABCD	75	73.04

groups, the main effects and the interaction effects, and contrasts in one group are assigned the same weights. Four preparation designs are generated with algorithm JE5 using four different pairs of weights. The contrasts matrix consists of the main effects shown in Table 7.2 and the corresponding interactions.

For the first three preparation designs the contrasts related to the main effect of factor P are all 100%, while the pairwise contrasts related to factor F are either 100% or 94%. The contrasts of the first and second preparation design have essentially the same efficiencies and differ only in the assignment of the efficiencies to the contrasts. For both designs, all main effect contrasts have higher efficiencies than the interaction contrasts. For the third design, which was constructed with weight 0 for the interaction effects, the interaction efficiencies are much more dispersed than for the first and second design and the efficiency of the interaction contrast (F1-F2)(P1-P3) with 95% is even slightly higher than the efficiency of the main effect contrast (F1-F2) with 94%. Other interaction contrasts for this design have much lower efficiencies though, with a minimum of 61% for (F1-F4)(P2-P3). This

Table 7.5: Complete sensory design with factorial structure for Example 7.1

	Session 1				Session 2				Session 3				Session 4			
Assessor 1	9	8	7	10	12	5	4	13	11	7	2	9	10	6	1	14
Assessor 2	11	9	6	1	14	12	3	2	16	11	5	3	15	10	8	4
Assessor 3	16	1	10	6	15	2	13	3	13	3	9	5	12	4	14	8
Assessor 4	11	10	1	7	14	13	2	4	16	9	3	2	15	14	4	1
Assessor 5	6	10	16	7	3	13	15	4	5	9	13	2	8	14	12	1
Assessor 6	8	7	16	11	5	4	15	14	7	2	13	16	6	1	12	15
Assessor 7	9	7	1	16	12	4	2	15	11	2	3	13	10	1	4	12
Assessor 8	1	8	11	16	2	5	14	15	3	7	16	13	4	6	15	12
Assessor 9	7	6	9	11	4	3	12	14	2	5	11	16	1	8	10	15
Assessor 10	10	11	8	6	13	14	5	3	9	16	7	5	14	15	6	8
Assessor 11	6	16	8	9	3	15	5	12	5	13	7	11	8	12	6	10
Assessor 12	1	9	10	8	2	12	13	5	3	11	9	7	4	10	14	6

	Session 5				Session 6				Session 7				Session 8			
Assessor 1	16	4	3	11	13	1	7	15	14	2	5	16	15	3	8	12
Assessor 2	12	16	2	8	9	13	6	5	10	14	4	6	13	15	1	7
Assessor 3	10	8	11	2	14	5	15	6	9	6	16	4	11	7	12	1
Assessor 4	12	11	8	3	9	15	5	7	10	16	6	5	13	12	7	8
Assessor 5	2	11	10	3	6	15	14	7	4	16	9	5	1	12	11	8
Assessor 6	4	3	10	12	1	7	14	9	2	5	9	10	3	8	11	13
Assessor 7	16	3	8	10	13	7	5	14	14	5	6	9	15	8	7	11
Assessor 8	8	4	12	10	5	1	9	14	6	2	10	9	7	3	13	11
Assessor 9	3	2	16	12	7	6	13	9	5	4	14	10	8	1	15	13
Assessor 10	11	12	4	2	15	9	1	6	16	10	2	4	12	13	3	1
Assessor 11	2	10	4	16	6	14	1	13	4	9	2	14	1	11	3	15
Assessor 12	8	16	11	4	5	13	15	1	6	14	16	2	7	15	12	3

Table 7.6: Efficiencies of the factorial contrasts for Example 7.4
Column 2 to 6 show the efficiency of the four preparation designs. In the last column the efficiencies from the complete sensory design, constructed from P1 and and the panel design from Example 6.6 are listed.

	Preparation designs				Complete design
Weights	(2,1)	(4,1)	(1,0)	(1,2)	(2,1)
Factor level	P1	P2	P3	P4	C1
F1–F2	94.12	94.12	94.12	71.79	90.94
F1–F3	100.00	94.12	94.12	80.00	97.10
F1–F4	94.12	100.00	100.00	87.50	91.20
F2–F3	94.12	100.00	100.00	87.50	91.20
F2–F4	100.00	94.12	94.12	80.00	97.10
F3–F4	94.12	94.12	94.12	71.79	90.33
P1–P2	100.00	100.00	100.00	82.35	93.97
P1–P3	100.00	100.00	100.00	82.35	94.50
P2–P3	100.00	100.00	100.00	82.35	94.00
(F1–F2)(P1–P2)	84.42	84.42	67.27	96.55	81.41
(F1–F2)(P2–P3)	84.42	84.42	76.42	96.55	80.78
(F1–F2)(P1–P3)	84.42	84.42	95.11	96.55	81.01
(F1–F3)(P1–P2)	83.33	84.42	82.58	100.00	80.27
(F1–F3)(P2–P3)	83.33	84.42	64.20	100.00	80.80
(F1–F3)(P1–P3)	83.33	84.42	72.60	100.00	81.11
(F1–F4)(P1–P2)	84.42	83.33	62.54	96.55	81.40
(F1–F4)(P2–P3)	84.42	83.33	61.12	96.55	82.37
(F1–F4)(P1–P3)	84.42	83.33	89.80	96.55	80.85
(F2–F3)(P1–P2)	84.42	81.25	90.90	96.55	81.69
(F2–F3)(P2–P3)	84.42	81.25	73.46	96.55	79.88
(F2–F3)(P1–P3)	84.42	81.25	64.98	96.55	81.53
(F2–F4)(P1–P2)	81.25	84.42	77.95	100.00	78.27
(F2–F4)(P2–P3)	81.25	84.42	76.39	100.00	77.98
(F2–F4)(P1–P3)	81.25	84.42	84.32	100.00	79.60
(F3–F4)(P1–P2)	84.42	84.42	73.77	96.55	82.28
(F3–F4)(P2–P3)	84.42	84.42	66.52	96.55	81.47
(F3–F4)(P1–P3)	84.42	84.42	82.75	96.55	81.96

Table 7.7: Factorial preparation designs, $D^{2;3\times4}(12,6,6)$, for Example 7.2 Design P1 for qualitative factors and design P5 for quantitative factors.

P1

12	7	9	5	2	1
9	8	11	4	1	3
7	11	12	3	5	4
10	12	8	1	6	5
8	10	7	6	3	2
11	9	10	2	4	6

P5

12	8	7	5	4	3
10	9	11	1	2	6
9	7	11	2	6	4
12	8	10	5	1	3
10	9	8	5	6	1
7	12	11	2	3	4

shows that it is preferable to include all contrasts that are estimated in the analysis, and assign to them non-zero weights. If the interactions are of any interest at all, however minute, the first two designs are preferable over the third.

The fourth design, which was constructed with weights giving priority to interaction contrasts, reflects this ordering in the efficiencies. Interaction contrasts have an efficiency of either 100% or 97%, the main effect contrasts of factor P have an efficiency of 82% and the main effect contrasts of factor F are in the range 72 to 88%. This design is preferable to the other three if interactions are of primary and main effects of secondary interest.

From these four preparation designs, the first design, P1, is selected and combined with panel design $D^2_{co}(6,12,4)$*, which has already been used in Example 6.6. For step three of the three-step procedure, optimising the column structure of the preparation design, the average efficiency of all main effect contrasts is chosen as optimality criterion. Average efficiencies of the main effects vary between 85.98% and 86.60% for the different column permutations. The efficiencies for the factorial contrasts are shown in Table 7.6; the factorial preparation design with the optimal column order is shown in Table 7.7 and the final sensory design is presented in Table 7.8.*

Table 7.8: Complete sensory design $D_R^{3\times 4}(12, 12, 6, 4; 6)$ with factorial structure and qualitative levels for Example 7.2

	Session 1				Session 2				Session 3				Session 4				Session 5				Session 6			
Assessor 1	9	7	5	2	11	8	4	1	12	11	3	5	8	12	1	6	7	10	6	3	10	9	2	4
Assessor 2	7	2	12	5	8	1	9	4	11	5	7	3	12	6	10	1	10	3	8	6	9	4	11	2
Assessor 3	5	12	2	9	4	9	1	11	3	7	5	12	1	10	6	8	6	8	3	7	2	11	4	10
Assessor 4	9	2	1	7	11	1	3	8	12	5	4	11	8	6	5	12	7	3	2	10	10	4	6	9
Assessor 5	1	5	7	12	3	4	8	9	4	3	11	7	5	1	12	10	2	6	10	8	6	2	9	11
Assessor 6	2	12	9	1	1	9	11	3	5	7	12	4	6	10	8	5	3	8	7	2	4	11	10	6
Assessor 7	7	1	9	5	8	3	11	4	11	4	12	3	12	5	8	1	10	2	7	6	9	6	10	2
Assessor 8	1	2	7	9	3	1	8	11	4	5	11	12	5	6	12	8	2	3	10	7	6	4	9	10
Assessor 9	12	9	5	1	9	11	4	3	7	12	3	4	10	8	1	5	8	7	6	2	11	10	2	6
Assessor 10	2	5	9	12	1	4	11	9	5	3	12	7	6	1	8	10	3	6	7	8	4	2	10	11
Assessor 11	5	1	12	7	4	3	9	8	3	4	7	11	1	5	10	12	6	2	8	10	2	6	11	9
Assessor 12	12	7	1	2	9	8	3	1	7	11	4	5	10	12	5	6	8	10	2	3	11	9	6	4

Table 7.9: Efficiencies for Example 7.5

Contrast	Efficiency Preparation design P5	 Complete design C5
F_L	100.00	92.01
F_Q	88.89	85.40
F_C	100.00	95.91
P_L	100.00	96.79
P_Q	100.00	90.26
F_LP_L	100.00	88.58
F_QP_L	55.56	53.61
F_CP_L	100.00	92.52
F_LP_Q	100.00	81.76
F_QP_Q	55.56	49.84
F_CP_Q	100.00	82.02

Example 7.5 Designs for Example 7.2 with quantitative and equally spaced levels. *If the factors are quantitative and their levels equi-distant, a preparation design for the orthogonal contrasts listed in Table 7.3 could be used instead. Choosing weights 4 and 1 for the main effects and interaction respectively, the preparation design P5, shown in Table 7.7, was produced by algorithm JE5. Its efficiencies for the quantitative contrasts and the efficiencies from the resulting complete sensory design (listed in Table 7.10) are given in Table 7.9. For the preparation design, only three contrasts are partially confounded, the quadratic contrast of factor F (resulting in an efficiency of 89%) and the cubic-by-linear and the linear-by-quadratic interactions with an efficiency of 56% each. The same relation is visible for the final design, with effciencies between 86% and 96% for main effects and 50% to 92% for interactions. The average efficiencies for the complete design vary between 87.69% and 91.88% for the column permutations of the preparation design.*

Table 7.10: Complete sensory design $D_R^{3\times4}(12,12,6,4;6)$ with factorial structure and quantitative levels for Example 7.2

	Session 1				Session 2				Session 3				Session 4				Session 5				Session 6			
Assessor 1	5	4	8	12	1	2	9	10	2	6	7	9	5	1	8	12	5	6	9	10	2	3	12	7
Assessor 2	7	3	12	4	11	6	10	2	11	4	9	6	10	3	12	1	8	1	10	6	11	4	7	3
Assessor 3	8	5	3	7	9	1	6	11	7	2	4	11	8	5	3	10	9	5	1	8	12	2	4	11
Assessor 4	3	4	12	8	6	2	10	9	4	6	9	7	3	1	12	8	1	6	10	9	4	3	7	12
Assessor 5	5	12	3	7	1	10	6	11	2	9	4	11	5	12	3	10	5	10	1	8	2	7	4	11
Assessor 6	8	3	5	12	9	6	1	10	7	4	2	9	8	3	5	12	9	1	5	10	12	4	2	7
Assessor 7	4	7	8	5	2	11	9	1	6	11	7	2	1	10	8	5	6	8	9	5	3	11	12	2
Assessor 8	7	8	4	3	11	9	2	6	11	7	6	4	10	8	1	3	8	9	6	1	11	12	3	4
Assessor 9	12	5	7	4	10	1	11	2	9	2	11	6	12	5	10	1	10	5	8	6	7	2	11	3
Assessor 10	4	8	7	3	2	9	11	6	6	7	11	4	1	8	10	3	6	9	8	1	3	12	11	4
Assessor 11	3	12	5	8	6	10	1	9	4	9	2	7	3	12	5	8	1	10	5	9	4	7	2	12
Assessor 12	12	7	4	5	10	11	2	1	9	11	6	2	12	10	1	5	10	8	6	5	7	11	3	2

7.3 Conclusions

The examples shown in this chapter only give a small glimpse of the range of sensory factorial designs. When products have a factorial structure, interest is in special contrasts and we have shown how the basic three-step procedure from Chapter 5 can be modified to reflect the contrasts of interest and their relative importance. While in the earlier chapters interest was always in pairwise contrasts of products, in this chapter efficient designs are found for other preplanned contrasts.

The factorial structure is incorporated in the first stage of the procedure, the construction of the preparation design and we have shown how factorial preparation designs can be constructed, either "by hand" or by the algorithm of Jones and Eccleston (1980) with an appropriate modification. An external source can be used to construct a factorial design, as long as it satisfies the constraints imposed on the preparation design. Alternatively, the preparation design can be constructed within the three-step procedure by specifying the contrasts of interest.

When contrasts of interest and the associated weights are specified the introduced method can be used to construct equi-replicated designs of any kind. This includes factorial treatment-control designs and designs for experiments in which some products have a factorial setting and others do not.

It has been shown how designs with a higher replication for a single control product can be constructed. This procedure can be extended to designs with more than one control product, although this is more difficult because products are divided into sets. It will be fairly straightforward to modify the algorithms for a special design example, in which more than one product has a higher replication. To find a general rule that assigns any number of test and control products to sets will be more difficult, though.

Chapter 8

Conclusions and further work

Designs used for sensory studies are generally repeated measurement designs that take order and carry-over effects into account. Complete and incomplete block designs for such studies are well known and have been discussed in Chapter 3. When a large number of products are compared in a trial, the experiment is split into sessions and resolvable row-column or cross-over designs are required. Algorithms for such designs have been reviewed in Chapter 4.

A common problem for sensory experiments is that only a subset of all products can be prepared for a session. The usual solution to this problem has been so far to use designs, in which the number of products for each session equals the number of servings in the session, so that every assessor tastes the same set of products in a session. While there is a limit on the number of products that can be prepared for a session, it is often larger than the number of servings in a session. It has been shown in this dissertation that it is beneficial to exploit the preparation constraint and construct more efficient designs than the simple split-plot designs.

The main purpose of this thesis was to create an algorithm to construct efficient sensory multi-session designs with preparation constraints. Building on an initial idea of Ian Wakeling and Roland Carpenter, a fully automated procedure was created to construct designs in which all pairwise product comparisons are of equal interest. All search algorithms have been programmed as macros in SAS/IML. These macros have been incorporated in a web-based design service provided (internally) by Unilever.

In the first step of the procedure, a preparation design is constructed. This is an incomplete block design with a special column structure that ensures the resolvability of the resulting sensory design. This preparation design assigns products to sessions. In this thesis, three search algorithms, modified to construct designs with the required column structure, have been compared and the algorithm based on the one by Jones and Eccleston (1980) performed best. The algorithm by Jones (1980) is then used to re-sort the products into the required column structure.

The second step is the construction of a panel design, a cross-over design with a special structure on the contents of the rows. It is required to ensure the resolvability of the complete design. It assigns columns from the preparation design to assessors. The panel design is constructed in two steps, first an efficient IBD of the required size and with the row-constraint is constructed, then the order and carry-over structures are optimised. While it is known that optimal cross-over designs are not necessarily built from efficient incomplete block designs, the two-step approach is chosen, since it is the easiest to provide the row structure for efficient panel designs, while still creating efficient designs. The IBD for the panel design is constructed with the same algorithm as is used for the preparation designs. The algorithm by Ball (1997) is used to optimise the estimation of order and carry-over effects.

Another stage of the three-step procedure is required, when the final design is not balanced. The efficiency of the complete sensory design varies for different column permutations of the preparation design. Thus, the column order with the highest efficiency is searched for. If the number of permutations is small enough, all column permutations are explored, otherwise the best permutation from a random subset is used.

The three-step procedure produces efficient multi-session designs with preparation constraints within minutes. When efficient preparation and panel designs are stored it is often possible to re-use these component designs by combining them with stored designs of different sizes. If, for example, a design is needed for a number of assessors different from the stored one, then only steps two and three need to be re-run. This time saving feature is an issue only for large designs. Otherwise, it is recommended, from a randomisation point of view, to construct a new designs for each experiment. Assignment of products to design entries and assessors to rows should be randomised for every new experiment anyway.

The basic procedure was extended to cover treatment-control and factorial designs. Treatment-control designs are useful, when the greatest interest is in product comparisons involving the control product. Increased precision of these contrasts can most easily be achieved by increased replication of the control product. This can be done in keeping with the convention that all non-control products are tasted by all assessors the same number of times. With the adjusted three-step procedure a method has been developed for the construction of efficient treatment-control designs for multi-session trials with preparation constraints. Treatment-control designs have so far rarely been used in sensory trials, because this class of designs is generally not well known and has not been easily available.

In experiments with a factorial structure, designs optimised for all pairwise product comparisons are not necessarily the best option any longer. In factorial experiments, interest is usually in the main effects and lower-order interactions, and designs are optimised for the selected contrasts of interest. With the extension of the three-step procedure to factorial designs, sensory multi-session designs with preparation constraints can be constructed for any set of contrasts relevant for a specific experiment. Since the matrix of contrasts has to be provided by the user, this is a flexible tool for a wide range of experiments.

A potential extension for the three-step procedure, which has not been considered in this thesis, are designs for experiments with more than one control product. For such trials specifying meaningful weights for the algorithm is more complex: weights are required for the contrasts of each control with the test products and for contrasts between the different controls. Weights for control-test contrasts do not need to be identical and can differ between the controls. The problem for the construction of treatment-control preparation designs with several controls is due to the special column structure of the preparation designs. Products with different replication have to be assigned to sets. While this can easily be solved on a case by case basis, it is much more difficult to do so in a general algorithm. However, one advantage of the three-step procedure is that special preparation designs can be constructed externally and can then be incorporated into the procedure if they follow the two necessary constraints. Thus, specialised non-equally replicated designs can be constructed "by hand" or can be taken from the literature and can be included as preparation designs.

All designs in this thesis are constructed as cross-over designs. If carry-over effects are regarded as not relevant for an experiment, this algorithm can be used also for the construction of row-column designs by using a row-column panel design.

The designs in this thesis have been optimised for two simple fixed-effects models. The procedure can be adapted for other linear models by altering the third step, in which the efficiency could be calculated according to other models. This might be relevant especially for models with random assessor effects, which are frequently used in sensory trials.

Appendix A

Notation

A.1 Notation for models in Chapter 2

The following is an overview of occurring effects:

α_a assessor location effect, $a = 1, \ldots, A$

β_a "stretching" factor for each assessor, $a = 1, \ldots, A$

γ intercept in dispersion model

δ_a assessor effect in dispersion model, $a = 1, \ldots, A$

ε_{atr} residual of assessor a on treatment t in replication r, $a = 1, \ldots, A$, $t = 1, \ldots, T$ and $r = 1, \ldots, R$

ζ_s session effect, $s = 1, \ldots, S$

θ vector of all unknown fixed parameters

π_i serving position / period effect, $p = 1, \ldots, P$

ρ_t carry-over effect of product t, $\quad t = 1, \ldots, T$

σ^2 variance of observation y_{atr}

τ_t product/treatment effect, $t = 1, \ldots, T$

ϕ_a scale parameter, $a = 1, \dots, A$

ξ_k cut-points, $k = 1, \dots, K$

μ intercept in location model

r_{at} is the discrete ordinal observation of assessor a and product t with possible values between 1 and K

u_a vector of random effects for each assessor a, $a = 1, \dots, A$

x_{at} vector of known explanatory variables for fixed effects

y_{atr} observation of assessor a on treatment t in replication r, $a = 1, \dots, A$, $t = 1, \dots, T$ and $r = 1, \dots, R$. In the Erichsen model it is the unobserved underlying continuous normal variate instead, see r_{at}.

z_{at} vector of known explanatory variables for random effects

A.2 Notation for designs in Chapters 3 to 7

Summary of design notation

a = number of **assessors** taking part in the trial

p = number of **products** / treatments compared in the trial

r = number of **replicates**, i.e.the number of times each assessor tests all p products

t = number of **products tested per session** by each assessor

s = number of **sessions**

k = number of **products available at each session**

q = number of **sets** products are divided into

r_p = **product replication**, i.e. the number of times each product appears in the preparation design

α_j	assessor effect
β_k	session effect
π_i	serving order effect
τ_l	product effect
ρ_l	residual effect
X	model/design matrix
X_1	design matrix related to nuisance effect
X_2	design matrix related effects of interest
X_A	design matrix related to assessor effects
X_P	design matrix related to product effects
X_R	design matrix related to residual effects
X_S	design matrix related to session effects
X_T	design matrix related to serving order effects

N_A	$=$	$X_P'X_A$	assessor incidence matrix
N_S	$=$	$X_P'X_S$	session incidence matrix
N_T	$=$	$X_P'X_T$	serving order incidence matrix
$\tilde{N}_A$	$=$	$X_R'X_A$	assessor incidence matrix for residual effects
$\tilde{N}_S$	$=$	$X_R'X_S$	session incidence matrix for residual effects
$\tilde{N}_T$	$=$	$X_R'X_T$	serving order incidence matrix for residual effects
$\boldsymbol{r}_R$	$=$	$X_R'1_{at}$	replication vector of the carry-over effects
Y_{++}	$=$	$Y1_{ats}$	sum of all observations
Y_P	$=$	$X_P'Y$	vector of product sums
Y_A	$=$	$X_A'Y$	vector of assessor sums
Y_S	$=$	$X_S'Y$	vector of session sums
Y_T	$=$	$X_T'Y$	vector of serving position sums
$\bar{Y}_{++}$	$=$	$\frac{1}{ats}Y_{++}$	mean of all observations
$\bar{Y}_P$	$=$	$\frac{1}{ar_p}Y_P$	vector of product means
$\bar{Y}_A$	$=$	$\frac{1}{ts}Y_A$	vector of assessor means
$\bar{Y}_S$	$=$	$\frac{1}{at}Y_S$	vector of session means
$\bar{Y}_T$	$=$	$\frac{1}{as}Y_T$	vector of serving position means
Λ_A	$=$	N_AN_A'	assessor concurrence matrix

Appendix B

Reduced normal equations

B.1 Generalised inverses of $X'X$

Definition B.1 (Generalised inverse) *A generalised inverse of a matrix A is defined as any matrix G that satisfies the equation*

$$AGA = A$$

Characteristics of a generalised inverses of $X'X$

Let $G = (X'X)^{-}$ be any generalised inverse of $X'X$, then

a) $X'XGX'X = G$

b) $XGX'X = X$

c) XGX' is invariant to G

d) XGX' is symmetric

Calculating generalised inverses

When deriving the information matrices for the models introduced in this thesis we will make use of three calculating methods for generalised inverses of partioned symmetric matrices. Two ways are given in Searle (1971): Let XX', the matrix of cross-products of the designs matrix be partitioned as

$$XX' = \begin{bmatrix} X_1' \\ X_2' \end{bmatrix} [X_1 \; X_2] = \begin{bmatrix} X_1'X_1 & X_1'X_2 \\ X_2'X_1 & X_2'X_2 \end{bmatrix} = \begin{bmatrix} A & B \\ B' & D \end{bmatrix}$$

Calculation Method B.1:

If rank(XX')=rank(D) then

$$(XX')^{-} \begin{bmatrix} 0 & 0 \\ 0 & D^{-}1 \end{bmatrix} \qquad \text{(B.1)}$$

Calculation Method B.2:

If both $A^{-} = (X_1X_1')^{-}$ and $D^{-} = (X_2X_2')^{-}$ can be calculated then

$$\begin{aligned}(XX')^{-} &= \begin{bmatrix} A^{-} + A^{-}BQ^{-}B'A^{-} & -A^{-}BQ^{-} \\ -Q^{-}B'A^{-} & Q^{-} \end{bmatrix} \qquad \text{(B.2)} \\ &= \begin{bmatrix} A^{-} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -A^{-}B \\ I \end{bmatrix} Q^{-}[-A^{-}B \quad I]\end{aligned}$$

Calculation Method B.3:

Another way of calculating a generalised inverse of the information matrix is given by Shah (1959) as

$$C^{-} = (C + xJ)^{-1}, \qquad \text{where } x \neq 0,$$

if a matrix $C + xJ$ can be found that is non-singluar. This solution is most useful for the special case when $X'X$ has the form $X'X = a(I_b - \frac{1}{b}J_{b,b})$. Then

$$(XX')^{-} = \frac{1}{a} I_b \qquad \text{(B.3)}$$

B.2 Projection matrices

Let

$$XX' = \begin{bmatrix} X_1' \\ X_2' \end{bmatrix} [X_1 \; X_2]$$

The projection matrix from X_1 into X_2 is $P = I - X_1 \left(X_1'X_1\right)^{-} X_1'$

a) P is idempotent $(P = PP)$

b) P is symmetric $(P = P')$

B.3 Information matrices for incomplete block designs

In the following sections the information matrices are derived for the models described in this thesis. If incomplete block designs are used for sensory experiments the number of observations per trial is $r_p \; p = a \; t$ and the product replication is $r_p = a\,t/p$

For the partitions of the design matrix holds:

$$\begin{array}{llllll}
\text{Assessors:} & X_A' \mathbf{1}_{at} & = & t\mathbf{1}_a, & X_A' X_A & = & tI_a \\
\text{Serving positions:} & X_T' \mathbf{1}_{at} & = & a\mathbf{1}_t, & X_T' X_T & = & aI_t \\
\text{Products:} & X_P' \mathbf{1}_{at} & = & r_p\mathbf{1}_p, & X_P' X_P & = & r_pI_p \\
\text{Carry-over:} & X_R' \mathbf{1}_{at} & = & \boldsymbol{r}_R, & X_R' X_R & = & \boldsymbol{r}_R^{\delta}
\end{array}$$

B.3.1 Randomised incomplete block designs

The **linear model** (3.16) appropriate for analysing data from randomized complete block designs is

$$Y_{jk} = \mu + \alpha_j + \tau_j + \varepsilon_{jk} \quad 1 \leq j \leq a, \quad 1 \leq k \leq p$$

The **design matrix** X for this model is

$$X = [1_{ap} \;\; X_A \;\; X_P] = [X_1 \;\; X_2]$$

where

$$X_1 = [1_{ap} \;\; X_A] \quad \text{and} \quad X_2 = [X_P]$$

with $\text{rank}(X_1) = a$. The **matrix of crossproducts of the nuisance effects** $X_1'X_1$ has the form

$$X_1'X_1 = \begin{bmatrix} 1_{at}'1_{at} & 1_{at}'X_A \\ X_T'1_{at} & X_A'X_A \end{bmatrix} = \begin{bmatrix} at & t\,1_a' \\ t\,1_a & tI_a \end{bmatrix}$$

A **general inverse** for $X_1'X_1$ is given with method B.3:

$$(X_1'X_1)^- = \frac{1}{t}\begin{bmatrix} 0 & 0_{1,a} \\ 0_{a,1} & I_a \end{bmatrix}$$

The **projection matrix** P_1 is independent of the choice of the general inverse and can be expressed as

$$\begin{aligned} P_1 &= I_{at} - X_1\,(X_1'X_1)^-\,X_1' \\ &= I_{at} - [1_{at}X_A]\frac{1}{t}\begin{bmatrix} 0 & 0_{1,a} \\ 0_{a,1} & I_a \end{bmatrix}\begin{bmatrix} 1_{at} \\ X_A' \end{bmatrix} \\ &= I_{at} - \frac{1}{t}\,[0 \;\; X_A]\begin{bmatrix} 1_{at} \\ X_A' \end{bmatrix} \\ &= I_{at} - \frac{1}{t}X_AX_A' \end{aligned}$$

The information matrix for product effects and the right hand-side of the reduced normal equations Q_P in model (3.16) are therefore

$$\begin{aligned} C_{3.16} &= X_P'P_1X_P \\ &= X_P'X_P - X_P'X_AX_A'X_P \end{aligned}$$

$$
\begin{aligned}
&= r_p\, I_p - \frac{1}{t} N_A N_A' \\
Q_{3.16} &= X_P' P_1 Y \\
&= X_P' Y - \frac{1}{t} N_A X_A' Y \\
&= Y_P - \frac{1}{t} N_A Y_A'
\end{aligned}
$$

B.3.2 Row-column designs

The model equation for row-column designs is model (3.21):

$$Y_{ijk} = \mu + \pi_i + \alpha_j + \tau_k + \varepsilon_{ijk} \quad 1 \leq i \leq t, \quad 1 \leq j \leq a, \quad 1 \leq k \leq p$$

The design matrix X and the matrix of crossproducts of the nuisance effects for this model are

$$
\begin{aligned}
X &= [1_{at}\ X_T\ X_A\ X_P] \\
X_1 &= [1_{at}\ X_T\ X_A] \\
X_2 &= [X_P] \\
X_1'X_1 &= \begin{bmatrix} 1_{at}'1_{at} & 1_{at}'X_T & 1_{at}'X_A \\ X_T'1_{at} & X_T'X_T & X_T'X_A \\ X_A'1_{at} & X_A'X_T & X_A'X_A \end{bmatrix} \\
&= \begin{bmatrix} at & a1_t' & t1_a' \\ a1_t & aI_t & J_{t,a} \\ p1_a & J_{a,t} & tI_a \end{bmatrix}
\end{aligned}
$$

A generalised inverse for $X_1'X_1$ can be found using Calculation Method B.2, for which $X_1'X_1$ is devided into 4 sub-matrices:

$$
\begin{aligned}
X_1'X_1 &= \begin{bmatrix} A & B \\ B' & D \end{bmatrix} \\
A &= a\begin{bmatrix} p & 1_p' \\ 1_p & I_p \end{bmatrix} \\
B &= \begin{bmatrix} p1_a' \\ J_{p,a} \end{bmatrix} \\
D &= pI_a \\
A^- &= \frac{1}{a}\begin{bmatrix} 0 & 0_{1,p} \\ 0_{p,1} & I_p \end{bmatrix} \\
B'A^- &= [p1_a \; J_{ap}]\frac{1}{a}\begin{bmatrix} 0 & 0_{1,p} \\ 0_{p,1} & I_p \end{bmatrix} = \frac{1}{a}[0_{a,1} \quad J_{a,p}] \\
A^-B &= (B'A^-)' = \frac{1}{a}\begin{bmatrix} 0_{1,a} \\ J_{p,a} \end{bmatrix} \\
Q &= D - B'A^-B = pI_a - \frac{1}{a}[0_{a,1} \quad J_{a,p}]\begin{bmatrix} p1_a' \\ J_{p,a} \end{bmatrix} \\
&= pI_a - \frac{1}{a}J_{a,p}J_{p,a} = p(I_a - \frac{1}{a}J_{a,a}) \\
Q^- &\overset{B.3}{=} \frac{1}{p}I_a \\
A^-BQ^- &= \frac{1}{a}\begin{bmatrix} 0_{1,a} \\ J_{p,a} \end{bmatrix}\frac{1}{p}I_a = \frac{1}{ap}\begin{bmatrix} 0_{1,a} \\ J_{p,a} \end{bmatrix} \\
Q^-B'A^- &= (A^-BQ^-)' = \frac{1}{ap}[0_{a,1} \quad J_{a,p}] \\
A^-BQ^-B'A^- &= \frac{1}{ap}\begin{bmatrix} 0_{1,a} \\ J_{p,a} \end{bmatrix}\frac{1}{a}[0_{a,1} \quad J_{a,p}] = \frac{1}{a^2p}\begin{bmatrix} 0 & 0_{1,p} \\ 0_{p,1} & J_{p,a}J_{a,p} \end{bmatrix} \\
&= \frac{1}{ap}\begin{bmatrix} 0 & 0_{1,p} \\ 0_{p,1} & J_{p,p} \end{bmatrix}
\end{aligned}
$$

With these calculations, a generalised inverse of $X_1'X_1$ is

$$(X_1'X_1)^- = \begin{bmatrix} A^- + A^-BQ^-B'A^- & -A^-BQ^- \\ -Q^-B'A^- & Q^- \end{bmatrix} = \frac{1}{at}\begin{bmatrix} 0 & 0_{1,t} & 0_{1,a} \\ 0_{t,1} & t\,I_t + J_{t,t} & -J_{t,a} \\ 0_{a,1} & -J_{a,t} & a\,I_a \end{bmatrix}$$

The projection and information matrices and Q_P for model (3.21) are:

$$\begin{aligned}
P_1 &= I_{ap} - X_1\,(X_1'X_1)^-\,X_1' \\
&= I_{ap} - [1_{at}X_TX_A]\,\frac{1}{at}\begin{bmatrix} 0 & 0_{1,t} & 0_{1,a} \\ 0_{t,1} & tI_t + J_{t,t} & -J_{t,a} \\ 0_{a,1} & -J_{at} & aI_a \end{bmatrix}\begin{bmatrix} 1_{at}' \\ X_T' \\ X_A' \end{bmatrix} \\
&= I_{at} - \frac{1}{at}\,[0 \;\; X_T(tI_t + J_{t,t}) + X_A(-J_{a,t}) \;\; X_T(-J_{t,a}) + X_A(aI_a)]\begin{bmatrix} 1_{at}' \\ X_T' \\ X_A' \end{bmatrix} \\
&= I_{at} - \frac{1}{at}\,[X_T(tI_t + J_{t,t})X_T' - X_AJ_{a,t}X_T' - X_TJ_{t,a}X_A' + X_A(aI_a)X_A'] \\
&= I_{at} - \frac{1}{a}X_TX_T' - \frac{1}{t}X_AX_A' - \frac{1}{at}\,(X_T1_t1_t'X_T' - X_A1_a1_t'X_T' - X_T1_t1_a'X_A') \\
&= I_{at} - \frac{1}{a}X_TX_T' - \frac{1}{t}X_AX_A' + \frac{1}{at}J_{at,at} \\
C_{3.17} &= X_P'P_1X_P \\
&= X_P'\left(I_{ap} - \frac{1}{a}X_TX_T' - \frac{1}{p}X_AX_A' + \frac{1}{at}J_{ap,ap}\right)X_P \\
&= X_P'X_P - \frac{1}{a}X_P'X_TX_T'X_P - \frac{1}{t}X_P'X_AX_A'X_P + \frac{1}{at}X_P'J_{at,at}X_P \\
&= r_pI_p - \frac{1}{a}N_TN_T' - \frac{1}{t}N_AN_A' + \frac{r_p}{p}J_{p,p} \\
Q_{3.17} &= X_P'P_1Y \\
&= X_P'Y - \frac{1}{a}X_P'X_TX_T'X_P - \frac{1}{t}X_P'X_AX_A'X_P + \frac{1}{at}X_P'J_{at,at}Y \\
&= Y_P - \frac{1}{a}N_TY_P' - \frac{1}{t}N_AY_A' + \frac{r_pY_{++}}{p}1_{at}
\end{aligned}$$

B.3.3 Cross-over designs

Model (3.22):

$$Y_{d(i,j)} = \mu + \pi_i + \alpha_j + \tau_{d(i,j)} + \rho_{d(i-1,j)} + \varepsilon_{ij} \qquad 1 \le i \le t, \quad 1 \le j \le a$$

$$\begin{aligned}
X &= [1_{at} X_T \;\; X_A \;\; X_P] \\
X_1 &= [1_{at} X_T \;\; X_A] \\
X_2 &= [X_P \;\; X_R] \\
X_1'X_1 &= \begin{bmatrix} at & a1_t' & t1_a' \\ a1_t & aI_t & J_{t,a} \\ p1_a & J_{a,t} & tI_a \end{bmatrix} \\
P_1 &= I_{at} - \frac{1}{a} X_T X_T' - \frac{1}{t} X_A X_A' + \frac{1}{at} J_{at,at} \\
C_{11;3.22} &= X_P' P_1 X_P \\
&= r_p I_p - \frac{1}{a} N_T N_T' - \frac{1}{t} N_A N_A' + \frac{r_p}{p} J_{p,p} \\
C_{12;3.22} &= X_P' P_1 X_R \\
&= X_P' X_R - \frac{1}{a} N_T \tilde{N}_T' - \frac{1}{t} N_A \tilde{N}_A' + \frac{1}{at} X_P' J_{at,at} X_R \\
&= X_P' X_R - \frac{1}{a} N_T \tilde{N}_T' - \frac{1}{t} N_A \tilde{N}_A' + \frac{r_p}{at} 1_p \boldsymbol{r}_R' \\
&= C_{21}' \\
C_{22;3.22} &= X_R' P_1 X_R \\
&= X_R' X_R - \frac{1}{a} \tilde{N}_T \tilde{N}_T' - \frac{1}{t} \tilde{N}_A \tilde{N}_A' + \frac{1}{at} X_P' J_{at,at} X_R \\
&= X_R' X_R - \frac{1}{a} \tilde{N}_T \tilde{N}_T' - \frac{1}{t} \tilde{N}_A \tilde{N}_A' + \frac{1}{at} \boldsymbol{r}_R \boldsymbol{r}_R' \\
Q_{1;3.22} &= X_P (I_{at} - \frac{1}{a} X_T X_T' - \frac{1}{t} X_A X_A' + \frac{1}{at} J_{at,at}) Y \\
&= Y_P - \frac{1}{a} N_T Y_T - \frac{1}{t} N_A Y_A + r_p 1_p Y_{++} \\
Q_{2;3.22} &= X_R (I_{at} - \frac{1}{a} X_T X_T' - \frac{1}{t} X_A X_A' + \frac{1}{at} J_{at,at}) Y \\
&= Y_R - \frac{1}{a} \tilde{N}_T Y_T - \frac{1}{t} \tilde{N}_A Y_A + \frac{Y_{++}}{at} \boldsymbol{r}_R 1_p
\end{aligned}$$

$$\begin{aligned}
C_{D;3.22} &= C_{11} - C_{12}C_{22}^{-}C_{21}\\
C_{R;3.22} &= C_{22} - C_{21}C_{11}^{-}C_{12}\\
Q_{D;3.22} &= Q_{1} - C_{12}C_{22}^{-}Q_{2}\\
Q_{R;3.22} &= Q_{2} - C_{21}C_{11}^{-}Q_{1}
\end{aligned}$$

B.4 Information matrices for complete block designs

The information matrices of complete block designs are special cases of the information matrices $C_{3.16}, C_{3.21}$ and $C_{3.22}$ for incomplete block design with

$$\begin{aligned}
t &= p\\
r_p &= a\\
N_A &= J_{a,p}
\end{aligned}$$

Number of observations in a trial from a complete block design: ap
Replication: $r = 1$, $r_p = a$

$$\begin{aligned}
&\text{Assessors:} & X_A'1_{ap} &= p1_a, & X_A'X_A &= pI_a\\
&\text{Serving position:} & X_T'1_{ap} &= a1_p, & X_T'X_T &= aI_p\\
&\text{Products:} & X_P'1_{ap} &= a1_p, & X_P'X_P &= aI_p
\end{aligned}$$

For the following three models the information matrices can be derived from the information matrices above:

$$\begin{aligned}
&\text{Model (3.1):} & Y_{jk} &= \mu + \alpha_j + \tau_k + \varepsilon_{jk}\\
&\text{Model (3.2):} & Y_{ijk} &= \mu + \pi_i + \alpha_j + \tau_k + \varepsilon_{ijk}\\
&\text{Model (3.3):} & Y_{d(ij)} &= \mu + \pi_i + \alpha_j + \tau_{d(i,j)} + \rho_{d(i-1,j)} + \varepsilon_{ij}\\
& & & 1 \le j \le a, 1 \le i, k \le p.
\end{aligned}$$

$$
\begin{aligned}
C_{3.1} &= a\,I_p - \frac{1}{p}N_A N_A' \\
&= a\left(I_p - \frac{1}{p}\,J_{pp}\right) \\
C_{3.2} &= aI_p - \frac{1}{a}N_T N_T' - \frac{1}{p}N_A N_A' + \frac{a}{p}J_{p,p} \\
&= aI_p - \frac{1}{a}N_T N_T' \\
C_{11;3.3} &= aI_p - \frac{1}{a}N_T N_T' \\
C_{12;3.3} &= X_P' X_R - \frac{1}{a}N_T \tilde{N}_T' - \frac{1}{p}J_{p,a}\tilde{N}_A' + \frac{1}{p}1_p \boldsymbol{r}'_R \\
C_{22;3.3} &= X_R' X_R - \frac{1}{a}\tilde{N}_T \tilde{N}_T' - \frac{1}{p}\tilde{N}_A \tilde{N}_A' + \frac{1}{ap}\boldsymbol{r}_R \boldsymbol{r}'_R \\
C_{D;3.3} &= C_{11} - C_{12}C_{22}^{-}C_{21} \\
C_{R;3.3} &= C_{22} - C_{21}C_{11}^{-}C_{12} \\
Q_{3.1} &= Y_P - \frac{1}{t}N_A Y_A' \\
Q_{3.2} &= Y_p - \frac{1}{a}N_T Y_P' - \frac{1}{t}N_A Y_A' + \frac{r_p Y_{++}}{p}1_{at} \\
Q_{1;3.3} &= X_P(I_{at} - \frac{1}{a}X_T X_T' - \frac{1}{t}X_A X_A' + \frac{1}{at}J_{at,at})Y \\
&= Y_P - \frac{1}{a}N_T Y_T - \frac{1}{t}N_A Y_A + r_p 1_p Y_{++} \\
Q_{2;3.3} &= X_R(I_{at} - \frac{1}{a}X_T X_T' - \frac{1}{t}X_A X_A' + \frac{1}{at}J_{at,at})Y \\
&= Y_R - \frac{1}{a}\tilde{N}_T Y_T - \frac{1}{t}\tilde{N}_A Y_A + \frac{Y_{++}}{at}\boldsymbol{r}_R 1_p \\
Q_{D;3.3} &= Q_1 - C_{12}C_{22}^{-}Q_2 \\
Q_{R3.3} &= Q_2 - C_{21}C_{11}^{-}Q_1
\end{aligned}
$$

B.5 Information matrices for resolvable incomplete block designs

Number of observations in a trial: $apr = ast$

Replication: $r = \frac{st}{p}, r_p = ar$

$$\begin{array}{lrcl} \text{Assessors:} & X_A' 1_{ast} & = & st1_a,\ X_A'X_A \ = \ stI_a \\ \text{Sessions:} & X_S' 1_{ast} & = & at1_s,\ X_S'X_S \ = \ atI_s \\ \text{Serving positions:} & X_T' 1_{ast} & = & as1_t,\ X_T'X_T \ = \ asI_t \\ \text{Products:} & X_P' 1_{apr} & = & ar1_p,\ X_P'X_P \ = \ arI_p \end{array}$$

$$\begin{array}{rcll} N_A' = X_A'X_P & = & rJ_{a,p} & \text{every assessor tastes all products } r \text{ times} \\ X_A'X_S & = & tJ_{a,s} & \text{every assessor tastes } t \text{ products in every session} \\ X_A'X_T & = & sJ_{a,t} & \text{every assessor tastes } s \text{ products in every serving position} \\ X_S'X_T & = & aJ_{s,t} & a \text{ products are tasted in every serving position in a session} \end{array}$$

B.5.1 Information matrix for a randomised RIBD

The simplest form for a linear model for multi-session trials is model (4.3):

$$Y_{jkl} = \mu + \alpha_j + \beta_k + \tau_l + \varepsilon_{jkl}$$

$$1 \leq j \leq a,\ 1 \leq k \leq s,\ 1 \leq l \leq p$$

The design matrix X and the matrix of crossproducts for the nuisance effects $X_1'X_1$ are:

$$\begin{aligned} X &= [1_{st}\ X_S\ X_A\ X_P] \\ X_1 &= [1_{st}\ X_S\ X_A] \\ X_1'X_1 &= \begin{bmatrix} 1_{ast}'1_{ast} & 1_{ast}'X_S & 1_{ast}'X_A \\ X_S'1_{ast} & X_S'X_S & X_S'X_A \\ X_A'1_{ast} & X_A'X_S & X_A'X_A \end{bmatrix} \end{aligned}$$

$$= t\begin{bmatrix} as & a1'_s & s1'_a \\ a1_s & aI_s & J_{s,a} \\ s1_a & J_{a,s} & sI_a \end{bmatrix}$$

The information matrix for this model can be derived from row-column model (3.21), which includes a serving order effect. If this effect and respectively its model matrix X_T is replaced by a session effect with its model matrix X_S the information matrix for model (4.3) can be derived from information matrix $C_{3.21}$:

$$\begin{aligned}
(X_1'X_1)^- &= \frac{1}{ast}\begin{bmatrix} 0 & 0_{1,s} & 0_{1,a} \\ 0_{s,1} & s\,I_s + J_{s,s} & -J_{s,a} \\ 0_{a,1} & -J_{a,s} & a\,I_a \end{bmatrix} \\
P_{1;4.3} &= I_{ast} - X_1\,(X_1'X_1)^-\,X_1' \\
&= I_{ast} - \frac{1}{ast}\,(sX_SX_S' + aX_AX_A' - J_{ast,ast})
\end{aligned}$$

$$\begin{aligned}
C_{4.3} &= X_P'P_{1;4.3}X_P \\
&= arI_p - \frac{1}{at}N_SN_S' \\
Q_{4.3} &= Y_P - \frac{1}{at}N_SY_S
\end{aligned}$$

B.5.2 Information matrix for resolvable row-column designs

Model (4.5):

$$Y_{ijklm} = \mu + \pi_i + \alpha_j + \beta_k + \tau_l + \varepsilon_{ijklm}$$

$$1 \le i \le t, 1 \le j \le a, 1 \le k \le s, 1 \le l \le p, 1 \le m \le r$$

The design matrix X_1 for this model and the matrix of cross-products for the nuisance effects $X_1'X_1$ are

$$X = [1_{st}\;\; X_S\;\; X_A\;\; X_T\;\; X_P]$$

$$
\begin{aligned}
X_1 &= [1_{st}\ X_S\ X_A\ X_T] \\
X_1'X_1 &= \begin{bmatrix} 1_{sta}'1_{ast} & 1_{ast}'X_S & 1_{ast}'X_A & 1_{ast}'X_T \\ X_S'1_{ast} & X_S'X_S & X_S'X_A & X_S'X_T \\ X_A'1_{ast} & X_A'X_S & X_A'X_A & X_A'X_T \\ X_T'1_{ast} & X_T'X_S & X_T'X_A & X_T'X_T \end{bmatrix} \\
&= \begin{bmatrix} ast & at1_s' & st1_a' & as1_t' \\ at1_s & atI_s & tJ_{s,a} & aJ_{s,t} \\ st1_a & tJ_{a,s} & stI_a & sJ_{a,t} \\ as1_t & aJ_{t,s} & sJ_{t,a} & asI_t \end{bmatrix}
\end{aligned}
$$

A generalised inverse for $X_1'X_1$ can be calculated by method B.2. The auxiliary matrices are

$$
\begin{aligned}
A &= t\begin{bmatrix} as & a1_s' & s1_a' \\ a1_s & aI_s & J_{s,a} \\ s1_a & J_{a,s} & sI_a \end{bmatrix} \\
B &= \begin{bmatrix} as1_t' \\ aJ_{s,t} \\ sJ_{a,t} \end{bmatrix} \\
D &= asI_t \\
A^- &= \frac{1}{ast}\begin{bmatrix} 0 & 0_{1,s} & 0_{1,a} \\ 0_{s,1} & s\,I_s + J_{s,s} & -J_{s,a} \\ 0_{a,1} & -J_{a,s} & a\,I_a \end{bmatrix} \\
A^-B &= \frac{1}{ast}\begin{bmatrix} 0 & 0_{1,s} & 0_{1,a} \\ 0_{s,1} & s\,I_s + J_{s,s} & -J_{s,a} \\ 0_{a,1} & -J_{a,s} & a\,I_a \end{bmatrix}\begin{bmatrix} as1_t' \\ aJ_{s,t} \\ sJ_{a,t} \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{ast}\begin{bmatrix} 0_{1,t} \\ a(sI_s + J_{s,s})J_{s,t} - sJ_{s,a}J_{a,t} \\ -aJ_{a,s}J_{s,t} + asJ_{a,t} \end{bmatrix} \\
&= \frac{1}{ast}\begin{bmatrix} 0_{1,t} \\ asJ_{s,t} + asJ_{s,t} - saJ_{s,t} \\ -asJ_{a,t} + asJ_{a,t} \end{bmatrix} \\
&= \frac{1}{t}\begin{bmatrix} 0_{1,t} \\ J_{s,t} \\ 0_{a,t} \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
B'A^{-} &= (A^{-}B)' = \frac{1}{t}\,[0_{t,1}\ J_{t,s}\ 0_{t,a}] \\
Q &= D - B'A^{-}B = asI_t - \frac{1}{t}\,[0_{t,1}\ J_{t,s}\ 0_{t,a}]\begin{bmatrix} as1'_t \\ aJ_{s,t} \\ sJ_{a,t} \end{bmatrix} \\
&= asI_t - \frac{a}{t}J_{t,s}J_{s,t} \\
&= as(I_t - \frac{1}{t}J_{t,t}) \\
Q^{-} &= \frac{1}{as}\,I_t \\
A^{-}BQ^{-} &= \frac{1}{t}\begin{bmatrix} 0_{1,t} \\ J_{s,t} \\ 0_{a,t} \end{bmatrix}\frac{1}{as}\,I_t = \frac{1}{ast}\begin{bmatrix} 0_{1,t} \\ J_{s,t} \\ 0_{a,t} \end{bmatrix} \\
Q^{-}B'A^{-} &= (A^{-}BQ^{-})' = \frac{1}{ast}\,[0_{t,1}\ J_{t,s}\ 0_{t,a}] \\
A^{-}BQ^{-}B'A^{-} &= \frac{1}{ast}\begin{bmatrix} 0_{1,t} \\ J_{s,t} \\ 0_{a,t} \end{bmatrix}\frac{1}{t}\,[0_{t,1}\ J_{t,s}\ 0_{t,a}] \\
&= \frac{1}{ast}\begin{bmatrix} 0_{1,1} & 0_{1,s} & 0_{1,a} \\ 0_{s,1} & J_{s,s} & 0_{s,a} \\ 0_{a,1} & 0_{a,s} & 0_{a,a} \end{bmatrix}
\end{aligned}
$$

With these matrices, a generalised inverse for $X_1'X_1$ is

$$
\begin{aligned}
(X_1'X_1)^- &= \begin{bmatrix} A^- + A^-BQ^-B'A^- & -A^-BQ^- \\ -Q^-B'A^- & Q^- \end{bmatrix} \\
&= \frac{1}{ast}\begin{bmatrix} 0 & 0_{1,s} & 0_{1,a} & 0_{1,t} \\ 0_{s,1} & s\,I_s + 2J_{s,s} & -J_{s,a} & -J_{s,t} \\ 0_{a,1} & -J_{a,s} & a\,I_a & 0_{a,t} \\ 0_{t,1} & -J_{t,s} & 0_{t,a} & tI_t \end{bmatrix}
\end{aligned}
$$

Projection matrix for model (4.5) and (4.8) :

$$
\begin{aligned}
P_{1;4.5,4.8} &= I - X_1\,(X_1'X_1)^-\,X_1' \\
&= I_{ast,ast} - \frac{1}{ast}\,[1_{ast}\; X_S\; X_A\; X_T] \begin{bmatrix} 0 & 0_{1,s} & 0_{1,a} & 0_{1,t} \\ 0_{s,1} & s\,I_s & -J_{s,a} & -J_{s,t} \\ 0_{a,1} & -J_{a,s} & a\,I_a & 0_{a,t} \\ 0_{t,1} & -J_{t,s} & 0_{t,a} & tI_t \end{bmatrix} \begin{bmatrix} 1_{ast}' \\ X_S' \\ X_A' \\ X'T \end{bmatrix} \\
&= I_{ast,ast} - \frac{1}{ast}\,[0\;\; sX_S + 2J_{s,s} - X_AJ_{a,s} - X_TJ_{t,s}\;\; - X_SJ_{s,a} + aX_A \\
&\qquad -X_SJ_{s,t} + tX_T\,] \begin{bmatrix} 1_{ast}' \\ X_S' \\ X_A' \\ X'T \end{bmatrix} \\
&= I_{ast,ast} - \frac{1}{ast}\,(sX_S + 2X_SJ_{s,s} - X_AJ_{a,s} - X_TJ_{t,s})\,X_S' \\
&\quad - \frac{1}{ast}\,(-X_SJ_{s,a} + aX_A)\,X_A' \\
&\quad - \frac{1}{ast}\,(-X_SJ_{s,t} + tX_T)\,X_T' \\
&= I_{ast,ast} - \frac{1}{ast}\,(sX_SX_S' + aX_AX_A' + tX_TX_T' + 2X_sJ_{s,s}X_s' - X_AJ_{a,s}X_S' \\
&\quad - X_TJ_{t,s}X_S' - X_SJ_{s,a}X_A' - X_SJ_{s,t}X_T') \\
&= I_{ast,ast} - \frac{1}{ast}\,(sX_SX_S' + aX_AX_A' + tX_TX_T' - J_{ast,ast})
\end{aligned}
$$

The information matrix for product effects for model (4.5) is

$$\begin{aligned}
C_{4.5} &= X'_P P_1 X_P \\
&= X'_P X_P - \frac{1}{ast}\left(sN_S N'_S + aN_A N'_A + tN_T N'_T - X'_P J_{ast,ast} X_P\right) \\
&= arI_p - \frac{1}{at}N_S N'_S - \frac{1}{st}N_A N'_A - \frac{1}{as}N_T N'_T + \frac{ar}{p}J_{p,p} \\
&= arI_p - \frac{1}{at}N_S N'_S - \frac{r}{p}J_{p,a}J_{a,p} - \frac{1}{as}N_T N'_T + \frac{ar}{p}J_{p,p} \\
&= arI_p - \frac{1}{at}N_S N'_S - \frac{1}{as}N_T N'_T
\end{aligned}$$

B.5.3 Information matrices for resolvable cross-over designs

Model (4.8):

$$Y_{d(ijk)} = \mu + \pi_i + \alpha_j + \beta_k + \tau_{d(i,j,k)} + \rho_{d(i-1,j,k)} + \varepsilon_{ijk}$$

$$1 \le i \le p,\ 1 \le j \le n,\ 1 \le k \le s$$

Information matrices for direct product and first-order residual effects:

$$\begin{aligned}
X &= [1_{st}\ X_S\ X_A\ X_T\ X_P\ X_R] \\
X_1 &= [1_{st}\ X_S\ X_A\ X_T] \\
X_2 &= [X_P\ X_R] \\
P_{1,4.8} &= I_{ast,ast} - \frac{1}{ast}\left(sX_S X'_S + aX_A X'_A + tX_T X'_T - J_{ast,ast}\right) \\
C_{11,4.8} &= arI_p - \frac{1}{at}N_S N'_S - \frac{1}{as}N_T N'_T \\
C_{12,4.8} &= X'_P X_R - \frac{1}{at}N_S \tilde{N}'_S - \frac{1}{p}J_{p,a}\tilde{N}'_A - \frac{1}{as}N_T \tilde{N}'_T + \frac{r_p}{ap}1_p \boldsymbol{r}'_R \\
&= X'_P X_R - \frac{1}{at}N_S \tilde{N}'_S - \frac{r_p}{p}1_p \mathrm{r}'_R - \frac{1}{as}N_T \tilde{N}'_T + \frac{r_p}{ap}1_p \boldsymbol{r}'_R \\
C_{22,4.8} &= \boldsymbol{r}^{\delta}_R - \frac{1}{at}\tilde{N}_S \tilde{N}'_S - \frac{1}{st}\tilde{N}_A \tilde{N}'_A - \frac{1}{as}\tilde{N}_T \tilde{N}'_T + \frac{1}{ap}\boldsymbol{r}_R \boldsymbol{r}'_R \\
C_{D,4.8} &= C_{11,4.8} - C_{12,4.8}C^{-}_{22,4.8}C_{21,4.8} \\
C_{R,4.8} &= C_{22,4.8} - C_{21,4.8}C^{-}_{11,4.8}C_{12,4.8}
\end{aligned}$$

Bibliography

[1] Abel, R.J.R., Brouwer, A.E., Colbourn, C.J. and Dinitz, J.H. (1996) Mutually orthogonal Latin squares (MOLS). In: *The CRC Handbook of Combinatorial Designs* (eds.: Colbourn, C. and Dinitz, J.H.), CRC Press.

[2] Avery, P.J. and Masters, G.A. (1999) Advice on the design and analysis of sensory evaluation using a case-study with cooked pork. *The Statistician*, **48(3)**, 349–359.

[3] Bailey, R.A. (1977) Patterns of confounding in factorial design. *Biometrika*, **64**, 597–603.

[4] Ball, R.D. (1997) Incomplete block designs for the minimisation of order and carry-order effects in sensory analysis. *Food Quality and Preference*, **8**, 111–118.

[5] Bechhofer, R.E. and Tamhane, A.C. (1981) Incomplete block designs for comparing treatments with a control: General theory. *Technometrics*, **23**, 45–57; Corrigendum (1982) **24**, 171.

[6] Brockhoff, P.M. (1997) Statistical testing of individual differences in sensory profiling. *Invited paper for the 5th European Conference on Food Industry and Statistics, Versailles, France.*

[7] Brockhoff, P.M. and Skovgaard, I.M. (1994) Modeling individual differences between assessors in sensory evaluations. *Food Quality and Preferences*, **5**, 215–224.

[8] Chambers, E. and Wolf, M.B. (Eds.) (1996) *Sensory Testing Methods*, (2nd edn.). ASTM, West Conshohocken, PA.

[9] Cochran, W.G. and Cox, G.M. (1957) *Experimental Designs.* 2nd ed., Wiley, New York.

[10] Das, M.N. (1958) On reinforced incomplete block designs. *Journal of the Indian Society of Agricultural Statistics*, **10**, 73–77.

[11] Deppe, C., Carpenter, R. and Jones, B. (2001). Nested incomplete block designs in sensory testing: construction strategies. *Food Quality and Preference*, **12**, 281–290.

[12] *Design Express* (2001) Reference Manual, Version 1.0. QiStatistics, Ruscombe.

[13] Dijksterhuis, G.B. (1997) *Multivariate Data Analysis in Sensory and Consumer Science.* Food Nutrition Press, Inc., Trumbull, CT.

[14] Donev, A.N. (1997) Algorithm AS 313: An algorithm for the construction of crossover trials. *Applied Statistics*, **46**, 288–298.

[15] Durier, C., Monod, H. and Bruetschy, A. (1997) Design and analysis of factorial sensory experiments with carry-over effects. *Food Quality and Preference*, **8**, 141–149.

[16] Eccleston, J.A. and Hedayat, A.S. (1974) On the theory of connected designs: Characterization and optimality. *Annals of Statistics*, **2**, 1238–1255.

[17] Eccleston, J.A. and McGilchrist, C.A. (1985) Algebra of a row-column design. *Journal of Statistical Planning and Inference*, **12**, 305–310.

[18] Eccleston, J.A. and Whitaker, D. (1999) On the design of optimal changeover experiments through multi-objective simulated annealing. *Statistics and Computing*, **9**, 37–41.

[19] Erichsen, L. (1998) Generalised mixed models for sensory data — quantitative or ordinal scale? In: *Regression Methods in Sensometrics.* Ph.D. Thesis, Royal Veterinary and Agricultural University, Denmark.

[20] Fahrmeier, L. and Tutz, G. (1994) *Multivariate Statistical Modeling Based on Generalized Linear Models.* Springer, New York.

[21] Gay, C. and Mead, R. (1992) A statistical appraisal of the problem of sensory measurement. *Journal of Sensory Studies*, **7**, 205–228.

[22] Gupta, V.K., Ramana, D.V.V. and Parsad, R. (1999) Weighted A-efficiency of block designs for making treatment-control and treatment-treatment comparisons. *Journal of Statistical Planning and Inference*, **77**, 301–319.

[23] Harshbarger, B. (1949) Triple rectangular lattices. *Biometrics*, **5**, 1–13.

[24] Hedayat, A.S., Jacroux, M. and Majumdar, D. (1988) Optimal designs for comparing test treatments with controls (with discussions). *Statistical Science*, **3**, 462–491.

[25] Hedayat, A.S. and Majumdar, D. (1984) A-optimal incomplete block designs for control-test treatment comparisons. *Technometrics*, **26**, 363–370.

[26] Hinkelmann, K. and Kempthorne, O. (1994) *Design and Analysis of Experiments: Volume I, Introduction to Experimental Design.* Wiley, New York.

[27] Hunter, E.A. (1996) Experimental design. In: *Multivariate analysis of data in sensory sciences*, (eds.: Næs, T. and Risvik, E.), Elsvier, Amsterdam, 37–69.

[28] Jacroux, M. (1989) The A-optimality of block designs for comparing test treatments with a control. *Journal of the American Statistical Association*, **94**, 310–117.

[29] Jarrett, R.G. (1977) Bounds for the efficiency factor of block designs. *Biometrika*, **23**, 67–72.

[30] Jarrett, R.G. (1983) Definitions and properties for m-concurrence designs. *Journal of the Royal Statistical Society B*, **45**, 1–10.

[31] John, J.A. (1981) Efficient cyclic designs. *Journal of the Royal Statistical Society B*, **43**, 76–80.

[32] John, J.A. and Eccleston, J.A. (1986) Row-column α-designs. *Biometrika*, **73**, 301–306.

[33] John, J.A. and Street, D.J. (1992) Bounds for the efficiency of row-column designs. *Biometrika*, **79**, 658–661.

[34] John, J.A. and Whitaker, D. (1993) Construction of resolvable row-column designs using simulated annealing. *Australian Journal of Statistics*, **35**, 237–245.

[35] John, J.A., Whitaker, D. and Triggs, C.M. (1993) Construction of cyclic designs using integer programming. *Journal of Statistical Planning and Inference*, **36**, 357–366.

[36] John, J.A. and Williams, E.R. (1995) *Cyclic and Computer Generated Designs.* Chapman & Hall, London.

[37] John, J.A., Wolock, F.W. and David, H.A. (1972) *Cyclic Designs.* National Bureau of Standards, Applied Mathematics Series 62.

[38] Jones, B. (1976) An algorithm for deriving optimal block designs. *Technometrics*, **18(4)**, 451–458.

[39] Jones, B. (1980) Combining two component designs to form a row-and-column design. *Applied Statistics*, **29**, 334–337.

[40] Jones, B. and Deppe, C. (2001) Recent developments in the design of cross-over trials: A brief review and bibliography. In: *Recent Advances in Experimental Designs and Related Topics. Papers presented at the conference in honor of*

Professor Damaraju Raghavarao, (ed. Altan, S. and Singh, J.). Nova Science Publishers, Inc., Huntington, NY.

[41] Jones, B. and Eccleston, J.A. (1980) Exchange and interchange procedures to search for optimal designs. *Journal of the Royal Statistical Society B*, **42**, 238–243.

[42] Jones, B. and Kenward, M.G. (1990) *Design and Analysis of Cross-Over Trials.* Chapman and Hall, London.

[43] Jones, B. and Wang, J. (2000) The analysis of repeated measurements in sensory and consumer studies. *Food Quality and Preferences*, **11**, 35–41.

[44] Kempton, R.A., Ferris, S.J. and David, O. (2001) Optimal change-over designs when carry-over effects are proportional to direct effects of treatments. *Biometrika*, **88(2)**, 391–399.

[45] Kunert, J.(1998) Sensory Experiments as Crossover Studies. *Food Quality and Preference*, **9(4)**, 243–253.

[46] Lamacraft, R.R. and Hall, W.B. (1982) Tables of cyclic incomplete block designs: $r = k$. *Australian Journal of Statistics*, **24**, 350–360.

[47] Lee, Y. and Nelder, J.A. (1996) Hierarchical generalized linear models. *Journal of the Royal Statistical Society B*, **58(4)**, 619–656.

[48] Lundahl, D.S. and McDaniel, M.R. (1988) The panelist effect - fixed or random? *Journal of Sensory Studies*, **3**, 113–121.

[49] MacFie, H.J.H., Bratchell, N., Greenhoff, K. and Vallis, L.V. (1989) Designs to balance the effect of order of presentation and first-order carry-over effects in Hall tests. *Journal of Sensory Studies*, **4**, 129–148.

[50] Majumdar, D. (1996) Optimal and efficient treatment control designs. In: Gosh, S. and Rao, C.R. (Eds.) *Handbook of Statistics, Vol. 13.* North Holland, Amsterdam, 1007–1053.

[51] Majumdar, D. and Notz, W. (1983) Optimal incomplete block designs for comparing treatments with a control. *Annals of Statistics*, **11**, 258–266.

[52] Majumdar, D. and Tamhane, A.C. (1996) Row-column designs for comparing treatments with a control. *Journal of Statistical Planning and Inference*, **49**, 387–400.

[53] Martens, M. and Martens, H. (1986) In: Piggott, J.R. (Ed.) *Statistical Procedures in Food Research*. Elsevier, London.

[54] Mathon, R. and Rosa, A. (1996) $2-(v,k,\lambda)$ designs of small order. In: *The CRC Handbook of Combinatorial Designs*. Eds.: Colbourn, C. and Dinitz, J.H., CRC Press.

[55] McCullagh, P. and Nelder, J.A. (1989) *Generalized Linear Models*. 2nd ed., Chapman and Hall, London.

[56] Mead, R. and Gay, C. (1995) Sequential design of sensory trials. *Food Quality and Preferences*, **6**, 271–280.

[57] Muir, D.D. and Hunter, E.A. (1991/92) Sensory evaluation of cheddar cheese: Order of tasting and carryover effects. *Food Quality and Preference*, **3(2)** 141–145.

[58] Næs, T. (1990) Handling individual differences between assessors in sensory profiling. *Food Quality and Preference*, **2**, 187–199.

[59] Næs, T. and Solheim, R. (1991) Detection and interpretation of variation within and between assessors in sensory profiling. *Journal of Sensory Studies*, **6**, 159.

[60] Nguyen, N.-K. (1993) An algorithm for constructing optimal resolvable block designs. *Communications in Statistics B*, **22**, 911–923.

[61] Nguyen, N.-K. (1994) Construction of cyclic incomplete block designs by heuristic method. CSIRO-IAPP Biometrics Unit, Tech. Rep. 9402.

[62] Nguyen, N.-K. and Dey, A.(1990) Computer-aided construction of small (M,S)-optimal incomplete block designs. *Australian Journal of Statistics*, **32(3)**, 399–410.

[63] Nguyen, N.-K. and Williams, E.R. (1993) An algorithm for constructing optimal resolvable row-column designs. *Australian Journal of Statistics* **35**, 363–370.

[64] Notz, W. (1985) Optimal designs for treatment-control comparisons in the presence of two-way heterogeneity. *Journal of Statistical Planning and Inference*, **12**, 61–73.

[65] O'Mahony, M. (1998) Comments on Næs, Langsrud and Steinsholt. *Food Quality and Preferences*, **9(3)**, 165.

[66] O'Mahony, M. (2001) personal communication.

[67] Park, D.K. and Dean, A.M. (1990) Average efficiency factors and adjusted orthogonality in multidimensional designs. *Journal of the Royal Statistical Society B*, **52**, 361–368.

[68] Paterson, L.J. (1983) Circuits and efficiency in incomplete block design. *Biometrika*, **70**, 215–225.

[69] Patterson, H.D. and Williams, E.R. (1976) A new class of resolvable incomplete block designs. *Biometrika*, **63**, 83–92.

[70] Pearce, S.C. (1960) Supplemented balance. *Biometrika*, **47**, 263–271.

[71] Pigeon, J.G. and Raghavarao, D. (1987) Crossover designs for comparing treatments with a control. *Biometrika*, **74 (2)**, 321–328.

[72] Pritchett-Mangan, P.A. (1992) Performance assessment of sensory panelists. *Journal of Sensory Studies*, **7**, 229–252.

[73] Ragharavao, D. (1971) *Construction and Combinatorial Problems in Design of Experiments*. Wiley, New York.

[74] Raghavarao, D. and Blaisdell, E.A. (1985) Efficiency bounds for partially balanced change-over designs based on M-associate class PBIB designs. *Journal of the Royal Statistical Society B*, **47**, 132–135.

[75] Russel, K.G. (1991) The construction of good change-over designs when there are fewer units than treatments. *Biometrika* **78**, 305–313.

[76] Russel, K.G, Eccleston, J.A. and Knudsen, G.J. (1981) Algorithm for the construction of (M,S)-optimal block designs and row-column designs. *Journal of Statistical Computing and Simulation*, **12**, 93–105.

[77] Schlich, P. (1993) Uses of change-over designs and repeated measurements in sensory and consumer studies. *Food Quality and Preferences*, **4**, 223–235.

[78] Searle, S.R. (1971) *Linear Models*. Wiley, New York.

[79] Shah, B.V. (1959) On a generalisation of the Kronecker product designs. *Annals of Mathematical Statistics*, **30**, 48–54.

[80] Snedecor, G.W. and Cochran, W.G. (1980) *Statistical Methods*. Iowa State University Press, Ames, IA.

[81] Spurrier, J.D. and Edwards, D. (1986) An asymptotically optimal subclass of balanced treatment incomplete block designs for comparison with a control. *Biometrika*, **73**, 191–199.

[82] Steinsholt, K. (1998) Are assessors levels of a split-plot factor in the analysis of variance of sensory profile experiments? *Food Quality and Preference*, **9(3)**, 153–156.

[83] Stufken, J. (1987) A-optimal block designs for comparing test treatments with a control. *Annals of Statistics*, **15**, 1629–1638.

[84] Stufken, J. (1988) On bounds for the efficiency of block designs for comparing test treatments with a control. *Journal of Statistical Planning and Inference*, **19**, 361–372.

[85] Tjur, T. (1990) A new upper bound for the efficiency factor of a block design. *Australian Journal of Statistics*, **32**, 231–237.

[86] Türe, T.E. (1994) Optimal row-column designs for multiple comparisons with a control: A complete catalog. *Technometrics* **36**, 292–299.

[87] Wakeling, I.N. and Buck, D. (2001) Balanced incomplete block designs useful for consumer experimentation. *Food Quality and Preference*, **12(4)**, 265–268.

[88] Wakeling, I.N., Hasted, A. and Buck, D. (2001) Cyclic presentation order designs for consumer research. *Food Quality and Preference*, **12(1)**, 39–46.

[89] Wakeling, I.N. and Macfie, H.J.H. (1995) Designing consumer trials for first orders of carry-over effects when only a subset of k samples from p may be tested. *Food Quality and Preference*, **6**, 299–308.

[90] Whitaker, D. (1995) A nested simulated annealing algorithm. *Journal of Statistical Computing and Simulation*, **53**, 233–241.

[91] Whitaker, D. and Triggs, C.M. (1990) Construction of block designs using mathematical programming. *Journal of the Royal Statistical Society B*, **52 (3)**, 497–503.

[92] Wilkinson, C. and Yuksel, D. (1997) Modeling differences between panelists in use of measurement scales. *Journal of Sensory Studies*, **12**, 55–68.

[93] Williams, E.J. (1949) Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Scientific Research*, **2**, 149–168.

[94] Williams, E.R. and Talbot, M. (1993) *ALPHA+: Experimental Designs for Variety Trials*. Design User Manual. CSIRO, Canberra and SASS, Edinburgh.

[95] Yates, F. (1936) A new method of arranging variety trials involving a large number of varieties. *Journal of Agricultural Science*, **26**, 424–455.

ELSEVIER

Food Quality and Preference 12 (2001) 281–290

Food Quality
and Preference

www.elsevier.com/locate/foodqual

Nested incomplete block designs in sensory testing: construction strategies

Carola Deppe[a,*], Roland Carpenter[b], Byron Jones[a]

[a]*Department of Medical Statistics, De Montfort University, The Gateway, Leicester LE1 9BH, UK*
[b]*Unilever Research Laboratory, Colworth House, Sharnbrook, Bedford MK44 1LQ, UK*

Received 15 September 2000; received in revised form 15 December 2000; accepted 22 December 2000

Abstract

In most cases the choice of sensory designs is driven solely by the limitation of assessor fatigue. However, often there is an additional but unrelated kitchen constraint, which limits the number of products that can be prepared for a given session. Nested incomplete block designs provide the opportunity to take proper account of both of these constraints, and in so doing, identify opportunities for more effective product comparisons. Here we describe a new three-stage method for constructing such designs and identify criteria for assessing their quality. We illustrate an extension of the method for situations where there is a factorial product structure and some contrasts are judged more important than others. © 2001 Elsevier Science Ltd. All rights reserved.

Keywords: Nested incomplete block design; Three-stage procedure; Sensory design

1. Introduction

One of the key aims of sensory analysis is to objectively assess and compare product properties based upon responses from a panel of trained assessors.

The reliability and relevance of the inference will depend crucially on the quality of the underlying design. In the sensory context, the design process takes on additional significance in attempting to cope with the inherent limitations of human response data. These limitations are especially evident when a large number of products are to be compared in a given study, as is frequently the case.

It is therefore appropriate to search for new systematic approaches for generating designs that are better able to address these additional design objectives, whilst at the same time fully recognising the practical operational constraints. For this reason, we focus on nested incomplete block designs.

The special requirements for sensory designs are discussed in Section 2. The suggested three-stage procedure to construct such designs is introduced in Section 3, explained in detail in Section 4 and further expanded to include factorial designs in Section 5. A summary of results is given in Section 6.

2. Practical constraints and objectives in sensory design

The number of products (p) for comparison in a sensory experiment is usually fixed by the client, and will therefore be called the *client constraint*.

If the number of products is too large to be presented on a single occasion without sensory fatigue, only a subset of products (t) will be tasted by an assessor in one session. Products have to be presented over multiple sessions (s). This will be called the *assessor constraint*.

In addition, a *kitchen constraint* arises if it is only possible to prepare a subset of products (k) for a given session. This is a common situation when assessing food products, especially those requiring carefully controlled cooking or other preparation prior to assessment. The constraint may depend upon the available kitchen facilities — e.g. the number of hobs or cooking utensils, or upon the problems of achieving tight control of temperatures, volumes and particularly of timings, when too many products are being processed simultaneously.

Sensory assessors cannot be trained to perform in an entirely homogeneous and consistent manner. A further

[*] Corresponding author. Tel.: +44-116-255-1551, ext. 8510; fax: +44-116-250-6114.
E-mail address: cd@dmu.ac.uk (C. Deppe).

0950-3293/01/$ - see front matter © 2001 Elsevier Science Ltd. All rights reserved.
PII: S0950-3293(01)00013-1

consideration is thus the number of replications (r) that is assumed to be necessary for a valid experiment. To accommodate the differences among assessors' scores, assessors are used as blocks, with the requirement that every assessor tastes every product at least twice, and with equal frequency. This results in *equal replication* of all products, namely each product is tasted $a \times r$ times, where a is the number of assessors in a panel.

Often all pairwise product comparisons are of equal importance, as might be the case when there is no underlying product structure. For efficient designs for models with independent error terms, products have to be arranged in blocks in such a way that they occur the same number of times alongside all other products within sessions (*concurrence*).

Balance for serving position within a session will additionally provide efficient estimates when order effects are expected and *balance for precedence* will cater for possible carryover effects between one product and its successor, this is particularly associated with strong and persistent aroma, flavour, or astringency.

Complete balance for concurrence, precedence and serving position is rarely possible due to the given constraints, so a strategy to reach near balance will be derived. These latter requirements will determine the sequence of products that will be assigned to each assessor.

Designs suggested for sensory experiments are mostly constructed from Williams Latin squares (Williams, 1949), which are balanced for order and carryover effects. Balance for first order carryover effects requires that each product is preceded equally frequently by every other product and balance for order requires that each product occurs equally frequently in each period of the design. The rows of the Latin square are assigned randomly to assessors, the columns specify the test sequence. The labels within the cells of the Williams square are assigned randomly to the names of the products in the test.

If the number of products (p) is larger than the number of assessors in the panel (a), only a subset of the rows of the square will be used. If $p < a$, rows from a second square will be used. If incomplete Williams squares are used, balance for order and carryover effects is lost. Further information on such designs in the sensory context can be found in Hunter (1996) and Wakeling and MacFie (1995), for example. A more general review is given by Jones and Deppe (2000).

The linear model assumed here for a sensory trial includes additive assessor, session, serving-order, product and first-order carryover effects. All observations are assumed to be independent.

$$y_{ijk} = \text{assessor}_i + \text{session}_j + \text{serving position}_k \text{ (within session}_j\text{)} + \text{product}_{d[ijk]} + \text{carryover}_{d[ij(k-1)]} + \varepsilon_{ijk}$$

where $\text{product}_{d[ijk]}$ is the product served to assessor i in session j in serving position k. $\text{Carryover}_{d[ij1]}$ is defined as 0 for products in the first serving position of every session, as any carryover effects are assumed to occur, if at all, only within a session. The general ANOVA table for this model has the following form:

Source of variation	*Degrees of freedom*
Between subjects:	
Assessors	a-1
Within subjects:	
Session	s-1
Serving position within a session	t-1
Product	p-1
Carryover within a session	p-1
Residual	$a \times s \times t\text{-}a\text{-}s\text{-}t\text{-}2p+4$

Often the interest of analysis is on all pairwise comparisons between products. The three-step procedure ensures that the comparisons are all estimable, when the kitchen and the panel design are connected designs.

3. A useful design strategy

The construction process for a nested incomplete block design will be illustrated using a typical example: a 16 product design (client constraint $p=16$) with $k=8$ products available at every session (kitchen constraint) and $t=4$ products to be tasted by an assessor within a session (assessor constraint). This particular example has been chosen as it can also be used to demonstrate designs with factorial structure.

As mentioned before, a block in the sensory design refers to the number of products an assessor is given within the whole trial, with the constraint that each assessor will taste every product r times. This assignment can be viewed as r complete block designs. Each block is split into s sessions, such that an assessor within a session can be viewed as a sub-block. Ignoring assessors, these sub-blocks form an incomplete block design. The two design structures therefore form a nested incomplete block design. The kitchen constraint adds extra difficulties to the construction, as only a subset of all products can be available for a session. The construction process is therefore split into three stages:

1. A suitable kitchen/preparation design is devised (Table 1). This defines which products will be available in a session. It is an incomplete block design with an additional and design-specific constraint on its columns, which will be explained in Section 4.1. A cyclic construction method for the design will be demonstrated in the example.

2. A suitable panel design is built (Table 2). The panel design is constructed as a cross-over design that allocates columns of the kitchen design to serving positions for the assessors. The design tries to balance for concurrence, serving position, and precedence. Again a cyclic design will be used.
3. The kitchen and the panel design are combined in an optimum way (Tables 3 and 4). In cases where it is necessary to compromise the structure of the panel design, e.g. because there are insufficient assessors to achieve perfect balance, the exact sequence of columns in the given kitchen design can have a profound influence.

In Section 4, this three-stage approach is applied to the construction of a sensory design in which all product contrasts are equally important.

In Section 5 more complex sensory design situations are identified in which some selected product contrasts may be more important than others. The three-stage approach is then adapted to cope with one such case, namely 16 products in a full 2×2×2×2 factorial design.

Table 1
Explanation of the three step procedure — step 1: the kitchen design

	Products available in each session							
Session 1	1	10	3	11	2	9	5	13
Session 2	2	11	4	12	3	10	6	14
Session 3	3	12	5	13	4	11	7	15
Session 4	4	13	6	14	5	12	8	16
Session 5	5	14	7	15	6	13	1	9
Session 6	6	15	8	16	7	14	2	10
Session 7	7	16	1	9	8	15	3	11
Session 8	8	9	2	10	1	16	4	12
Column	1	2	3	4	5	6	7	8

Table 2
Explanation of the three step procedure — step 2: the panel design

Assessor	Serving position			
	1	2	3	4
1	3	6	8	7
2	4	7	1	8
3	5	8	2	1
etc...				

Table 3
Explanation of the three step procedure — step 3: combining the kitchen and the panel design assignments for assessor 1

	Serving 1	Serving 2	Serving 3	Serving 4
Session1	3	9	13	5
Session 2	4	10	14	6
Session 3	5	11	15	7
Session 4	6	12	16	8
Session 5	7	13	9	1
Session 6	8	14	10	2
Session 7	1	15	11	3
Session 8	2	16	12	4

4. Building a nested incomplete block design

4.1. Stage 1 — the kitchen design

The first task is to construct an efficient incomplete block design for 16 products in eight blocks of size 8, where each block of the kitchen design represents the products available for the corresponding session. The number of blocks in the kitchen design is the number of sessions (s); the size of the blocks represents the number of products that can be prepared for one session (k). The number of sessions is $s=p\times r/t$, determined by the number of products (p), the number of replications (r), and the number of servings within a session (t); for this example $p=16$, $s=8$, $t=4$ and $r=2$. The construction method only works when s is an integer.

The method for constructing a kitchen design is based upon cyclic incomplete block designs, John and Williams (1995). A cyclic design is fully determined by its initial block. All other blocks are generated from the initial block, adding one to each element and reducing modulo p.

The 16 products are randomly assigned the numbers 1–16. As there are 16 products but only eight sessions, the products are divided into two separate sets, products 1–8 form the first set and products 9–16 the second. Half of the columns of the kitchen design will contain all numbers from the first set, the other half of the columns the numbers from the second set. Two columns picked randomly from each set will therefore always contain all 16 products. This ensures that assessors will taste all products the same number of times.

Each set is the basis of a different cyclic incomplete block design, and the two block designs are combined to build the kitchen design (Table 5). In general p products are separated into $p/s=t/r$ sets, where both fractions have to be integers. Columns are then arranged alternating from each set, such that p/s consecutive columns contain all p products. In this example $16/8=2$ consecutive columns include the 16 products.

With careful choice of the initial block it is possible to achieve an optimum concurrence matrix for a cyclic design (Table 6). The elements of the concurrence matrix (λ) denote how often each product occurs in the same session with every other product. The design will be constructed such that a product will never occur twice in the same block, resulting in a diagonal of 0s. If all products occur the same number of times with all other products, the design is called balanced for product effects. If balance is not possible, a concurrence matrix with all entries as equal as possible will be called optimal. As the concurrence matrix is symmetric it is sufficient to refer to the upper diagonal only. From this, a summary of the concurrence matrix can be constructed which gives the number of times each different entry of the concurrence matrix is repeated in the matrix (Table 7).

As explained before, the kitchen design is created from two different cyclic designs (A and B in Table 5). The first cyclic design (A) refers to products 1–8, the second (B) to the products 9–16. For the first cyclic design the initial row is reduced modulo 8. The second design is constructed equivalently to the first, with the exception that 8 is added to each entry so that the entries in the second design refer to the products 9–16.

To find an optimal initial block for the first part of the kitchen design (A in Table 5), the concurrence matrix and its summary are calculated for all possible initial blocks for this part. From all initial rows with optimal concurrence (all elements as equal as possible, eight products occur once in a block with another product and 20 products occur twice, Table 7) the initial row chosen for this case is 1 2 3 5.

The initial row for the second design (9 10 12 13) is also chosen from the list of optimal initial rows, but with 8 added to each entry. The row is chosen so that a combination of the two initial rows results in an optimal concurrence matrix for the combined kitchen design.

The first and the fourth quadrants of the concurrence matrix in Table 6 are determined through each of the single designs, with the second quadrant determined by their combination. For this example it is not possible to have a concurrence matrix that does not contain a 3. The resulting concurrence matrix is as balanced as possible under the given constraints, i.e. all of the elements are as equal as possible. All comparisons are represented, and none occurs more than three times.

With these two initial rows the kitchen design is almost fully determined. Only the order of the columns has to be chosen, which will not affect the efficiency of the kitchen design but that of the complete sensory design. This will be referred to in Section 4.3.

Efficiency for a comparison of two products is defined as the ratio between the minimal variance of this comparison and the variance under the given design. The minimum variance for the example is given as the variance of a comparison from an orthogonal design, in this case a complete block design, where each product is replicated $k \times s/p$ times. To calculate an average efficiency, the average over all possible pairwise comparisons is calculated. The average efficiency from the resulting incomplete block design is 92.7%.

4.2. Stage 2 — the panel design

There are three design issues to be optimised within the panel design.

1. for optimum overall concurrence, each of the entries, representing columns 1 to 8, must occur with equal frequency in the panel design;

Table 4
Explanation of the three step procedure — the complete sensory design

	Session 1: serving poistion				Session 2: serving poistion				...	Session 8: serving poistion			
	1	2	3	4	1	2	3	4	...	1	2	3	4
Assessor 1	3	9	13	5	4	10	14	6		2	16	12	4
Assessor 2	11	5	1	13	12	6	2	14		10	4	8	12
Assessor 3	2	13	10	1	3	14	11	2		1	12	9	8
etc...	.	.	.	.	.	.	.	.	...	.	.	.	.
Available products	1, 2, 3, 5,9,10,12,13				2,3,4,6,10,11,12,14				...	1,2,4,8,9,11,12,16			

Table 5
Construction principle of the kitchen design[a]

Session	First set — design A				Second set — design B			
1[b]	1	2	3	5	9	10	12	13
2	2	3	4	6	10	11	13	14
3	3	4	5	7	11	12	14	15
4	4	5	6	8	12	13	15	16
5	5	6	7	1	13	14	16	9
6	6	7	8	2	14	15	9	10
7	7	8	1	3	15	16	10	11
8	8	1	2	4	16	9	11	12
	Generated modulo 8				Generated modulo 8 + 8			

[a] For the final kitchen design colums are arranged alternatley from both sets and permuted in step 3. The final kitchen design is shown in Table 1.

[b] Intial rows for A and B.

Table 6
Concurrence matrix of the kitchen design

	Design A — first set								Design B — second set							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Design A																
1		2	2	1	2	1	2	2	3	2	2	2	2	1	1	3
2			2	2	1	2	1	2	3	3	2	2	2	2	1	1
3				2	2	1	2	1	1	3	3	2	2	2	2	1
4					2	2	1	2	1	1	3	3	2	2	2	2
5						2	2	1	2	1	1	3	3	2	2	2
6							2	2	2	2	1	1	3	3	2	2
7								2	2	2	2	1	1	3	3	2
8									2	2	2	2	1	1	3	3
Design B																
9										2	1	2	2	2	1	2
10											2	1	2	2	2	1
11												2	1	2	2	2
12													2	1	2	2
13														2	1	2
14															2	1
15																2
16																

2. if tasting order is to be balanced overall, then the panel design itself must be balanced with respect to serving position; and
3. if the carryover effect is to be balanced overall, then the panel design matrix must itself be balanced with respect to carryover.

It is rarely possible to completely satisfy all three of these objectives.

The assignment of kitchen design columns to assessor serving positions is achieved using a crossover design to achieve balance for serving position, product and carryover effects. The panel design consists of 16 rows, one for each assessor, and four columns, one for each serving (Table 8).

One possibility for creating the panel design is to use a similar cyclic structure as for the kitchen design, but checking for precedence as well as concurrence. The panel design for this example requires two cyclic designs to assign columns of the kitchen design to 16 assessors, each cycle providing eight rows.

The entries in the panel design (numbers 1–8) specify which of the columns from the kitchen design are to be used by each assessor at that serving. So, assessor 1 will be allocated products from columns 3, 6, 8 and 7 of the kitchen design in that order (Table 3).

To ensure that every assessor tastes every product twice, the initial row of the panel design must consist of two even and two odd entries, representing columns of the kitchen design. The odd columns of the kitchen design contain only products 1–8, and the even columns contain only products 9–16.

Again, cyclic designs from all possible initial rows are constructed and the concurrence matrix and its summary calculated. For a carryover design it is not sufficient

Table 7
Summary for the cyclic kitchen design

	λ		
	1	2	3
Design A	8	20	0
Design B	8	20	0
Kitchen design	32	72	16

Table 8
The panel design

Assessor	Serving position			
	1	2	3	4
1	3	6	8	7
2	4	7	1	8
3	5	8	2	1
4	6	1	3	2
5	7	2	4	3
6	8	3	5	4
7	1	4	6	5
8	2	5	7	6
9	1	2	6	3
10	2	3	7	4
11	3	4	8	5
12	4	5	1	6
13	5	6	2	7
14	6	7	3	8
15	7	8	4	1
16	8	1	5	2

to optimize only concurrence, as precedence should also be optimized. The precedence matrix (Tables 10–13) refers to successive pairs of products and indicates how often one entry in the design is followed by another entry. The precedence matrix is, in general, not symmetric. To check for balance, the summary matrix of the precedence matrix is sufficient. For the first cycle an initial row is chosen that is as balanced as possible for concurrence and precedence (Tables 9 and 10), for this example this is 3 6 8 7.

One cyclic design supplies assignments for eight assessors; therefore a second cyclic design is necessary to provide additional and different assignments for assessors 9–16. (Refer to Section 4.4 for cases with less/more than 16 assessors.)

There is no other initial row for the second cycle containing two odd and two even numbers, which gives a better-balanced concurrence matrix for the complete design than the one chosen (Tables 11 and 12, summaries in Table 13). All alternatives also include zero concurrences. The initial row for the second cyclic design is therefore chosen to include no zero concurrences, and to create a precedence matrix with zeros and ones only.

In the panel design each of the values 1–8 is compared alongside each other value on either two or four occasions (Table 9). It is nearly balanced for carry-over with each of the 56 possible carryover sequences represented on one occasion or not at all (Table 12). The average efficiency of all pairwise product contrasts of the panel design is 76.0%, calculated in comparison to an 8×8 Latin Square.

Table 9
Concurrence summary for the first cycle of the panel design, the second cycle and the complete panel design

	λ			
	1	2	3	4
First cycle	8	20	0	0
Second cycle	8	20	0	0
Complete panel design	0	8	0	20

Table 10
Precedence matrix for the first cycle of the panel design

	1	2	3	4	5	6	7	8
1		0	1	1	0	0	0	1
2	1		0	1	1	0	0	0
3	0	1		0	1	1	0	0
4	0	0	1		0	1	1	0
5	0	0	0	1		0	1	1
6	1	0	0	0	1		0	1
7	1	1	0	0	0	1		0
8	0	1	1	0	0	0	1	

Table 11
Precedence matrix for the second cycle of the panel design

	1	2	3	4	5	6	7	8
1		1	0	0	1	1	0	0
2	0		1	0	0	1	1	0
3	0	0		1	0	0	1	1
4	1	0	0		1	0	0	1
5	1	1	0	0		1	0	0
6	0	1	1	0	0		1	0
7	0	0	1	1	0	0		1
8	1	0	0	1	1	0	0	

4.3. Stage 3 — *the optimum combination*

Having fixed the kitchen design and the panel design, there remains one further opportunity for optimisation. This opportunity arises because the properties of the kitchen design are not changed if the columns are placed in a different order. In fact, with eight columns there are 8! = 40320 possible column sequences. However, only those with alternating 'high' and 'low' columns are considered, leaving 4!×4! = 576 possibilities.

So the final step in design optimisation is to work through each of the sequences of the kitchen design in turn, combining it with the panel design, and then to investigate the properties of each resulting design by calculating overall concurrence and precedence matrices and efficiencies. The efficiency of the complete sensory design is calculated by comparing it to a randomised complete block design with a product replication of 32 for each product, i.e. two 16×16 Latin Squares.

Table 14 summarises the concurrence and precedence matrices and average efficiencies from all possible resulting designs. All 576 different design matrices fall into one of six sets.

Table 12
Precedence matrix for the complete panel design

	1	2	3	4	5	6	7	8
1		1	1	1	1	1	0	1
2	1		1	1	1	1	1	0
3	0	1		1	1	1	1	1
4	1	0	1		1	1	1	1
5	1	1	0	1		1	1	1
6	1	1	1	0	1		1	1
7	1	1	1	1	0	1		1
8	1	1	1	1	1	0	1	

Table 13
Precedence summaries

	λ	
	0	1
First cycle	32	24
Second cycle	32	24
Complete panel design	8	48

In cases where it is necessary to compromise the structure of the panel design the exact sequence of columns in the given kitchen design can have a more profound influence than in this example, where the efficiencies are all quite close.

A column sequence for the kitchen design is chosen that results in an efficiency of 80.5%. This final kitchen design is shown in Table 1.

The resulting complete sensory design is presented in Table 15. Each of the 16 rows in the sensory design shows the serving sequence for an assessor, split into eight sessions. The design efficiency is 90.6%. Sixteen direct product comparisons are never made within a session and eight product comparisons on the other hand, will be made 16 times (see last line Table 14). All other product comparisons are made between two and eight times. Thiry-two of the possible carry-over sequences do not occur, while 16 carry-over sequences will occur four times. All other carryover sequences either occur once or twice.

Cyclic designs are only one possible type of design to use for the kitchen and panel designs. Existing algorithms for efficient incomplete block and crossover designs can only be used to construct designs where the number of products equals the number of sessions. For all other cases these algorithms have to be adjusted to incorporate the extra constraints resulting from dividing the products into separate sets.

4.4. Compromising the panel design to deal with additional or fewer assessors

Taking the cyclic panel design in Table 8, some consideration will be given on how to adjust the design to the exact number of assessors taking part in the experiment. Balance for serving position will be lost, and the changes in the concurrence and precedence matrix are discussed.

If two more assessors are available, the two rows 7, 8, 1, 2 and 3, 4, 5, 6 are a possibility for the panel design. These rows use 6 of the unused carry-over pairings (i.e. consecutive treatments), leaving only pairs (2,3) and (6,7) unused. Concerning concurrence, these additions result in six column pairs occurring three times within a block, 20 column pairs four times and two column pairs five times.

The problem of restricting the number of assessors is evidently likely to result in a sub-optimal panel design and it has to be decided which part of the 16-assessor design should be used.

For 15 assessors it does not matter which row is taken out. Reducing the design for 14 assessors, two rows have to be taken out. To preserve equal occurrence of each column

Table 14
Summary measures for the resulting sensory designs from different permutations

Concurrence						Precedence				Efficiency
0	2	4	6	8	16	0	1	2	4	
16	0	56	16	40	8	32	96	112	16	90.3
16	8	44	16	44	8	40	80	120	16	90.4
16	8	40	24	40	8	32	96	112	16	90.5
16	16	32	16	48	8	48	64	128	16	90.4
16	16	28	24	44	8	40	80	120	16	90.5
16	16	24	32	40	8	32	96	112	16	90.6

Table 15
Final sensory design

Assessor	Session 1				Session 2				Session 3				Session 4				Session 5				Session 6				Session 7				Session 8			
1	3	9	13	5	4	10	14	6	5	11	15	7	6	12	16	8	7	13	9	1	8	14	10	2	1	15	11	3	2	16	12	4
2	11	5	1	13	12	6	2	14	13	7	3	15	14	8	4	16	15	1	5	9	16	2	6	10	9	3	7	11	10	4	8	12
3	2	13	10	1	3	14	11	2	4	15	12	3	5	16	13	4	6	9	14	5	7	10	15	6	8	11	16	7	1	12	9	8
4	9	1	3	10	10	2	4	11	11	3	5	12	12	4	6	13	13	5	7	14	14	6	8	15	15	7	1	16	16	8	2	9
5	5	10	11	3	6	11	12	4	7	12	13	5	8	13	14	6	1	14	15	7	2	15	16	8	3	16	9	1	4	9	10	2
6	13	3	2	11	14	4	3	12	15	5	4	13	16	6	5	14	9	7	6	15	10	8	7	16	11	1	8	9	12	2	1	10
7	1	11	9	2	2	12	10	3	3	13	11	4	4	14	12	5	5	15	13	6	6	16	14	7	7	9	15	8	8	10	16	1
8	10	2	5	9	11	3	6	10	12	4	7	11	13	5	8	12	14	6	1	13	15	7	2	14	16	8	3	15	9	1	4	16
9	1	10	9	3	2	11	10	4	3	12	11	5	4	13	12	6	5	14	13	7	6	15	14	8	7	16	15	1	8	9	16	2
10	10	3	5	11	11	4	6	12	12	5	7	13	13	6	8	14	14	7	1	15	15	8	2	16	16	1	3	9	9	2	4	10
11	3	11	13	2	4	12	14	3	5	13	15	4	6	14	16	5	7	15	9	6	8	16	10	7	1	9	11	8	2	10	12	1
12	11	2	1	9	12	3	2	10	13	4	3	11	14	5	4	12	15	6	5	13	16	7	6	14	9	8	7	15	10	1	8	16
13	2	9	10	5	3	10	11	6	4	11	12	7	5	12	13	8	6	13	14	1	7	14	15	2	8	15	16	3	1	16	9	4
14	9	5	3	13	10	6	4	14	11	7	5	15	12	8	6	16	13	1	7	9	14	2	8	10	15	3	1	11	16	4	2	12
15	5	13	11	1	6	14	12	2	7	15	13	3	8	16	14	4	1	9	15	5	2	10	16	6	3	11	9	7	4	12	10	8
16	13	1	2	10	14	2	3	11	15	3	4	12	16	4	5	13	9	5	6	14	10	6	7	15	11	7	8	16	12	8	1	9

from the kitchen design, each entry from 1 to 8 should be taken out. However, there are no two rows that include together all eight entries, so two rows will be chosen that include only one common entry. A possibility would be to take out rows 10 and 16, where 2 occurs in both rows and 6 is missing (**2**, 3, 7, 4 and 8, 1, 5, **2**).

Possible options are:

1. no substitution; or
2. substitute a remaining 6 for a 2 for example in row 14 or 12

In both cases this would mean introducing a carry-over sequence that has already occurred (7 following 2 for row 14, or 2 following 1 in row 12). All other substitutions would mean that serving position has to be further compromised by having 2 three times in a column. When a substitution is made, the structure of two odd and two even entries has to be preserved.

The summaries of the concurrence and precedence matrices in Table 16 show little difference between the three options, but there are other designs where adding or taking out can create large differences in the concurrence and precedence values. The design with no substitution and the design with a substitution in row 14 both result in an efficiency of 89.8%. One key consequence of this is that designs should be optimised to take accurate account of the number of assessors who are expected to participate.

5. Emphasising particular product contrasts

Within the set of products presented for sensory analysis it often happens that there are particular product contrasts which are of special relevance, so all contrasts may no longer be of equal importance and fully balanced designs may no longer be the best choice. This may arise when individual products in the study have particular experimental or commercial relevance, e.g. an experimental reference or control product, or the market or brand leader.

In other cases there may be specific structural relationships within the product set such that some or all of the products constitute a full or fractional factorial design. In these cases, there is usually more importance attached to the estimates of main effects and the lower order interactions.

Just one example from this potentially vast range of sensory designs has been selected as an illustration — a design again involving 16 products, but this time arranged as a 2×2×2×2 factorial.

Two different strategies are compared:

1. ignore the factorial structure: assign the products randomly to a design constructed as described above; and
2. confound the three and four-factor interactions with the rows of the kitchen design.

5.1. *Ignoring the factorial structure*

For this method the kitchen design in Table 1 is used. For a design with 16 products there are 16! ways to assign product levels to the design. As there are too many to compare them all, a randomly chosen subset of 10 000 assignments is used for comparison.

The efficiencies of the 10 000 different incomplete block designs that result from the different product assignments to the kitchen design vary over all contrasts between 75 and 100%. Main effect contrasts A, B, D and two-factor interaction contrasts BD, and BCD have minimum efficiencies of 80.0%, while all the other factorial contrasts have a minimum efficiency of 75% (Table 17, columns 1 and 2).

As a next step, efficiencies for the complete sensory design are calculated. Again, maxima and minima are tabulated for every contrast (Table 17, columns 3 and 4). The same pattern as for the kitchen design is evident, but with reduced efficiencies compared to the kitchen design. The maximum efficiency is 98.7%, and the minimum efficiency is either 80 or 74.4%. So, the worst case than can happen for a design that has been optimised to have all pairwise product contrasts as equal as possible, is that a factorial product contrast of interest is estimated with 74.4% efficiency.

An idea might be to search for the design from the 10 000 that gives the highest efficiency for: main effects (M); two-factor interactions (I); and main effects and two-factor interactions (B).

The efficiencies for these designs are tabulated in Table 17, columns 5–7. For these three designs all contrasts of interest have an efficiency higher than 89%, and those of lesser interest higher than 80%.

Table 16
The summary for the concurrence and the precedence matrix and efficiencies for the complete sensory designs resulting from the three alternative panel designs for 14 assessors

Design	Concurrence										Precedence					Efficiency
	0	1	2	3	4	5	6	7	8	14	0	1	2	3	4	
No substitution	16	8	8	16	8	24	8	32	8	8	40	120	80	8	8	89.8
Substitution in row 12	16	8	8	16	8	24	16	16	16	8	40	120	80	8	8	89.7
Substitution in row 14	16	0	20	8	12	24	24	8	16	8	48	104	88	8	8	89.8

Whereas previously assessors were randomly assigned to sequences and products randomly assigned to labels, for this strategy only the first randomisation can take place, because this design is optimised for the best product-label allocation.

5.2. *Confounding three and four-factor interactions*

If interest is mostly in main effects and two-factor interactions, another possibility is to confound higher order interactions (ABCD, ABD, ABC, ACD in the example) with blocks (sessions) in the kitchen design. The first step is to write out the confounding scheme for the blocks (Table 18) and translate it into product numbers.

As a second step, a computer algorithm for creating row-column designs will be used to create a kitchen design with the correct column constraint (Jones, 1980).

The efficiencies for the kitchen design are, as expected, 100% for the unconfounded effects and 75% for the partially confounded effects (Table 17, column 8). Using different column permutations for the kitchen design results in very similar efficiencies.

This strategy of confounding higher order interactions results in higher efficiencies for the contrast of interest by reducing the efficiencies of the higher order interactions compared to the previous design strategy (Table 7, column 9).

6. Conclusions and implications

A strategy for the construction of nested incomplete block designs where only a subset of products is available in a session has been developed and evaluated. The construction strategy is applicable in all areas of sensory research where monadic assessment is undertaken.

Even though our procedure uses cyclic designs, the whole process of constructing and assessing a design is still very time consuming. While algorithms to check the design qualities have been developed, the process itself needs to be automated and optimised and alternative

Table 17
Efficiencies for the factorial contrasts

Contrast	Ignoring the factorial structure							Confounding interactions	
	Kitchen design		Complete design						
	Maximum	Minimum	Maximum	Minimum	*M*	*I*	*B*	Kitchen design	Complete design
A	100	80.8	98.7	80.0	95.2	94.7	92.7	100	99.0
B	100	80.8	98.7	80.0	96.4	88.0	94.4	100	96.1
C	100	75.0	98.7	74.4	94.5	89.5	89.9	100	96.2
D	100	80.8	98.7	80.0	94.4	85.6	91.5	100	96.1
AB	100	75.0	98.7	74.4	91.7	95.0	91.2	100	97.5
AC	100	75.0	98.7	74.4	87.4	94.5	91.0	100	98.5
AD	100	75.0	98.7	74.4	93.0	94.7	94.7	100	97.3
BC	100	75.0	98.7	74.4	88.2	94.5	94.3	100	97.2
BD	100	80.8	98.7	80.0	88.6	93.1	94.2	100	99.5
CD	100	75.0	98.7	74.4	85.3	94.6	91.5	100	97.2
ABC	100	75.0	98.7	74.4	87.4	86.1	80.7	75	73.7
ABD	100	75.0	98.7	74.4	88.8	86.4	88.7	75	73.1
ACD	100	75.0	98.7	74.4	87.5	89.6	94.4	75	74.0
BCD	100	80.8	98.7	80.0	94.5	86.1	83.7	100	96.1
ABCD	100	75.0	98.7	74.4	87.5	88.2	88.3	75	73.4

Table 18
Confounding scheme for blocks

Session	Products available in each session								Confounded
1	1	ab	ac	ad	bc	bd	cd	abcd	ABCD
2	a	b	c	d	abc	abd	acd	bcd	
3	a	b	d	ac	bc	cd	abd	abcd	ABD
4	1	c	ab	ad	bd	abc	acd	bcd	
5	a	b	c	ad	bd	cd	abc	abcd	ABC
6	1	d	ab	ac	bc	abd	acd	bcd	
7	a	c	d	ab	bc	bd	acd	abcd	ACD
8	1	b	ac	ad	cd	abc	abd	bcd	

construction methods for the kitchen as well as for panel designs have to be evaluated.

In cases where individual product comparisons are *not* equally important, the second approach offers an opportunity to concentrate on those product differences of main interest, and estimate these with higher precision than would be the case with the general procedure, where the special structure is ignored. How far this procedure can be generalised to include control products that are outside the factorial structure has still to be investigated.

Acknowledgements

We are grateful to Dr. Ian Wakeling for early discussions, which helped to formalise the interdependence of kitchen and panel design.

References

Hunter, E. A. (1996). Experimental design. In T. Naes, & E. Risvik, *Multivariate analysis of data in sensory science* (pp. 37–69). Amsterdam: Elsevier Applied Sciences.

John, J. A., & Williams, E. R. (1995). *Cyclic and computer generated Designs* (2nd ed.). London: Chapman and Hall.

Jones, B. (1980). Combining two component designs to form a row-and-column design. *Applied Statistics, 29*, 334–337.

Jones, B., & Deppe, C. (2000). Recent developments in the design of cross-over trials: a brief review and bibliography. *Proceedings of a conference on Recent Advances in Design of Experiments and Related Topics*. Edited by J. Singh. Published by Nova Science Publishers, Huntington, NY: (in press).

Wakeling, I. N., & MacFie, H. J. (1995). Designing consumer trials for first and higher orders of carryover effects when only a subset of k samples from p may be tested. *Food Quality and Preference, 6*, 299–308.

Williams, E. R. (1949). Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Science Research, A2*, 149–168.