Data-Driven Self-Optimizing Control: Unconstrained Optimization Problem

*¹Alhaji S. Grema, ²Yi Cao and ³Modu B. Grema
¹Department of Chemical Engineering, University of Maiduguri, Maiduguri, Nigeria
²College of Chemical and Biological Engineering, Zhejiang University, Hangzhou, P. R. China
³Department of Mathematics and Statistics, Ramat Polytechnic, Maiduguri, Nigeria
a.grema@unimaid.edu.ng | caoyi2018@zju.edu.cn | gremamodubako@gmail.com

Received: 06-SEP-2020; Reviewed: 29-OCT-2020; Accepted: 23-NOV-2020

http://dx.doi.org/10.46792/fuoyejet.v6i3.573

Abstract- Controlled variable (CV) selection plays an important role in determining the performance of a process plant. Existing methods for CV selection through self-optimizing control requires linearization of rigorous models around nominal operating points. This is a very difficult task which results to large losses. This work presents a novel method for CV design. A necessary condition of optimality (NCO) was proposed to be the CV. The approach does not require the analytical expression of the NCO to be derived but is approximated through a single regression step based on data. Finite difference was used to approximate the NCO (gradient) using data; three finite difference schemes were employed for this purpose, which are forward, backward and central differences. Seven different cases with respect to number of sampling points, neighborhood and finite difference schemes were investigated. To demonstrate the efficacy of the method in simplest way, it is applied to a hypothetical unconstrained optimization problem. The proposed method was found to have outperformed some existing approaches in many instances. A zero loss was recorded by some designed CVs. Central difference was found to be the best schemes among the three.

Keywords- controlled variable, disturbance, finite difference, monotonicity, regression.

1 INTRODUCTION

A critical stage in any control structure design is the selection of Control Variables (CVs) and Manipulated Variables (MVs), and the linkage of these variables. This stage affects to a larger extend the economy and the safety of any plant operation (Umar et al., 2012). Skogestad (2000), proposed a method of CV selection that places an emphasis on optimal operation of plant. This concept is called self-optimizing control (SOC). As stated by Umar et al. (2012), the main idea of this method is to find CVs which can be implemented, and despite the presence of uncertainties and disturbances, keep the operation of the process near-optimal. That is to say, the process becomes ‘self-optimizing’ with the control of the selected CVs at constant set-points. Self-optimizing control is when a near-optimality is achieved with an acceptable loss at a constant set-point value of CV without the need to reoptimize in the presence of disturbance (Skogestad, 2000).

A MV, control variable or control input is a quantity that can be adjusted directly, and which hopefully influences the output in a favourable way. The CV which is sometime called process variable is the quantity that is to be controlled. The controlled variable is needed to track a certain set-point (Janert, 2013). The concept of MV and CV is experienced in the process of maintaining the temperature of a vessel containing some material at a specific value. Here, the MV may be the flow of heating oil or the voltage applied to a heating element. The temperature of the vessel is CV (Janert, 2013). Basically, SOC methods can be classified into local and global methods. These were developed for either static or dynamic processes.

The local methods depend on linearizing a non-linear model around a nominal operating point and quadratic approximation of the loss function; this renders the solution to be local (Umar et al., 2012). A linearized model between $y$, $u$ and $d$ is given as in (Ye et al. 2013a)

$$ y = G^r u + G^s y d + W_n e $$

(1) where $G^r$ and $G^s$ are steady-state gain matrices of input and disturbance respectively. $W_n$ and $W_e$ are magnitude diagonal matrices to normalize $d$ and $e$ respectively. Again $y$ and $u$ are respectively the measurements and manipulated variables while $d$ and $e$ are disturbances and errors respectively. Equation (1) is a linear form of a nonlinear model. The CV, $c$ is given as a function of the measurements, $y$ as in (Ye et al. 2013a)

$$ c = H y $$

(2) where $H$ is a combination or selection matrix. In the latter, individual elements of measurements are selected as CVs while measurements are linearly combined in the former to form the CV function.

One of the first local methods for CV selection is based on Minimum Singular Value (MSV) rule presented by Skogestad and Postlethwaite (1996) and Halvorsen et al. (2003). The main idea relies on the CV that maximizes the MSV of a scale gain matrix (Halvorsen, et al., 2003; Umar et al., 2012). Unfortunately, this method can lead to wrong identification of CVs (Hori and Skogestad, 2008). A method based on the assumption that ‘the set-point around which linearization is done to obtain the approximate model is optimal’ was developed to overcome the shortcoming of MSV rule by Halvorsen et al. (2003). This is termed exact local method. Obtained loss expressions are used to screen CVs (Umar et al., 2012). The above two local methods are used to select CVs as a subset of measurements. A lower loss and therefore improved self-optimizing properties can be obtained by selecting CVs as linear combination of measurements. One of such methods that was developed to obtain optimal combination matrix is null space method (Alstad and Skogestad, 2007). Here, the implementation error is ignored.

*Corresponding Author

Section D: MATERIALS/ CHEMICAL ENGINEERING & RELATED SCIENCES

Can be cited as:


http://dx.doi.org/10.46792/fuoyejet.v6i3.573
and the set-point error is considered. The idea is to have an optimal value of CVs which is not affected by disturbance so that loss due to set-point error is reduced to zero. The method is suboptimal since implementation error is ignored, although this can be beneficial to complex systems where consideration of implementation error will be difficult to solve. There is also a requirement for minimum number of measurements to be satisfied in order to obtain the combination matrix which may sometimes lead to a complex control structure (Umar et al., 2012).

The selection of a subset of measurements as CV alternatives or their combination is regarded as a combinatorial optimization problem. Branch and bound methods for efficient solution of such problem has been proposed (Kariwala and Cao, 2010, 2009; Cao and Kariwala, 2008). Two expressions for loss in objective function have been defined by Ye et al. (2013a), they are used as criteria for CV selection. These are worst case and average losses for uniformly distributed disturbance given as follows (Ye et al., 2013a).

\[
L_{\text{worst}} = \frac{1}{2} \sigma_{\text{max}}^2
\]

\[
L_{\text{average}} = \frac{1}{6} \sigma_{\text{max}}^2 + \|M\|_F^2
\]

where \(\sigma_{\text{max}}(\cdot)\) and \(\|\cdot\|_F\) are the maximum singular value and Frobenius norm of a matrix, respectively. The matrix \(M\) was defined as

\[
M = \left[ J_{uu}^{-1/2} J_{ud}^{-1} G^{-1} C_d W_d J_{uu}^{-1/2} G^{-1} H W_a \right]
\]

where \(G = H G^T\), \(G_d = H G^T\) and the Hessian matrices are given as in (Ye et al., 2013a)

\[
J_{uu} = \frac{\partial^2 J}{\partial u^2}
\]

\[
J_{ud} = \frac{\partial J}{\partial u \partial d}
\]

The loss expressions given by Equations (3) and (4) are used to select the right CV as a subset of measurements defined in Equation (2) (Umar et al., 2012). It can then be said that CV selection procedure involves minimizing the loss expressions with respect to \(H\) (Ye et al., 2013a). Expressions for \(H\) have been derived as stated by Ye et al. (2013a).

The local methods depend on linearizing a nonlinear system around a nominal point to obtain a local solution which introduces losses in the SOC framework. To avoid this shortcoming, gradient functions were proposed to be used as the CVs directly so as to achieve a global optimal operation. In the work of Cao (2003, 2005), the gradient function was proposed to be determined analytically and be used directly as the CV. For cases where the gradient is not available or difficult to be obtained, the idea was extended to select CVs based on sensitivity of the gradient function to disturbances and implementation errors. The major drawback of this method is that finding the analytical gradient of some systems is not trivial. The gradient may also be nonlinear in state variables and unknown disturbances. In a related work, chain rule differentiation was used to explicitly express the gradient function in terms of system’s Jacobian (Cao, 2004). The gradient function was used as CV for a constrained optimization problem. A cascade control structure was proposed to handle the active constraint.

Methods were also developed to select CVs to approximate Necessary Condition of Optimality (NCO) with zero set-points to achieve near-optimality globally (Ye et al., 2013a, 2012). This is similar to NCO tracking technique where the NCO is selected as controlled variables. The difference is for SOC, CVs are selected offline based on output measurements and then a feedback controller with an integral action track the selected CVs online at their set-points. On the other hand, the components of NCO which are active constraints and reduced gradients are computed online and chosen as CVs in NCO tracking (Ye et al., 2013a). In the work of Ye et al. (2012), a two-step regression approach was used. In the first step, the economic objective was approximated using operational data while CVs were determined in the second step by incorporating NCO. The main advantage of this approach is that CVs are obtained that cover a wide operational range based on data; process model is not required. However, a shortcoming to this is the large error that results from the two regression steps.

A one-step regression approach was reported where CVs were used to approximate the NCO or reduced gradient with zero set-points (Ye et al., 2013a). The NCO was split into two parts: active constraint, \(g_a\) (constraint with strict equality) and reduced gradients \(\nabla_J f^{\text{red}}\) given as (Ye et al., 2013a)

\[
g_a = 0, \quad g_a \in \mathbb{R}^{n_a}
\]

and

\[
\nabla_J f^{\text{red}} = \frac{\partial f}{\partial u} \left[ I - \left( \frac{\partial g_a}{\partial u} \right) \frac{\partial g_a}{\partial u} \right] = 0
\]

The reduced gradient has \(n_a\) components which is compressed to \(n_u - n_a\) dimensions using singular value decomposition and is given by (Ye et al., 2013a)

\[
\nabla_J f^{\text{red}} = \frac{\partial f}{\partial u} V_2 = 0, \quad \nabla_J f^{\text{red}} \in \mathbb{R}^{n_u - n_a}
\]

where \(\nabla_J f^{\text{red}}\) is the compressed reduced gradient, and \(V_2\) are \(n_u - n_a\) right singular vectors.

An assumption made by Ye et al. (2013a) is that the active constraints are measurable and are controlled perfectly with an aim to control the remaining compressed reduced gradient at zero set-point. Despite the fact that a global optimal operation is achievable with this approach, a system model is still required to evaluate the NCO. This is actually a hiccup to systems with unknown or complicated model.

The local and global methods listed above were developed for continuous processes at steady state. Having recognised the impacts of batch processes in chemical plant operation, some authors (Dahl-Olsen et al., 2008; Dahl-Olsen and Skogestad, 2009; Hu et al., 2012; Ye et al., 2013b) strived to develop CV selection methodologies of dynamic processes applicable to batch process. In the work of Dahl-Olsen et al. (2008) for instance, the principle of maximum gain rule for the selection of CV is extended to batch process tracking problems, thus, poor controls can be screened out which leads to selection of only good ones according to scaled gains.
Ye et al. (2013b) extended the technique of NCO approximation (for static optimization of continuous processes) to approximating invariants. The invariants were also modelled as functions of output measurements where a case of unconstrained dynamic optimization was considered. Although, the application of the approach to a simplistic case was successful, the method will be cumbersome when applied to complex processes. More importantly, it is not applicable to processes with unknown models. A better approach is to approximate the NCO with measurements based on data only without the need of process equations. This type of data-driven SOC for constrained optimization problems was presented by Grema et al., (2018). A typical constrained optimization problem was given in Grema et al., (2020). This study sought to explore the development of the methodology for unconstrained problems.

2 METHODOLOGY
The method of self-optimizing control based on data will be developed here for unconstrained cases (Grema, 2014).

2.1 METHOD DEVELOPMENT
For an unconstrained case, the optimization problem is of the form.

\[
\min J(u, d)
\]

where \( J \) is the objective function, \( u \) the manipulative variable and \( d \) the disturbance. Here, we assume the target CVs be measurement functions, \( C = C(y, \theta) \) with parameters, \( \theta \) to be determined through regression using measurements, \( y \). The CV can be expressed as

\[
\frac{df}{du} = C(y, \theta)
\]

Equation (11) can be approximated using finite difference. For a reference point \( k \) and using forward difference, the approximation is given as

\[
C(y_k, \theta) = \frac{f_{k+1} - f_k}{u_{k+1} - u_k}
\]

Using backward difference, we have

\[
C(y_k, \theta) = \frac{f_k - f_{k-1}}{u_k - u_{k-1}}
\]

For central difference, the approximation is written as

\[
C(y_k, \theta) = \frac{f_{k+1} - f_{k-1}}{2(u_{k+1} - u_{k-1})}
\]

The above formulations are for one degree of freedom (DOF). For DOF other than one, Equation (12) – (14) are respectively written as

\[
f_{k+1} - f_k = C^T(y_k, \theta)(u_{k+1} - u_k)
\]

\[
f_k - f_{k-1} = C^T(y_k, \theta)(u_k - u_{k-1})
\]

and

\[
f_{k+1} - f_{k-1} = 2C^T(y_k, \theta)(u_{k+1} - u_{k-1})
\]

Various types of models such as polynomials, neural network model and so on can be used to approximate the target CV function, \( C \) depending on the complexity of the system. The performance of the method is evaluated using average loss of Ye et al., (2013a) defined by

\[
L = \frac{1}{d^2 - d} \int_{d}^{d^2} Ldd
\]

for which the loss, \( L \) is given by

\[
L = f(u_{yo}, d) - J_{opt}(d)
\]

In Equation (19), \( u_{yo} \) is the feedback control law and \( J_{opt} \) the theoretically obtainable optimum value of the cost function. The developed methodology will now be tested on a hypothetical problem, named toy example (Umar et al., 2012).

2.2 ILLUSTRATIONS
The objective function for the toy problem is

\[
J = \frac{1}{2} (u - d)^2
\]

with two available measurements

\[
\begin{align*}
   y_1 &= u \\
   y_2 &= \frac{1}{4} u^2 + d
\end{align*}
\]

The disturbance, \( d \) is assumed to vary in the range \( d \in [-1, 1] \). The manipulative variable, \( u \) is also bounded in this range. With this set up, optimum operation is achieved if Equation (22) (gradient)

\[
\frac{df}{du} = u - d
\]

is maintained at zero.

Two sets of polynomials were used for regression purposes to approximate the target CV:

First-order polynomial

\[
C_1(y_{yo}, \theta) = \theta_1 y_{1,0} + \theta_2 y_{2,0} + \theta_3
\]

Second-order polynomial

\[
C_2(y_{yo}, \theta) = \theta_1 y_{1,0}^2 + \theta_2 y_{1,0} y_{2,0} + \theta_3 y_{1,0} + \theta_4 y_{2,0} + \theta_5
\]

where the subscript, 0 in Equations (23) and (24) indicates measurements taken at reference points. The following steps were followed to determine the CV parameters, \( \theta \) through linear regression:

i. A set of data is collected by sampling the whole space of manipulative variables and disturbances.

ii. At each reference point, the gradient of the objective function with respect to manipulative variable, \( \frac{df}{du} \) is computed using one of the finite difference schemes presented in Equations (12) – (14), and the measurements, \( y_1 \) and \( y_2 \).

iii. Regressions were performed by minimizing the value of squared 2-norm of the residual with the parameters, \( \theta \) being adjusted to fit in the computed gradient to either of the measurement functions in Equations (23) and (24);

If we let the right-hand sides of Equations (18) – (20) to be \( q \), the regression problem can be expressed as

\[
\min_{\theta} \frac{1}{2} \| C(y_{yo}, \theta) - q \|^2_2
\]

R-squared value is used to measure the performance of the regression. This is sometimes referred to as coefficient of determination which is an indicator for goodness of fit. It ranges from 0 to 1 with 1 indicating best regression fit (Ye
et al., 2013a). If $N$ samples of $x_i$, $i = 1, 2, \ldots, N$ is approximated, then,

$$R^2 = \frac{S_T - S_E}{S_T}$$

(26)

where

$$S_T = \sum (x_i - \bar{x})^2$$

(27)

is total sum of squares and Equation (28)

$$S_E = \sum (\hat{x}_i - \bar{x})^2$$

(28)

is error sum of squares. In the above equations, $\bar{x}_i$ are the approximated $x_i$ and $\bar{x}_i$ the mean of $x_i$ for which $i = 1, 2, \ldots, N$.

The CV implementation can simply be visualized in Figure 1. Measurements obtained from the process are used to evaluate the CV, $C(y, \theta)$ while a feedback controller with a simple integral action is used to update the feedback control, $u_{\beta b}$ for every disturbance, $d$ so that the CV is kept at a set-point, $C_0$.

Different configurations with respect to number of sampling points, neighbourhood points and finite difference schemes are tested next. This novel method was compared to NCO approximation (Ye et al., 2012) and local methods. CVs found using NCO approximation method are denoted by $C_{\text{NCO}}$ and $C_{2\text{NCO}}$ for first and second order polynomials respectively, while that which resulted from using local method is as $C_{\text{local}}$.

1) Case 1: Forward and Backward Finite Difference with Multiple Neighbourhood Points

Here, $u$ and $d$ are divided into 11 equal points in the range $[-1, 1]$. Each point of $u$ was taken as a reference point and the interval between each successive point was divided into 10. These subdivisions were used as neighbourhood points. For each reference point, $d$ was varied over its entire range. Precisely, the edge reference points have 10 neighbours while the 9 inner points each have 20 neighbourhood points (considering backward and forward neighbours) as illustrated in Figure 2. So, in summary, the following were considered:

- 11 reference points in the range $[-1, 1]$
- 10 neighbourhood points each for boundary references and 20 for inner references
- 11 disturbance points in the range $[-1, 1]$
- Number of data points, $N_p$ is given by the expression given in Equation (29)

$$N_p = [bn + 2n(N-b)]n_d$$

(29)

- Where $N$ is number of reference points, $n$ is number of neighbor points, $b$ is number of boundaries and $n_d$ the number of disturbance divisions. So, for this case, $b = 2$, $n = 10$, $N = 11$ and $n_d = 11$. Therefore, using Equation (29), $N_p = 2200$

Effects of numbers of reference points, $N$ and neighbourhood points, $n$ on the method’s efficacy are studied in subsequent cases.

2) Case 2: Forward and Backward Finite Difference with Single Neighbourhood Point

In this configuration, combination of forward and backward finite differences is used as in Configuration 1. Here, $N$ is increased from 11 (Configuration 1) to 21 while $n$ is reduced from 10 to 1. The following are the parameters used:

- 21 reference points in the range $[-1, 1]$
- One neighbourhood point each for boundary references and two for inner references
- 11 disturbance points in the range $[-1, 1]$

Therefore, using Equation (29), number of data points, $N_p$ for this configuration is 440. The arrangement of references and neighbours for this configuration is shown in Figure 3.

3) Case 3: Forward and Backward Finite Difference with Reference Points used as Neighbours

In this set up, all available sampling points are used as reference points with one neighbour point each for boundary references and two for inner references. There was no subdivision in the references to obtain neighbours, but the reference points were actually used as the Neighbours. For this case, we have
101 reference points in the range [-1, 1]
One neighbourhood point each for boundary references and two for inner references
11 disturbances

\( N_p \) is therefore 2200 as computed from Equation (29). This is shown in Figure 4.

![Fig. 4: Reference and Neighbourhood Points for Case 3](image)

4) Case 4: Central Finite Difference with Reference Points used as Neighbours
Here central difference scheme is employed with the aim to improve the performance over a mix of forward and backward differences. For central difference, Equation (29) is modified as

\[ N_p = (N - b)n_d \]  

(30)

The following were used:
- Six sampling points in the range [-1, 1]
- 11 disturbance points in the range [-1, 1]
- Using Equation (30), \( N_p = 44 \)

5) Case 5: Variables Ranking using Separable Rule
In some practical situations, there are instances that the disturbance is totally unknown. In the present case we will test the robustness of the method by ranking the variables according to the effect of disturbance. In Case 6, the disturbance was assumed to be unknown.

Disturbance information was used to ensure the finite difference is calculated between data points at the same disturbance. In this configuration, ranking is used to sort variables based on the effect of disturbance on \( J \). Here, \( f_j \)s are ranked according to their magnitude when disturbance is changed. All measurements were also ranked according to the order of \( J \). For this configuration, the following were employed:
- 201 sampling points in the range [-1, 1]
- 11 disturbance points in the range [-1, 1]
- Using Equation (30), \( N_p = 2189 \)

6) Case 6: Variables Ranking with Monotonicity Rule
To improve the performance of the ranking method, an appropriate variable needs to be selected for sorting. The selection is done by following a certain rule. The rule actually used is monotonicity of the measurements and objective functions to disturbance. For the toy example, the cost function is square to disturbance while measurement \( y_1 \) is independent of the disturbance. In the case of \( y_2 \), all its values are well distributed; hence it is used as the sorting variable. The method is further validated by using randomised disturbance between -1 and 1. Central difference scheme was used.
- Six sampling points in the range [-1, 1]
- 11 random disturbance points in the range [-1, 1]

\( N_p \) is therefore 44 using Equation (30)

7) Case 7: Separable Rule for Random Disturbance
This setup uses separable rule as in Configuration 5 but here random disturbances were generated using
- 201 sampling points in the range [-1, 1]
- 11 disturbance points in the range [-1, 1]
- \( N_p \) is therefore 2189 by using Equation (30)

3 Results and Discussion
For Case 1, where multiple neighbourhood points were used in conjunction with the combination of forward and backward finite differences, the two CVs obtained are

\[ C_1 = 0.9838y_1 - 0.9850y_2 + 0.0837 \]  

(31)

and

\[ C_2 = 0.2500y_1^2 - 0.0037y_1y_2 + 0.9844y_1 - y_2 + 0.0000 \]  

(32)

with R²-values of 0.9867 and 0.9949 respectively. Comparing this to NCO approximation method, CVs obtained are, for first-order model

\[ C_{1NCO} = y_1 - 0.9809y_2 + 0.0981 \]  

(33)

and second-order model

\[ C_{2NCO} = 0.2500y_1^2 - 2.976x10^{-17}y_1y_2 + y_1 - y_2 \]  

(34)

with R²-values of 0.9903 and 1.0000 respectively.

Applying local SOC method to the toy problem, the following CV was obtained

\[ C_{local} = y_1 - y_2 \]  

(35)

If the bilinear term in Equation (34) is ignored, \( C_{2NCO} \) is the true gradient and hence its loss is 0. Knowing that at optimal operation point, the obtained CV functions must all equal to zero, therefore, by substituting the measurements \( y_1 \) and \( y_2 \) according to Equation (21) in the CV functions (Equations 31 –35), equivalent feedback control laws can be obtained as summarized in Table 1. It can be observed from Table 1 that the proposed method provides self-optimizing CV with a better performance than local SOC without the much needed effort to determine the gradient equation as with NCO approximation.

Results for Case 2, where effects of numbers of reference points, \( N \) and neighbourhood points, \( n \) on the method’s efficacy are investigated are shown in Table 2. Comparing this configuration with only 440 data points to Configuration 1 where \( N_p = 2200 \), it can be concluded that increasing the number of reference points helps in improving the performance of the methods. However, using multiple neighbours does not have effect on the
performance. This is because, the CV function is only evaluated at reference points, and more neighbourhood points do not supply additional information to the CV function evaluation. A simple finite difference can provide similar but more consistent results than multiple neighbourhood points. For this reason, all data points were considered as reference points in Case 3. Case 3 results obtained are shown in Table 3. Comparing this case to the first configuration of equal data points (2200), a tremendous improvement was obtained with reduction in loss for the second-order model of up to 99.96%.

It is interesting to note from Table 4 which shows Case 4 findings that, using only six sampling points, a zero loss for second-order polynomial was recorded. This justifies the importance for selecting a right model structure.

The results in Table 5 (for Case 5) indicate that even if the disturbance is totally unknown, we can still achieve acceptable loss if we have sufficient data. Table 6 for Case 6 results shows an improvement in the ranking method with performance similar to that of Case 4. The results have proven the efficacy of using monotonicity as a rule to selecting sorting variable; with only 6 sampling points, a zero loss was achieved even with random disturbance points. The separable variable approach for Case 7 does work but not as perfect as Case 6 according to Table 7. This is because the ranking has some inherent error for certain disturbance range.

Table 1. Comparison between Data-Driven SOC and other Methods for Case 1

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Control Law Equivalent to</th>
<th>Average Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>(C_1)</td>
<td>(1.9798 \times 10^{-2})</td>
<td>0.0054</td>
</tr>
<tr>
<td>(C_2)</td>
<td>1.0154</td>
<td>4.2135x10^{-3}</td>
</tr>
<tr>
<td>(C_{DCO})</td>
<td>(2.03894 - 0.03894)</td>
<td>0.0038</td>
</tr>
<tr>
<td>(C_{SOC})</td>
<td>(-0.03894 + 0.9962 - 0.9962)</td>
<td>0</td>
</tr>
<tr>
<td>(C_{local})</td>
<td>2.0 - 2.0(\sqrt{1 - 0.25d})</td>
<td>0.0935</td>
</tr>
</tbody>
</table>

Table 2. Data-Driven SOC – Case 2

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>First-Order Polynomial</th>
<th>Second-Order Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\theta_1)</td>
<td>0.9963</td>
<td>0.2500</td>
</tr>
<tr>
<td>(\theta_1)</td>
<td>-0.9860</td>
<td>-0.0009</td>
</tr>
<tr>
<td>(\theta_2)</td>
<td>0.0826</td>
<td>0.9964</td>
</tr>
<tr>
<td>(\theta_3)</td>
<td>-</td>
<td>-1.0000</td>
</tr>
<tr>
<td>(\theta_4)</td>
<td>-</td>
<td>0.0000</td>
</tr>
<tr>
<td>(\theta_5)</td>
<td>2.0209</td>
<td>1.0036d</td>
</tr>
<tr>
<td>Control Law</td>
<td>(-0.03894)</td>
<td>(1.9976 \times 10^{-2})</td>
</tr>
<tr>
<td>Average Loss</td>
<td>0.0046</td>
<td>2.1600x10^{-4}</td>
</tr>
<tr>
<td>R(^2)</td>
<td>0.9915</td>
<td>0.9992</td>
</tr>
</tbody>
</table>

Table 3. Data-Driven SOC – Case 3

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>First-Order Polynomial</th>
<th>Second-Order Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\theta_1)</td>
<td>0.9997</td>
<td>0.2500</td>
</tr>
<tr>
<td>(\theta_1)</td>
<td>-0.9863</td>
<td>-0.0001</td>
</tr>
<tr>
<td>(\theta_3)</td>
<td>0.0822</td>
<td>0.9997</td>
</tr>
<tr>
<td>(\theta_4)</td>
<td>-</td>
<td>-1.0000</td>
</tr>
<tr>
<td>(\theta_5)</td>
<td>-</td>
<td>0.0000</td>
</tr>
<tr>
<td>Control Law</td>
<td>(-0.03894)</td>
<td>(1.9976 \times 10^{-2})</td>
</tr>
<tr>
<td>Average Loss</td>
<td>0.0044</td>
<td>1.5000x10^{-4}</td>
</tr>
<tr>
<td>R(^2)</td>
<td>0.9924</td>
<td>0.9999</td>
</tr>
</tbody>
</table>

Table 4. Data-Driven SOC – Case 4

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>First-Order Polynomial</th>
<th>Second-Order Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\theta_1)</td>
<td>0.5000</td>
<td>0.2500</td>
</tr>
<tr>
<td>(\theta_1)</td>
<td>-0.4980</td>
<td>-0.0000</td>
</tr>
<tr>
<td>(\theta_3)</td>
<td>0.0249</td>
<td>0.5000</td>
</tr>
<tr>
<td>(\theta_4)</td>
<td>-</td>
<td>-0.5000</td>
</tr>
<tr>
<td>(\theta_5)</td>
<td>-</td>
<td>0.0000</td>
</tr>
<tr>
<td>Control Law</td>
<td>(-0.03894)</td>
<td>(1.9976 \times 10^{-2})</td>
</tr>
<tr>
<td>Average Loss</td>
<td>0.0077</td>
<td>0</td>
</tr>
<tr>
<td>R(^2)</td>
<td>0.9973</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

Table 5. Data-Driven SOC – Case 5

<table>
<thead>
<tr>
<th>Coefficients</th>
<th>First-Order Polynomial</th>
<th>Second-Order Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\theta_1)</td>
<td>0.5000</td>
<td>0.1215</td>
</tr>
<tr>
<td>(\theta_2)</td>
<td>-0.4797</td>
<td>0.0000</td>
</tr>
<tr>
<td>(\theta_3)</td>
<td>0.0396</td>
<td>0.5000</td>
</tr>
<tr>
<td>(\theta_4)</td>
<td>-</td>
<td>-0.862</td>
</tr>
<tr>
<td>(\theta_5)</td>
<td>-</td>
<td>0.0000</td>
</tr>
<tr>
<td>Control Law</td>
<td>(-0.03894)</td>
<td>(1.9976 \times 10^{-2})</td>
</tr>
<tr>
<td>Average Loss</td>
<td>0.0032</td>
<td>1.2696x10^{-4}</td>
</tr>
<tr>
<td>R(^2)</td>
<td>0.9780</td>
<td>0.9850</td>
</tr>
</tbody>
</table>

Table 6. Data-Driven SOC – Case 6

<table>
<thead>
<tr>
<th>Coefficients</th>
<th>First-Order Polynomial</th>
<th>Second-Order Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\theta_1)</td>
<td>0.5000</td>
<td>0.2500</td>
</tr>
<tr>
<td>(\theta_2)</td>
<td>-0.4935</td>
<td>0.0000</td>
</tr>
<tr>
<td>(\theta_3)</td>
<td>0.0244</td>
<td>0.5000</td>
</tr>
<tr>
<td>(\theta_4)</td>
<td>-</td>
<td>-0.5000</td>
</tr>
<tr>
<td>(\theta_5)</td>
<td>-</td>
<td>0.0000</td>
</tr>
<tr>
<td>Control Law</td>
<td>(-0.03894)</td>
<td>(1.9976 \times 10^{-2})</td>
</tr>
<tr>
<td>Average Loss</td>
<td>0.0066</td>
<td>0.0000</td>
</tr>
<tr>
<td>R(^2)</td>
<td>0.9964</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

Table 7. Data-Driven SOC – Case 7

<table>
<thead>
<tr>
<th>Coefficients</th>
<th>First-Order Polynomial</th>
<th>Second-Order Polynomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\theta_1)</td>
<td>0.4991</td>
<td>0.1220</td>
</tr>
<tr>
<td>(\theta_2)</td>
<td>-0.4805</td>
<td>0.0003</td>
</tr>
<tr>
<td>(\theta_3)</td>
<td>0.0396</td>
<td>0.4994</td>
</tr>
<tr>
<td>(\theta_4)</td>
<td>-</td>
<td>-0.4886</td>
</tr>
<tr>
<td>(\theta_5)</td>
<td>-</td>
<td>0.0000</td>
</tr>
<tr>
<td>Control Law</td>
<td>(-0.03894)</td>
<td>(1.9976 \times 10^{-2})</td>
</tr>
<tr>
<td>Average Loss</td>
<td>0.0033</td>
<td>7.7895x10^{-4}</td>
</tr>
<tr>
<td>R(^2)</td>
<td>0.9803</td>
<td>0.9901</td>
</tr>
</tbody>
</table>
4 CONCLUSIONS AND RECOMMENDATIONS
A novel method of data-driven SOC was developed where the gradient of the objective function with respect to control was used as the target CV. The method does not require the gradient information (explicit expression of the gradient) but is computed based on data through finite difference scheme. An improved performance was seen with the method which is far better than local SOC when applied to unconstrained optimization problem.

It was found out that, the more the number of reference points, the better the performance, although this has a detrimental effect on the computational time. On the other hand, the use of multiple neighbourhood points does not contribute to the superior performance of the method; this is because CV functions are only computed at reference points. Also, using central difference scheme produced the best performance than forward and backward differences. The methodology was also tested for situations where the disturbance is completely unknown. Here, variables ranking based on separable and monotonicity rules were employed to deal with the situation. Again, a remarkable performance was recorded with a loss close to zero. The monotonicity rule used for selecting sorting variable has some setbacks; in practice, we may not know which of the variables is monotonous to disturbance. Furthermore, two or more variables can be monotonous. Hence, it is recommended to find out a numerically realizable algorithm for choosing a sorting variable.

ACKNOWLEDGMENT
The financial support of Petroleum Technology Development Fund (PTDF), Abuja is appreciated.

REFERENCES