Simulation of Top Oil Temperature for Transformers

Masters Thesis and
Final Project Report

Power Systems Engineering Research Center

Simulation of Top-Oil Temperature for Transformers

Thesis and Final Report

Yong Liang

February 2001

Information about this Thesis and Project Report

This thesis was prepared under the direction of:

Daniel Tylavsky
Associate Professor
School of Electrical Engineering
Arizona State University
Tempe, AZ 85287-5706
Phone: 480-965-3460
Fax: 480-965-0745
e-mail: tylavsky@asu.edu

The thesis serves as the final report for the PSERC project "On-Line Peak Loading of Substation Distribution Transformers Through Accurate Temperature Prediction."

Additional Copies of the Report

Notice Concerning Reproduction

Acknowledgements

Executive Summary

When a transformer's winding gets too hot, either the load has to be reduced as a short-term solution or another transformer bay needs to be installed as a long-term solution. To decide on whether and when to employ either of these two strategies, one needs be able to predict the transformer temperature accurately. There are several transformer models for predicting top-oil temperature. These include the traditional top-oil-rise model, top-oil model and semi-physical model.

From previous research we have found that the best model for predicting TOT is the semi-physical model. Based on this model, we created and developed a software tool TOTPS (Top-Oil Temperature Prediction System) using LabVIEW, which gave SRP's engineers a friendly user interface with which to use the semi-physical model to predict TOT (Top-Oil Temperature) and calculate the maximum load that a transformer is capable of carrying.

Once we began using this tool we found that the semi-physical model yielded a wide range of coefficients for data that appeared, to the eye, to be similar. We hypothesized that the wide range of coefficients was caused by the absence of significant drive variables. We consequently added wind and solar radiation as driving variables to our model. When we used linear regression with the revised model to calculate the model coefficients, we found that this range of coefficient decreased, indicating that including these other variables produced a more reliable model. This increase in reliability was however still insufficient to allow us to have confidence that our model would calculate coefficients that represent the real system.

There are several outstanding issues that need to be solved before a production grade program can be produced. These tasks include:

- Identifying sources of error in the data we are working with.

- Developing a quality control algorithm that will eliminate as much erroneous data as possible, e.g., data samples taken during rain storms change the thermodynamics; hence these data points will not conform to a model based on dry weather.

- Improving our algorithm so that it can more robustly handle erroneous data.

- Implementing strategies that can deal effectively with the ill-conditioned nature of the matrices that result from linear regression.

- Determine the effect on the reliability of the model of including additional driving variables.

- Determine the effect of the nonlinearity on our model.

Table of Contents

Table of Contents
(continued)

1. INTRODUCTION

1.1 Introduction

The Salt River Project (SRP), a local power utility company based in Phoenix, Arizona, has many substation distribution transformers (SDT's) throughout the valley. When a transformer's winding gets too hot, either the load must be reduced, as a short-term solution, or another transformer needs to be installed to share the load, as a long-term solution. To decide on whether to use either of these two strategies, one should be able to predict the transformer's top-oil temperature (TOT) accurately. To do that, SRP came to the department of Electrical Engineering of Arizona State University and asked us to develop a tool that could be used in system planning to predict the maximum load that a transformer is capable of carrying by predicting a transformer's TOT, which can in turn be used to predict the temperature of the transformer winding.

There are several fundamental models for predicting transformer temperature, such as the top-oil-rise model and top-oil model. All models are designed to model the same phenomenon: when an increase in the load (current) of the transformer occurs, an increase in the losses within the device provides more heat and results in an increase in the overall TOT. Earlier work by another student [1] provided a model (semi-physical model) which could accurately predict TOT using load and ambient temperature as driving variables. Using this semi-physical model, our goal was to create and develop a

software tool (virtual instrument) that could allow engineer in SRP's planning group to easily predict TOT.

As the research progressed, we noticed that the results of the semi-physical model yielded coefficients that varied widely even though the input data sets used for constructing the model were similar, similar enough that the coefficients calculated should have been within five or ten percent of each others; consequently a second goal of our research was to investigate the cause of the widely varying coefficients.

1.2 Literature review

Two temperatures are used in estimating the loss of insulation life of a transformer: the hot-spot temperature (HST) and the top-oil temperature (TOT). Prediction of HST and TOT in transformers has gained momentum recently because of the deregulation of power distribution system. Accurately predicting HST and TOT can save money since it allows maximal loading of transformers while keeping the loss of life of the insulation within operator-defined limits. There are several fundamental models for predicting a transformer's TOT. In this section the various models which are relevant to our research are discussed.

Model #1 Top-oil-rise Model ANSC C57.92, C57.115

The classical model for predicting a transformer's TOT is described in IEEE/ANSI standard C57.92 (updated as C57.115) and is called IEEE/ANSI C57.115 model. The IEEE/ANSI C57.115 model for TOT rise over ambient temperature [4] is governed by the first order differential equation,

$$T_o \frac{d\theta_o}{dt} = -\theta_o + \theta_u \qquad (1.1)$$

which has the solution,

$$\theta_o = (\theta_u - \theta_i)(1 - e^{-t/T_o}) + \theta_i \qquad (1.2)$$

where,

$$\theta_u = \theta_{fl}\left(\frac{K^2 \cdot R + 1}{R + 1}\right)^n \qquad (1.3)$$

$$T_o = \frac{C\theta_{fl}}{P_{fl}} \qquad (1.4)$$

where,

θ_o : top-oil <u>rise</u> over ambient temperature, °C,

θ_{fl} : top-oil <u>rise</u> over ambient temperature at rated load, °C,

θ_u : ultimate top-oil <u>rise</u> for load L, °C,

θ_i : initial top-oil <u>rise</u> for t=0, °C,

θ_{amb} : ambient air temperature, °C,

T_o : time constant at rated KVA (hr.),

P_{fl}: rated load (MVA),

C: thermal capacity (Wh/^{o}C),

n: oil exponent,

K: ratio of load to rated load,

R: Ratio of load loss to no-load loss at rated load.

If (1.1) is solved, the TOT, θ_{top}, is then given by,

$$\theta_{top} = \theta_o + \theta_{amb}$$
$$= (\theta_u - \theta_i)(1 - e^{-t/T_o}) + \theta_i + \theta_{amb} \tag{1.5}$$

Our goal is to find the parameters of (1.5) (namely T_0, R and θ_{fl}) that will most accurately predict θ_{top} as measured by field data. There are several ways to do this. One way is using linear regression along with measured data. To use linear regression we must first construct a discrete-time form of (1.1). Applying the forward Euler discretization rule,

$$\frac{d\theta_0[k]}{dt} = \frac{\theta_0[k] - \theta_0[k-1]}{\Delta t}, \tag{1.6}$$

where Δt is the sampling period, and solving we get,

$$\theta_0[k] = \frac{T_0}{T_0 + \Delta t}\theta_0[k-1] + \frac{\Delta t \theta_{fl}}{T_0 + \Delta t}\left(\frac{(\frac{I[k]}{I_{rated}})^2 R + 1}{R+1}\right)^n \tag{1.7}$$

where $I[k]$ is the per-unit transformer current (based on the rated value of the transformer) at time-step index k.

When the load current is near its rating, or $R > 1$ and $K^2 R > 1$, TOT rise over ambient temperature is given by,

$$\theta_0[k] = \frac{T_0}{T_0 + \Delta t}\theta_0[k-1] + \frac{\Delta t \theta_{fl} R}{(T_0 + \Delta t)(R+1)}(\frac{I[k]}{I_{rated}})^{2n} + \frac{\Delta t \theta_{fl}}{(T_0 + \Delta t)(R+1)} \qquad (1.8)$$
$$= K_1 \theta_0[k-1] + K_2 I[k]^{2n} + K_3$$

But this simplified model does not accurately account for the effects of ambient temperature dynamics on TOT [2,5].

Model #2 Top-oil model

The top-oil model (as opposed to top-oil-<u>rise</u> model) proposed by Lesieutre et al. [2], accounts for dynamic variations in ambient temperature. It is based on the differential equation that defines the TOT θ_{top} by,

$$T_o \frac{d\theta_{top}}{dt} = -\theta_{top} + \theta_u + \theta_{amb} , \qquad (1.9)$$

and has the solution,

$$\theta_{top} = (\theta_u + \theta_{amb} - \theta_{topi})(1 - e^{-\frac{t}{T_o}}) + \theta_{topi} , \qquad (1.10)$$

where θ_{topi} is the initial TOT for *t=0*.

To improve upon the manufacturer supplied coefficients, Lesieutre et al. used linear regression algorithm to get optimal coefficients obtained from measured data. To use linear regression, the defining equation must first be written in a discretized linear form. To achieve this, Lesieutre et al. [2] applied the forward Euler discretization,

$$\frac{d\theta_{top}[k]}{dt} = \frac{\theta_{top}[k] - \theta_{top}[k-1]}{\Delta t},$$ (1.11)

and solved (1.9), with n=1, to get the following discretized form:

$$\theta_{top}[k] = \frac{T_o}{T_o + \Delta t}\theta_{top}[k-1] + \frac{\Delta t}{T_o + \Delta t}\theta_{amb}[k] + \frac{\Delta t \theta_{fl} R}{(T_o + \Delta t)(R+1)}\left(\frac{I[k]}{I_{rated}}\right)^2 + \frac{\Delta t \theta_{fl}}{(T_o + \Delta t)(R+1)}.$$ (1.12)

where $I[k]$ is the per-unit transformer current (based on the rated value of the transformer) at time-step index k. Equation (1.12) can also be written in a form amenable to linear regression with coefficients $k_1 - k_3$ as,

$$\theta_{top}[k] = k_1\theta_{top}[k-1] + (1-k_1)\theta_{amb}[k] + k_2 I[k]^2 + k_3.$$ (1.13)

Lesieutre et al. showed that the top-oil model outperforms the top-oil-rise model [2].

Model #3 Semi-physical Model

It has been shown [1,5,6] that if $(1-k_1)$ in equation (1.13) is replaced by another coefficient, k_4, then,

$$\theta_{top}[k] = k_1\theta_{top}[k-1] + k_4\theta_{amb}[k] + k_2 I[k]^2 + k_3.$$ (1.14)

Re-assigning the subscripts of the coefficients we get,

$$\theta_{top}[k] = k_1 I[k]^2 + k_2\theta_{amb}[k] + k_3\theta_{top}[k-1] + k_4.$$ (1.15)

The resulting model is known as a semi-physical model because it is not based entirely on physical principles.

Another student compared these three different models for the prediction of TOT and found that Model #3 gave the best prediction [1]. She tried several attempts to improve the semi-physical model used for TOT prediction. She also studied some of the sources of error that affect TOT prediction, such as database quantization, remote ambient temperature monitoring and low sampling rate. After thorough study, we selected the semi-physical model for the prediction of a transformer's TOT.

1.3 Thesis overview

The study focuses on the development of a graphical tool for predicting TOT using a semi-physical model and on the studies conducted to assess the effect of solar radiation and wind velocity on the prediction of TOT. In Chapter 2, three algorithms needed for the simulation of TOTPS are discussed. In Chapter 3, the GUI (graphical user interface) for predicting TOT is provided. In Chapter 4, we investigate the causes of coefficient variability to improve the TOTPS reliability. In Chapter 5, the results of solar radiation and wind velocity on the resultant TOT predicted by linear regression model are reported. Finally some conclusions and proposals for future work are presented in Chapter 6.

2. ALGORITHM NEEDED FOR THE SIMULATION OF TOTPS

2.1 Introduction

In this study, we use the so-called semi-physical model proposed by Tylavsky, Qing et. al [1,5,6] as the basis for developing TOTPS (Transformer Top-Oil Temperature Prediction System). In order to build TOTPS, several algorithms were studied. These included an algorithm needed to determine the coefficients in the semi-physical model, an algorithm needed to calculate the HST and an algorithm needed to use the semi-physical model to predict the maximum load that a transform can endure without exceeding operating limits.

2.2 Linear regression algorithm

In Chapter 1 it was shown that we could configure our model so that we could use linear regression to get optimal coefficients for a semi-physical model from measured data. The typical criteria for linear regression is a least-squares criteria to minimize the error squared over all data points.

For a data set with one input, X, and one output, Y, and assuming there is a linear relationship between the input and output, linear regression can be used to find the optimal coefficients, A and k, which give a best fit (in the least-square sense) to the single-variable equation,

$$Y = A + k * X,\tag{2.1}$$

where,

X: independent variable,

k: a coefficient to be determined,

Y: dependent variable.

Using linear regression, the value of k is determined by selecting the value such that the square of the error between the actual Y and the predicted Y is minimized. But in our case, (1.15), there are three independent variables: load, ambient temperature and TOT. Then a multiple-regression is necessary. The method used for multiple-regression is an extension of that used for single regression.

For a model with three independent variables, the generalization of equation (2.1) can be written in the form,

$$Y = k_1 * X_1 + k_2 * X_2 + k_3 * X_3 + k_4.\tag{2.2}$$

Equation (2.2) can be expressed in normalized mathematical format,

$$\begin{aligned} Y &= k_1 * (X_1 - \bar{X}_1) + k_2 * (X_2 - \bar{X}_2) + k_3 * (X_3 - \bar{X}_3) + k_4 + \varepsilon \\ &= \hat{Y} + \varepsilon \end{aligned}\tag{2.3}$$

Where,

Y : actual value of TOT (dependent variable),

$\hat{Y}$: predicted (estimated) value of TOT,

ε : the error between the actual and the predicted TOT,

X_1 : load value (independent variable or driver variable),

$\bar{X}_1$: mean value of X_1,

X_2 : ambient temperature (independent variable or driver variable),

$\bar{X}_2$: mean value of X_2,

X_3 : TOT (independent variable or driver variable),

$\bar{X}_3$: mean value of X_3,

k_1, k_2, k_3, k_4 : coefficients to be determined,

N: the number of sample data.

Equation (2.3) can be written in matrix notation,

$$y_j = \lfloor x_j \rfloor [K] + \varepsilon_j \qquad (2.4)$$

where,

$$x_j = X_j - \bar{X}_j \qquad (2.5)$$

y_j : normalized dependent variable,

$\lfloor x_j \rfloor$: vector of independent (driver) variables with element $x_{j,i}$,

$[K]$: vector of coefficients with element k_i,

j : subscript index for data points.

Note that when the equation for multiple-regression is written in matrix notation, it appears identical to that of single regression.

Equation (2.3) also can be written using scalar notation,

$$y_j = \sum_i x_{j,i} k_i + \varepsilon_j, \tag{2.6}$$

where,

i: subscript index for the ith regression coefficient.

In order to determine the coefficients, we select those values when the error squared between the actual TOT and the predicted TOT is minimized. This criterion can be expressed as,

$$\min \sum (y_j - \hat{y}_j)^2 = \min \sum \varepsilon_j^2, \tag{2.7}$$

By substituting equation (2.6) into (2,7), we get,

$$\min(\sum \varepsilon_j^2) = \min\{\sum_j (y_j^2 - 2y_j(\sum_i x_{j,i} k_i) + (\sum_i x_{j,i} k_i)^2)\}$$

$$= \min\{ \sum_j (y_j^2 - 2y_j[x_j][K] + ([x_j][K])^2)\} \tag{2.8}$$

In order to find the coefficients that minimize the error squared, the equation (2.8) can be differentiated with respect to the ith coefficient, k_i, with the result set equal to zero.

$$\sum_j 2y_j x_{j,i} = \sum_j \frac{d}{dk_i}(x_{j,1}k_1 + x_{j,2}k_2 + \cdots)^2$$
$$= 2\sum_j x_{j,i}(x_{j,1}k_1 + x_{j,2}k_2 + \cdots) \tag{2.9}$$

$$i = 1, \cdots, 4$$

$$\sum_j y_j x_{j,i} = (\sum_j x_{j,1}x_{j,i})k_1 + \cdots + (\sum_j x_{j,i}^2)k_i + \cdots \tag{2.10}$$

$$i = 1, \cdots, 4$$

The notation of equation (2.10) can be simplified by two definitions:

Definition I:

$$W_{1i} = \sum_j x_{j,1} x_{j,i} = [X^T][X]_{1i} \tag{2.11}$$

= matrix of rank equal to the number of regression coefficients,

Definition II:

$$X_i^T Y = \sum_j x_{j,i} y_j = [X^T][Y]_i \tag{2.12}$$

= vector of rank equal to the number of regression coefficients, where, T is the matrix transpose operation.

Using equation (2.11) and equation (2.12), equation (2.10) can be rewritten as,

$$X_i^T Y = \sum_P W_{iP} k_P \tag{2.13}$$

for $i = 1, \cdots, 4$.

Using matrix notation, (2.10) maybe written as,

$$[X^T Y] = [W][K] \tag{2.14}$$

The vector of coefficients can be solved by inverting the W matrix,

$$[K] = [W]^{-1}[X^T Y] \tag{2.15}$$

Using (2.15) along with measured data, where,

$X_1 = TOT$: top-oil temperature ($^o C$),

$X_2 = \theta_{amb}$: ambient temperature ($^o C$),

$X_3 = Load$: transformer load (p.u.), (base load is 22.4MVA),

we can find the optimal coefficients of the semi-physical model that minimize the error between the actual TOT and the predicted TOT.

2.3 Hot-spot temperature prediction

An algorithm is needed to use the semi-physical model to predict the maximum load that a transform can endure without exceeding operating limits. One of the operating limits is the hot-spot temperature limit. The IEEE Transformer Loading Guide (IEEE/ANSI C57.115) [4] equations use the TOT rise over ambient temperature to determine the winding hot-spot temperature during an overload. The IEEE Loading Guide (IEEE/ANSI C57.115) equations are based on the assumption that the temperature of the oil exiting the winding ducts is the same temperature as the top-oil in the tank. The hot-spot temperature θ_{hs} is assumed to consist of three components given by the following equation [4],

$$\theta_{hs} = \theta_0 + \theta_{amb} + \theta_g \tag{2.16}$$

Where,

θ_0 : top-oil rise <u>over</u> ambient temperature,

θ_{amb} : ambient air temperature,

θ_g : hot-spot conductor rise over top-oil temperature.

By substituting equation (1.5) into equation (2.16), we get,

$$\theta_{hs} = \theta_{top} + \theta_g \tag{2.17}$$

The ultimate hot-spot conduct rise over top-oil for load is [2],

$$\theta_g = \theta_{g(fl)} K^{2m} \tag{2.18}$$

Where,

$\theta_{g(fl)}$: hot-spot conductor rise over top-oil temperature at rated load,

K: ratio of load to rated load,

m: exponential power of winding loss versus winding temperature rise over oil.

By substituting equation (2.18) into equation (2.17), we get,

$$\theta_{hs} = \theta_{top} + \theta_{g(fl)} K^{2m} \tag{2.19}$$

The parameter m is determined by the cooling modes used in liquid filled transformer. The guide dictate the typical value for m as:

m=0.8:for self-cooled (OA), forced-air-cooled (FA) operation,

m=1: for directed flow forced-air, forced-oil-cooled operation.

The observed transformers use natural convection of oil in the transformer and natural convection of cooling air over the radiator (OA), so we set m=0.8 to predict the HST.

2.4 Iteration algorithm for calculating maximum load

Once we have the optimal coefficients for predicted TOT, our next task (and our ultimate goal) is to use this model to find the maximum load that a transformer can tolerate without exceeding the maximum TOT and HST provided by SRP.

Because the input data to this process is not available in a simple form, to find the maximum load not violating SRP's maximum TOT and HST criteria, we must employ an iterative algorithm. We will describe a Newton-type iterative procedure, where the derivative (inherent in a Newton scheme) is approximated using the difference between successive iterates. Precisely what this means is described in the language of

mathematics. We will provide a conceptual view of this procedure by using the plots shown in Figure 2.1. This figure shows various curves that are useful in describing the process by which we can use our model to find the maximum load a transformer can withstand.

Figure 2.1 (a) shows the initialized load plot (load(0)) for 2 days at iteration index P=0, the increment of load at iteration index P=0 (*Incr*(0)) and the load plot (load(1)) for the next iteration index, P=1. The relationship between load(0) and load(1) is given by,

$$load(1)=load(0)*Incr(0). \tag{2.20}$$

The TOT plot for 2 days (TOT(0)) in Figure 2.1 (b) and HST plot (HST(0)) in Figure 2.1(c) at time step t=0 are calculated using equations (1.15) and (2.19), respectively. The maximum TOT value of TOT(0) is given by TOTmax(0) and the maximum HST value of HST(0) is given by HSTmax(0) at iteration index P=0. The TOT limit and HST limit given by SRP are also shown in Figure 2.1 (b) and (c) as dotted lines. We can get the load increment at iteration index P=0, Incr(0), by comparing the maximum TOT value (TOTmax(0)), to the TOT limit, and the maximum HST value (HSTmax(0)) to the HST limit. We can get load(1) by equation (2.20) and then get TOT(1) and HST(1) using (1.15) (along with our optimized coefficients) for the iteration index P=1. Then we can get the load increment *Incr*(1) for next time step. We keep calculating until the maximum TOT or HST hit the limit.
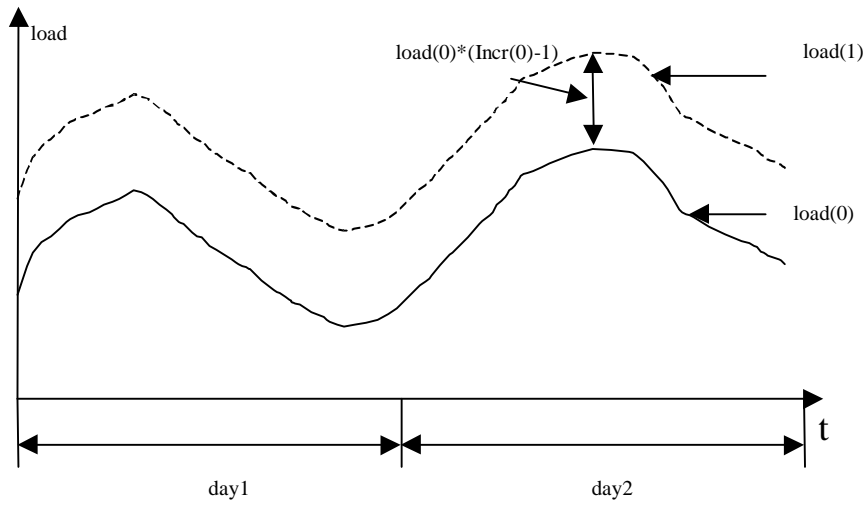
(a)
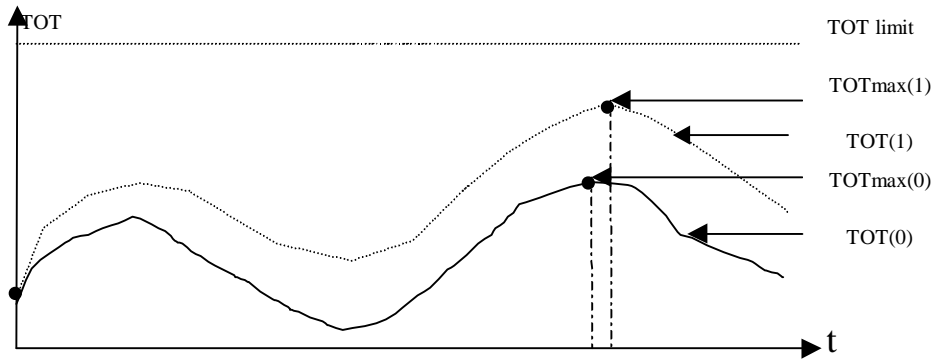
(b)
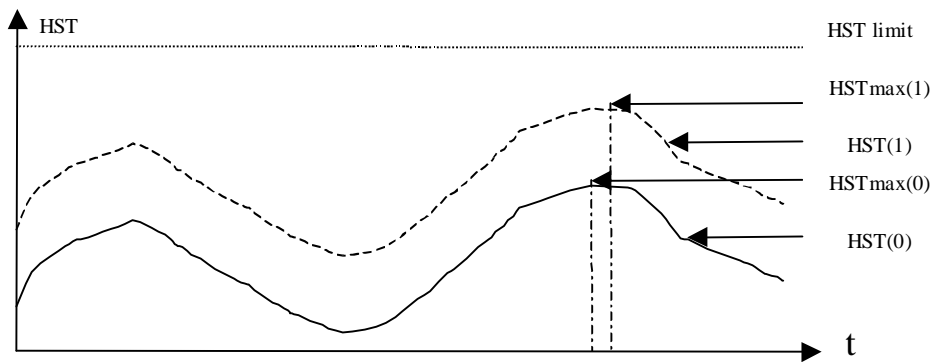
(c)

Figure 2.1 Iteration algorithm scheme

To completely define this iterative scheme, we must define the TOT and HST initial conditions at t=0 for any given loading level, the equation that estimates the load at successive iterations, and the convergence criteria.

Initial estimate of maximum load

We use the semi-physical model to predict the maximum load that a transformer is capable of carrying. For a selected load/temperature profile, we can get the TOT by using (1.15) and the calculated optimal coefficients. There are three initial conditions that we must define for the iteration scheme. First we must define the initial TOT value at t=0, then we must define the initial HST value at t=0. Finally we must define the initial load increment at t=0. These values will be used to start the TOT simulation for any maximum load estimate. To do this, we assume that the transformer has been operating in an environment where all driving variables have been constant for a sufficiently long time that the transformer is in a steady state.

Equation (1.15) and (2.19) can be rewritten as,

$$\theta_{top}(t,0) = k_1 I(t,0)^2 + k_2 \theta_{amb}(t,0) + k_3 \theta_{top}(t-1,0) + k_4, \qquad (2.21)$$

$$\theta_{hs}(t,0) = \theta_{top}(t,0) + \theta_{g(fl)} I(t,0)^{2m}, \qquad (2.22)$$

where, t is the time step index for the data points, and 0 is the iteration index. Assume at time index j, we get the initial maximum value of TOT and HST for a given load profile. That means,

$TOT \max(0) = \theta_{top}(j,0)$

$$= k_1 I(j,0)^2 + k_2 \theta_{amb}(j,0) + k_3 \theta_{top}(j-1,0) + k_4, \quad (2.23)$$

$HST \max(0) = \theta_{top}(j,0) + \theta_{g(fl)} I(j,0)^{2m}$

$$= TOT \max(0) + \theta_{g(fl)} I(j,0)^{2m}, \quad (2.24)$$

where, TOTmax(0) and HSTmax(0) are the maximum values at iteration index t=0. We also assume that at the time index j, we get the maximum value of ambient temperature,

$$Amb \max(0) = \theta_{amb}(j,0) \quad (2.25)$$

Our approach to finding the factor, $Incr1(0)$ (that when multiplied by the load approximates the maximum load to loaded by a transformer), is to assume that the TOTmax(0) is approximately the <u>steady</u> <u>state</u> TOT caused by $I(j,0)$ and Ambmax(0).

This means that a reasonable approximation between TOTmax(0), $I(j,0)$ and Ambmax(0) can be obtained using (1.15) and the steady state assumption that,

$$TOT(j,0) = TOT(j+1,0) = TOT \max(0), \forall j \quad (2.26)$$

Substituting (2.26) into (1.15) yields,

$$TOT \max(0) = k_3 TOT \max(0) + k_1 I(j,0) + k_2 Amb \max(0) + k_4 \quad (2.27)$$

Solving for TOTmax(0) yields,

$$TOT \max(0) = \frac{k_1}{1-k_3} I(j,0)^2 + \frac{k_2}{1-k_3} Amb \max(0) + \frac{k_4}{1-k_3} \quad (2.28)$$

We know from (1.12) and (1.13) that $k_2 + k_3 \approx 1$. Substituting this into (2.28) and rearranging yields,

$$TOT\max(0) - Amb\max(0) \approx \frac{k_1}{1-k_3}I(j,0)^2 \tag{2.29}$$

This equation says that the steady state *TOT*max rise above ambient is approximately

given by $\frac{k_1}{1-k_3}I(j,0)$. If this is true, then the steady state *TOT*limit rise above ambient

should be approximately given by,

$$TOT\text{limit} - Amb\max(0) \approx \frac{k_1}{1-k_3}I(j,0)^2 \, Incr1(0) \tag{2.30}$$

Dividing equation (2.29) by (2.30), we get,

$$Incr1(0) = \frac{TOT\text{limit} - Amb\max(0)}{TOT\max(0) - Amb\max(0)}. \tag{2.31}$$

This load increment would be the load increment to use if the TOT limit provided was the

only applicable limit; however, we must also obey the HST limit. To get an estimate of

the load increment that would cause the HST to reach its limit, we make the same

assumptions (i.e. of coincidence in time of *HST*max(0), *TOT*max(0) and *Amb*max(0), and

the steady state assumption). Using these assumptions, (2.19) and (2.29) yields,

$$HST\max(0) = TOT\max(0) + \theta_{g(fl)}I(k,0)^2$$

$$= (\frac{k_1}{1-k_3} + \theta_{g(fl)})I(j,0)^2 + \frac{k_2}{1-k_3}Amb\max(0) + \frac{k_4}{1-k_3} \tag{2.32}$$

Assuming $k_2 + k_3 \approx 1$, and substituting this into (2.32) while rearranging yields,

$$HST\max(0) - Amb\max(0) = (\frac{k_1}{1-k_3} + \theta_{g(fl)})I(j,0)^2, \tag{2.33}$$

This equation says that the steady state *HST*max rise above ambient is approximately

given by $(\dfrac{k_1}{1-k_3}+\theta_{g(fl)})I(j,0)$. If this is true, then the steady state *HST* limit rise above

ambient should be approximately given by,

$$HST\text{limit} - Amb\max(0) = (\frac{k_1}{1-k_3}+\theta_{g(fl)})I(j,0)^2\,Incr2(0),$$ (2.34)

From equation (2.33) and (2.34), we get,

$$Incr2(0) = \frac{HST\text{limit} - Amb\max(0)}{HST\max(0) - Amb\max(0)}.$$ (2.35)

To avoid exceeding either the HST or TOT limits, we select the load multiplier to be,

$$Incr(0) = \min(Incr1(0), Incr2(0))$$ (2.36)

This load multiplier is applied to the initial load profile, $I(k,0)$, to get load(1).

Subsequent estimates of maximum load

The candidate load multiplier at successive iterations, P, is derived similarly and is given

by,

$$Incr1(P) = \frac{TOT\text{limit} - Amb\max(P)}{TOT\max(P) - Amb\max(P)},$$ (2.37)

and,

$$Incr2(P) = \frac{HST\text{limit} - Amb\max(P)}{HST\max(P) - Amb\max(P)}.$$ (2.38)

The load multiplier at iteration index P is selected as the minimum value of $Incr1(P)$ and

$Incr2(P)$,

$$Incr(P) = \min(Incr1(P), Incr2(P)),\qquad(2.39)$$

At iteration index P+1, the load curve is defined by,

$$I(t, P+1) = Incr(P)I(t, P).\qquad(2.40)$$

This load curve is used along with (1.15) to calculate $TOT(t, P+1)$, and $HST(t, P+1)$.

Convergence Criteria

The iteration scheme is said to converge at iteration P when $TOT\max(P) < TOTlimit$, $HST\max(P) < HSTlimit$ and either $TOTlimit - TOT\max(P+1)$ or $HSTlimit - HST\max(P+1)$ is less than $0.5\ ^oC$.

TOT & HST initial conditions at t=0

Accurately calculating $TOT(t, P)$ or $HST(t, P)$ requires that we use $I(t, P)$ and a reasonable initial condition for $TOT(0, P)$ that is based on $I(0, P)$. We estimate this initial condition by assuming that for $t < 0$ the driving variables have been constant for a sufficiently long period of time that the transformer is in a steady state. Using this assumption (i.e. $TOT(t, P) = TOT(0, P), t < 0$) and (1.15) for t=0 yields,

$$TOT(0, P) = k_3 TOT(0, P) + k_1 I(0, P)^2 + k_2 Amb(0, P) + k_4,\qquad(2.41)$$

Solving for $TOT(0, P)$ gives,

$$TOT(0, P) = \frac{k_1}{1 - k_3} I(0, P)^2 + \frac{k_2}{1 - k_3} Amb(0, P) + \frac{k_4}{1 - k_3}.\qquad(2.42)$$

21

3. TOTPS DEVELOPMENT

3.1 LabVIEW overview

LabVIEW, much like the text-based languages such as C and BASIC, is a graphical program development environment. It looks like Visual BASIC language and has several similar features such as main function and subroutines. It contains functions analogous to library functions in conventional languages which can be readily implemented by 'pointing and clicking' with a mouse. However, LabVIEW uses a graphical programming language, G, to create programs in a block diagram form instead of creating lines of code. G simplifies scientific computation, process monitoring/control and application measurement and can be used for a wide variety of other applications. The graphical nature of the LabVIEW environment makes it significantly more intuitive than conventional text-based languages.

LabVIEW programs are called virtual instruments (VIs) because of their appearance and the way their operation imitates actual instruments. Generally, a VI has three main parts: front panel, block diagram and connector.

♦ Front panel: the interactive user interface of a VI, which simulates the front panel of a physical instrument. It contains knobs, push buttons, graphs, controls (user inputs) and indicators (program outputs).

Figure 3.1. Front panel of LabVIEW

♦ Block diagram: the VI's source code in G language compared to the text-based language. It includes lower-level Vis (subVIs), built-in functions, constants and program execution control structures (loop, while, case, sequence structure).
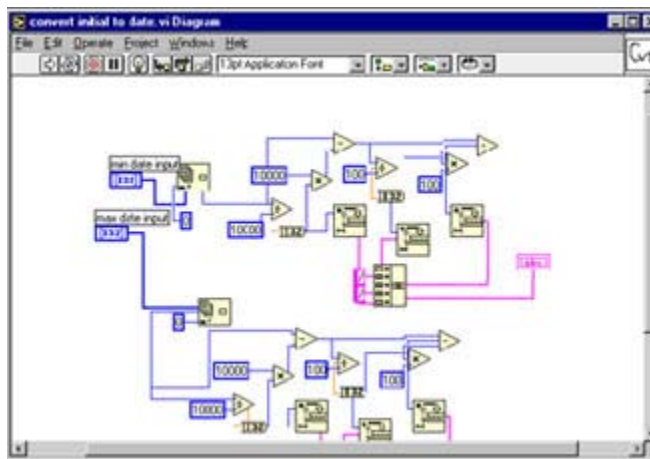
Figure 3.2. Block diagram of LabVIEW

♦ Connector: be used to wire data into the VI from other block diagrams when the VI is used as a subVI (a VI that is used within another VI).

With these three parts, LabVIEW creates the concept of modular programming. One can divide an application into series of tasks, which can be divided again until a complicated application becomes series of simple subtasks.

LabVIEW can be used not only as a instrument control language, but also as a general purpose programming language with extensive libraries of functions. One can set breakpoints and make single-step debugging through the program to see how data passes through the program.

3.2 Overview of TOTPS development

Development of the TOTPS required that two different types of VI's be developed,

- Analytical VI's: VI's that perform linear regression, HST prediction and maximum load prediction,

- Interface VI's: VI's that drive the interface.

Many of the analytical VI's were developed in coordination with another student's work [1] and thus will not be reported upon here. In other words, I will focus on the interface VI's developed to drive the TOTPS user interface.

3.3 Customer requirement

The simulation of Transformer Top-Oil Temperature Prediction System (TOTPS) has been sponsored by SRP (Salt River Project) for two years. The overall goal of this work is to produce an instrument that SRP can use in the field. The goal of this work was to

produce a friendly graphical user interface so that the system planner or operator can easily use the load, ambient temperature and TOT of the substation to predict the maximum load that a transformer is capable of carrying. Additional requirements are listed as followings:

Inputs

- Need to read data collected from the EMS (Energy Management System) UNIX platform, normally saved as a text file,

- Given a user specific day, a moving window can display 10 days' data centered on the user specified date,

- The user must be given the choices to select training data to be used to calculate the coefficient for the semi-physical model,

- There must be a way to highlight bad data,

- There must be a method of approximating missing data.

General

- Friendly graphical user interface for Windows NT/98 operation system,

- Flexible,

- Simple to use,

- Y2K compatible.

Output

- Predict the maximum load, for a given temperature and load profile.

3.4 Software selection

Selecting software for providing a tool for SRP is an important decision. From the point of view established by the goals of the project, the software we considered fell into one of two categories:

- Software that could create a technical user interface (including controls and graphical charts) with ease;

- Software that could allow complex algorithmic development with ease.

There are many issues involved in deciding which category our software should belong to. Ultimately we chose LabVIEW, which belongs to the first category (ease of user interface development). We chose software from the first category because:

- The user interface for the production software is critical if it is to find acceptance at SRP. We believe that using software that could allow us to create a powerful user interface would eventually save use time.

- We expected that as the project evolved we could have to make a lot of changes to the interface. (We were right!) Software that allowed us to do this quickly, accurately and easily, would save us development time.

- We felt that we could use any software (from first category) to develop the technical algorithm and the interface without too much difficulty.

Of the software in the first category we choose LabVIEW, a powerful visual programming development environment, because:

- It is excellently suited to developing a GUI that contains the features we know we would need.

- LabVIEW's graphical programming environment has several features that allow the developer to easily create simulation and prediction software. These features include natural representation, easy readability and interaction, language independence, programming at higher levels of abstraction and rapid prototyping.

- The graphical nature of the LabVIEW environment is significantly more intuitive than conventional text-based languages. Compare with other text-based languages, one can learn the basics of LabVIEW in an extremely short period of time. Upon completion of the tutorial (approximately 50 hours), one can implement several small applications including file manipulation, data manipulation and GPIB instrumentation control.

- In its ultimate incarnation, SRP expected that the tool we are developing would be executed using date collected in real-time. LabVIEW is designed to be good at developing virtual instrument that access real time data.

- LabVIEW's friendly graphical environment makes it possible to effectively perform fairly complex array and string manipulations, which are typically more difficult to do using text-based languages.

- There is a perception that software based on graphical programming environment are not well suited to large-scale applications, and that the traditional text-based programming languages are better suited for developing large applications. We were told that LabVIEW uses hierarchical programming methodologies and abstraction mechanisms that allow an efficient management of program complexity in large

applications. So far, we have been satisfied with the way LabVIEW handles hierarchical programming and complexity management.

- We also noted that troubleshooting and debugging in LabVIEW is also easier than in conventional text-based languages. In LabVIEW the edit/compile/link/run sequence of traditional programming languages is replaced by the draw/run cycle.

- The most impressive feature of LabVIEW's graphical programming environment is that LabVIEW programs do not have to be complied into executables prior to running.

3.5 Designing the GUI (Graphic User Interface)

In the LabVIEW program environment, the front panel (or GUI) is the window through which the user interacts with the program. The front panel is primarily a combination of controls and indicators. Controls simulate typical input objects you might find on a conventional instrument, such as knobs and switches. Indicators show output values produced by the program. In general,

Controls = Input from the User = Source Terminals,

Indicators = Outputs to the User = Destinations.
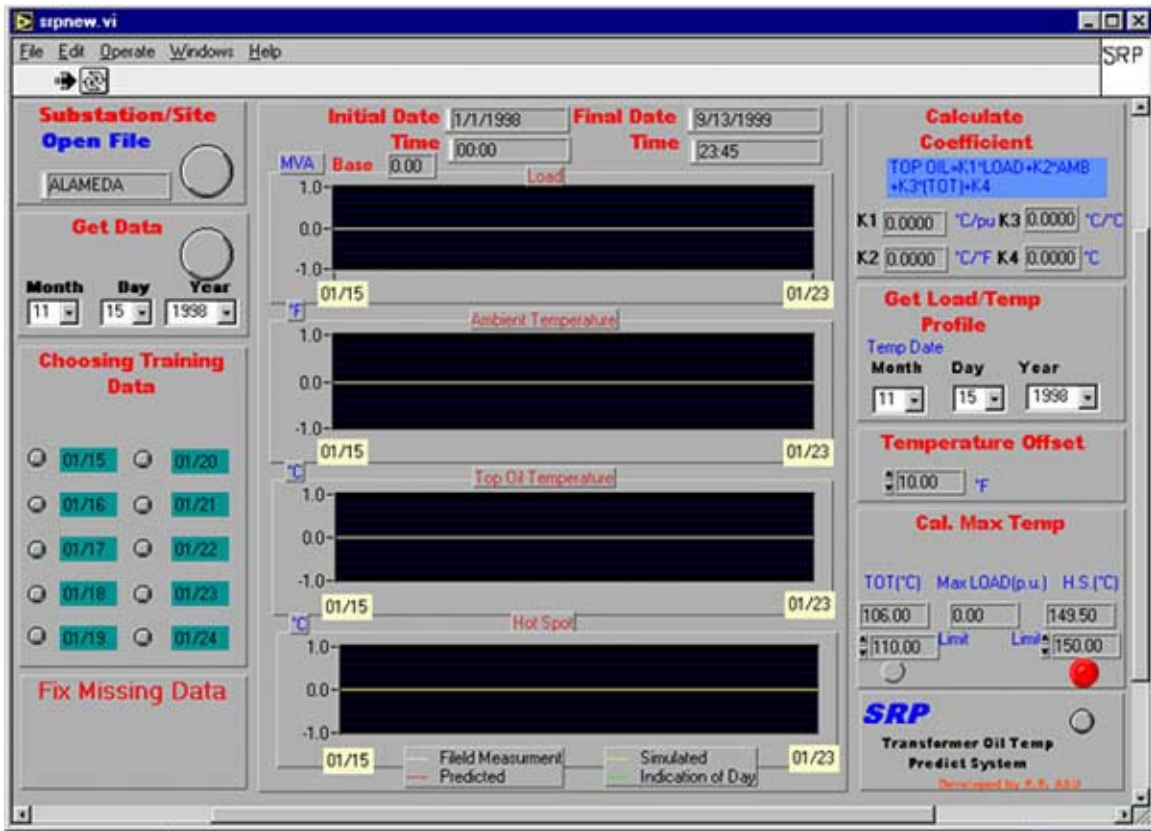
Figure 3.3. TOTPS interface

Once we had established the basic front panel for the TOTPS (shown in Figure 3.3), we presented it to SRP's planners and EMS operators. The indicators and controls were made to appear operational through the use of 'dummy' data. We showed SRP operators how they would be able to select menu items, push buttons and use the interface just as if it were connected to the actual EMS system. We were able to get feedback from the users and immediately make changes based on their comments. This was a very effective method for obtaining agreement by all the relevant parties, since it gave the users a real voice in the development process.

Many traditional text-based software development tools allow the developer to quickly build a GUI for demonstration purposes, but then the GUI must be redone in a

conventional programming language for implementation.   An advantage of using LabVIEW was that we were able to take the GUI developed with the users' inputs, and use it directly in our final product. All that was required was to strip out the 'dummy' data, and make the appropriate connections to the actual acquired data. Thus, we saved significant development time by using LabVIEW as both a rapid prototype tool and as our programming language during creation of the TOTPS software.

3.5.1 Virtual instrument architecture

LabVIEW's modularity greatly streamlines the process of developing TOTPS software. Each of the functions that make up TOTPS has a main virtual instrument (VI) which contains subVIs representing steps in that function. Likewise, each subVI may contain additional subVIs and so on. Each subVI performs a single function or a group of associated functions, which also facilitates the opportunity for code reuse.   The modularity can be seen in Figure 3.4, which shows the VI hierarchy of TOTPS.
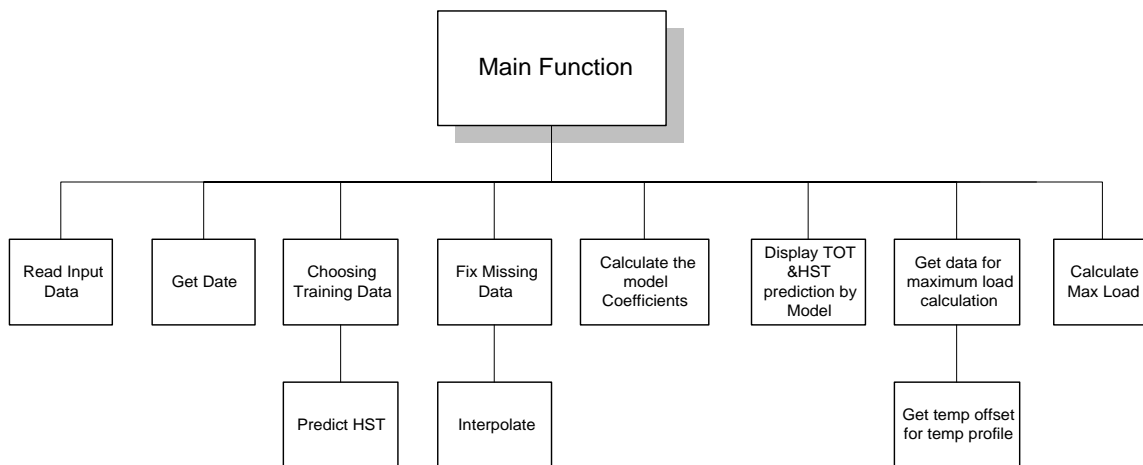
Figure 3.4. VI hierarchy
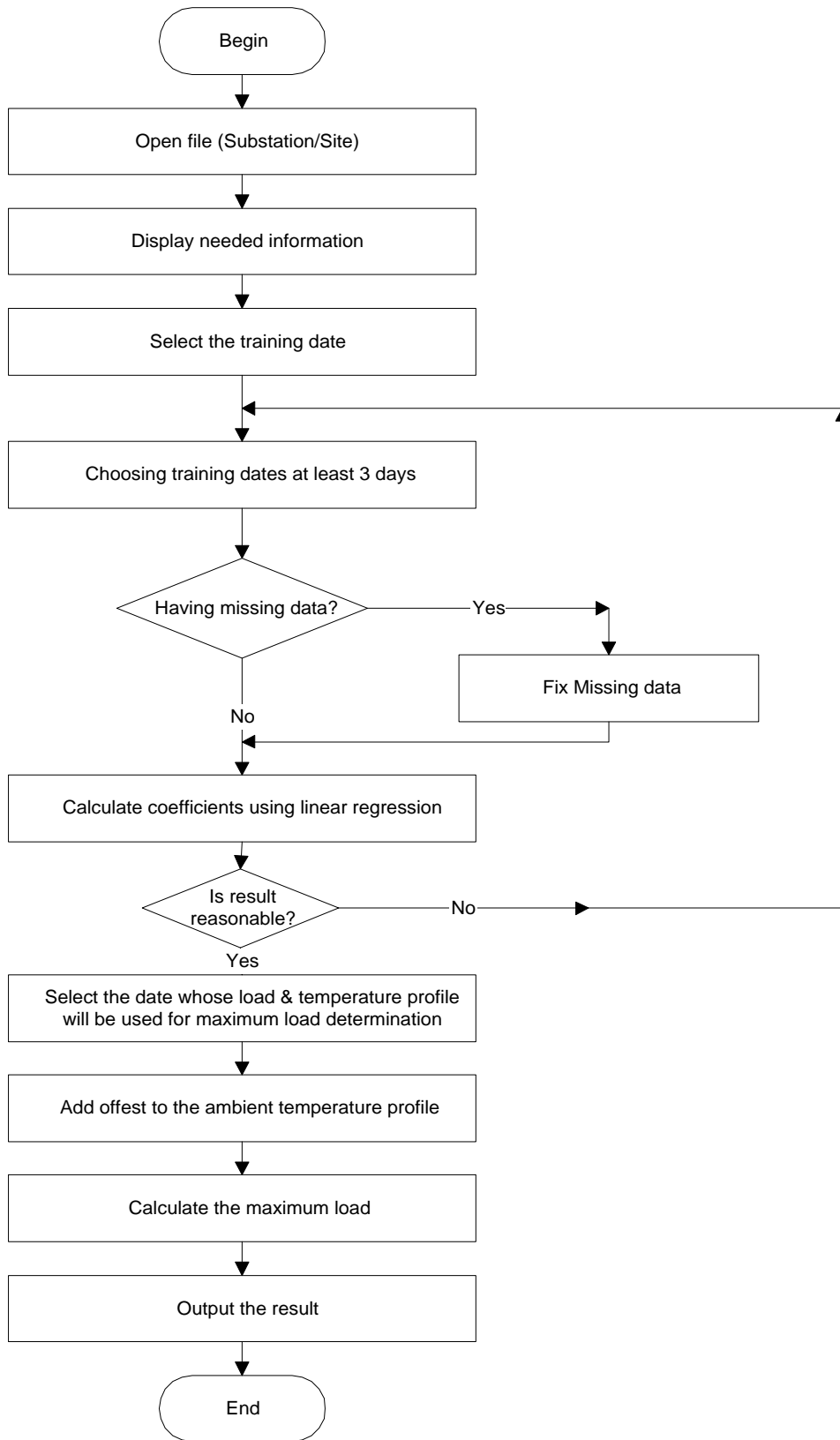
Figure 3.5 Flow chart of TOTPS
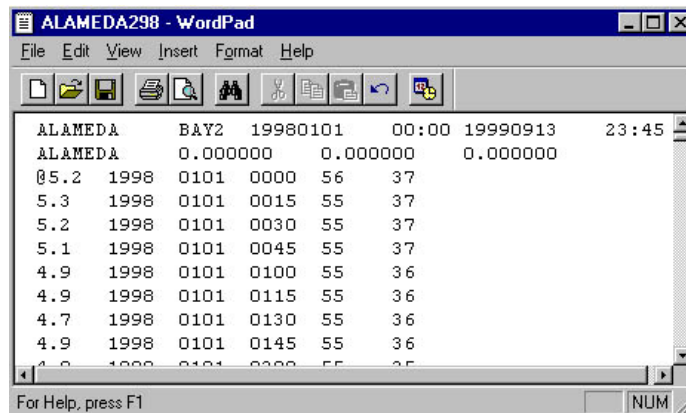
Figure 3.5 shows the flow chart of the TOTPS process.

3.6 User interface

3.6.1 Data preparation

At SRP, the load, ambient temperature, and TOT for each substation distribution transformer are monitored and recorded in the EMS (Energy Management System) on a UNIX platform. They are transferred to a database as a text file annually.

The structure of the data set files is formed by a description file header and the values of load, ambient temperature and top-oil temperature as shown in Figure 3.6.

Figure 3.6 Alameda substation data file

The definition of each variable we assigned to the data in the data file, its type and an example consistent with the data set shown in Fig. 3.6. is shown in Table 3.1.

Table 3.1. Definition of input data file

Variable	Example	Variable Type
Substation Name	Alameda	String
Bay Name	Bay2	String
Initial Date	19980101	String
Initial Time	00:00	String
Final Date	19990913	String
Final Time	23:45	String
Substation for Ambient Temperature Measured	Alameda	String
Load	5.2	Float
Year	1998	String
Date	0801	String
Time	000	String
Ambient Temperature	56.0	Float
Top Oil Temperature	37.0	Float
TGFL	45.5	Float
M1	0.8	Float
M2	0.9	Float
M3	1.3	Float

Note that some of the data:

TGFL: no-load loss at rated load,

M1: self-cooling mode 'm' coefficient for calculating HST,

M2: first stage cooling mode 'm' for calculating HST,

M3: second stage cooling mode coefficient 'm' for calculating HST,

are not containing in the data file. These data must be obtained from SRP transformer

reports.

There are two different kinds of data in the input ".txt" data file: one is the header,

including the description of the substation, transformer name and parameters; the other is

the raw data of load (measured in MVA), ambient temperature (measured in Fahrenheit),

top-oil temperature and hot-spot temperature (measured in Centigrade/Celcius).

LabVIEW cannot automatically distinguish the header and raw data using the I/O file

function. In order to separate these two parts, SRP added a special character '@' to the

end of first part, just before the first load value of second part. The program looks for

this special character and, if found, stores the strings before and after '@' into two

different arrays for later use.

3.6.2 Data initialization

There are certain functions that our software performs before the user pushes the first

button. First of all, several TOTPS software variables must be initialized to respective

default values. The Date is set to Jan. 20, all the coefficients of semi-physical top-oil

model are set to 0.0. The fields of 'Initial Date', 'Final Date', 'Initial Time', 'Final Time' and 'Base' are made blank. The TOT limit value is set to 110.00 ^oC and HST limit value is set to 150.00 ^oC. Only the 'Open File' and 'Get Data' buttons are shown to the user, the others are made invisible. The 'Temperature Offset' default value is set to 10.00.

3.6.3 Open file (Substation/Site)

The first function users must perform is to open the data file from which they wish to return data. To do this, they must click on the button displayed in the 'Substation/Title' control box (see Figure 3.3). This will open a Windows directory window that the user can use to locate the designated site. Normally all the data files are stored on the data directory of a local hard drive. The user must select an existing data file whose type is defined as a 'text document', as Figure 3.7 shows,
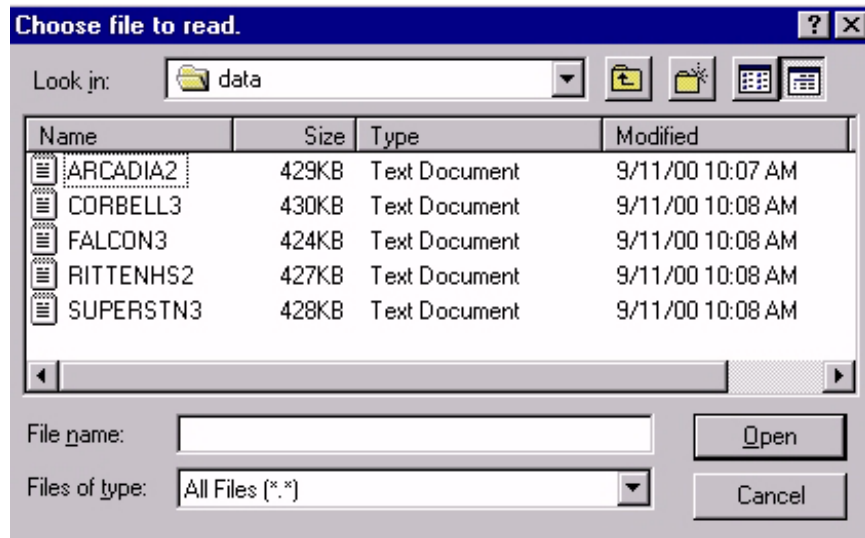
Figure 3.7 Choose file to read

If the user tries to move on to the next step (selecting a date) before opening a data file, a dialog box (Figure 3.8) will pop up to remind him to first select an input data file. After selecting the file, another dialog box (Figure 3.9) will appear asking him if he want to replace the "temp.txt" file. The user should hit the "Replace" button to go to next step.
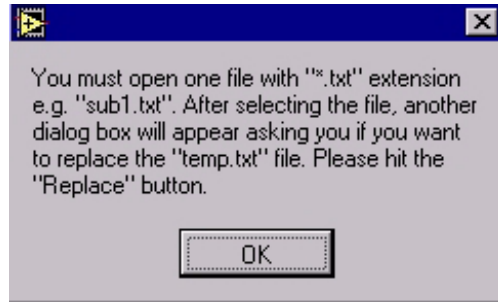
Figure 3.8 Message reminds opening a file
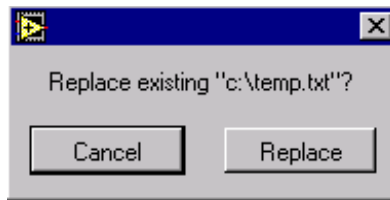
Figure 3.9 Dialog for the replace button

3.6.4 Get data

To choose the date for training, the 'Get Data' entry box is used. In this box the user specifies the Month in the range of 1 to 12 (representing January to December), the Day in the range of 1 to 31 and the Year (using four digits). Figure 3.10 shows the structure rings of labVIEW, which are special numeric objects used to make these selections.

Figure 3.10 Choosing data

If the user selects a date beyond the initial date and final date associated with date in the data file, an error message will pop up (Figure 3.11) to indicate that the selected date is not between the initial date and final date. In this case, the user has to choose another date.
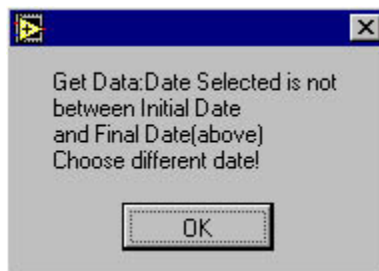
Figure 3.11 Error message

Once the 'Get Data' button is pushed, load, TOT, ambient temperature and HST data will be displayed for a 10 day period centered on the date selected as shown in Figure 3.13. For example, if you select 10/15/1998, you will see data displayed for dates from 10/10/1998 to 10/19/1998 (Figure 3.12). It includes 5 days before and 4 days after the

selected date. The plots in the middle of the screen will display the load, ambient

temperature (AMB) and top-oil temperature (TOT) curves of that 10 days (Figure 3.13).

Figure 3.12 Display 10 days' date
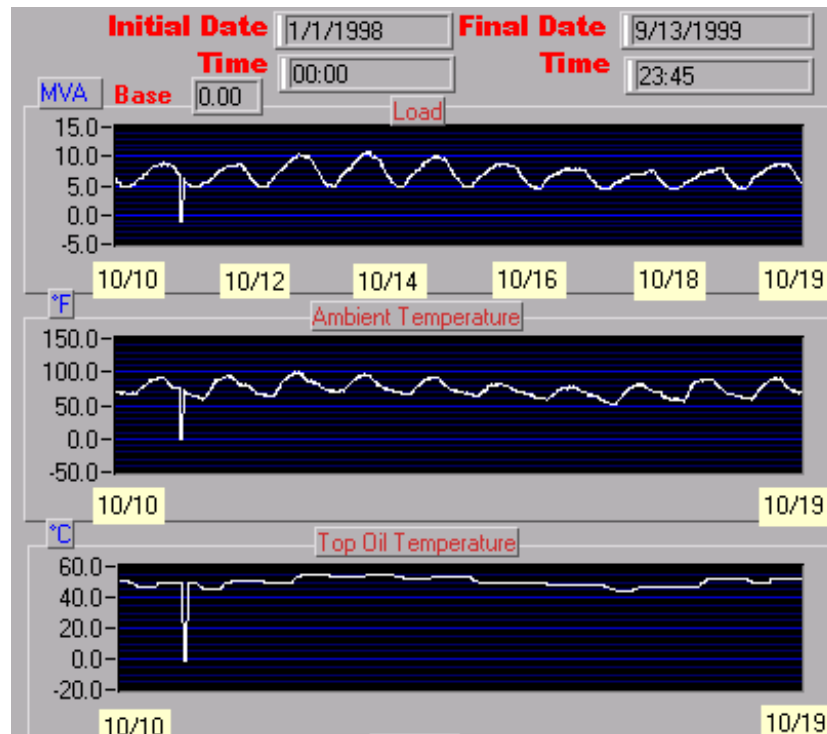
Figure 3.13 Plot for the Load, Amb and TOT

Our model does not incorporate sophisticated calendar development. For example, we have not included a leap year checking function in the program. So if the user chooses February 29 and unfortunately that year is not a leap year, a problem will occur when the program tries to read the data file, and all the displays will be set to 0.

3.6.5 Choosing training data

The user chooses training data by clicking the 'radio button' in the 'Choosing Training Data' entry box. The color of the radio button will change from gray to black if the user clicks on it, as shown in Figure 3.14. (On the screen, the radio button changes from gray to orange.)
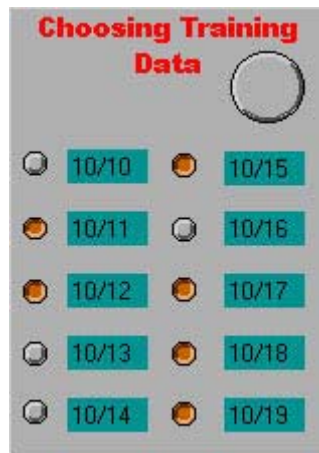
Figure 3.14 Pick 6 days for training

It was the conclusion of SRP that the user should be given the responsibility for choosing the training data for the semi-physical model. Our research shows that 3 days is the minimum time period to get optimal coefficients that can accurately predict TOT. The

user must choose at least 3 dates to get the optimal coefficients, otherwise TOTPS will

pop up an error message (Figure 3.15).

Figure 3.15 Error message for at least 3 dates

In order to give users a friendly user interface, we put four different kinds of input and

calculated data on the same plot with different colors (Figure 3.16). In the display area,

the green curve shows the dates the user has chosen. In this way, users can distinguish

which data they have picked for the training. Table 3.2 shows the color specifications,

for the four plots of TOTPS from top to bottom.
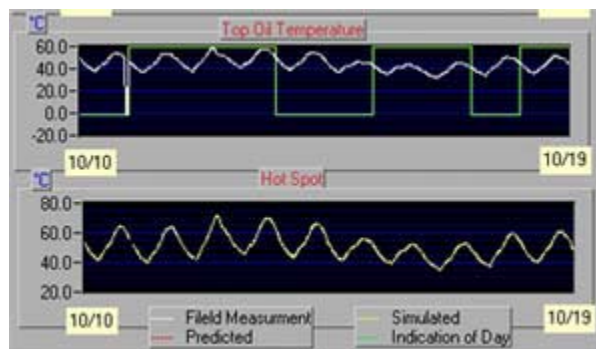
Figure 3.16 Display date of TOT, predicted H.S.

Table 3.2. Different color indication

Color	Load	Amb	TOT	H.S.T	Data Type
White	X	X	X	X	Input
Yellow				X^*	Simulated
Red	X	X	X	X	Predicted
Dark Blue	X	X	X		Limit

$*$Note that the HST data is simulated using the algorithm described in chapter 2 only when no HST data is

available in the input data file.

3.6.6 Fix missing data

When the computer sensors are measuring data, some unexpected events may interrupt the sampling of data, such as power shut down, communication block etc. When this happens, the equipment will record a value of –1 instead of the true value; hence a –1 in the data indicates bad data.

It is preferable that the user chooses good data for model training, but sometimes there will be a requirement to use a specified date's data, which could contain bad data. If such dates are selected, we have shown that the algorithm will give acceptable results if we 'fix' the missing (bad) data using linear interpolation. Table 3.3 shows a simple example of corrected data that results from interpolating.

Table 3.3 Linear interpolate

Point	1	2	3	4	5	6	7
Data value before	10	-1	-1	-1	-1	-1	22
Data value after	10	12	14	16	18	20	22

Figure 3.17 and Figure 3.18 show the results of fixing missing data. Note that the scale of plot changes in Figure 3.18 because of fixing the missing data with linear interpolate.
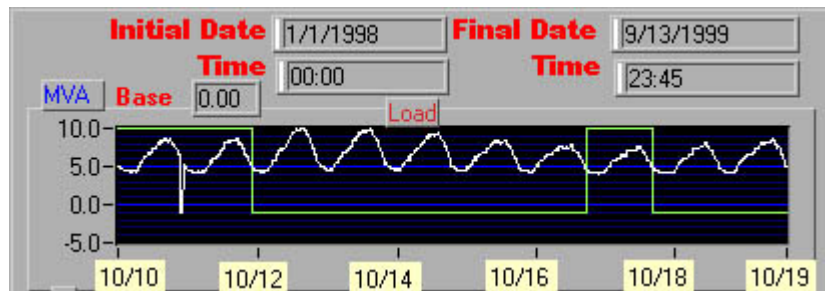
Figure 3.17 Load before fixing missing data
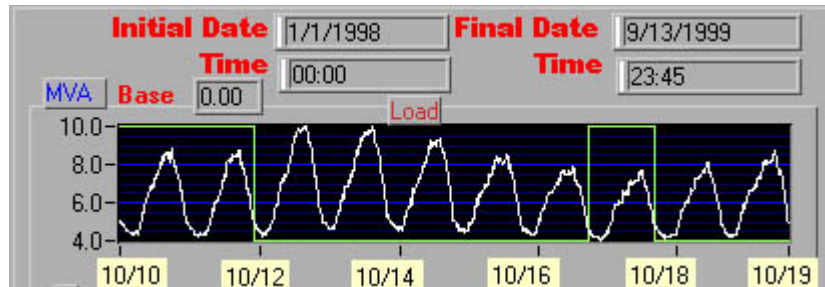
Figure 3.18 Load after fixing missing data (Note the change in scale.)

Not all of the missing data needs to be fixed. Data only needs to be fixed if the missing data occurs in the days selected. To fix missing data the user must simply click on the button in the 'Fix Missing Data' entry box. If missing data does not occur during the

dates selected, it is not necessary for the user to click the 'Fix Missing Data' button. However, if missing data does occurs at other (non-selected) dates, clicking the 'Fix Missing Data' button can change the scale and give the user more resolution as described earlier. If missing data is not fixable in the selected days, the user must return to the previous step to choose other training data; otherwise an error message will pop up (shown in Figure 3.19) when the user tries to calculate model coefficients with the 'Calculate Coefficients' button
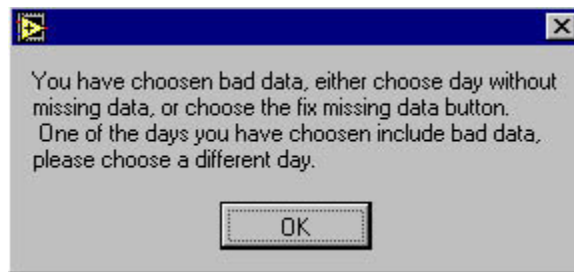
Figure 3.19 Remind for the bad data chosen

3.6.7 Calculate coefficient

After choosing the data for training and fixing the bad data (if desired and/or necessary), the linear regression algorithm discussed earlier can be used to calculate the 'k' coefficients of equation (3.1).

$$TOT(k+1) = k_1 * Load(k+1) + k_2 * Amb(k+1) + k_3 * TOT(k) + k_4 \qquad (3.1)$$

The user can get these coefficients by clicking the button in the 'Calculate Coefficient' program control box. The predicted top-oil temperature based on the input load, ambient temperature and calculated coefficients are also plotted on the same set of axis (in

yellow) that display the input TOT. The prediction of TOT is usually extremely accurate, the input TOT and predicted TOT are almost the same as shown in Figure 3.20.
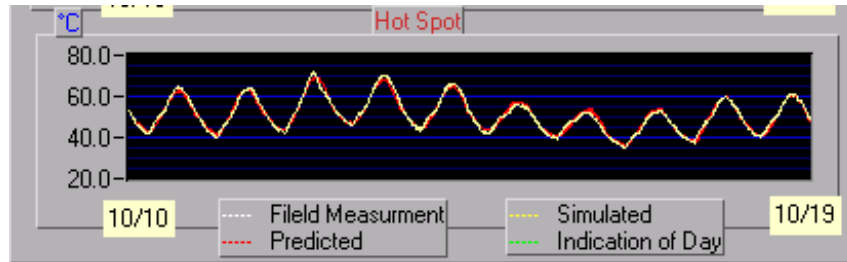
Figure 3.20 Calculate coefficient

3.6.8 Get the load/temperature profile

After the coefficients are obtained, the user can choose the date for the load and temperature profiles that he wants to use to calculate maximum transformer load. He can get the predicted ambient temperature profile from historic data. We give users the freedom to adjust the temperature profile (Figure 3.21). For example, the temperature for the same day of next year may be 10 degrees higher. To adjust the temperature profile the user must simply chooses a numeric number (could be negative) and click on the button in the 'Temperature Offset' entry box.
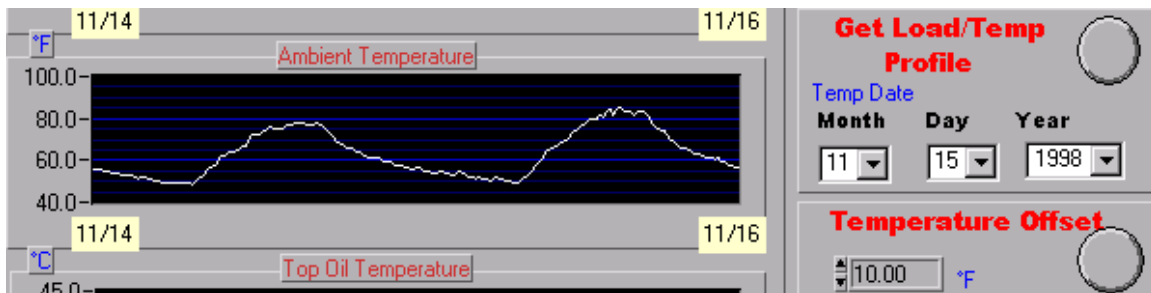
Figure 3.21 Get ambient temperature profile

3.6.9 Calculate the maximum load

After getting the load and ambient temperature profiles of selected date, TOTPS uses the iteration algorithm described in Chapter 2 to calculate the maximum load that this transformer can endure without exceeding the operation limits. The TOT limit is 110 ^{o}C. The HST limit is 150 ^{o}C. Both of these limits may be changed by the user using the lines in the 'Cal. Max Temp' box as shown in Figure 3.22.   To calculate the maximum load the user must simply click on the button in the 'Cal. Max Temp' entry box shown in Figure 3.22.

Figure 3.22 Red button for maximum value

Once the maximum load is obtained, TOTPS will display the maximum load, ambient temperature and TOT curves (Figure 3.23). The white color curve represents the current selected load, ambient temperature, TOT and HST profile, the red color curve represents the predicted maximum load, TOT, HST profile. The dark blue color curve represents the TOT limit and HST limit.
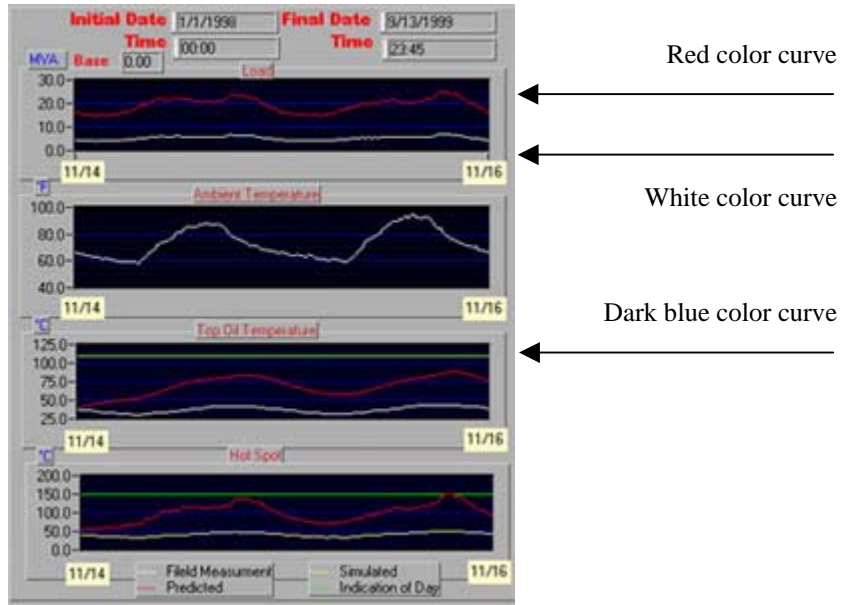
Figure 3.23 Get maximum temperature

The maximum load limit can be dictated by either the TOT or HST limits. (The example shown in Figure 3.22 is one in which the maximum load was limited by the HST limit.) The limit (HST or TOT) that reaches first, as the load is increased, is indicated by the 'radio button' in Figure 3.22. The one that turns red marks the thermal limit.

4. ASSESSING TOTPS RELIABILITY

4.1 Introduction

We built the TOTPS program with LabVIEW software using the semi-physical model to predict the transformer's top-oil temperature. We got good predictions from a wide range of coefficients by using TOTPS. For example, the load coefficient, k_1, ranged from about –0.5 to 8.0 for ten different SRP transformers; even though this range was large, the TOT prediction obtained using these coefficients was good and the maximum error between calculated TOT and predicted TOT was less than $\pm 3^o C$, which was the criterion set by SRP for the good prediction. Each substation had similar transformers under similar operating conditions; hence we expected to get similar k_1 coefficients, not the wide range of k_1 coefficients we obtained. We hypothesized that there were several possible causes that led to this wide range of k_1 coefficients: quantization noise, other noise and absence of significant driving variables. In this chapter, we investigate the effects on the k_1 coefficient of quantization noise and other noise added to the driving variable.

4.2 Semi-physical model unreliable coefficients

An example of the range of k_1 coefficients we obtained during 9 consecutive 10-day intervals on a single transformer is shown in Table 4.1 (a). Table 4.1 (a) shows a list of coefficients calculated by a linear regression algorithm for different data sets taken from

the same transformer (Pringle substation bay 3). Table 4.1 (b) shows a list of coefficients

for 10 similar but different transformers.

Table 4.1 Coefficients list

Day	k1	k2	k3	k4
01-10	-0.2285	-0.0004	0.8333	7.8057
11-20	0.1527	0.0010	0.9977	0.0430
21-30	4.9471	-0.0002	0.9722	0.0629
31-40	0.5939	0.0045	0.9847	0.2680
41-50	-0.1299	0.0005	0.9940	0.2076
51-60	0.0686	0.0021	1.0005	-0.1341
61-70	-0.6978	0.0018	0.9991	0.0149
71-80	0.1047	0.0009	0.9892	0.4021
81-90	0.1789	0.0001	0.9549	1.7676

(a) Pringle substation bay 3 coefficients

Substation	k1	k2	k3	k4
alemeda1	4.1792	0.0293	0.9416	0.0448
arcadia2	1.8429	0.0209	0.9582	-0.0496
collier2	0.2322	0.0026	0.9844	0.3766
corbell3	1.4172	0.0194	0.9576	0.1771
falcon3	1.9766	0.0281	0.9468	-0.1884
fountain2	3.8987	0.0284	0.9325	0.6973
kay2	2.5245	0.0259	0.9476	0.1635
pringle3	-0.2285	-0.0004	0.8333	7.8057
sheely1	-0.5028	0.0059	0.9783	0.5261
sunlakes2	8.0832	0.1182	0.7769	0.8206

(b) List of coefficients for different substation

One source of noise that we thought might cause the large k_1 range was the TOT

measurement error caused by old-style mechanical temperature transducers. Earlier

research [1] showed that a type of hysteresis would create problems when we tried to

apply the algorithms to transformers. To avoid this problem, we limited our analysis to

data taken from the substations whose data were generated from solid-state temperature

transducers; hence the range of coefficients displayed in table 4.1 cannot be caused by the

hysteresis of mechanical temperature transducers.

One possible cause of the wide range of k_1 coefficients, we hypothesized, might be due to quantization measurements and other noise.

4.3 Quantized and non-quantized data

In the semi-physical model of the transformers that were studied, the optimal coefficients we calculated were based on measured data: TOT, ambient temperature and load. The measured TOT and ambient temperature data that was available for the majority of the duration of this research was rounded to two significant digits, load (measured in MVA) was rounded to three significant digits. We refer to this as 'old-style' data. More recently SRP has been updating their EMS system and the data logged by the new EMS system has 4 digits of precision. This data we refer to as 'new-style' data. Table 4.2 and Table 4.3 show two typical data segments from the old-style and the new-style data.

Table 4.2 Old-style data

Load	Year	Data	Time	Tamb(F)	TOT(C)
3.9	1998	0101	0015	55	37
3.8	1998	0101	0030	55	37
3.8	1998	0101	0045	55	36
3.8	1998	0101	0100	55	36
3.8	1998	0101	0115	55	36
3.8	1998	0101	0130	55	36

Table 4.3 New-style data

Load	Year	Data	Time	Tamb(F)	TOT(C)
3.8716	1998	0101	0015	54.6712	37.4861
3.8123	1998	0101	0030	54.9073	36.9368
3.7821	1998	0101	0045	55.0129	36.4567
3.8231	1998	0101	0100	55.0794	35.6877
3.8251	1998	0101	0115	54.8811	36.3689
3.7627	1998	0101	0130	54.5864	36.1392

The old-style data's TOT is rounded to the nearest integer (such as 65) and the new-style data's TOT is rounded to 4 decimal digits (such as 64.5318). To estimate the effects of using greater precision on the range of k_1 value obtained from similar transformers under similar operating condition, we asked SRP to supply us with old-style and new-style data for a transformer, both measured over the same time interval. Tables 4.4 and 4.5 show that the range, over which the k_1 coefficient varies, decreases from

$(k_{1\min}, k_{1\max}) = (0.3992, 2.5907)$ for the old-style data to $(k_{1\min}, k_{1\max}) = (0.9395, 2.3027)$ for the new-style data.

Table 4.4 Alameda1 old-style (quantized) data

Day	k1	k2	k3	k4
01-10	2.5907	0.0162	0.9659	0.1921
11-20	1.4386	0.0187	0.9560	0.5012
21-30	1.6416	0.0170	0.9633	0.2492
31-40	0.4979	0.0204	0.9570	0.3441
41-50	0.3992	0.0198	0.9500	0.6405
51-60	1.2813	0.0194	0.9635	0.0984

Table 4.5 Alameda1 new-style (non-quantized) data

Day	k1	k2	k3	k4
01-10	2.3027	0.0156	0.9726	-0.0355
11-20	1.1995	0.0182	0.9619	0.2911
21-30	2.0906	0.0150	0.9723	0.0220
31-40	1.0586	0.0182	0.9667	0.1000
41-50	0.9395	0.0187	0.9610	0.2953
51-60	1.4472	0.0180	0.9738	-0.1804

Unfortunately, the range of k_1 values obtained from the new-style data is still too large to be useful in predicting transformers' peak loading. We hypothesized that one possible cause of the wide range of k_1 values could be noise in the measured data.

4.4 Simulated data with noise

Our experiments with large sets of real data from SRP showed that our model was good for predicting the TOT, but the optimal coefficients we got varied over an unacceptably wide range. We suspected that this may be due to the noise portion of the driving variables. In order to study the effects of noise added to the driving variables, we generated a set of simulated data for testing by using the equation,

$$TOT[k] = k_1 Load[k] + k_2 Amb[k] + k_3 TOT[k-1] + k_4 \qquad (4.1)$$

Where,

$$k_1 = \frac{\Delta t \theta_{fl} R}{(T_o + \Delta t)(R+1)},$$

$$k_2 = \frac{\Delta t}{T_o + \Delta t},$$

$$k_3 = \frac{T_o}{T_o + \Delta t},$$

$$k_4 = \frac{\Delta t \theta_{fl}}{(T_o + \Delta t)(R+1)},$$

θ_{fl} : top-oil rise over ambient temperature at rated load, °C,

T_o : time constant at rated KVA (hr.),

R: ratio of load loss to no-load loss at rated load,

Δt : the sampling period.

We selected the values of parameters T_o, θ_{fl}, R from the transformer test report: $\theta_{fl} = 50$, $T_o = 5$, $R = 10$ and $\Delta t = 0.25$. Using these values, the simulated coefficients we obtained were the following:

$k_1 = 2.1645$,

$k_2 = 0.0476$,

$k_3 = 0.9524$,

$k_4 = 0.2165$.

We picked the load, initial value of TOT and ambient temperature from a real data set (Arcadia substation), then calculated the theoretically exact result (to within the error of the Forward Ruler rule used in (4.1)) for TOT as shown in Figure 4.1.

Simulated Data

Figure 4.1 Simulated data by simulator

We then tested our simulation and regression algorithms by putting the simulated data into our semi-physical model linear regression algorithm. Our linear regression algorithm gave us the optimal coefficients shown in Table 4.6. As should be the case, we got exactly the same results as the simulated coefficients for all 10-day time intervals. This proved to us that our simulator and regression algorithm were working correctly.

Table 4.6 Results for simulated data

Day	k1	k2	k3	k4
01-10	2.1645	0.0476	0.9524	0.2165
11-20	2.1645	0.0476	0.9524	0.2165
21-30	2.1645	0.0476	0.9524	0.2165
31-40	2.1645	0.0476	0.9524	0.2165
41-50	2.1645	0.0476	0.9524	0.2165
51-60	2.1645	0.0476	0.9524	0.2165

To simulate the effects of noise, we added random noise (uniformly distributed over a fixed interval) to the simulated data. We added random noise separately to load, ambient temperature and TOT, then used linear regression to calculate the semi-physical model coefficients. We conducted this study using several different noise amplitudes. Our results, shown in Table 4.7, showed that our algorithm was insensitive to noise in the load and ambient temperature variables but that even small amounts of noise added to TOT would cause a significant change in the coefficients.

Note that the noise amplitude in Table 4.7 is specified as the percent of maximum magnitude of the signal to which the noise was added. For example, if the maximum TOT value was 50 ^{o}C and the noise amplitude was 1%, then the random noise added to TOT was between –0.5 and 0.5.

Table 4.7 Coefficients after adding noise

Noise =1%

	k1	k2	k3	k4
Without Noise	2.1645	0.0476	0.9524	0.2165
Load Noise	2.1540	0.0476	0.9523	0.2177
Tamb Noise	2.1537	0.0476	0.9524	0.2166
TOT Noise	2.0658	0.0476	0.9524	0.2164

Noise =2%

	k1	k2	k3	k4
Without Noise	2.1645	0.0476	0.9524	0.2165
Load Noise	2.1379	0.0476	0.9522	0.2206
Tamb Noise	2.1429	0.0476	0.9524	0.2167
TOT Noise	1.5640	0.0476	0.9524	0.2168

Noise =4%

	k1	k2	k3	k4
Without Noise	2.1645	0.0476	0.9524	0.2165
Load Noise	2.0459	0.0471	0.9512	0.1904
Tamb Noise	2.1023	0.0470	0.9537	0.1845
TOT Noise	0.3329	0.0636	0.9273	0.6219

4.5 Conclusion

The above research showed that in order to improve the reliability of optimal coefficients

of the semi-physical model, we must use the new-style data and choose data from

substations whose data were generated from solid-state temperature transducers. We also

showed that our algorithm was insensitive to noise added to load and ambient

temperature measurement, but was very sensitive to noise added to TOT. But we still can

not solve the problem of unreliable coefficients, so we thought it was due to the absence

of driving variables.

The wide range of k_1 coefficients that we get from our algorithm is still too wide to allow

us to use our results to extrapolate (maximum transformer load), although they are

sufficient for interpolating (i.e. predicting TOT values) for conditions similar to those used in obtaining the model coefficients.

The unacceptable range of k_1 coefficients obtained from our algorithm must be due to one of two causes. Either the noise in the measured (driving) TOT variable is too large or we are missing some significant driving variable from our model.

5. RADIATION AND WIND VELOCITY

5.1 Introduction

The semi-physical model is a good predictor of transformer top-oil temperature, however the semi-physical model yields a wide range of coefficients for data that appears good to the eye. One reason for that wide range of coefficients may be the lack of driving variables in our model. In this chapter, we investigate the effects of wind velocity and radiation as driving variables on the reliability of coefficient estimation.

5.2 Semi-physical model

5.2.1 Prediction and planning

Figure 5.1 shows a typical example of the TOT error (defined as the difference between actual TOT value and predicted TOT value by the semi-physical model) using the calculated coefficients of load and ambient temperature. These results were obtained using non-quantized tier-one data from the Alameda substation. Tier-one data is limited to cooling due to natural convection in the oil and core. (If we try to model all three tiers of cooling (OA/FA, FA, FOA) using one set of k_1 coefficients, the errors are not nearly as low.) The training data set included 10 days of data as shown in Figure 5.2. The error shown is for the training data. SRP has defined acceptable error as $\pm 3^0 C$. Since the

maximum error in this plot is only 1^oC, (which accounts for about 5% of the TOT) prediction results like these are acceptable to SRP.

Semi-physical model training data prediction
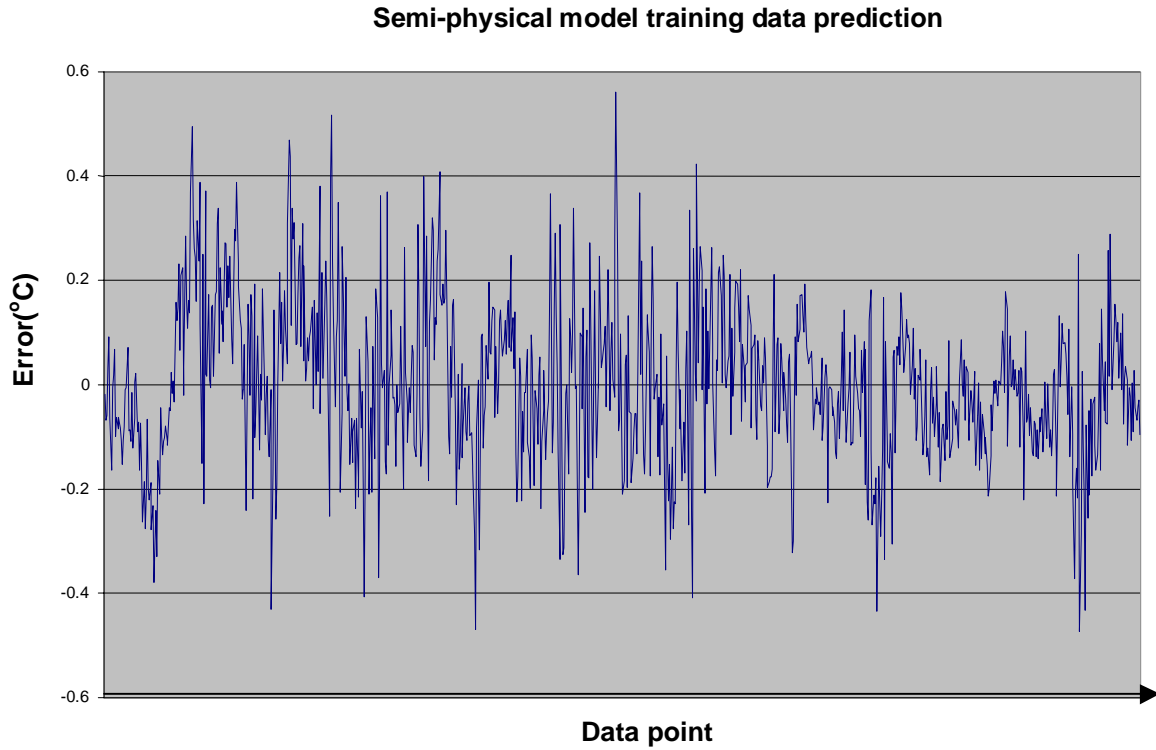
Figure 5.1. Semi-physical model training data prediction

Then we chose another 10 days of data (testing data) to predict the TOT value by the semi-physical model using the calculated coefficients of training data. The testing data set included 10 days of data as shown in Figure 5.3.

10 days' training data
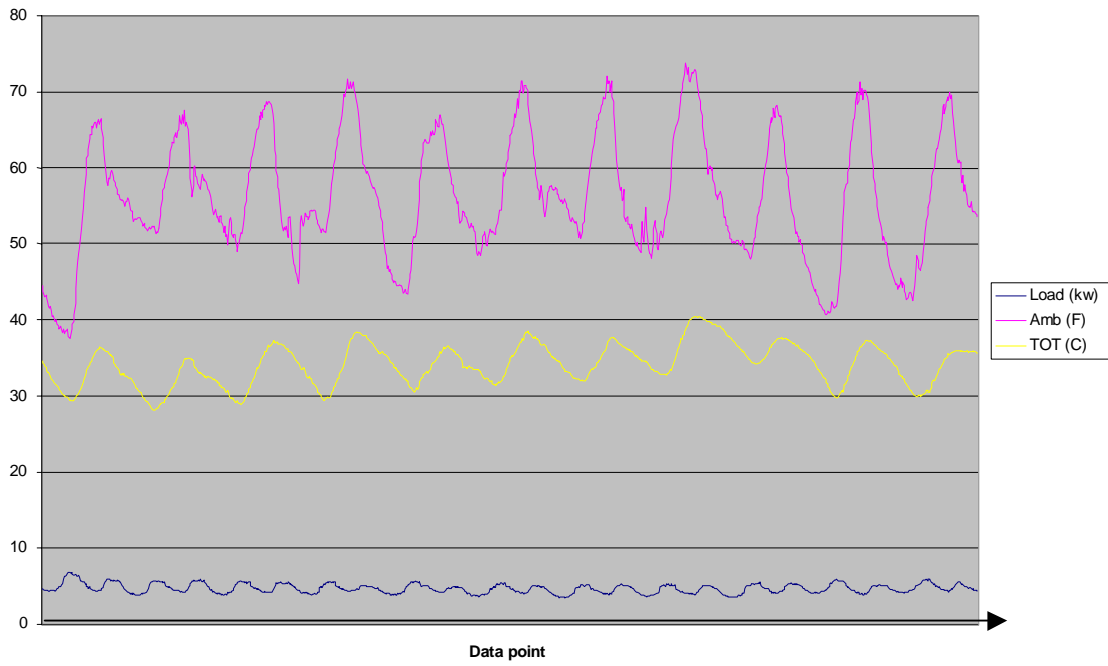
Figure 5.2 10 days of training data

10 days' testing data
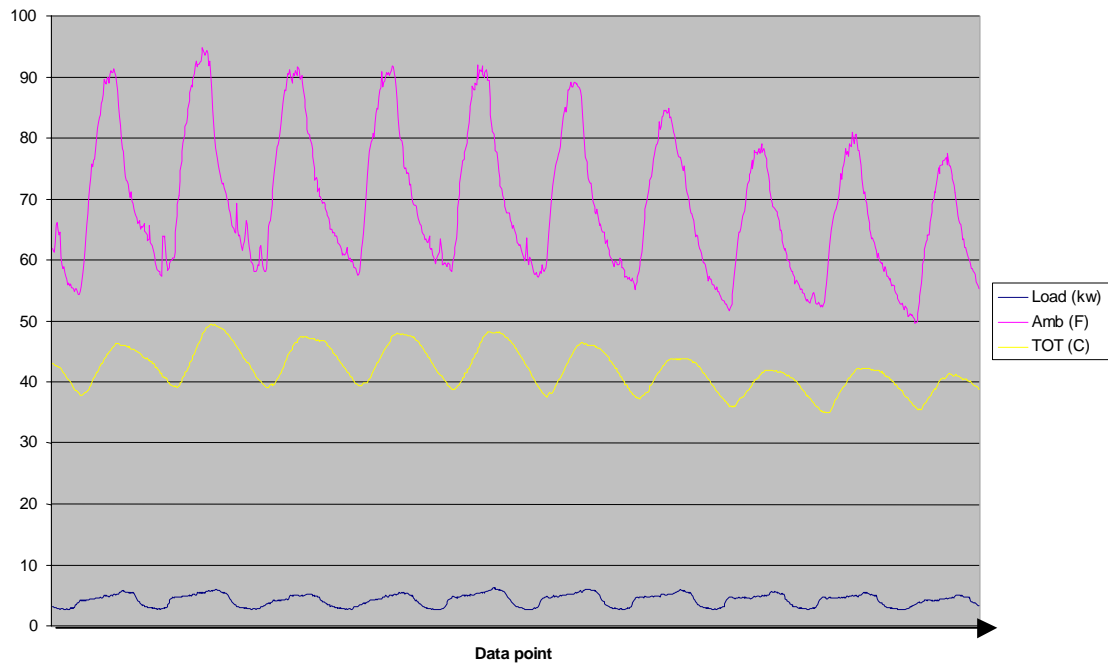
Figure 5.3 10 days of testing data

Figure 5.4 shows the TOT error (defined as the difference between actual TOT value and predicted TOT value by the semi-physical model) of testing data using the calculated coefficients of training data.

Semi-physical model testing data prediction
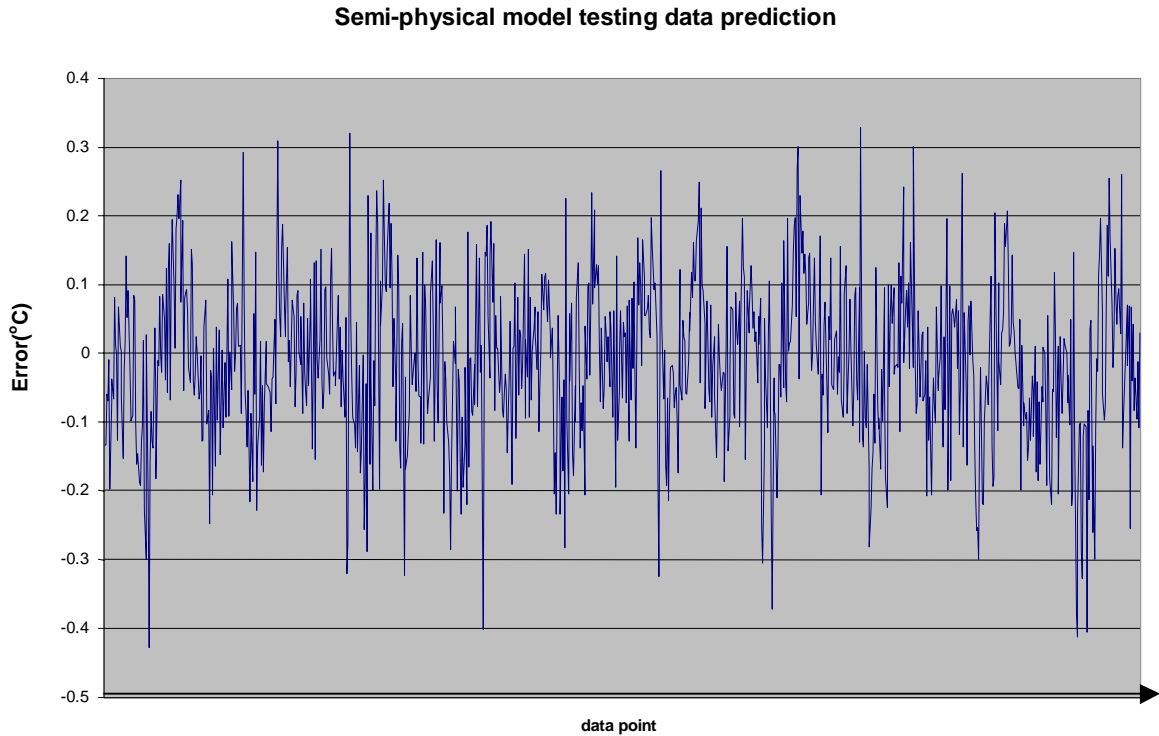
Figure 5.4 Semi-physical model testing data prediction

Comparing the results from Figure 5.1 and Figure 5.4, we can see that the semi-physical model is a good predictor of transformer top oil temperature.

The results of Figure 5.1 and Figure 5.4 were generated using the TOTPS tool. When we ran the TOTPS tool again and again for different transformers and different training data, we found that the calculated coefficients would vary greatly and sometimes the k_1 coefficient would be negative. The negative k_1 coefficient implies that as the load of the transformer increases, the transformer gets cooler. This is clearly a non-physical result.

The truth is, the more load losses consumed by a transformer, the higher the TOT will be. We decided to determine whether the prediction error increases as the k_1 coefficients vary. We used the real data, along with the semi-physical model and obtained a wide range of k_1 values. We then plotted the peak value of the TOT prediction error (for the various k_1 values we obtained) versus k_1. Error was defined here as the difference between the actual TOT value and the predicted TOT value. These errors for the k_1 coefficients obtained for one substation (Alemada) are shown in Figure 5.5. We used the above k_1 coefficients and the real testing data (shown in Figure 5.3), along with the semi-physical model and then plotted the peak value of the TOT predication error versus k_1. These errors for the k_1 coefficients are shown in Figure 5.6.
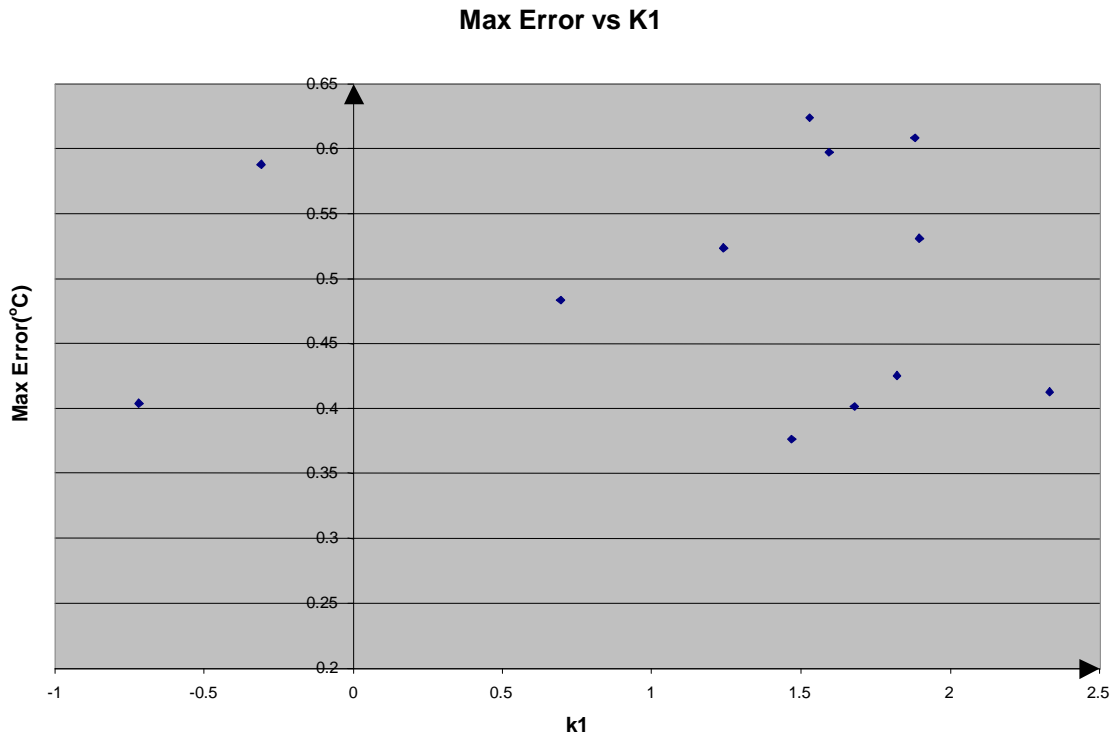
Max Error vs K1

Figure 5.5 Maximum error vs K1 (training data)
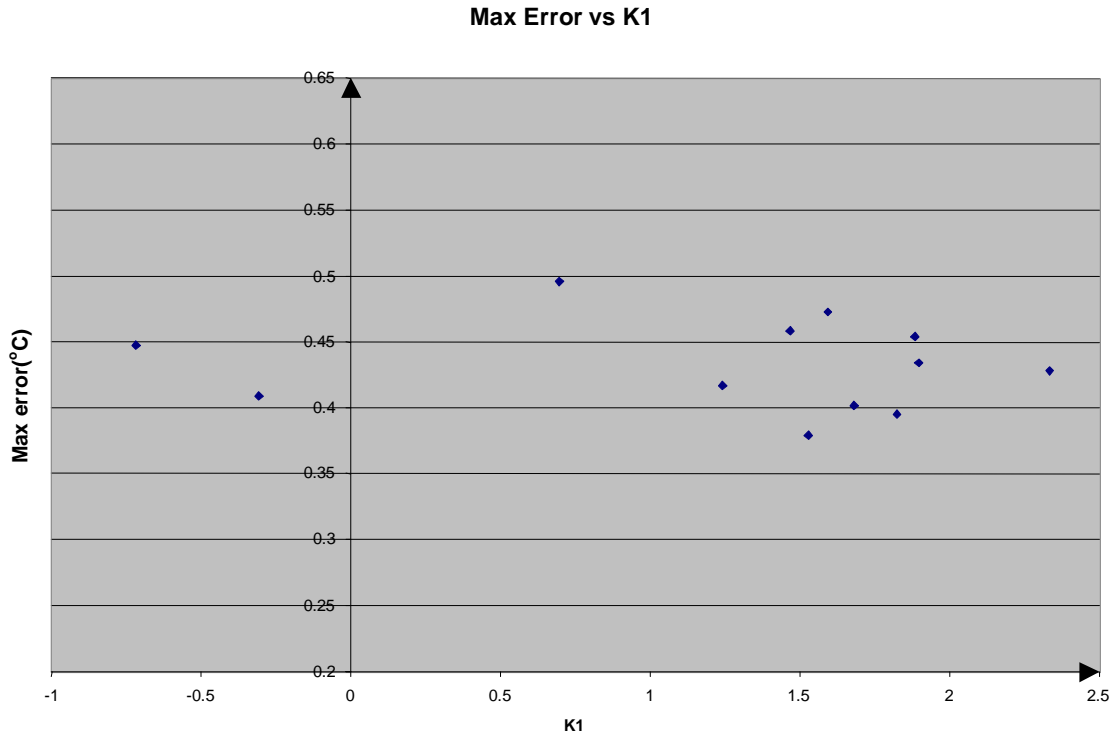
Max Error vs K1

Figure 5.6 Maximum error vs K1 (testing data)

Although the k_1 coefficient vary widely, the maximum error of TOT is within the acceptable range ($\pm 3^0 C$), which indicates that the prediction of TOT is acceptable. Extrapolation of the peak load with such a wide range of k_1 coefficients will however produce widely varying and unusable results.

We expected that regardless of what methods we employ to limit the range of k_1 coefficients, there would always be some variability in this, and the other coefficients. We wanted to determine what range of k_1 coefficients was acceptable, so that we have a target to shoot for in our research. In discussions with SRP we determined that if we can reliably predict maximum load to within $\pm 5\%$ using various sets of training data, then

our algorithm would be useful to in a production mode. Because peak load prediction is

dominated by the k_1 (load) coefficient, we conducted a study of the effect of varying k_1

on peak load predicted by our algorithm. The result of this study, shown as a plot of peak

load versus k_1, is shown in Figure 5.7. This figure shows that, for a transformer whose

true k_1=1.816, the peak load predicted by our algorithm will vary by $\pm 5\%$ provided the

k_1 varies by no more than $\pm 10\%$. Our conclusion is that if we can reliably predict k_1 to

within $\pm 10\%$, we should be able to give meaningful peak load predictions.
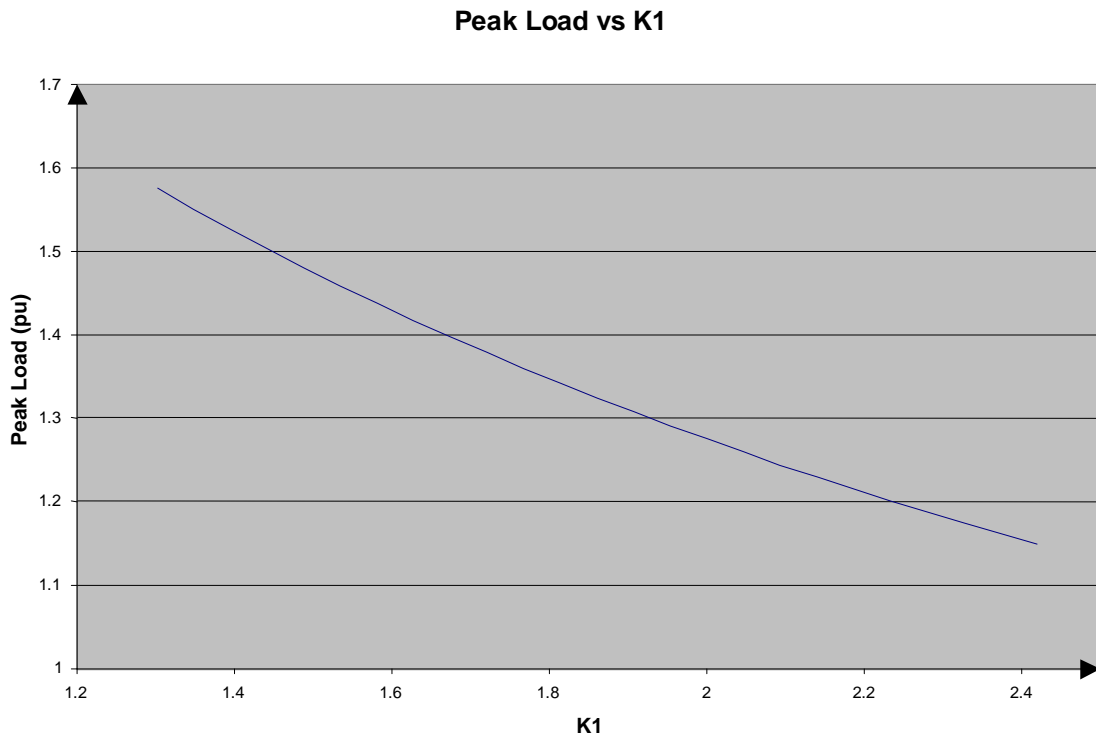
Peak Load vs K1

Figure 5.7 Peak load vs k_1

In this study, we only change k_1, while keeping $k_2 - k_4$ constant. We know from our

work that when we use different data sets, all coefficients obtained from linear regression

vary; hence our target of achieving a $\pm 10\%$ change in k_1 is taken as an approximate

target. Once we are getting more reliable sets of coefficients, we will need to reassess whether this range is acceptable.

5.2.2 Matrix analysis

There may be several causes that create the wide range of coefficients from apparently similar data sets. One cause is rooted in the numerical operations used in finding the model coefficients. Referring to the definition of the solution of the linear regression problem, (2.14), you will see that in order to get the model coefficients we need to solve a problem of the form Ax=b, where A is the W matrix and $b = X^T Y$. For a typical 2x2 matrix A, which has the elements,

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix},$$

the inverse matrix of A is given by,

$$A^{-1} = \frac{1}{a_{11}a_{22} - a_{12}^2} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{12} & a_{11} \end{bmatrix}.$$

The sensitivity of x to the A matrix coefficients is given by,

$$x_1 = \frac{1}{a_{11}a_{22} - a_{12}^2}(a_{22}b_1 - a_{12}b_2).$$

If the A matrix is ill conditioned, small variations in the entries of A or b will lead to large changes in the solution for x. Condition number is used to indicate whether the calculated matrix is ill conditioned or not. The L-2 norm condition number is the ratio of the largest singular value of matrix to the smallest. Large condition numbers indicate an ill-conditioned matrix. Figure 5.8 shows the matrix and the condition number of matrix.

$$W = 10^4 * \begin{bmatrix} 0.00366067 & 0.09540826 & 0.11520432 \\ 0.09540826 & 3.55900006 & 3.47906712 \\ 0.11520432 & 3.47906712 & 6.31256265 \end{bmatrix}$$

$$X^T Y = 10^4 * \begin{bmatrix} 0.12332244 \\ 4.20098862 \\ 5.46187065 \end{bmatrix}$$

$$cond = 9260.641582$$

Figure 5.8 Condition number of matrix

When we re-scale the driving variables and divide them by their maximum values and recalculate the coefficients, the condition number decreases a lot as shown in Figure 5.9. The coefficients obtained from the natural input and scaled input are the same as shown in Table 5.1. This is to be expected since the condition number of the matrix in Figure 5.8 and 5.9 is sufficiently good to make the linear regression reliable.

$$W = \begin{bmatrix} 51.99708637 & 25.60847835 & 16.18003270 \\ 25.60847835 & 18.05117730 & 9.23321946 \\ 16.18003270 & 9.23321946 & 8.76614921 \end{bmatrix}$$

$$X^T Y = \begin{bmatrix} 17.32019393 \\ 11.14915239 \\ 5.46187065 \end{bmatrix}$$

$$cond = 23.376808$$

Figure 5.9 Condition number of re-scale matrix

Table 5.1 Comparing of scale and re-scale coefficients

Day	k1	k2	k3	k4
01-10	0.59644	0.05351	0.93946	0.46635
11-20	-2.79214	0.06186	0.92134	0.85565
21-30	-7.88257	0.06655	0.91793	1.06169
31-40	1.29597	0.05634	0.93596	0.45895
41-50	1.22133	0.06014	0.92908	0.62849
51-60	1.64116	0.05747	0.93432	0.51152

(a). Natural coefficients

Day	k1	k2	k3	k4
01-10	0.59644	0.05351	0.93946	0.46635
11-20	-2.79214	0.06186	0.92134	0.85565
21-30	-7.88257	0.06655	0.91793	1.06169
31-40	1.29597	0.05634	0.93596	0.45895
41-50	1.22133	0.06014	0.92908	0.62849
51-60	1.64116	0.05747	0.93432	0.51152

(b). Re-scaled natural coefficients from scale data

Matrix W is ill conditioned, which means the small changes in the matrix $X^T Y$ will cause a significant change in the result. Table 5.2 shows the change of k_1 coefficient in percent when there are 10% small changes in the item of the matrix.

Table 5.2 Percentage change in k_1 due to change in $X^T Y$ vector and W matrix

k1 \ Case	%Change					
	1	2	3	4	5	6
2.33316	18.081	11.809	1.533	446.184	264.246	171.938
-0.71771	12.871	5.263	0.227	356.572	1527.062	1160.49
1.68014	19.488	12.009	3.609	1571.957	870.68	691.277
1.46798	17.258	12.334	0.122	1010.668	1039.218	38.55
1.59359	10.932	1.762	0.545	80.409	382.223	472.631
1.88384	11.022	2.577	0.125	326.745	421.711	104.966
-0.3069	11.314	1.364	1.431	1883.692	3066.9	4940.592
1.52877	10.394	0.63	0.983	200.689	203.817	414.506
1.24062	9.809	0.014	0.898	262.217	96.643	368.86
0.6955	11.901	0.283	3.942	1443.377	150.839	1604.216
1.89612	9.864	0.346	0.603	296.282	76.862	229.42
1.82294	9.577	0.215	0.379	294.554	83.803	220.751

For the real 3x3 matrix W, which has the items,

$$W = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix},$$

and matrix $X^T Y$, which has items,

$$X^T Y = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix},$$

Table 5.2 shows 6 different cases:

Case1: a_{11} increase by 10%,

Case2: a_{12} increase by 10%,

Case3: a_{13} increase by 10%,

Case4: b_1 increase by 10%,

Case5: b_2 increase by 10%,

Case6: b_3 increase by 10%.

Small changes in a_{ij} values cause proportional changes in k_1 values, which small changes in $X^T Y$ values cause disproportionately large changes in the k_1 values.

5.3 Solar radiation

A significant driving variable in our model may be solar radiation. In this section we develop a solar radiation model.

5.3.1 Fundamentals of solar radiation

To account for solar radiation, we first need to be able to predict quantitatively the Sun's motion and its effect on the energy flux on a given location on the earth. Specifically, we need to understand the following aspects of solar radiation [3],

Beam radiation

Solar radiation intercepted by a surface with negligible direction change and scattering in the atmosphere. Beam radiation is also referred to as direct radiation.

Diffuse radiation

Solar radiation scattered by aerosols, dust, and by the Rayleigh mechanism, it does not have a unique direction.

Total radiation

The total of diffuse and beam radiation, sometimes is referred to as global radiation.

5.3.2 Calculation of solar radiation

Solar radiation is a significant source of heat flux in Phoenix, Arizona (which is the location of the transformers under study). Preliminary hand calculations indicate that heat flux could contribute as much heat as that produced by 25% of the full load losses of the transformer. Modeling this heat source could consequently improve the performance of the model. The simplest method of calculating atmospheric absorption of solar radiation for clear skies is using Bouger's law equation [3],

$$I_b = I_o * \tau_{atm} \tag{5.1}$$

Where,

I_b: solar radiation at the surface in the direction of the sun (the direct normal radiation),

I_0: solar constant, the average amount of solar radiation in the near earth space,

τ_{atm}: atmospheric transmittance.

For an observed substation, we first can get the solar constant I_0 by,

$$I_o = 1.353 * (1 + 0.034 * \cos(2 * \pi * n / 365)) \tag{5.2}$$

Where,

n: the date number countered from January 1.

Its measured value is $1.353\, kW / m^2$. However, since the sun-earth orbit is elliptical, the sun-earth distance varies by ± 1.7 percent during a year, and the extraterrestrial radiation also varies slightly by the inverse-square law [3].

The atmospheric transmittance τ_{atm} is given by,

$$\tau_{atm} = 0.5 * (\exp(-0.65 * m(0, \alpha)) + \exp(-0.095 * m(0, \alpha)) \tag{5.3}$$

where,

$m(0, \alpha)$: the air mass (radiation attenuation in the atmosphere) at sea level (altitude 0).

The air mass at sea level (altitude 0) is given by,

$$m(0, \alpha) = [1229 + (614 * \sin \alpha)^2]^{0.5} - 614 * \sin \alpha \tag{5.4}$$

Where,
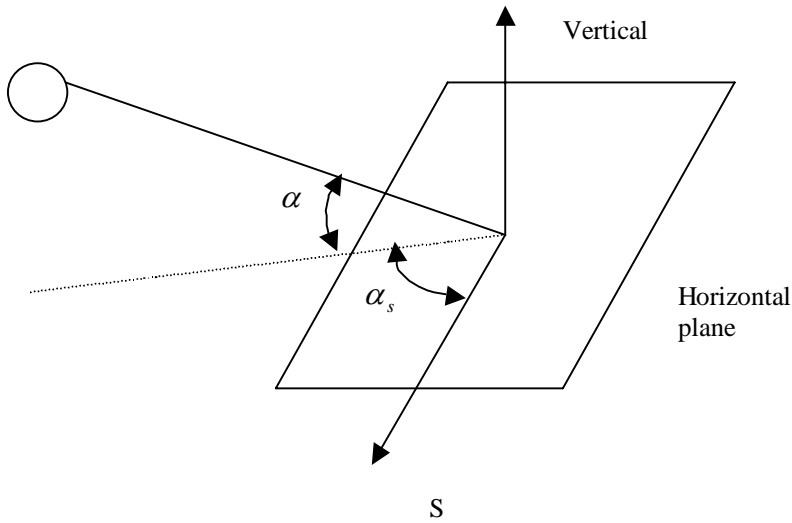
α: altitude angle.

Figure 5.10 Horizontal map

Figure 5.10 shows the relationship between altitude angle α and solar-azimuth angle α_s. The altitude angle α is measured from the local horizontal plane upward to the center of sun. The azimuth angle α_s is measured in the horizontal plane between a due south line and the projection of the site-to-sun line on the horizontal plane.

$$\sin(\alpha) = \sin(L) * \sin(\delta_s) + \cos(L) * \cos(\delta_s) * \cos(h) \qquad (5.5)$$

$$\sin(\alpha_s) = \cos(\delta_s) * \sin(h) / \cos(\alpha)$$

where,

h: the solar hour angle, is equal to 15° times the number of hours from local solar noon,

δ_s: declination of sun,

L: the latitude of the location.
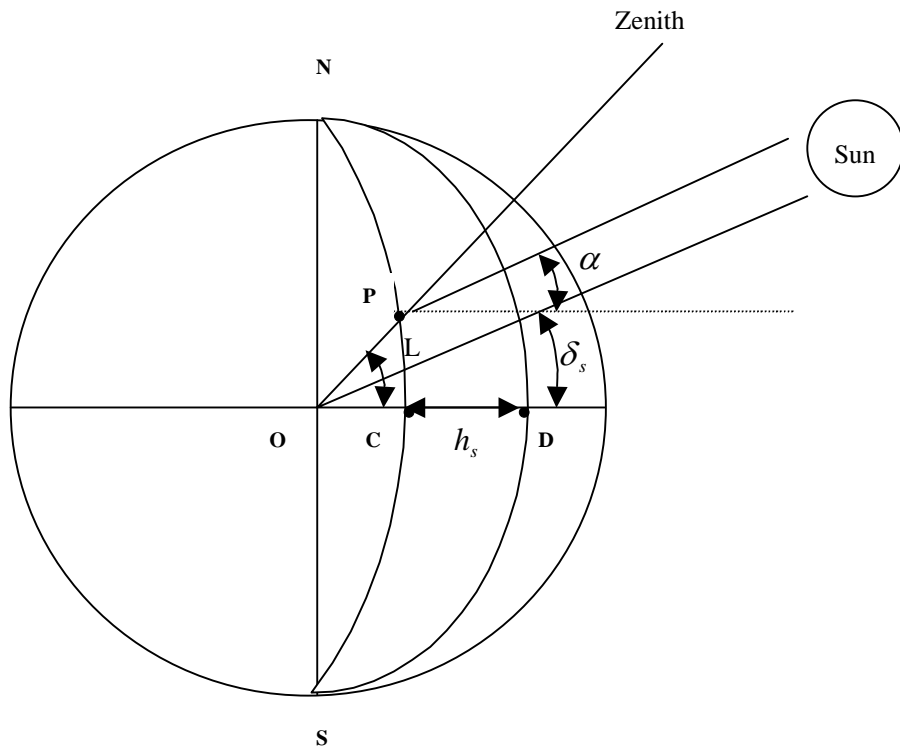
Figure 5.11 Zenith map

Figure 5.11 shows the map of solar radiation over the point P on the earth. We can get the degree of declination (dh) and the minute of declination (dm) from Table 5.3 [3], then we get the declination of the sun δ_s,

$$\delta_s = \frac{(dh + dm/60) * \pi}{180} \tag{5.6}$$

Table 5.3 Summary Solar Ephemeris

Date	Degree(dh)	Minute(dm)
Sept.1-4	8	35
Sept.5-8	7	7
Sept.9-12	5	37
Sept.13-16	4	6
Sept.17-20	2	34
Sept.21-24	1	1
Sept.25-28	0	32
Sept.29-30	-2	6

These equations can be used to calculate the heat flux as a function of latitude and time of day and year. Determining the amount of heat flux received requires that the size and orientation of each surface of the transformer be known. The total heat flux available to the transformer is then the sum of heat flux available to each surface.

We get the size and orientation of each surface of the transformer of Arcadia substation by SRP's transformer structure map as shown in Figure 5.12. The specification for the length a, width b and height c is the following, a=164in, b=64in and c=131in.
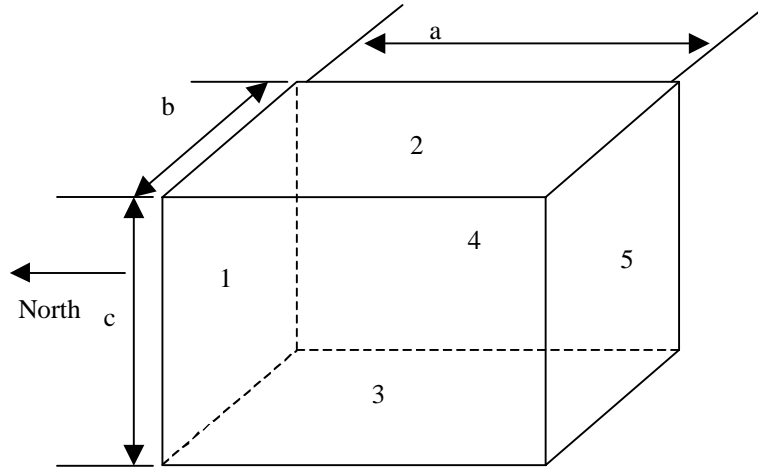
Figure 5.12 Transformer face 1 orientation toward due north

The component of solar radiation, $I_{b,c}$, intercepted by the surface of a collector is given

by,

$$I_{b,c} = I_b * \cos(i) \tag{5.7}$$

where, i, the incidence angle for a fixed surface is given by,

$$\cos(i) = -\sin(\delta_s) * (\sin L * \cos \beta - \cos L * \sin \beta z * \cos(\alpha_w))$$
$$+ \cos(\delta_s) * \cos(h) * (\cos L * \cos \beta + \sin L * \sin \beta * \cos(\alpha_w)) \tag{5.8}$$
$$+ \cos(\delta_s) * \sin \beta * \sin(\alpha_w) * \sin(h)$$

where,

α_w: wall-azimuth angle defined in the same manner as the solar-azimuth angle α_s

β: collect tilt angle defined as positive for surface facing south.

72
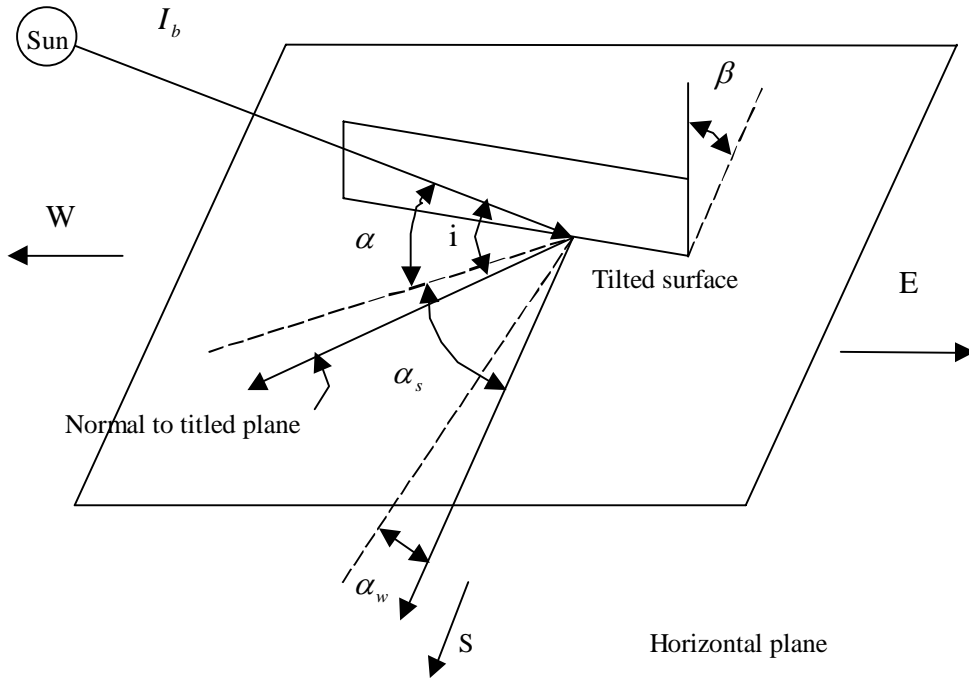
Figure 5.13 Definition of incidence angle i

Figure 5.13 shows the definition of incidence angle i, surface title angle β, solar-altitude angle α, wall-azimuth angle α_w, and solar-azimuth angle α_s for a non-south-facing, titled surface. The incidence angle i depends on the three basic solar angles: δ_s, h, L as well as on the two angles that characterize the surface orientation: $-\beta$ (surface facing north) and α_w.

There are several special cases that we encountered. The incidence angle for them is given by the following,

- Vertical surface facing due south

$$-\sin(\delta_s) * \cos(L) + \cos(\delta_s) * \sin(L) * \cos(h)$$

- Horizontal surface

$$\sin(\delta)*\sin(L)+\cos(\delta)*\cos(L)*\cos(h)$$

- Vertical surface facing west

$$-\sin(\alpha_s)*\cos(\alpha\ )$$

- Vertical surface facing east

$$\sin(\alpha)*\cos(\alpha)$$

- Vertical surface facing due north

$$-(-\sin(\delta_s)*\cos(L)+\cos(\delta_s)*\sin(L)*\cos(h))$$

The total solar radiation (SE) is the sum of all of the solar radiation intercepted by all surfaces of a collector.

The total solar radiation for a transformer of dimensions a=164in, b=64in, c=131in is shown in Figure 5.15.   Because SRP (and other utilities) may not be interested in determining the orientation of their transformers, we also investigated the total solar radiation collected using the transformers orientated of Figure 5.14.
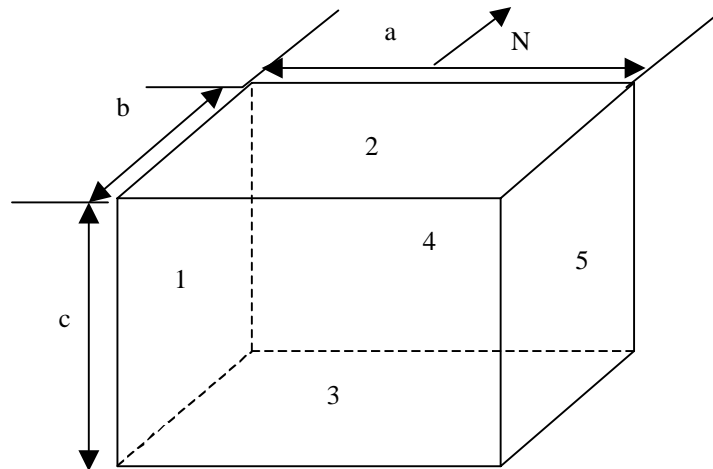
Figure 5.14 Transformer face 1 orientation toward due west

The solar radiation collected by this orientation is shown in Figure 5.16. Comparing the results seen in Figure 5.15 and Figure 5.16 shows that the orientation of the transformer has a relatively minor effect on the total solar radiation profile experienced by the transformer.

Solar radiation (North)
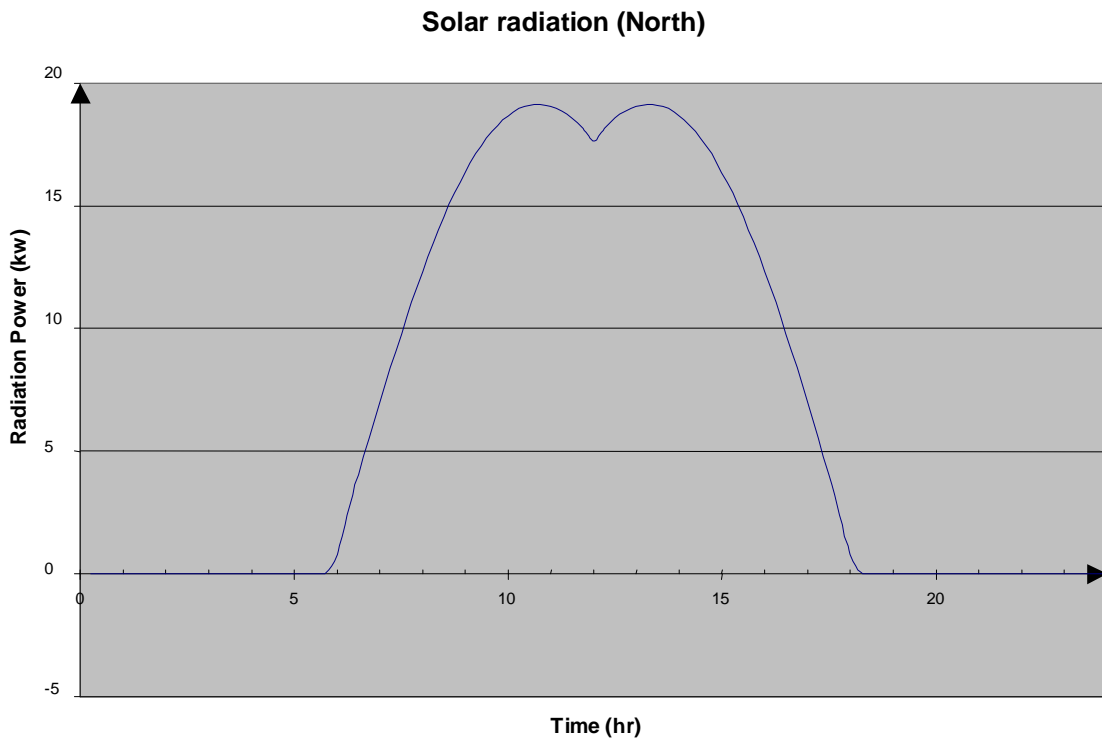
Figure 5.15 Solar radiation (face 1 orientation toward due north)

Solar radiation (west)
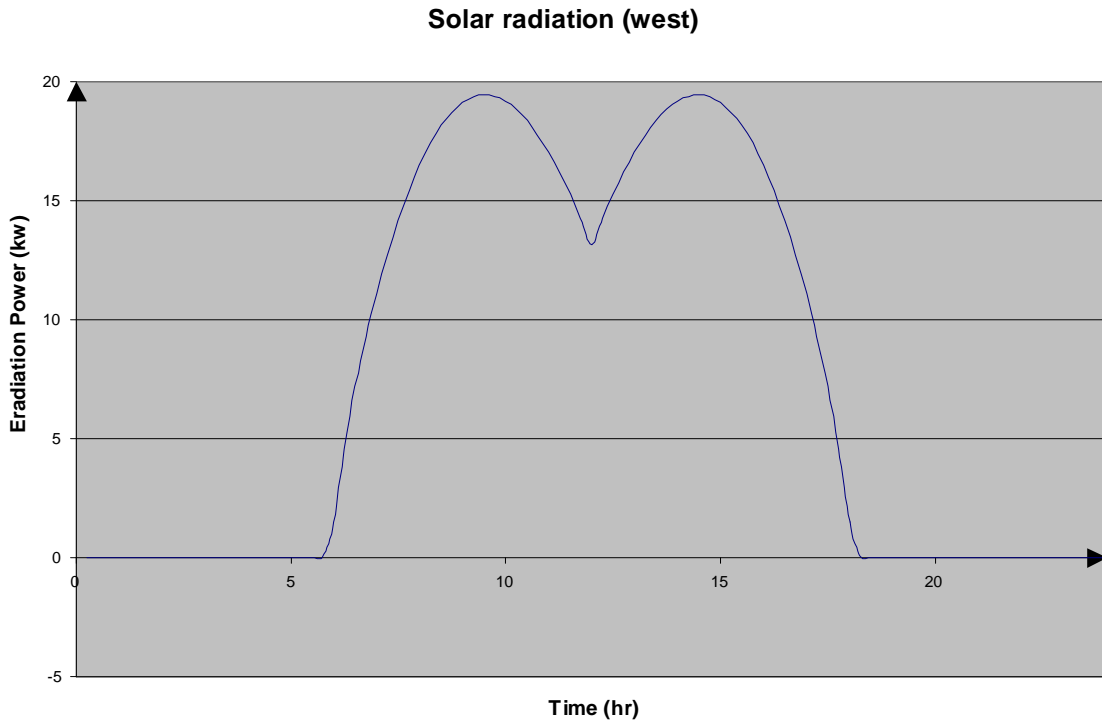
Figure 5.16 Solar radiation (face 1 orientation toward due west)

Note that accurately determining the heat flux received by the transformer requires that the effects of reflection, elevation and air-pollution be taken into account. Because we will be including the radiation heat flux (Figure 5.15) in our linear regression procedure in the form,

$$\theta_{top}[k] = k_3\theta_{top}[k-1] + k_2\theta_{amb}[k] + k_1I[k]^2 + k_5P_{Rad}[k] + k_4, \qquad (5.10)$$

getting the magnitude of the heat flux is unimportant; the linear regression algorithm will determine the optimal multiplier value, k_5, that will ultimately control the magnitude of the radiation term.

5.4 Wind velocity

5.4.1 Introduction and data preparation

Just as external fans cool the SDT by changing its thermodynamics, so too does wind change the thermal performance of an SDT. We suspected that wind may be a significant driving variable and decided to include it in our model.

Our first task was to acquire wind data for the SRP prism site. We obtained the wind velocity data from Nancy Selover, Assistant State Climatologist of the office of Climatology at Arizona State University. The wind velocity data we received for 17 SRP prism substations (Alamada, Arcadia, Collier, Corbell etc.) included measurements collected from July 1999 to Feb. 2000. The structure of data file is showed in Table 5.4.

Table 5.4 Structure of input wind data file

	Explanation	Value
Year	Year	1999
Mo	Month	07
Da	Day	01
Time	5 minutes sample period from 00:00AM	55
St	Station number	3
Temp	Temperature (Fahrenheit, whole degree)	56
Dew	Dew point (Fahrenheit, whole degree)	76
RH	Rain in hundredths of an inch (2 decimal place %)	88
Rain	Rain or Sun/Cloudy	1
Wdir	Wind direction in whole compass degree 0 through 360, (clockwise) where 0=no direction, 360=north	247
WSpd	Wind speed (mph, 5 minutes average)	7
PkGst	Peak gust speed (mph, 5 minutes average)	14

5.4.2 Developing a wind model

Developing a model that accounts for cooling due to wind flow is a complicated problem.

To do this properly, we need to know the wind velocity near the SDT as well as the

detailed geometry of the SDT. The next step is then to develop equations that approximate the heat flow from the transformer to the surrounding air. This problem, in itself, is a major research problem, which is beyond the scope of this project. Further, we do not have wind velocity data near the surface of the SDT. We only have wind velocity measured somewhere in the substation in which the SDT resides. Also, we do not know the geometry of the SDT. While we could, with some research, find this geometry, SRP has no interest in developing geometric models for all of their SDTs – and (we suspect) neither do other utilities who might use the techniques that we're researching.

Based on these facts we decided to try a simple wind model that would fit in our linear regression algorithm. We decided to model the effect of wind on TOT as linear. That is, we approximating the TOT defining differential equation to be,

$$T_o \frac{d\theta_{top}}{dt} = -\theta_{top} + \theta_u + \theta_{amb} + \theta_{wx} + \theta_{wy} . \qquad (5.11)$$

Notice that there are two terms in this model that account for wind, θ_{wx} and θ_{wy}. Realizing that SDT are not cubes, we felt we needed to model cooling separately along the surfaces running North-South, and along surfaces running East-West.

Discretizing (5.11) using the forward Euler discretization rule gives the linear form,

$$\theta_{top}[k] = k_3\theta_{top}[k-1] + k_2\theta_{amb}[k] + k_1I[k]^2 + k_4 + k_6V_x[k] + k_7V_y[k]$$

$$(5.12)$$

where,

k_1: coefficient for the input load,

k_2: coefficient for the input ambient temperature,

k_3: coefficient for the simulated TOT,

k_4: constant coefficient,

k_6: coefficient for the input wind velocity x direction,

k_7: coefficient for the input wind velocity y direction.

5.4.3 Transforming wind data to fit our model

To use linear regression with measured data with the model of (5.12), we needed to resolve our wind velocities into two orthogonal directions. Since the transformers we studied were oriented along N-S/E-W grid lines, we resolved the wind velocities in our data file into two orthogonal components along these directions. The velocity that runs in the E-W direction is given by,

$$V_x = |V * \cos(\theta)| \tag{5.13}$$

The wind velocity that runs in the N-S direction is given by,

$$V_y = |V * \sin(\theta)| \tag{5.14}$$

where,

θ: angle of wind direction with respect to the North in degree,

V: peak wind velocity, in 15 minutes interval.

Notice that the wind velocities are amplitudes, rather than vectors. We use amplitudes because the cooling takes place as long as the wind is blowing, regardless of the direction

in which it is blowing. Also in chosing the wind velocity, we choose the average wind speed instead of the peak gust speed.

5.5 New model for TOTPS

The semi-physical model can give us good prediction but can not give reliable coefficients for the real data. A modified semi-physical model was needed to accurately predict the top oil temperature, so we added two additional variables (Figure 5.13), wind velocity and solar radiation to the semi-physical model and obtained,
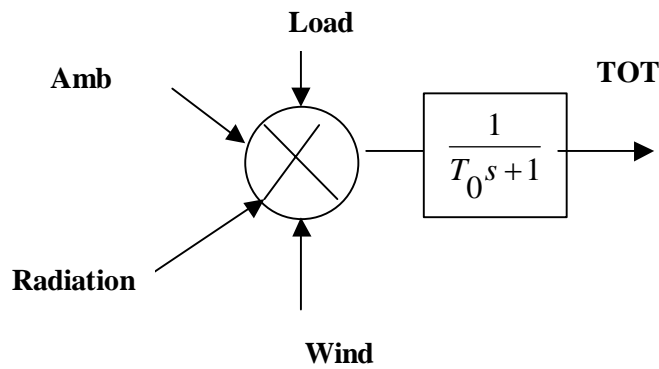
Figure 5.17 Transformer model

$$T_o \frac{d\theta_{top}}{dt} = -\theta_{top} + \theta_u + \theta_{amb} + \theta_R + \theta_{wx} + \theta_{wy}. \qquad (5.15)$$

Discretizing (5.15) using the forward Euler discretization rule gives the linear form,

$$\theta_{top}[k] = k_3 \theta_{top}[k-1] + k_2 \theta_{amb}[k] + k_1 I[k]^2 + k_4 + k_5 R[k] + k_6 V_x[k] + k_7 V_y[k]$$

$$(5.16)$$

where,

k_1: coefficient for the input load,

k_2: coefficient for the input ambient temperature,

k_3: coefficient for the simulated TOT,

k_4: constant coefficient,

k_5: coefficient for the solar radiation,

k_6: coefficient for the input wind velocity x direction,

k_7: coefficient for the input wind velocity y direction.

Using the linear regression algorithm discussed in chapter 1 to calculate the optimal coefficients ($k_1 - k_7$) with the assigned model of (1.15) and the new model of (5.16), we can get the results shown in Table 5.5 (a) - (d).

Table 5.5 Compare coefficients with/without wind, radiation

(a) Coefficients for semi-physical model

	K1	K2	K3	K4
1~10	2.8089	0.0392	0.9491	0.5054
11~20	3.4749	0.0518	0.9130	1.8320
21~30	2.4262	0.0096	0.9739	0.3050
31~40	2.6937	0.0299	0.9610	0.3670
41~50	1.9194	0.0367	0.9666	0.2376
51~60	4.2012	-0.0061	0.9396	2.2087
61~70	2.9176	0.0036	0.9680	0.7795
71~80	2.1243	0.0340	0.9726	-0.0225
81~90	1.5478	0.0278	0.9694	0.4080
91~100	1.6531	0.0308	0.9729	0.1834
101~110	2.0403	0.0305	0.9769	0.0850
111~120	1.8803	0.0336	0.9661	0.4054
121~130	1.7774	0.0336	0.9564	0.7226

(b) Coefficients with radiation

	K1	K2	K3	K4	K5
1~10	0.6842	0.0330	1.0120	-2.1612	0.0342
11~20	1.8955	0.0337	0.9565	0.4550	0.0234
21~30	1.0904	0.0112	0.9897	-0.3418	0.0245
31~40	1.6345	0.0304	0.9802	-0.4569	0.0147
41~50	1.5745	0.0295	0.9788	-0.1784	0.0085
51~60	3.4078	-0.0024	0.9513	1.6384	0.0087
61~70	2.4964	0.0022	0.9773	0.3994	0.0067
71~80	2.0605	0.0330	0.9746	-0.0932	0.0014
81~90	1.1858	0.0041	0.9956	-0.2228	0.0190
91~100	1.3020	0.0253	0.9784	0.0960	0.0051
101~110	1.6275	0.0194	0.9900	-0.1774	0.0066
111~120	1.2044	0.0108	0.9928	-0.1410	0.0143
121~130	0.9350	0.0077	0.9961	-0.2009	0.0204

(c) Coefficients with wind velocity

	K1	K2	K3	K4	K5	K6
1~10	3.1048	0.0290	0.9461	1.0215	-0.0269	-0.0185
11~20	3.5862	0.0513	0.9099	2.1114	-0.0192	-0.0298
21~30	2.4301	0.0059	0.9742	0.4873	-0.0213	-0.0169
31~40	2.6430	0.0311	0.9599	0.4321	-0.0078	-0.0037
41~50	1.8876	0.0362	0.9662	0.3116	-0.0069	-0.0080
51~60	4.1794	-0.0070	0.9407	2.2043	-0.0038	-0.0082
61~70	2.8889	0.0044	0.9683	0.7768	-0.0033	-0.0096
71~80	2.1063	0.0344	0.9725	0.0035	-0.0030	-0.0049
81~90	1.6840	0.0266	0.9686	0.4087	0.0058	0.0066
91~100	1.6493	0.0308	0.9728	0.1872	-0.0006	-0.0002
101~110	2.0357	0.0304	0.9770	0.0882	-0.0003	-0.0006
111~120	1.8887	0.0336	0.9659	0.4240	-0.0003	-0.0041
121~130	1.6543	0.0339	0.9552	0.8054	-0.0081	-0.0056

(d) Coefficients with radiation and wind velocity

	K1	K2	K3	K4	K5	K6	K7
1~10	0.9759	0.0271	1.0068	-1.7030	0.0323	-0.0144	-0.0131
11~20	2.1624	0.0356	0.9491	0.8176	0.0206	-0.0145	-0.0204
21~30	1.1385	0.0095	0.9893	-0.2422	0.0236	-0.0094	-0.0069
31~40	1.6351	0.0306	0.9799	-0.4374	0.0146	-0.0013	-0.0007
41~50	1.5477	0.0292	0.9783	-0.1024	0.0085	-0.0067	-0.0077
51~60	3.4119	-0.0031	0.9518	1.6499	0.0085	-0.0014	-0.0075
61~70	2.4686	0.0031	0.9776	0.3970	0.0067	-0.0032	-0.0097
71~80	2.0715	0.0338	0.9736	-0.0367	0.0008	-0.0029	-0.0047
81~90	1.2812	0.0037	0.9947	-0.2129	0.0187	0.0040	0.0042
91~100	1.2998	0.0253	0.9784	0.0981	0.0051	-0.0002	-0.0003
101~110	1.6308	0.0194	0.9902	-0.1889	0.0067	0.0005	0.0016
111~120	1.2148	0.0109	0.9925	-0.1246	0.0142	-0.0010	-0.0023
121~130	0.9441	0.0074	0.9968	-0.2248	0.0206	-0.0001	0.0026

We compare the maximum value, minimum value, average value and stand deviation of four different cases.

Case 1: use linear regression for the semi-physical model, get coefficients $k_1 - k_4$,

Case 2: use linear regression for the model with solar radiation, get coefficients $k_1 - k_5$,

Case 3: use linear regression for the model with wind velocity, get coefficients $k_1 - k_6$,

Case 4: use linear regression for the model with solar radiation and wind velocity, get coefficients $k_1 - k_7$.

Table 5.6 Comparing of k_1 coefficient

measure　＼　case	1	2	3	4
Max	4.2012	3.4078	4.1794	3.4119
Min	1.5478	0.6842	1.6493	0.9441
Ave	2.4204	1.6230	2.4414	1.6756
Std	0.7797	0.7260	0.8022	0.7000
Range	2.6534	2.7237	2.5302	2.4677

The results in Table 5.6 show that the modified semi-physical model improves the reliability of the coefficients a little bit. The range of coefficient decreases from 2.6534 to 2.4677.

5.6 Conclusion

We expected the addition of solar radiation and wind velocity to significantly reduce the range of k_1 variables obtained through linear regression. As shown in Table 5.6, this was not the case. Follow-on work will be needed to determine why this is the case and what other approach is needed to reduce the k_1 range obtained from linear regression.

6. CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

A software tool TOTPS (top-oil temperature prediction system) was created and developed to predict the TOT and maximum load a transformer is capable of carrying using the semi-physical model. In this work, several attempts are reported to improve the reliability of the optimal coefficients calculated by linear regression. These attempts include eliminating the effect of quantization and noise, and adding additional driving variables (such as solar radiation and wind velocity). It was shown that if the optimal coefficients vary within $\pm 10\%$ of the actual value, we still could get an acceptable peak load range. Further research showed that the revised semi-physical model with load, ambient temperature, TOT, solar radiation and wind velocity as driving variables tended to decrease slightly the range of k_1 coefficients obtained from similar data sets. Even with the addition of these variables, the range of k_1 is still too large to allow the peak transformer load to be estimated with confidence.

6.2 Future work

The peak load predicted by the TOTPS software is still not sufficiently accurate to be used as a production good tool. We have been told by the utility system operators that if the prediction are not accurate they will very quickly cease to use TOTPS. Consequently the objecting of future works is to improve the consistency with which we can calculate

k_1 and, hence the peak value of a transformer dynamic load. There are several other methods to get the optimized coefficients, that can be explored including weighted linear regression algorithm and neural network methods. Another model change that may lead to improvement is the use of different sets of model coefficients for different operating temperature ranges. A two-tiered/three-tiered model is essential when several different thermal dynamic conditions exist, such as when transformer fans and/or oil pumps turn on at high load values in summer. The biggest task remaining is to determine what is the cause the wide range of k_1 variable. Three possible driving variables that may account for this wide range of k_1 are rain, radiation to the night sky and cloud cover.

REFERENCES

[1]. Q. He, "Transformer semi-physical modeling and simulation", ASU dissertation, July 2000.

[2]. B.C. Lesieutre, W.H. Hagman, J.L. Kirtley Jr., "An improved transformer top oil temperature model for use in an on-line monitoring and diagnostic system", *IEEE Transactions On Power Delivery*, Vol. 12, No. 1, pp. 249-254, January 1997.

[3]. F.Kreith, J.F. Kreider, "Principles of Solar Engineer", Hemisphere Pub. Corp, Washington, 1978.

[4]. *IEEE Guide for Loading Mineral-Oil-Immersed Power Transformers rated in excess of 100 MVA (65 degree C winding rise)*, Transformers Committee of the IEEE Power Engineering Society, IEEE Std C57.115-1991.

[5]. D. J. Tylavsky, Q. He, G. A. McCulla, and J.R. Hunt, "Sources of Error and Quantization on Transformer Dynamic Thermal Modeling*", IEEE Transactions On Power Delivery,* Vol. 15, No. 1, (Jan. 2000), pp. 178-185

[6]. D. J. Tylavsky, Q. He, J. Si, G. A. McCulla, and J. R. Hunt, "Transformer Top-Oil Temperature Modeling and Simulation", *IEEE Transactions on Industry Application*, Vol. 36, No. 5, (Sep/Oct 2000), pp. 1219-1225.

[7]. Q. He, J. Si, and D. J. Tylavsky, "Prediction of Top-Oil Temperature for Transformers Using neural Networks", *IEEE Transactions on Power Delivery*, Vol. 15, No. 4, (Oct 2000), pp. 1205-1211.

[8]. Russ Turley, Matthew Wright, "Developing Engine Test Software in LabVIEW", CACI San Antonio, TX IEEE 1997.

[9]. Lisa K. Wells, Jeffrey Travis, "LabVIEW for everyone: Graphical Programming Made Even Easier", Prentice Hall PTR, Upper Saddle River, NJ.

[10]. Harry G.Stoll, "Least-Cost Electric Utility Planning", John Wiley & Sons Inc. 1989.

[11]. D. J. Tylavsky, Q. He, J. Si, G. A. McCulla, and J. R. Hunt, "Effect of Sampling and Quantization on Transformer Top-Oil Temperature Prediction*," Conf. Rec. IEEE IAS 1999 Annual Meeting (CD)*, Oct. 1999, Phoenix AZ.