CHAPTER II

THEORY AND LITERATURE REVIEW

This chapter considers the sun's position and Genetic Algorithm (GA). Finally, the design of the experiment and the statistical analysis of the data including factorial designs, the 2^k factorial design, 2^k fractional factorial Designs, 3^k factorial design, fractional replication of 3^k factorial design and analysis of variance (ANOVA) is described.

1. The Sun's Position

In order to understand how to collect energy from the sun, one must first be able to predict the location of the sun relative to the collection device. In this part describes the necessary equations by use unique vector approach. This approach will be used in this work to develop the equations for the sun's position relative to a tracking solar collector. (William & Michael, 2001)

1.1 The Hour Angle (ω)

To describe the earth's rotation about its polar axis, the concept of the hour angle (ω) is used. As shown in Figure 1, the hour angle is the angular distance between the meridian of the observer and the meridian whose plane contains the sun (-180, 180 degrees). The hour angle is zero at solar noon (when the sun reaches its highest point in the sky). At this time the sun is said to be 'due south' (or due north', in the Southern Hemisphere) since the meridian plan of the observer contains the sun. The hour angle increases by 15 degrees every hour.

Solar time is based on the 24-hour clock, with 12:00 as the time that the sun is exactly due south. The concept of solar time is used in predicting the direction of sunrays relative to appoint on the earth. Solar time is location (longitude) dependent and is generally different from local clock time, which is defined by politically define time zones and other approximations. An expression to calculate the hour angle from solar time is

$$\omega = 15\,(t_s - 12) \qquad \text{(degrees)} \qquad (1)$$

where t_s is solar time in hours

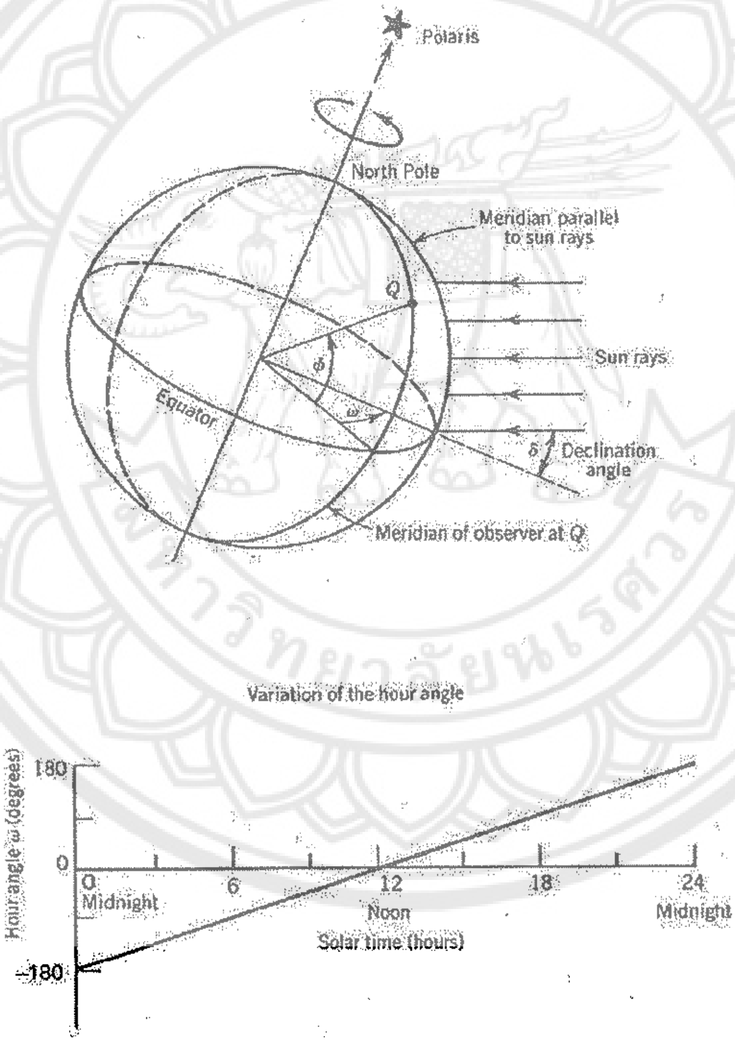

Figure 1 The hour angle (ω) (William & Michael, 2001)

1.2 Equation of Time (EOT)

The difference between mean solar time and solar time on a given date is shown in Figure 2. This difference is called equation of time (*EOT*). Since solar time is based on the sun being due south at 12:00 noon on any specific day, the accumulated difference between mean solar time and true solar time can approach 17 minutes either ahead of or behind the mean, with an annual cycle.

The level of accuracy required in determining the equation of time will depend on whether the designer is doing system performance or developing tracking equations. An approximation for calculating the equation of time in minutes is given by Woolf (1968) and is accurate to within about 30 seconds during daylight hours.

$$EOT = 0.258 \cos x - 7.416 \sin x - 3.648 \cos 2x - 9.228 \sin 2x \quad \text{(minutes)} \qquad (2)$$

where the angle x is defined as a function of the day number N

$$x = \frac{360(N-1)}{365.242} \quad \text{(degrees)} \qquad (3)$$

where the day number N is the number of days since January 1. Table 1 has been prepared as an aid in rapid determination of values of N from calendar dates.

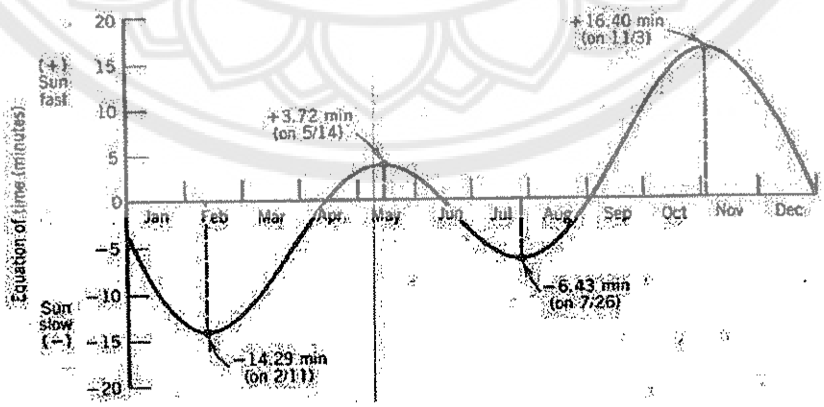

Figure 2 The Equation of Time (*EOT*) (William & Michael, 2001)

Table 1 Date-to-Day Number Conversion (William & Michael, 2001)

Month	Day Number, N	Notes
January	d	
February	$d + 31$	
March	$d + 59$	Add 1 if leap year
April	$d + 90$	Add 1 if leap year
May	$d + 120$	Add 1 if leap year
June	$d + 151$	Add 1 if leap year
July	$d + 181$	Add 1 if leap year
August	$d + 212$	Add 1 if leap year
September	$d + 243$	Add 1 if leap year
October	$d + 273$	Add 1 if leap year
November	$d + 304$	Add 1 if leap year
December	$d + 334$	Add 1 if leap year
Days of Special Solar Interest		
Solar Event	Date	Day Number, N
Vernal equinox	March 21	80
Summer solstice	June 21	172
Autumnal equinox	September 23	266
Winter solstice	December 21	355

Note :

1. d is the day of the month
2. Leap years are 2000, 2004, 2008 etc.
3. Solstice and equinox dates may vary by a day or two. Also, add 1 to the solstice and equinox day number for leap years.

1.3 Time Conversion

The conversion between solar time and clock time requires knowledge of the location, the day of the year, and the local standards to which local clocks are set. Conversion between solar time, t_s and local clock time *(LCT)* (in 24-hour rather than AM/PM format) takes the form

$$LCT = t_s - \frac{EOT}{60} + LC \qquad \text{(hours)} \tag{4}$$

where *EOT* is the equation of time in minutes and *LC* is a longitude correction defined as follows:

$$LC = \frac{(\text{local longitude}) - \begin{pmatrix}\text{longitude of standard}\\ \text{time zone meridian}\end{pmatrix}}{15} \qquad \text{(hours)} \tag{5}$$

1.4 The Declination Angle (δ)

The plane that includes the earth 's equator is called the *equatorial plane*. If a line is drawn between the center of the earth and the sun, the angle between this line and the earth's equatorial plane is called the *declination angle* (δ), as depicted in Figure 3.

Accurate knowledge of the declination angle is important in navigation and astronomy. One such approximation for the declination angle is

$$\sin \delta = 0.39795\cos[0.98563(N\text{-}173)] \tag{6}$$

where the argument of the cosine here is in degrees and *N* is the day number defined for Equation (3)

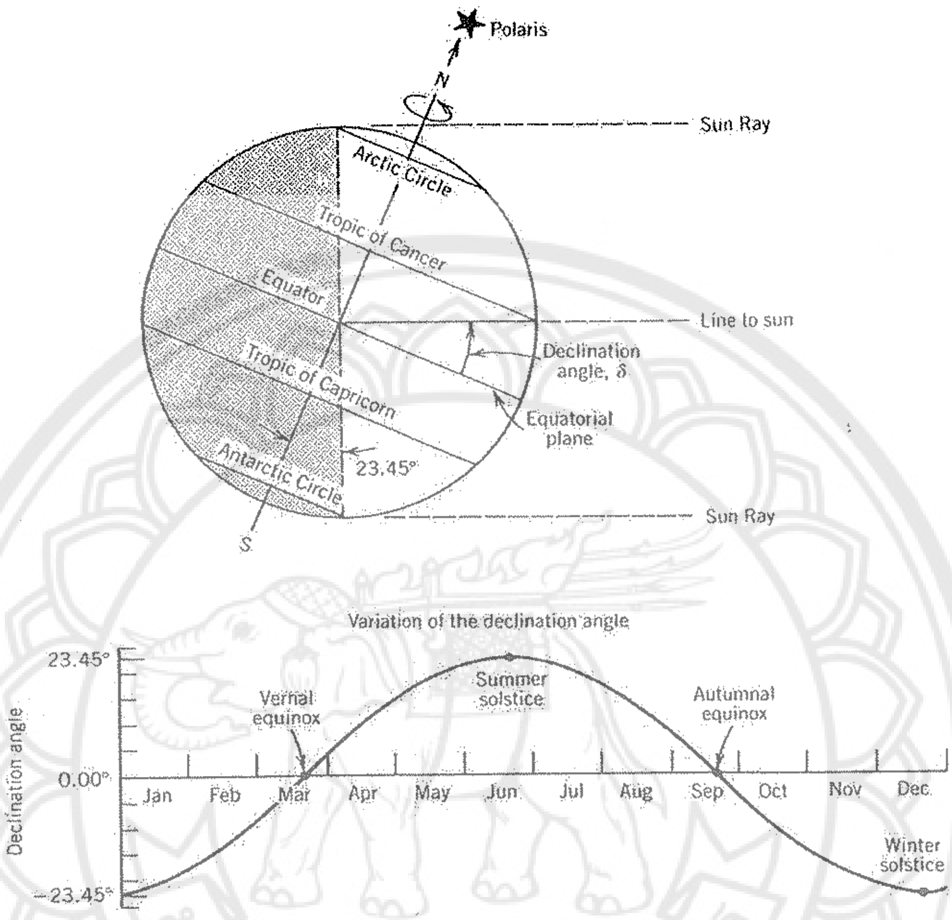

Figure 3 The Declination Angle (δ). The Earth is Shown in the Summer Solstice Position When δ = +23.45 Degrees. (William & Michael, 2001)

1.5 Latitude Angle (ϕ)

The latitude(ϕ) is the angle between a line drawn from a point on the earth's surface to the center of the earth, and the earth's equatorial plane. The intersection of the equatorial plane with the surface of the earth forms the equator and is designated as 0 degrees latitude. The earth's axis of rotation intersects the earth's surface at 90 degrees latitude (North Pole) and -90 degrees latitude (South Pole). Any location on the surface of the earth then can be defined by the intersection of a longitude angle and a latitude angle.

1.6 Solar Altitude (α), Zenith (θ), and Azimuth Angles (A)

The solar altitude angle (α) is defined as the angle between the central ray from the sun, and a horizontal plane containing the observer, as shown in figure 4. As an alternative, the sun's altitude may be described in terms of the solar zenith angle (θ) which is simply the complement of the solar altitude angle or

$$\theta_z = 90^\circ - \alpha \qquad \text{(degrees)} \qquad (7)$$

The other angle defining the position of the sun is the solar azimuth angle (A). It is the angle, measured clockwise on the horizontal plane, from the north-pointing coordinate axis to the projection of the sun's central ray.

There are other conventions for the solar azimuth angle in use in the solar literature. One of the more common conventions is to measure the azimuth angle from the south-pointing coordinate rather than from the north-pointing coordinate. Another is to consider the counterclockwise direction positive rather than clockwise. The information in Table 2 will be an aid in recognizing these differences when necessary.

It is of the greatest importance in solar energy systems design, to be able to calculate the solar altitude and azimuth angles at any location on the earth. This can be done using the three angles: latitude (ϕ), hour angle (ω), and declination (δ).

$$\alpha = \sin^{-1}(\sin\delta \sin\phi + \cos\delta \cos\omega \cos\phi) \qquad \text{(degrees)} \qquad (8)$$

$$A' = \cos^{-1}\left(\frac{\sin\delta\cos\phi - \cos\delta\cos\omega\sin\phi}{\cos\alpha}\right) \qquad \text{(degrees)} \qquad (9)$$

where if : $\sin\omega > 0$ then $A = 360^\circ - A'$

otherwise: $\sin\omega \le \left(\frac{\tan\delta}{\tan\phi}\right)$ and $A = A'$

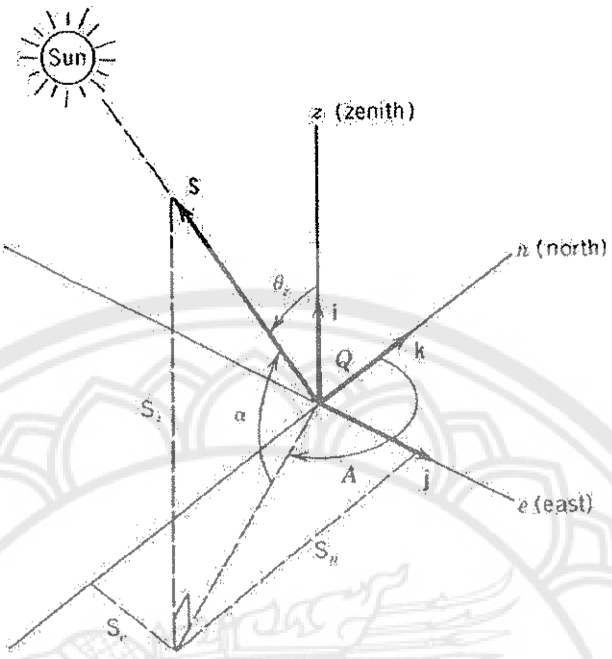

Figure 4 Earth Surface Coordinate System for Observer at Q.
(William & Michael, 2001)

Table 2 Sign Convention for Important Angles (William & Michael, 2001)

Title	Symbol	Zero	Positive Direction	Range
Earth-Sun Angles				
Latitude	ϕ	Equator	Northern hemisphere	$\pm 90^\circ$
Declination	δ	Equinox	Summer	$\pm 23.45^\circ$
Hour Angle	ω	Noon	afternoon	$\pm 180^\circ$
Observer-Sun Angles				
Sun Altitude	α	Horizontal	upward	0 to 90°
Sun Zenith	θ	Vertical	Toward horizon	0 to 90°
Sun Azimuth	A	Due north	clockwise	0 to 360°

2. Genetic Algorithm (GA)

Many optimization problems from the industrial engineering world, in particular the manufacturing systems, are very complex in nature and quite hard to solve by conventional optimization techniques. Since the 1960s, there has been an increasing interest in imitating living beings to solve such kinds of hard optimization problems. Simulating the natural evolutionary process of human beings results in stochastic optimization techniques called evolutionary algorithms, which can often outperform conventional optimization methods when applied to difficult real-world problems. Genetic algorithms are perhaps the most widely known type of evolutionary algorithms today.

Recently, genetic algorithms have received considerable attention regarding their potential as an optimization technique for complex problems and have been successfully applied in the area of industrial engineering. The well-known applications include scheduling and sequencing, reliability design, vehicle routing and scheduling, group technology, facility layout and location, transportation, and many others. (Gen and Cheng, 1997)

2.1 General Structure of Genetic Algorithms

The usual form of genetic algorithm was described by Goldberg. Genetic algorithms are stochastic search techniques based on the mechanism of natural selection and natural genetics. Genetic algorithms, differing from conventional search techniques, start with an initial set of random solutions called population. Each individual in the population is called a chromosome, representing a solution to the problem at hand. A chromosome is a string of symbols; it is usually, but not necessarily, a binary bit string. The chromosomes evolve through successive iterations, called generations. During each generation, the chromosomes are evaluated, using some measures of fitness. To create the next generation, new chromosomes, called offspring, are formed by either (a) merging two chromosomes from current generation using a crossover operator or (b) modifying a chromosome using a mutation operator. A new generation is formed by (a) selecting, according to the fitness valued, some of the parents and offspring and (b) rejecting others so as to keep the population size constant. Fitter chromosomes have higher probabilities of being selected. After several generations, the algorithms converge to the best chromosome, which hopefully represents the optimum or suboptimal solution to the problem. Let *P*(t) and *C*(t) be parents and offspring in current generation t; the general structure of genetic algorithms (see Figure 5) is described as follows:

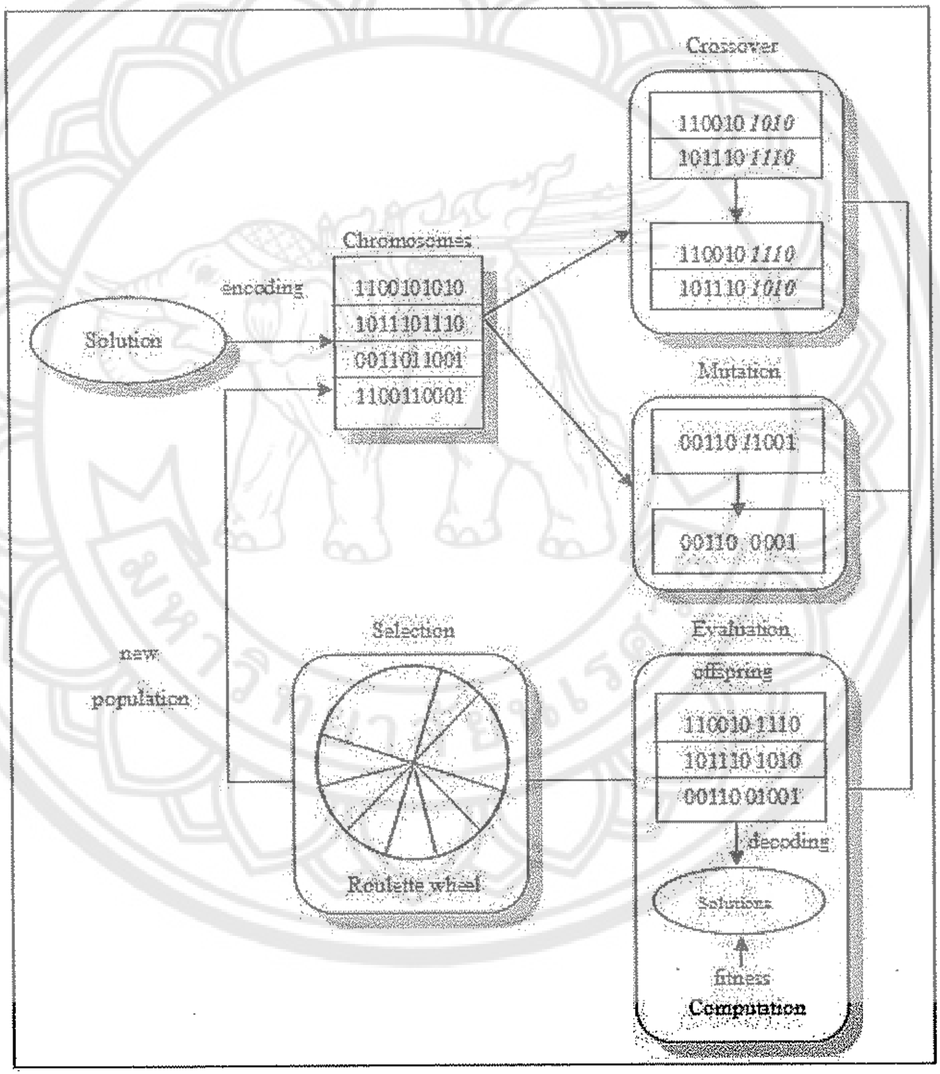

Figure 5 The Simple Structure of Genetic Algorithm (Gen and Cheng, 1997)

Procedure: Genetic Algorithms

Begin

$t \leftarrow 0$;

initialize *P(t)*;

Evaluate *P(t)*;

While (not termination condition) do

recombine *P(t)* to yield *C(t)*;

evaluate *C(t)*;

select *P(t+1)* from *P(t)* and *C(t)*;

$t \leftarrow t + 1$;

end

end

2.2 Operations in Genetic Algorithms

It is a modified version of Grefenstette and Baker's description. Usually, initialization is assumed to be random. Recombination typically involves crossover and mutation to yield offspring. In fact, there are only two kinds of operations in genetic algorithms:

2.2.1 Genetic operations: crossover and mutation

2.2.2 Evolution operation: selection

The genetic operations mimic the process of heredity of genes to create new offspring at each generation. The evolution operation mimics the process of Darwinian evolution to create populations from generation to generation. This description differs from the paradigm given by Holland, where selection is made to obtain parents for recombination.

Crossover is the main genetic operator. It operates on two chromosomes at a time and generates offspring by combining both chromosomes' features. A simple way to achieve crossover would be to choose a random cut-point and generate the offspring by combining the segment of one parent to the left of the cut-point with the segment of the other parent to the right of the cut-point.

This method works well the bit string representation. The performance of genetic algorithms depends, to a great extent, on the performance of the crossover operator used.

Chromosome 1	1	1	0	1	1	0	0	1	→	Offspring 1	1	1	0	1	1	1	1	0
Chromosome 2	1	0	0	1	1	1	1	0	→	Offspring 2	1	0	0	1	1	0	0	1

Figure 6 Crossover Operation for Chromosomes with Binary String.

The crossover rate (denoted by *Pc*) is defined ad the ratio of the number of offspring produced in each generation to the population size (usually denoted by pop_size). This ratio controls the expected number *Pc* x pop_size of chromosomes to undergo the crossover operation. A higher crossover rate allows exploration of more of the solution space and reduces the chances of settling for a false optimum; but if this rate is too high, it results in the wastage of a lot of computation time in exploring unpromising regions of the solution space.

Mutation is a background operator witch produces spontaneous random changes in various chromosomes. A simple way to achieve mutation would be to alter one or more genes. In genetic algorithms, mutation serves the crucial role of either (a) replacing the genes lost from the population during the selection process so that they can be tried in a new context or (b) providing the genes that were not present in the initial population.

Offspring 1	1	1	0	1	1	1	1	0
		↓						
Mutated offspring 1	1	0	0	1	1	1	1	0

Figure 7 Mutation Operation for Chromosomes with Binary String.

The mutation rate (denoted by *Pm*) is defined as the percentage of the total number of genes in the population. The mutation rate controls the rate at which new genes are introduced into the population for trial. If it is too low, many genes that would have been useful are never tried out; but if it is too high, there will be much random perturbation, the offspring will start losing their resemblance to the parents, and the algorithm will lose the ability to learn from the history of the search.

Genetic algorithms differ from conventional optimization and search procedures in several fundamental ways. Goldberg has summarized this as follows:

1. Genetic algorithms work with a coding of solution set, not the solutions themselves.

2. Genetic algorithms search from a population of solution, not a single solution.

3. Genetic algorithms use payoff information (fitness function), not derivatives or other auxiliary knowledge.

4. Genetic algorithms use probabilistic transition rules, not deterministic rules.

3. The Design of the Experiment and the Statistical Analysis of the Data

In general, experiments are used to study the performance of processes and systems. The process or system can be represented by the model shown in Figure 8. We can usually visualize the process as a combination of machines, methods, people, and other resources that transforms some input (often a material) into an output that has one or more observable responses. (Montgomery, 1997)

Some of the process variable $x_1, x_2, \ldots, x_p$ are controllable, whereas other variables $z_1, z_2, \ldots, z_q$ are uncontrollable (although they may be controllable for purposes of a test). The objectives of the experiment may include the following:

1. Determining which variables are most influential on the response *y*.

2. Determining where to set the influential x's so that y is almost always near the desired nominal value.

3. Determining where to set the influential x's so that variability in y is optimized.

4. Determining where to set the influential x's so that the effects of the uncontrollable variables $z_1, z_2, \ldots, z_q$ are minimized.

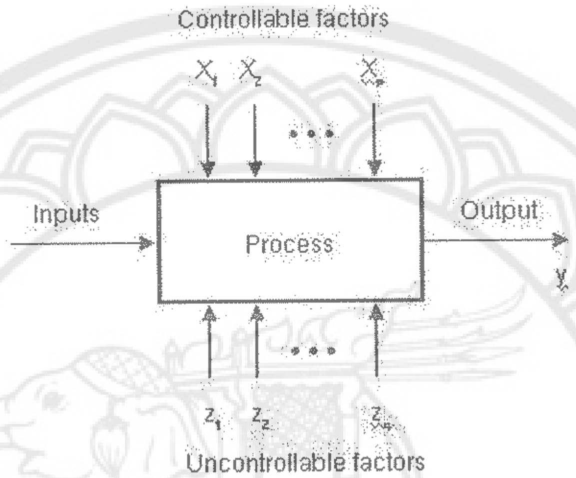

Figure 8 General Model of a Process or System. (Montgomery, 1997)

3.1 Factorial Design

Many experiments involve the study of the effects of two or more factors. In general, factorial designs are most efficient for this type of experiment. By a factorial design, we mean that in each complete trial or replication of the experiment all possible combinations of the levels of the factors are investigated. For example, if there are *a* levels of factor *A* and *b* levels of factor *B*, then each replicate contains all *ab* treatment combinations. When factors are arranged in a factorial design, they are often said to be crossed.

3.2 The 2^k Factorial Design

The most important of these special cases is that of k factors, each at only two levels. These levels may be qualitative, such as two machines, two operators, the "high" and "low" levels of a factor, or perhaps the presence and absence of a factor. A complete replicate of such a design require 2 X 2 X ... X 2 = 2^k observations and is called a factorial design.

The 2^k factorial design is particularly useful in the early stages of experimental work, when there are likely to be many factors to be investigated. It provides the smallest number of runs with which *k* factors can be studied in a complete factorial design.

3.3 The 2^k Fractional Factorial Design

As the number of factor in a 2^k factorial design increases, the number of runs required for a complete replicate of the design rapidly outgrows the resources of most experimenters. For example, a complete replicate of the 2^6 design requires 64 runs. In this design only 6 of the 63 degrees of freedom correspond to main effect, and only 15 degrees of freedom correspond to two-factor interactions. The remaining 42 degrees of freedom are associated with three-factor and higher interactions.

If the experimenter can reasonably assume that certain high-order interaction are negligible, then information on the main effects and low-order interactions may be obtained by running only a fraction of the complete factorial experiment. These Fractional factorial designs are among the most widely used types of designs for product and process design and for process improvement.

A major use of fractional factorials is in screening experiments. These are experiments in which many factors are considered with the purpose of identifying those factors (if any) that have large effects. Screening experiments are usually performed in the early stages of a project when it is likely that many of the factors initially considered have little or no effect on the response. The factors that are identified as important are then investigated more thoroughly in subsequent experiments.

3.4 The 3^k Factorial Design

The 3^k factorial design is a factorial arrangement with *k* factors each at three levels. Factors and interactions will be denoted by capital letter. The three levels are low, intermediate, and high. There are several different notations used to represent these factor levels; one possibility is to represent the factor levels by the digit 0 (low), 1 (intermediate), and 2 (high). Each treatment combination in the 3^k design will be denoted by k digits, where the first digit indicates the level of factor A, the second digit indicates the level of factor B, ..., and the k^{th} digit indicates the level of factor K. For example, in a 3^2 design, 00 denotes the treatment combination corresponding to A and B both at the low level, and 01 denotes the treatment combination corresponding to A at the low level and B at the intermediate level.

3.5 Fractional Replication of the 3^k Factorial Design

A complete replicate of the 3^k design can require a rather large number of runs even for moderate values of k, fractional replication of these designs is of interest.

3.5.1 The One-third Fraction of the 3^k Factorial Design

The largest fraction of the 3^k design is a one-third fraction containing 3^{k-1} runs. It is called 3^{k-1} fractional factorial design. To construct a 3^{k-1} fractional factorial design select a two-degrees-of-freedom component of interaction (generally, the highest-order interaction) and partition the full 3^k design into three blocks. Each of the three resulting blocks is a 3^{k-1} fractional design, and any one of the blocks may be selected for use.

3.5.2 Other 3^{k-p} Fractional Factorial Design

For moderate to large value of k, even further fractionation of the 3^k design is potentially desirable. In general, we may construct a $(\frac{1}{3})^p$ fraction of the 3^k design for $p < k$, where the fraction contains 3^{k-p} runs. Such a design is called a 3^{k-p} fractional factorial design. Thus, a 3^{k-2} design is one-ninth fraction, a 3^{k-3} design is a one-twenty-seventh fraction, and so on.

3.6 Analysis of Variance (ANOVA)

This topic was described by Pongcharoen (2001). The Analysis of Variance (ANOVA) is a commonly used approach for analyzing the results from factorial experiments. In general, a two factor factorial experiment is obtained as shown in Table 3 Where y_{ijk} is the observed value (response) obtained by using factor *A* at the i^{th} level (I = 1, 2, ..., *a*) and factor B at the j^{th} level (j = 1, 2, ..., *b*) for the k^{th} replicate (*k* = 1, 2, ..., *n*).

Table 3 Observed Response Arrangement for Two-Factor Factorial Design (Montgomery, 1997)

		Factor *B* 1	2	...	*b*
Factor *A*	1	$Y_{111}, Y_{112}, \ldots, Y_{11n}$	$Y_{121}, Y_{122}, \ldots, Y_{12n}$	...	$Y_{1b1}, Y_{1b2}, \ldots, Y_{1bn}$
	2	$Y_{211}, Y_{212}, \ldots, Y_{21n}$	$Y_{221}, Y_{222}, \ldots, Y_{22n}$	...	$Y_{2b1}, Y_{2b2}, \ldots, Y_{2bn}$
	⋮	⋮	⋮	⋮	⋮
	a	$Y_{a11}, Y_{a12}, \ldots, Y_{a1n}$	$Y_{a21}, Y_{a22}, \ldots, Y_{a2n}$	...	$Y_{ab1}, Y_{ab2}, \ldots, Y_{abn}$

The purpose of the ANOVA test is to establish whether a factor has a statistically significant effect on the variable being measured. ANOVA partitions the total variation within the results into its component parts, that is the variability due to each factor or interaction of interest and background uncertainty or error. There are a number of assumption behind ANOVA. Firstly, the results are independent of one another, that is, the result of one trial is not affected by another. Secondly, that differences in repeat trial would follow a normal distribution and finally that the error is normally distributed and is approximately equal over the whole experimental region (Kvanli et al., 1995). In general, the ANOVA table contains source of variation, sum of squares (*SS*), degrees of freedom (*DF*), mean squares (*MS*) and *F* value as summarized in Table 4.

Table 4 General ANOVA Table for the Two-Factor Factorial Design (Kvanli et al., 1995).

Source of Variation	Sum of Square	Degree of Freedom	Mean Square	F Value
A treatments	$SS_A = \frac{1}{bn}\sum_{i=1}^{a} y_{i..}^2 - \bar{y}^2$	*a*-1	$MS_A = \frac{SS_A}{a-1}$	$\frac{MS_A}{MS_E}$
B treatments	$SS_B = \frac{1}{an}\sum_{j=1}^{b} y_{.j.}^2 - \bar{y}^2$	*b*-1	$MS_B = \frac{SS_B}{a-1}$	$\frac{MS_B}{MS_E}$
Interaction	$SS_{AB} = \frac{1}{n}\sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij.}^2 - \bar{y}^2 - SS_A - SS_B$	(*a*-1) (*b*-1)	$MS_{AB} = \frac{SS_{AB}}{(a-1)(b-1)}$	$\frac{MS_{AB}}{MS_E}$
Error	$SS_E = SS_T - SS_A - SS_B - SS_{AB}$	*Ab*(*n*-1)	$MS_E = \frac{SS_E}{ab(n-1)}$	
Total	$SS_T = \sum_{i=1}^{a}\sum_{j=1}^{b}\sum_{k=1}^{n} y_{ijk}^2 - \bar{y}^2$	*Abn*-1		

The *F* test is used for comparing population variances. The *F* value is the ration of the mean square of the factor divided by the mean square of the error. It is therefore a ratio of two independent estimates of the population variance. The *F* value indicates the *p* value, which is the probability that a good model is falsely rejected. The *p* value is compared with a pre-specified significance level (α). It would lead to rejection of the null hypothesis (that the variances are the same) if $p > \alpha$ for example if *a* 95% confidence limit is used, rejection would occur if $p > 0.05$. The ANOVA can be conveniently performed using a statistical analysis packages such as Minitab and SPSS.