DESIGN OF EXPERIMENTS and ROBUST DESIGN

Problems in design and production environments often require experiments to find a solution. Design of experiments are a collection of statistical methods that, properly used, maximize the probability of finding the best solution at the lowest cost.

The fundamental idea is to vary a problem related factor and study changes in process or product performance. To increase the information content of the experiment, two or more factors are varied at the same time. Ten factors are not uncommon, forty has been used successfully. The concerted factor variations are specified by a matrix with carefully selected characteristics. Two or more matrixes are often used in a series of iterative steps to find the best solution. Design of experiments is used to identify important factors, to optimize physical and virtual systems and to create robust products and processes.

Many industrial businesses do not use the powerful resource of statistically designed experiments, in most cases due to insufficient knowledge of the methods potential.

Experience shows that systematic utilization of designed experiments often produces large dividends at relatively modest costs.

Theoretical foundation

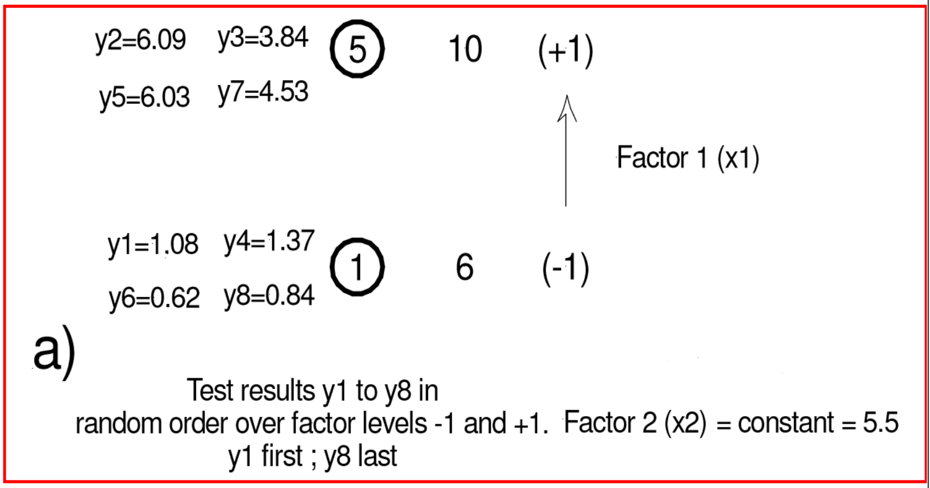
The theoretical foundation of designed experiments is mathematical statistics. The theoretical part of designed experiments includes a combination of a wide range of standard mathematical and statistical methods; and in some cases application-specific statistical tools. More rudimentary applications of design experiments do not require an extensive knowledge in theoretical statistics. It does, however, requires a good knowledge of basic statistical ideas and on what assumptions applied methods are derived from.

We will limit this presentation to the most common factorial designs.

Factorials

Design of experiments are best illustrated by example. Let us assume we have problem with the performance of a product or process we are developing. This could be high emissions from a boiler in a power plant, low structural yield in a laser weld or large variations in film thickness in a semiconductor production plant.

Let us further assume that we have found two factors that probably can solve the problem and that we will use an experiment to find out if and how. The intuitive way is to use a "one factor at a time approach". The more efficient way is to use a small factorial design.



We start with a rudimentary experimental design investigating "one factor at a time". Figure a) shows an experiment where a first factor (x_1) is investigated at two levels. The experiment is replicated for sufficient precision. Thus, the first factor requires a total of eight test runs, four at the 6 unit level and four at the 10 unit level. We start with adjusting x_1 to 6 units and read the performance value y_1 = 1.08. The y index indicates the time order. *This has been randomized to avoid bias due to noise factor correlation.* We then repeat the test procedure adjusting x_1 to 10 units and read y_2 = 6.09. The second factor x_2 has been set at a constant value of 5.5 during the investigation of x_1.

The levels are usually coded to create efficient statistical models. If we in this case use (x_1- 8)/2, the coding of the low level 6 is -1 and the coding of the high level 10 is +1.

The next step is to estimate a function between the factor settings and the product or process performance. This is usually done with appropriate least square modeling software. Based on the presented data set, the least square model would be

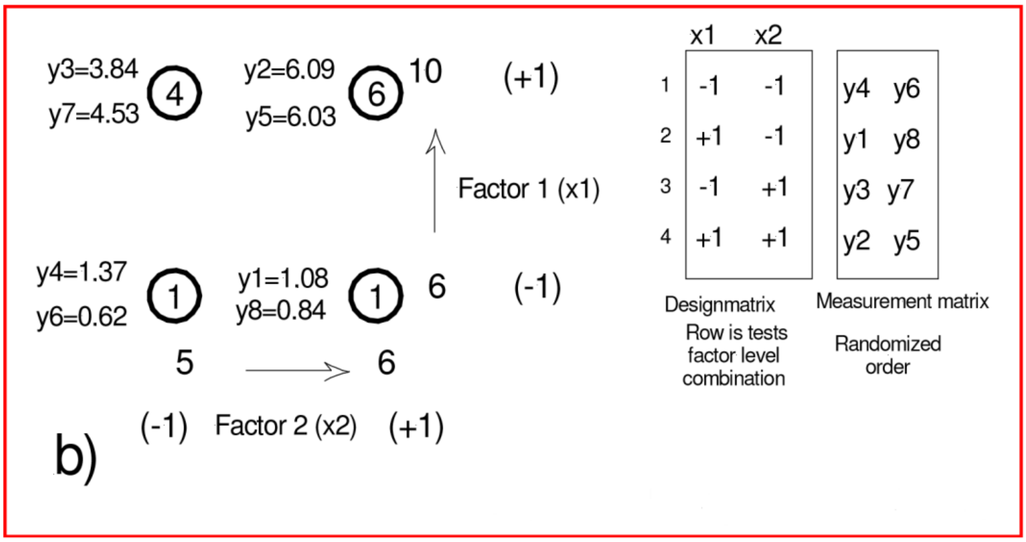
$$\hat{y} = 3 + 2x_1$$

where $\hat{y}$ is the least square estimate of the product performance and x_1 is factor adjustment in coded scale -1 to +1. If the coded value -1 is inserted into the model, the estimated response is 1 and if the coded value +1 is inserted, the estimated response is 5. For a more comprensive analysis see the least square analysis detailes in attachment A.

We notice that this result did cost us 8 test runs with x_2 set at a constant level. To model x_2 the procedure will be identical with x_1 held at a constant level. This will cost us 8 more test runs, thus a total of 16 runs.

If we assume that our project budget is limited to 8 test runs and we want to estimate both factors, we can distribute the test in a factorial as illustrated in figure *b*). The figure shows four rings arranged in a square, with vertical setting direction for our first factor x_1, and horizontal setting direction for our second factor x_2. We do not hold any factor constant in this case. Instead we vary all according to a predetermined pattern. A standard two level 2^2 design.

Both factors have been coded as low level -1 and high level +1. Test points are marked with circles and correspond to the code combinations lower left point {-1,-1}, lower right point {-1, +1}, upper left point {-1,+1}, upper right point {+1, +1}. The 8 test runs have been distributed evenly over the four points i.e. two test runs per factor level setting. The test runs are *randomized* in time order as indicated by the *y* index – 1 is the first, 2 the second etc. The design can alternatively be written on a matrix form as illustrated in figure b). Row 1 corresponds to the lower left point, row 2 lower right point row 3 the upper left point, and row 4 the upper right point. Figure b) shows two matrixes – the left is the factorial design and the right is the test run measurements matrix.



The least squares analysis best estimate model is in this case:

$$\hat{y} = 3 + 2x_1 + 0.5x_2 + 0.5x_1x_2$$

where x_1 and x_2 are in coded units. We note that in addition to the coefficients for each factor we get an additional coefficient measuring the factor interactions. Let us explain the model step by step.

The model is telling us that one unit increase in x_1 on average double the performance y. And one unit increase in x_2 on average increase the performance y with 50 %. The interaction term 0.5 x_1x_2 is a measurement of how much the setting of x_1 influence the effect of x_2 or vice versa. Thus, x_2 has no influence on performance y if x_1 is set at -1, as the x_2 coefficients candel, but has a positive effect of 1 if x_1 is set at +1. A study of the mean values inscribed in the four circles in figure *b*) verify this. For least square analysis details: see attachment B. (Note: The coded unit matrix is orthogonal implying that the mean of x_1 given x_2 is low do not change when x_2 is changed to a high value and vice versa; thus, we have *independent* coefficient estimates. Independent implies *uncorrelated* estimates. This is manifested in the correlation matrix at the bottom of attachment B. Orthogonality is a critical characterstic when selecting an experimental design).

A comparison of a fixed budget, eight test run experiment, shows that the “one factor at a time” alternative a) deliver one factor coefficient while b) deliver three coefficients. In addition, the a) result is only valid at the fixed value of the other factor x_2 while the b) result is, due to the interaction estimate, valid for all {x_1, x_2} coordinates of the design area.

We have briefly shown some advantages of choosing factorials instead of traditional “one factor at a time” designs. Generally, factorials give *better test economy*, *estimates of interactions* and more *generally valid results* with *improved repeatability*. When used in an optimizing scenario *they reduce the risk of sub optimizing*.

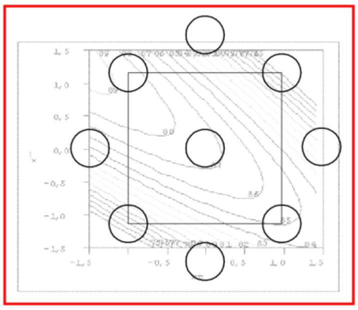
The number of factors can be theoretically arbitrarily chosen. However, the number of test runs in a two level full factorial design is determined by 2^k where k is the number of factors. This will quickly lead to a very large number of runs, in most cases not manageable within a normal project budget. Fractional factorials have been developed to limit experiment sizes when investigating a large number of factors. These will be discussed later.

Fractional factorials

As previously mentioned, fractional factorials have been developed to limit experiment test run sizes of full factorials when investigating more than a few number of factors. The idea behind these are that, by carefully selecting a small set of the runs in a full factorial, you get a design that generate most of the information required to solve the problem. Information is here least square coefficients. And if you do not, you usually can add a small set carefully selected additional test runs to solve your problem. Most factorials performed in industry are fractional factorials. Fractional factorials are often used as a screening tool when active factors and interactions are unknown. Their screening capability may be explained by their resolution characteristics in combination with the Pareto principle (in a large set of candidate factors, only a small set may be of importance). The fractional factorial is illustrated in *Attachment C* by a commonly used 8 factor 16 run design.

Response surface analysis

Designs of the type shown in figure b) is used to estimate the model $y = b_0 + b_1x_1 + b_2x_2 + b_{12}x_1x_2$. Often you want an estimate of the full second order model $y = b_0 + b_1x_1 + b_{11} x_1^2 + b_2x_2 + b_{22}x_2^2 + b_{12}x_1x_2$. This requires an additional set of test runs if we start with the design in figure *b*) or a completely new design (outside this presentation). Figure *c*) shows a design where *b*) has been complemented with a center point and two points along the vertical axis and two points along the horisontal axis. This additional set of test runs is called “start points”.



c)

This design that is usually executed in two steps. You start with a square with a replicated (repeated) centerpoint. If an analysis shows there is a deviation between the mean value of the center point and the mean value of the square points, you have curvature, and proceed with the starpoints. The design is illustrated in *Attachment D.* This procedure is of course scalable to three or more product parameters.
Experiments based on this design may be used to find extreme values of different types. Sometimes the investigated area is far from the extreme point of interest. Special designs for the most efficient move can then be used. When the new area of interest has been found a new response surface design (figure *c*)) can be used to get a more detailed surface estimate at the new location.

Response surfaces with more than two factors are difficult to visualize. It is therefore common practice to use eigenvalue analysis for surface identification and reduction of dimension. Eigenvalue analysis can also be used to search for mechanisms and probable mechanistic models.

Robust design

As designers we are not capable of controlling all the parameters and variation sources influencing our products functions. A number of factors are active – from variation in components from subcontractors, via variation in production processes to the product working environment and variation in customer usage. The goal of robust design is to find the combination of product parameter setting that is immune to this type of noise spectra. Experiments are often the most efficient way of finding these settings.

			x3: -1	x3: +1	
	x1	x2	-1	+1	
1	-1	-1	y2	y7	ym1; s1
2	+1	-1	y4	y8	ym2; s2
3	-1	+1	y5	y3	ym3; s3
4	+1	+1	y6	y1	ym4; s4

{x1,x2} = control factor

{x3} = noise factor

d)

Figure *d*) illustrates a robust design experimental design. We start with dividing the factors in control factors and noise factors. The control factors are chosen among the factors the designer can control (x_1 and x_2 in figure *d*). We then search for other factors with high probability to interfere with the products intended functions. These factors are designated noise factors (x_3 in figure *d*).
The matrix can then be divided into two parts: the inner control matrix and the outer noise matrix. This is illustrated in figure *d*). The experiment is then performed run by run until the measurement matrix is filled with data. Mean y_m and variance s^2 for each control factor level combination is then estimated. This is followed by estimation of control factor models $y = f(x)$ of location (mean) and dispersion (variance) respectively, and finally a search for best control factor setting. This is an optimization problem in that the location deviation from the

product specification is balanced against the dispersion value. Dispersion is usually harder to control than location. A two stage procedure, starting with minimising the dispersion (stabilize the process) followed by adjusting the mean value to the product specification, is often used.

Both static functions, such as a ball bearing fixation in a wheel, as dynamic functions, such as the wheel break moment at a road surface as a function of different pedal pressures, can be given a robust design with this method.

About factors (x) and designs

The designed experiment factors (*x*) may have very different characteristics. In our two factor example the factors where continuous unrestricted and varied at two levels. Depending on scenario the factors may be proportions / mixtures (sum of factor levels is 1), have discrete levels (e.g. different catalysts), have more than two levels etc. A matrix may include many different types of factors. Factor types and the goal of the experiment shape the final design matrix specification. Usually matrices are of standard type – as in figure *b*) – or generated through computer algorithms.

About system response (y)

Also system responses such as product or product performance can have many different characteristics. They may be continuous or categorical, be defined as positive numbers, be proportions or frequencies. In robust design you usually work with location and dispersion responses. In reliability experiments you usually work with time or number of cycles to failure where test runs often are intentionally interrupted before all objects have failed (censuring). It is very common to use more than one response, often of different types, in the same experiment.

About Evaluation and estimation methods

Analysis of experiments based on factorials is a *knowledge building process*, where data such as technical information from product /process, computer plots and estimates are synthesized. Estimation method is chosen based on factor and response characteristics. Often a least square (regression) or in more advanced cases, a Maximum Likelihood method is chosen.

XR TECH services

XR TECH has supplied training, software and project support services since 1987. For more information, please contact Jan Åkerström at XR TECH - 0709 – 210 686 / jan.akerstrom@xrtech.com.

Attachment A. Least square analysis details (JMP 12)

Factor 1	Factor2	x1	x2	y	yi
6	5,5	-1	0	1,08	y1
6	5,5	-1	0	0,84	y8
6	5,5	-1	0	1,37	y4
6	5,5	-1	0	0,62	y6
10	5,5	1	0	6,03	y5
10	5,5	1	0	6,09	y2
10	5,5	1	0	3,84	y3
10	5,5	1	0	4,53	y7

Response y

Summary of Fit

RSquare	0,894174
RSquare Adj	0,876537
Root Mean Square Error	0,823281
Mean of Response	3,05
Observations (or Sum Wgts)	8

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	1	34,362050	34,3621	50,6971
Error	6	4,066750	0,6778	**Prob > F**
C. Total	7	38,428800		0,0004*

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	3,05	0,291074	10,48	<,0001*
x1	2,0725	0,291074	7,12	0,0004*

Attachment B. Least square analysis details (JMP 12)

Response y

Effect Summary

Source	LogWorth	PValue
x1	4,029	0,00009
x1*x2	1,659	0,02192
x2	1,607	0,02469 ^

Summary of Fit

RSquare	0,98569
RSquare Adj	0,974958
Root Mean Square Error	0,370776
Mean of Response	3,05
Observations (or Sum Wgts)	8

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	3	37,878900	12,6263	91,8443
Error	4	0,549900	0,1375	**Prob > F**
C. Total	7	38,428800		0,0004*

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	3,05	0,131089	23,27	<,0001*
x1	2,0725	0,131089	15,81	<,0001*
x2	0,46	0,131089	3,51	0,0247*
x1*x2	0,4775	0,131089	3,64	0,0219*

Correlation of Estimates

Corr

	Intercept	x1	x2	x1*x2
Intercept	1,0000	0,0000	0,0000	0,0000
x1	0,0000	1,0000	0,0000	0,0000
x2	0,0000	0,0000	1,0000	0,0000
x1*x2	0,0000	0,0000	0,0000	1,0000

Attachment C A standard fractional factorial design for 8 factors in 16 runs

Design

Run	X1	X2	X3	X4	X5	X6	X7	X8
1	-1	1	-1	-1	-1	1	-1	-1
2	1	-1	-1	1	1	-1	1	-1
3	-1	1	-1	1	1	1	-1	1
4	1	-1	-1	-1	1	1	-1	1
5	-1	1	1	-1	1	1	1	-1
6	1	-1	-1	1	-1	1	-1	-1
7	-1	-1	1	1	-1	1	1	1
8	1	-1	1	-1	-1	1	1	-1
9	-1	-1	1	1	-1	-1	-1	1
10	1	1	-1	-1	-1	-1	1	1
11	-1	-1	1	-1	1	-1	-1	-1
12	1	1	1	-1	-1	-1	-1	1
13	-1	-1	-1	-1	1	-1	1	1
14	1	1	1	1	1	1	1	1
15	-1	1	-1	1	-1	-1	1	-1
16	1	1	1	1	1	-1	-1	-1

Attachment D A central composite design with square and center points in a first block (iteration) and, if data shows curvature, star points in a second block. The design is randomazed within blocks.

Run Order: Randomize

Pattern	Block	X1	X2	Y
00	1	0	0	.
−−	1	-1	-1	.
+−	1	1	-1	.
00	1	0	0	.
−+	1	-1	1	.
00	1	0	0	.
++	1	1	1	.
a0	2	-1,414213562	0	.
00	2	0	0	.
00	2	0	0	.
0a	2	0	-1,414213562	.
0A	2	0	1,4142135624	.
A0	2	1,4142135624	0	.
00	2	0	0	.