Qualitative Fault Detection and Hazard Analysis Based on Signed Directed Graphs for Large-Scale Complex Systems

Fan Yang[1,2], Deyun Xiao[1] and Sirish L. Shah[2]
[1] Tsinghua University
[2] University of Alberta
[1] China
[2] Canada

1. Introduction

Nowadays in modern industries, the scale and complexity of process systems are increased continuously. These systems are subject to low productivity, system faults or even hazards because of various conditions such as mis-operation, equipment quality change, external disturbance, and control system failure. In these systems, many elements are interacted, so a local fault can be propagated and probably spread to a wide range. Thus it is of great importance to find the possible root causes and consequences according to the current symptom promptly. Compared with the classic fault detection for local systems, the fault detection for large-scale complex systems concerns more about the fault propagation in the overall systems. And this demand is much close to hazard analysis for the system risks, which is a kind of qualitative analysis in most cases prior to quantitative analysis.

The signed directed graph (SDG) model is a kind of qualitative graphical models to describe the process variables and their cause-effect relations in continuous systems, denoting the process variables as nodes while causal relations as directed arcs. The signs of nodes and arc correspond to variable deviations and causal directions individually. The SDG obtained by flowsheets, empirical knowledge and mathematical models is an expression of deep knowledge. Based on the graph search, fault propagation paths can be obtained and thus certainly be helpful for the analysis of root causes and sequences (Yang & Xiao, 2005a). And with development of the computer-aided technology, graph theory has been implemented successfully by some graph editors, some of which, like Graphviz (2009), can transform text description into graphs easily. Hence the SDG technology can be easily combined with the other design, analysis and management tools.

The SDG definition and its application in fault diagnosis were firstly presented by Iri et al. (1979). Ever since then, many scholars have contributed to this area, including modeling, inference, software development and applications. Many efforts have been particularly made to implement the methods and to overcome the disadvantages, such as spurious solutions. Here we recognize some representatives among them. Kramer & Palowitch (1987) used rules to describe SDG arcs, which shows that expert systems can be employed as a tool in this problem. Oyeleye & Kramer (1988) took into account the qualitative simulation for the SDG inference. Shiozaki et al. (1989) improved the SDG model by adding fault revealing time. Yu et al. (Chang & Yu, 1990; Yu & Lee, 1991) introduced fuzzy information for arc signs to describe the steady state gains. Maurya et al. (2003a, 2003b, 2006) described the modeling method based on differential equations (DEs) and algebraic equations (AEs), analyzed the initial and final responses based on SDGs, and studied the description and analysis of control loops. SDG method has been combined with other data-driven methods to improve the diagnosis accuracy (Vedam & Venkatasubramanian, 1999; Lee et al., 2006). At first, the inference is based on single fault assumption, but multiple fault cases attract more and more attention (Vedam & Venkatasubramanian, 1997; Zhang et al. 2005; Chen & Chang, 2007). Up to now, SDG method has been implemented in some software tools (Mylaraswamy & Venkatasubramanian, 1997; McCoy et al. 1999; Zhang et al., 2005) and applied in various industrial systems.

Aiming at SDG applications in the area of fault detection and hazard analysis, the problems of description and inference are most important. As the system extends, the time consumption of graph search is heavy, so the single-level SDG model should be transformed into hierarchical model to improve the search efficiency. The root cause can be searched in this model level by level according to the initial response of the system. In control systems and many other cases, cycles exist in the graph, resulting in the truncation or misleading to the search. Thus the theoretic fundamentals and dynamic features of SDGs should be studied. We have analyzed the fault propagation principles by operations of corresponding qualitative matrices and obtained some typical rules of control systems.

Moreover, fault detection is performed based on sensor readings, thus the sensor location strategy affects the performance of fault detection. Due to the economical or technical limitations, the number of sensors should be limited while meeting the demands of fault detection. This can be considered in the SDG framework. We analyze main criteria such as detectability, identifiability and reliability in the framework of SDGs and presented algorithms, in order to guarantee that the faults can be detected and identified, and to optimize the fault detection ability.

This chapter is organized as follows: first, the SDG description is reviewed and hierarchical model is indicated; then the fault propagation rules and inference approaches are summarized to lead to the successful application of fault detection and hazard analysis; some considerations about sensor location are introduced next; finally a generator set process in a power plant is modeled and analyzed to illustrate the proposed model and method.

2. Model Description of Signed Directed Graph

2.1 Basic Form of SDG Model

SDGs are established by representing the process variables as graph nodes and representing causal relations as directed arcs. An arc from node A to node B implies that the deviation of A may cause the deviation of B. For convenience, “+”, “-” or “0” is assigned to the nodes in comparison with normal operating value thresholds to denote higher than, lower than or within the normal region respectively. Positive or negative influence between nodes is distinguished by the sign "+" (promotion) or "-" (suppression), assigned to the arc (Iri et al., 1979). The definition is as follows:

Definition 1: An *SDG model* γ is the composite (G,φ) of (1) a digraph G which is the quadruple $(N,A,\partial^+,\partial^-)$ of (a) a set of nodes $N=\{v_1,v_2,\cdots,v_n\}$, (b) a set of arcs $A=\{a_1,a_2,\cdots,a_m\}$, (c) a couple of incidence relations $\partial^+ : A \to N$ and $\partial^- : A \to N$ which make each arc correspond to its initial node $\partial^+ a_k$ and its terminal node $\partial^- a_k$, respectively; and (2) a function $\varphi : A \to \{+,-\}$, where $\varphi(a_k)\ (a_k \in A)$ is called *the sign of arc* a_k .

Usually we use a_{ij} to denote the arc from v_i to v_j.

Definition 2: A *pattern* on the SDG model $\gamma=(G,\varphi)$ is a function ψ: $N \to \{+,0,-\}$. $\psi(v)$ $(v \in N)$ is called the *sign of node* v , i.e.

$$\psi(v)=0 \text{ for } \left|x_v-\overline{x}_v\right|<\varepsilon_v$$
$$\psi(v)=+ \text{ for } x_v-\overline{x}_v \geq \varepsilon_v$$
$$\psi(v)=- \text{ for } \overline{x}_v-x_v \geq \varepsilon_v$$

where x_v is the measurement of the variable v , $\overline{x}_v$ is the normal value, and ε_v is the threshold.

Definition 3: Given a pattern ψ on a SDG model $\gamma=(G,\varphi)$, an arc a is said to be *consistent* (with ψ) if $\psi(\partial^+ a)\varphi(a)\psi(\partial^- a)=+$. A path, which is consisted of arcs $a_1,a_2,\cdots,a_k$ linked successively, is said to be *consistent* (with ψ) if $\psi(\partial^+ a_1)\varphi(a_1)\cdots\varphi(a_k)\psi(\partial^- a_k)=+$.

2.2 Modeling Methods of SDGs

2.2.1 SDG modeling by mathematical equations

In general, SDGs can be obtained either from operational data and process knowledge, or mathematical models. If we have the differential algebraic equations (DAEs), then we can derive the structure and signs of the graph by specific methods (Maurya, 2003a).

A typical dynamic system can be expressed as a set of DEs

$$\frac{\mathrm{d}x_i}{\mathrm{d}t}=f_i\left(x_1,\cdots,x_n\right) \tag{1}$$

where $x_1,\cdots,x_n$ are state variables. By Taylor expansion near normal state, we get

$$\frac{\mathrm{d}x_i}{\mathrm{d}t}\approx f_i\left(x_1^0,\cdots,x_n^0\right)+\sum_{j=1}^{n}\left.\frac{\partial f_i}{\partial x_j}\right|_{x_1^0,\cdots,x_n^0}\left(x_j-x_j^0\right) \tag{2}$$

where $x_1^0,\cdots,x_n^0$ are normal states. Eq. (2) can be written as the following matrix form

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \approx \begin{bmatrix} f_1\left(x_1^0,\cdots,x_n^0\right) \\ \vdots \\ f_n\left(x_1^0,\cdots,x_n^0\right) \end{bmatrix} + \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}\right|_{x_1^0,\cdots,x_n^0} \begin{bmatrix} x_1 - x_1^0 \\ \vdots \\ x_n - x_n^0 \end{bmatrix} \tag{3}$$

The Jacobian matrix

$$\boldsymbol{J} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \tag{4}$$

can be described by an SDG whose signs of arcs are defined as

$$\operatorname{sgn}\left(x_j \to x_i\right) = \operatorname{sgn}\left[\left.\frac{\partial f_i}{\partial x_j}\right|_{x_1^0,\cdots,x_n^0}\right] \tag{5}$$

if the nodes correspond to the state variables. Thus the SDG actually describes the direct influences or sensitivities between state variables.

In practical problems, the systems often have the following form as DEs:

$$a_n\left(\mathrm{d}^n x/\mathrm{d}t^n\right) + \cdots + a_2\left(\mathrm{d}^2 x/\mathrm{d}t^2\right) + a_1\left(\mathrm{d}x/\mathrm{d}t\right) + a_0 x = e \tag{6}$$

where x is the state and e is the disturbance. When $n=1$, it is a first-order system:

$$\left(\mathrm{d}/\mathrm{d}t\right)x = -\left(a_0/a_1\right)x + \left(1/a_1\right)e \tag{7}$$

The step response is shown as Fig. 1(a). An arc is constructed from the node e to x with a sign sgn[$1/a_1$] and a self-cycle on the node x with a sign -sgn[a_0/a_1]. For high-order systems, simplification can be made because the corresponding DE includes different order derivatives of the same variable, which can be avoided for the explicit physical meaning of the model. They can be approximated as first-order systems with delays:

$$\left(\mathrm{d}/\mathrm{d}t\right)x(t-\tau) = -\left(a_0{}'/a_1{}'\right)x(t) + \left(1/a_1{}'\right)e(t) \tag{8}$$

where τ is the equivalent pure delay. Its step response is shown as Fig. 1(b). The method of constructing SDGs is the same as the former one, and the delay can be embodied in dynamic SDGs (Yang & Xiao, 2006a).

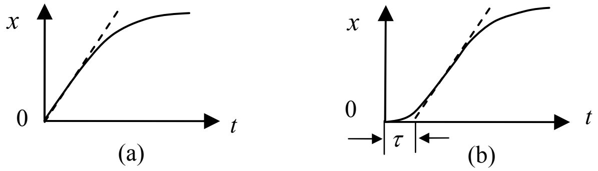

Fig. 1. Step response of different systems. (a) First-order system, (b) High-order system

Algebraic equations are usually included in the mathematical models as constraints which can also be transformed into SDGs (Maurya et al., 2003a) although they are noncausal in nature. Because there may be multiple perfect matchings between equations and variables, the corresponding SDGs may not be unique. Some treatment should be made to screen the unsteady or spurious SDGs (Oyeleye & Kramer, 1988; Maurya et al., 2003a).

For example, a tank system is shown as Fig. 2(a) where L is the level in the tank, R is resistance in the outlet pipe (can be manipulated by a valve), F_1 and F_2 are inlet and outlet flowrates respectively. The system is described as following DAEs:

$$F_2 = \frac{\alpha}{R}\sqrt{L} \tag{9}$$

$$A\frac{\mathrm{d}}{\mathrm{d}t}L = F_1 - F_2 \tag{10}$$

where A is the cross sectional area of the tank, and α is a constant. By the above method, the SDG is set up as Fig. 2(b).

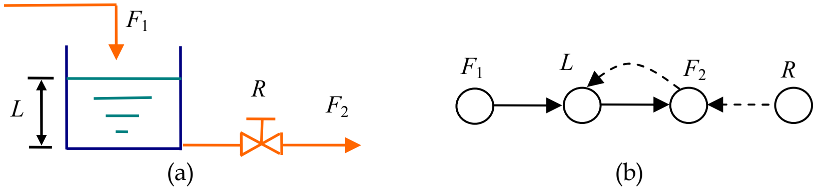

Fig. 2. Tank system and its SDG. (a) Schematic, (b) SDG

2.2.2 SDG modeling by qualitative process knowledge

In more cases, the SDG is established by qualitative process knowledge and experience. Fig. 3(a) shows a tank with one inlet and two outlets with control. The arcs from F_2 to V_2 and L to V_3 in Fig. 3(b) describe the flowrate control and level control respectively. Each control loop can be expressed by a negative cycle in SDG because of the negative feedback action. This qualitative SDG can be obtained directly from process knowledge and does need the exact mathematical equations. Sometimes the qualitative simulation and sensitivity experiments may also help. The SDGs obtained by this method often include indirect causalities besides direct ones, so the graph should be simplified and transformed so that all the arcs stand for direct causalities. Some rules are summarized by Yang & Xiao (2005b).

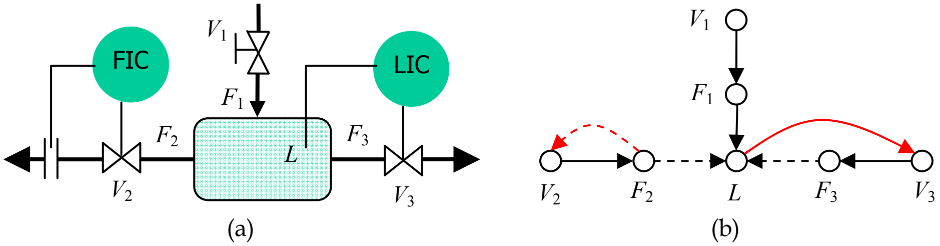

Fig. 3. Schematic and SDG of tank system with controlled flowrates. (a) Schematic, (b) SDG

Besides, P&ID diagrams and other flowsheets are very important topological process knowledge expression that can be standardized in XML (extensible markup language) format. It has been implemented in some commercial software products such as SmartPlant P&ID from Intergraph. The topology or connectivity obtained here includes both material flow and information flow, which are needed for SDG modeling. Although the granularity is entity-based, which is not enough for the variable-based SDG modeling, this kind of topological information is the fundamental of SDG and can be used as references as well (Thambirajah et al., 2009).

The SDG set up by the above methods can be validated by process data. For example, correlation is a necessary condition of causality, so the cross-correlation between every two measured variables can be used to validate the arcs in SDGs, and the directions can also be obtained by shifting the time series to find the maximal cross-correlation. Alternatively, probabilistic measure such as transfer entropy can be used to obtain the causality and directionality (Bauer et al., 2007).

In summary, the main steps of SDG modeling are: (1) Collect process knowledge, especially P&ID diagram and equations. (2) Set up the material flow diagraph by connectivity information between entities. (3) Choose the key variables and give them signs according to the process knowledge. (4) Add control arcs on the diagraph to constitute the SDG skeleton. (5) Add other variables and arcs to form the entire SDG. (6) Simplify and verify the SDG by graph theory. (7) Validate the SDG with process data and sensitivity experiments.

2.3 Hierarchical SDG Description of Large-Scale Complex Systems

Based on the decomposition-aggregation approach, a single-level SDG model can be transformed into a hierarchical model (Gentil & Montmain, 2004; Preisig, 2009). With this model, it is clear and easy to understand the system inherently. As such, the fault analysis method should also be modified from a centralized one to a distributed one.

The whole SDG model can be classified into 3 levels. If the scale of the whole system is too large, then more levels can be established, but 3-level model is enough for most cases. So we take it as a typical pyramid structure. The top level is called system level, where the system is divided into several sub-systems. Sometimes a large-scale system may include several independent sub-systems which can be dealt with separately. Also, in many cases, several components are operated in sequence or in parallel, with no recycle or other kind of interactions existed across the different components, then these components can also be regarded as sub-systems. Of course, if the SDG of the whole system is connected and cannot be separated, then it composes the only sub-system itself.

In the middle level, each control system is regarded as a super-node and the relations between control systems are expressed by arcs among controlled variables and a few important manipulated variables or other variables. The signs of arcs are determined according to the propagation rules to assure the consistency of the paths. The variables in some control loop and not appeared in other part of the system are usually invisible here. The SDG in this level is the backbone of the system which shows the main connectivity in the system flowsheet.

The bottom-level SDGs are the SDG units of all the control systems. The description is the most detailed qualitative expression because it shows the causalities between variables. Since most of the control systems are based on feedback actions, each SDG in this level usually contains at least a loop with various bias nodes attached on them.

2.4 Matrix Explanation of SDG Model

In this section, we look at the SDG model from another viewpoint. An SDG can be also described as an adjacency matrix ***X*** with the element 1/0 denoting the direct adjacency and direction between two variables. Actually it is the transpose of Jaccobian matrix in Eq. (4) with unsigned elements. By matrix computations, reachability matrix ***R*** can be obtained from ***X***, which shows the directed reachability from one variable to another, in which the element 1 means there are at least a path in the corresponding SDG (Jiang et al., 2008). It can be observed that the computation is just another form of graph traversal.

By simultaneous permutation of row and column (with variable order changed), ***X*** can be block triangulated as follows:

$$\boldsymbol{X}' = \boldsymbol{T}\boldsymbol{X}\boldsymbol{T}^T = \begin{bmatrix} \boldsymbol{A}_{11} & \boldsymbol{A}_{12} & \cdots & \boldsymbol{A}_{1m} \\ 0 & \boldsymbol{A}_{22} & \cdots & \boldsymbol{A}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{A}_{mm} \end{bmatrix} \tag{11}$$

Each block in the diagonal denotes a sub-system with a partial order meaning that the sub-system with larger number can not reach the one with smaller number. It can also be explained by the reachability matrix which is definitely also block triangulated with the same order as:

$$\boldsymbol{R} = \left(\boldsymbol{X}' + \boldsymbol{X}'^2 + \cdots + \boldsymbol{X}'^n\right)^{\#} = \begin{bmatrix} \boldsymbol{B}_{11} & \boldsymbol{B}_{12} & \cdots & \boldsymbol{B}_{1m} \\ 0 & \boldsymbol{B}_{22} & \cdots & \boldsymbol{B}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{B}_{mm} \end{bmatrix} \tag{12}$$

where the sign # is the Boolean equivalent (Mah, 1989). If the intersection block $\boldsymbol{B}_{ij}$ is a zero matrix, then the corresponding two sub-systems are independent (no arcs between them), otherwise they are in sequence. Thus we explain the decomposition between the top and middle level.

When we look at the relationship among control systems, we take a control loop as a super-node and add an arc from node *i* to node *j*, if the controller output of controller *i* can directly affect the controlled variable of controller *j* without going through controller output of any other nodes. This SDG as a part of the middle-level SDG is also named as control loop diagraph (Jiang, 2008).

For a feedback control system, there exists a loop in the corresponding SDG. Thus according to the controllability concept, all the variables within the loop are strongly connected, which can be found in the reachability matrix as a block with all the elements are ones.

Let us look at the tank example as Fig. 3 and get the adjacency matrix and reachability matrix by Eq. (12) as follows, both of which are block triangulated.

$$\boldsymbol{X} = \left[\begin{array}{cc:cc:ccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hdashline 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ \hdashline 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array}\right], \quad \boldsymbol{R} = \left[\begin{array}{cc:cc:ccc} 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ \hdashline 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ \hdashline 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{array}\right] \tag{13}$$

where the variable order is V_1, F_1, V_2, F_2, V_3, F_3, L. They are divided into 3 groups: inlet (V_1 and F_1), one outlet with flowrate control (V_2 and F_2) and another outlet with level control (V_3, F_3 and L). The elements of $\boldsymbol{R}_{22}$ and $\boldsymbol{R}_{33}$ are all ones because they are control loops, and the elements of $\boldsymbol{R}_{23}$ are ones showing the flowrate controller influences the level controller. Hence the control loop diagraph is consisted of two nodes corresponding to the two controllers and an arc corresponding to the influence between them. Moreover, if the variable order is changed to put V_1 and V_2 meaning the two controller outputs at the last, the corresponding block is just the adjacency matrix of the control loop diagraph. This is a useful property that links the concepts of SDG, control loop digraph and the matrices.

Matrix explanation helps us understand the SDG concept and its potential in applications. In fact, some results, such as propagation rules, are derived from matrix description.

3. Inference Approaches Based on SDGs

3.1 Fault Propagation Rules

Based on the SDG description, the fault propagation can be described qualitatively. There are two basic principles:

Proposition 1: The fault is propagated along the consistent paths.

Proposition 2: The node signs are determined by nodal balance, i.e. the sign on each node must be equal to the net influence on the node:

$$\psi(v_j) = \sum_i \varphi(a_{ij}) \cdot \psi(v_i) \tag{14}$$

where the qualitative operation rules are as Table 1. Due to the loss of quantitative information, some signs can not be determined shown as '?' in the table, which causes the uncertainty in the solutions.

No.	sgn[x]	sgn[y]	sgn[x]+sgn[y]	sgn[x]·sgn[y]
1	0	sgn[y]	sgn[y]	0
2	±	sgn[x]	sgn[x]	+
3	±	-sgn[x]	?	-

Table 1. Qualitative operation rules

The logic on a node in SDGs is OR in nature because any input deviation can result in the node sign. In some cases, however, there are other types of logics, for example, the logic is AND, XOR or high/low-selective, or arcs or nodes are conditional, some necessary logic nodes should be added to the SDG (Yang & Xiao, 2007).

Proposition 1 can be easily understood. By testing the consistency one can find the fault propagation paths based on the measurements, which form a sub-graph of the original SDG, called cause-effect graph (Iri et al., 1979). On the other hand, one can predict the next step response based on the measured and assumed variables.

Proposition 2, however, may have some limitations because it is only suitable for the dynamic trends near the initial state. When a fault occurs, the response of variables can be divided into three stages – initial, intermediate, and final responses. In large-scale complex systems, the intermediate response is very complex, but in most cases, we concentrate only on the initial and final stages. For stable systems with fixed input, the final response is a steady state. Thus the input and exogenous disturbances are assumed as step functions to show abrupt changes.

Initial response is the first response just after the exogenous input changes. In dynamic systems expressed by DAEs, initial response is the nonzero response of system variables predicted by propagation through all the shortest paths in the corresponding SDG if we define the length of arcs in AE and DE portion by 0 and 1 respectively (Maurya, 2003a). Final response is the steady states of variables obtained after the dynamic period ends. It can be solved simply by setting the derivatives as zeros in DE portion of DAE. For the obtained AEs, the final response can be predicted by propagation through all the directed (acyclic) paths in the corresponding SDG. However, there may exist more than one perfect matching between equations and variables, thus there may exist more than one SDG corresponding to the AEs. If there is only one perfect matching, the above method is correct; otherwise, the result may be wrong because the results based on different perfect matchings are inconsistent. There is an exception, however, if an SDG corresponding to a perfect matching contains only negative cycles, then any perfect matching (for which the SDG contains only negative cycles) can be chosen and the final response can be decided using the above method (Maurya, 2003a).

3.2 Control Action Influences on Fault Propagation

3.2.1 SDG description and fault propagation analysis of single control loop

Control actions should be considered particularly because they are forced actions that are different from process property itself and they may cause the truncation or misleading of fault propagation. We discuss this problem using the general methods and obtain some special results (Maurya, 2003b, 2006).

In the bottom level, SDG models are established for all kinds of control systems among which the most common and basic one is single PID loop shown as Fig. 4. The deviation *e* of the set point r and the measurement x_{m} of the controlled variable x, is inputted into the controller whose output u goes to the actuator and thus effects the controlled plant through the manipulated variable q. Hence they compose a closed loop. Because the controlled variable may be affected by some disturbances or be coupled with other system variables, the exogenous plant and variable x_j are also added. Assume that controlled plant and the controller are both linear amplifiers, i.e. proportion elements, with the positive gain k and k_{v} respectively. The control law of PID controller is:

$$\begin{cases} u = u_{\mathrm{P}} + u_{\mathrm{I}} + u_{\mathrm{D}} \\ u_{\mathrm{P}} = k_{\mathrm{c}} e \\ (\mathrm{d}/\mathrm{d}t) u_{\mathrm{I}} = k_{\mathrm{c}} e / \tau_{\mathrm{I}} \\ u_{\mathrm{D}} = k_{\mathrm{c}} \tau_{\mathrm{D}} \cdot (\mathrm{d}e/\mathrm{d}t) \end{cases} \tag{15}$$

where, k_{P} is the positive proportion parameter, τ_{I} and τ_{D} are integral and differential time constant respectively.

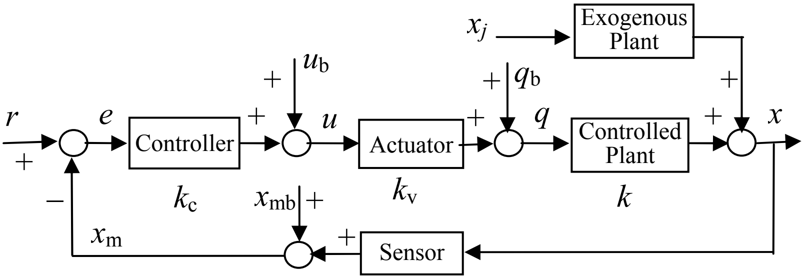

Fig. 4. Block diagram of a feedback control loop

According to the control law, the DAEs of the system are as follows:

$$x_{\mathrm{m}} = x + x_{\mathrm{mb}} \tag{16}$$

$$e = r - x_{\mathrm{m}} \tag{17}$$

$$u_{\mathrm{P}} = k_{\mathrm{c}} e \tag{18}$$

$$(\mathrm{d}/\mathrm{d}t) u_{\mathrm{I}} = k_{\mathrm{c}} e / \tau_{\mathrm{I}} \tag{19}$$

$$u_{\mathrm{D}} = k_{\mathrm{c}} \tau_{\mathrm{D}} \cdot \mathrm{d}e/\mathrm{d}t \tag{20}$$

$$u = u_{\mathrm{P}} + u_{\mathrm{I}} + u_{\mathrm{D}} + u_{\mathrm{b}} \tag{21}$$

$$q = k_{\mathrm{v}} u + q_{\mathrm{b}} \tag{22}$$

$$x = kq + a_j x_j \tag{23}$$

where subscript ‘b’ denotes bias. There are two perfect matchings between the equations and variables in AE portion, shown as Table 2, whose corresponding SDGs are shown as Fig. 5, in which the nodes with shadow are deviation nodes, arrows with solid and dotted lines denote signs “+” and “-” respectively. It is noted that the node $\mathrm{d}e/\mathrm{d}t$ is an individual node with special function, although it is the derivative of e. In applications, we generally assume that all changes on nodes are step functions, because the SDGs are only used to analyze the qualitative trends. Hence $\mathrm{d}e/\mathrm{d}t$ can be also replaced by e, but its effect is limited in initial response. Here the effect of $\mathrm{d}e/\mathrm{d}t$ on u_D is the same as the effect of e on u_P, but with shorter duration.

Equations	Matched variables in perfect matching No. 1	Matched variables in perfect matching No. 2
(16)	x_m	x
(17)	e	x_m
(18)	u_P	e
(20)	u_D	u_D
(21)	u	u_P
(22)	q	u
(23)	x	q

Table 2. Perfect matchings between the AEs and variables

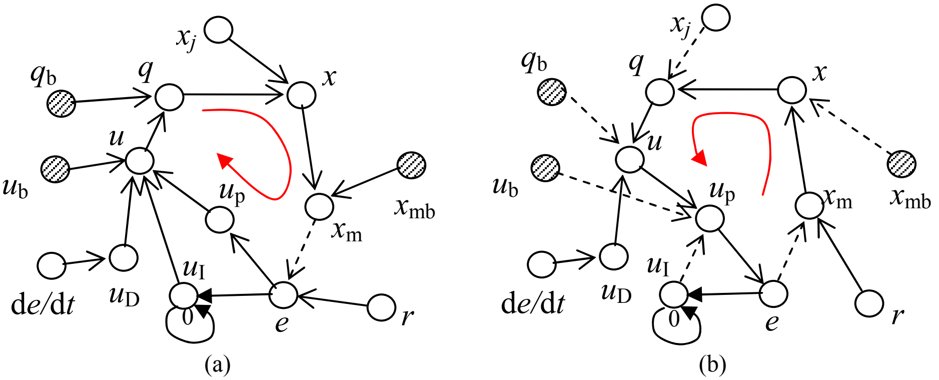

Fig. 5. Two SDGs of the PID control loop. (a) Case 1 (corrected), (b) Case 2 (spurious)

Eq. (23) describes the controlled plant, thus the arc direction should be from q to x according to the physical meaning, which shows the cause-effect relationship, so the case of Fig. 5(b) is removed. Moreover, if the plant shows some dynamic characteristic, for example, the left-hand of the equation is $\mathrm{d}x/\mathrm{d}t$, then the equation becomes a DE, hence there is only one perfect matching, and the case of Fig. 5(b) does not exist any more. Using Fig. 5(a), the initial response can be analyzed, for example, if the set point r increases, e, u_P, u, q, x and x_m will become "+" immediately, and u_I will become "+" gradually because the arc from e to u_I is a DE arc. This propagation path $r\rightarrow e\rightarrow u_\mathrm{P}\rightarrow u\rightarrow q\rightarrow x\rightarrow x_\mathrm{m}$ is consistent with the actual information transfer relations. Thus when we only consider the initial response of the system, the SDG of this control loop is obtained by transforming the blocks and links in block diagram into nodes and arcs while keeping the direction. However, in this example, no matter whether the case of Fig. 5(b) is reasonable, the analysis results of initial response by the two SDGs are the same because there are no positive cycles within them. We summarize the following rule:

Rule 1: The fault propagation path of the initial response in a control loop is the longest acyclic path starting from the fault origin in the path "set point → error → manipulated variable → controlled variable → measurement value → error", which is consistent with the information flow in the block diagram.

Final response is easier. The left-hand side of Eq. (19) is zero, so $e = 0$ in the steady state, which can be obtained from the concept. Hence u_P and u_D are both zeros. The above DAEs can be transformed into:

$$x_m = x + x_{mb} \tag{24}$$

$$x_m = r \tag{25}$$

$$u = u_I + u_b \tag{26}$$

$$q = k_v u + q_b \tag{27}$$

$$x = kq + a_j x_j \tag{28}$$

Now the perfect matching is exclusive and the corresponding SDG is shown as Fig. 6 that is the simplification of Fig. 5(b). There are two fault propagation paths: $r \to x_m \to x \to q$ and $x_j \to q \to u \to u_I$. If the set point r increases, then x_m, x, q, u and u_I will all increase in the steady state as long as the control action is effective. However, if only x_{mb} increases, then x_m will not be affected, but x will increase, that is the action of the control loop. We find that the Fig. 5(b) also makes sense for it reflects the information transfer relation in steady state. From the viewpoint of physical meaning, when control loop operates, the controlled variable is determined by the set point, and the controller looks like an amplifier with infinite gain, whose input equals to zero and whose output is determined by the demands. This logical transfer relation is opposite to the actual information relation.

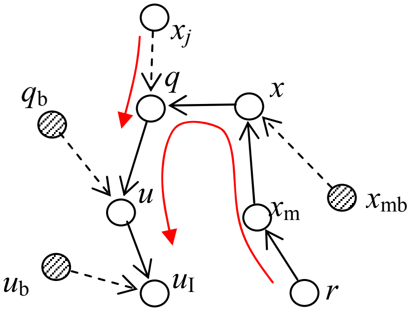

Fig. 6. Steady-state SDG of a PID control loop

Because the D action is only effective in the initial period, the fault propagation path of PI control is the same as the above one. Because of I action, some variables show compensatory response, for example, the response of x_m due to x_{mb} is limited in the initial stage. If there is only P action, then e is not zero in the steady state, thus u_I and related arcs in Fig. 5(a) are deleted, and both the initial response and steady-state response can be analyzed with this graph.

The rule of fault propagation analysis in steady state can be summarized as follows:

Rule 2: The fault propagation path of the steady-state response in a control loop is the path "set point → measurement value → controlled variable → manipulated variable" and "exogenous variable → manipulated variable".

When control loop operates, the above analysis shows the fault propagation principles due to the output deviation of sensor, controller, actuator and other exogenous variables. When control loop does not operate, there are two cases: (1) structural faults, e.g. the failure of sensor, controller or actuator causes the break of some arcs and the control loop becomes open, (2) excessive deviation causes the controller saturation, leading to the I action cannot eliminate the residual and let $e = 0$, which is similar with the P action case.

3.2.2 SDG description and fault propagation analysis of various control systems

Based on the above analysis of PID control loop, other control loops can be modeled as SDGs by the extension, combination, or transformation of the above SDG. Fault nodes are added according to the actual demands. Based on these models, fault propagation can also be shown explicitly.

Feedforward control is a supplement of feedback control. It is very familiar in actual cases, but it is easy to be treated according to the foregoing methods because it composes paths but not cycles, not leading to multiple perfect matchings.

Split-range control means the different control strategies are adopted in different value intervals. Here the sign of the arcs or even the graph structure may change with the variable values, which is realized by several controllers in parallel connection. This case is very hard for SDG to deal with. We have to do some judgments as making inference, and modify the structure or use conditional arcs to cover all the cases (Shiozaki et al., 1989).

Cascade control can be regarded as the extension of single loop case. It can be solved directly by AEs, or by the combination of two single loops. For example, the cascade control system in Fig. 7 has the steady-state SDG as shown in Fig. 8, where the controlled variable of the outer loop u_1 is the set point of the inner loop r_2.

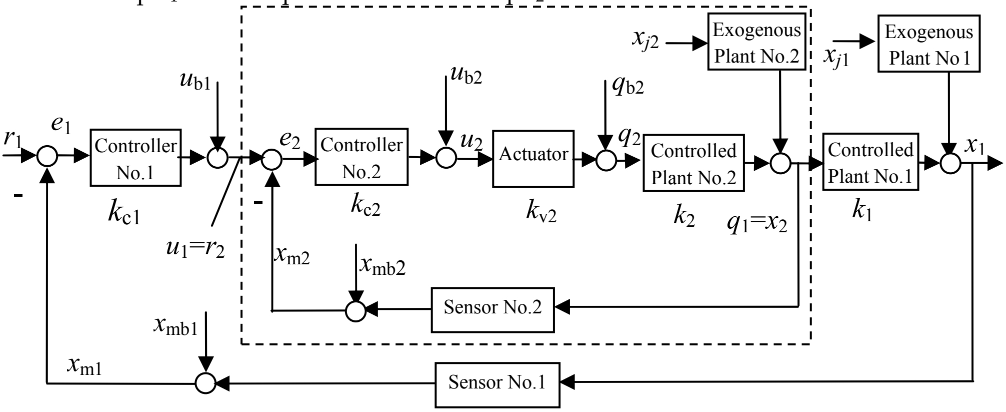

Fig. 7. Block diagram of a cascade control system

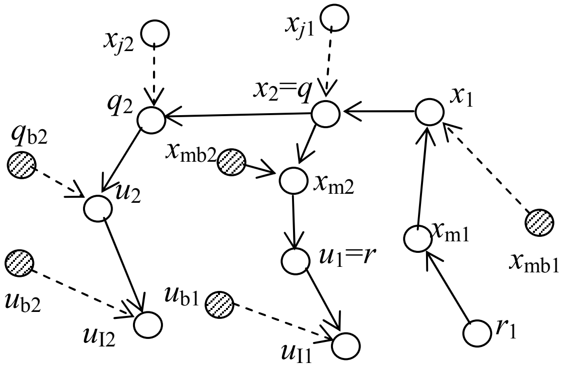

Fig. 8. Steady-state SDG of a cascade control system

Similar control methods are ratio control, averaging control, etc. Fig. 9 is a dual-element averaging control system whose objective is to balance two variables – level and flow, the block diagram of which is shown as Fig. 10. $P_x=P_L-P_F+P_S+c$, where P_x is the pressure signal of the adder output, P_L is the level measurement signal, P_F is the flow measurement signal, P_S is a tunable signal of the adder. In the simplest case, flow process and its measurement are both positive linear elements, and the level process is a negative linear element, so the steady-state SDG is shown as Fig. 11. Although there are several perfect matchings, SDG has only a negative cycle, thus we can analyze the fault propagation principle through the directed paths.

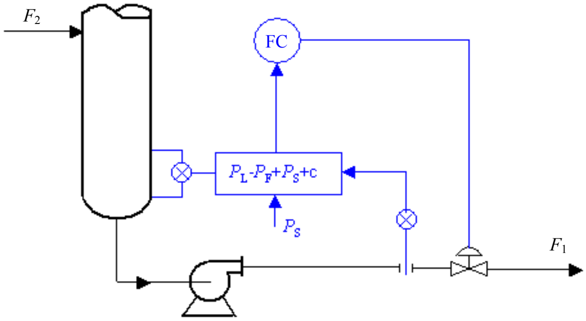

Fig. 9. A dual-element averaging control system

Thus we conclude:

Rule 3: The fault propagation path in a control system in steady state can be combined from the ones of single-loop by combining the same nodes and adding arcs by transforming AEs. In an industrial system, control systems play a special but important role. They compose information flow cycles in initial response but result in different flow in steady state response. Fig. 12 shows a system with a control loop. According to the above rules, the bias in x_1 propagates along the forward path (blue) in initial response while against the feedback path in steady state response.

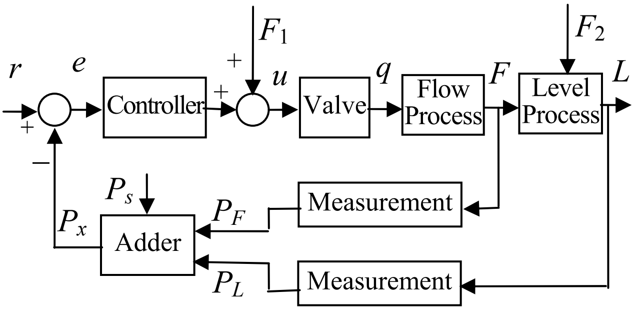

Fig. 10. Block diagram of a dual-element averaging control system

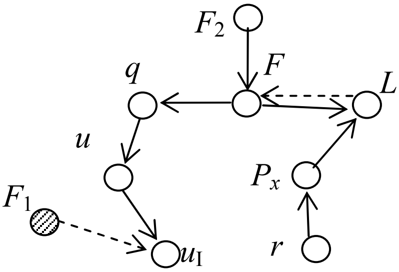

Fig. 11. Steady-state SDG of a dual-element averaging control system

The bias in x2 propagates along two paths until x3 and x4 in initial response, while PV and x4 restore to normal in the steady state because the steady state SDG changes the structure and directionality of the graph and thus PV becomes a compensatory variable.

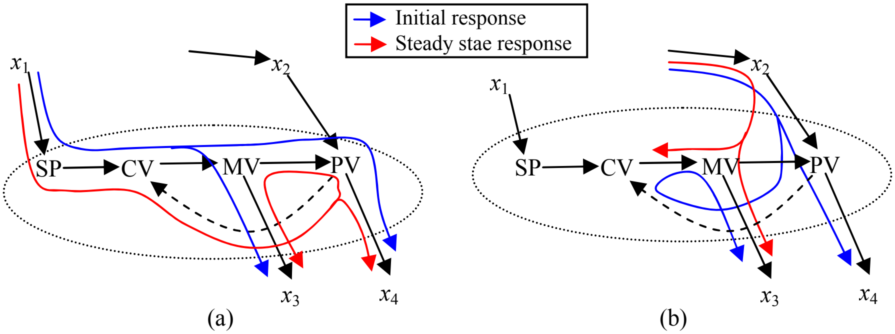

Fig. 13. Control system's effect on fault propagation in a system. (a) Bias in x_1, (b) Bias in x_2

3.2.3 Example

In a boiler system, the three-element control of the boiler water level is widely used, in which the main controlled variable is water level. If we take steam flow and inlet flow into account, the control system is a feedforward-cascade system, as shown in Fig. 13. In the initial stage of the disturbance, the SDG is shown as Fig. 14(a), which can be derived by original DAEs. Certainly the initial fault influence follows this SDG. The control action, however, is enrolled and some deviations are restored to the normal region after a complex intermediate process. If the control action is successful, the fault may be blocked in the control loop and does not spread any more. But for some kinds of faults, the situation is different, because the control action makes the fault propagation path change. According to the method in the foregoing sections, we can construct the backbone (ignoring the bias nodes) of the steady-state SDG model as Fig. 14(b) which is quite different from Fig. 14(a). Similar with Fig. 9, other fault nodes can be added to Fig. 14(b) and thus we can find the steady-state fault propagation paths.

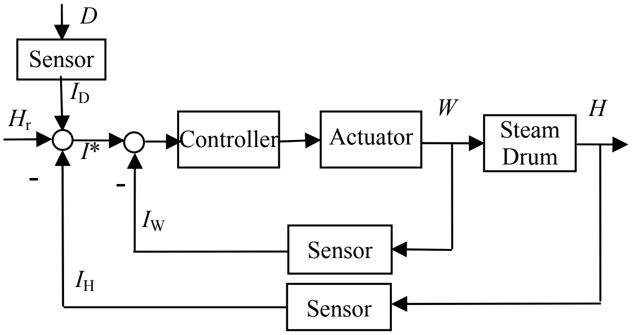

Fig. 13. Three-element control system

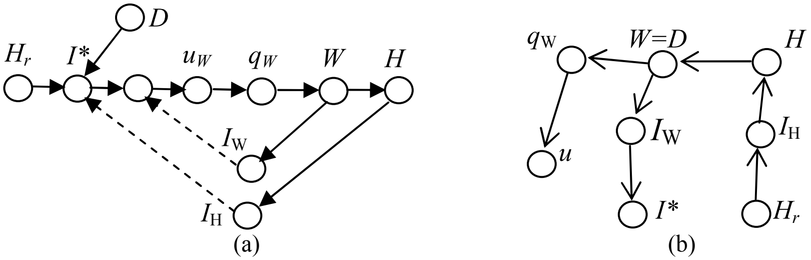

Fig. 14. SDGs of the three-element control system. (a) Initial response, (b) Steady state

3.3 Inference Approaches

In safety area, fault diagnosis and hazard assessment, especially hazard and operability analysis (HAZOP) are two different tasks. The former is to correctly find and identify the fault origin that is the cause leading to the symptom when fault occurs. It is based on measurements and is real-timed. While the latter, hazard assessment is to an off-line analysis whose purpose is to find the possible hazards due to all various causes. For this reason, we assume a series of departure nodes as fault origins, then analyze the possible consequences that are all the triggered departure nodes. Both fault diagnosis and hazard assessment need the interior mechanism of the system to express how the faults propagate. Thus the SDG model can be employed.

3.3.1 Graph traversal approach

The most common algorithm for searching the fault origin is depth-first traversal on the graph (Iri et al., 1979), which is a kind of efficient fault inference for both the single and multiple fault origin cases (Zhang et al., 2005). Its theoretical basis is nodal balance in Eq. (14). A depth-first traversal algorithm constructs a path by moving each time to an adjacent node until no further arcs can be found that have not yet been visited, the implementation of which is a recursive procedure.

For HAZOP purpose, forward traversal is applied from the assumed origin to predict all the variables based on consistency, which is deductive reasoning (Venkatasubramanian et al., 2000; Yang & Xiao, 2006b). For the fault detection purpose, backward traversal is applied within the causal-effect graph to find the maximal strongly connected component (Iri et al., 1979), which is abductive reasoning. Actually, the whole procedure includes two steps:

Step 1: Trace the possible fault origins back along the arcs.

Step 2: Make forward inference from these nodes to screen the candidates to choose which one is the real or most probable fault origin.

The time complexity of a traversal search is $O(n^2)$ in which n denotes the node number in the graph. When the system scale increases, the time for a traversal is too long to meet the demands of fault detection. Thus the model structure should be transformed from a single-level one to a hierarchical one. By this way, the search is first performed in the higher level to restrict the fault origin in a sub-system. Then the search is performed in the sub-graph of this sub-system.

For the hierarchical model, hierarchical inference from top to bottom is obtained naturally. The graph traversal is performed firstly in the higher level finding the possible super-node that includes the fault origin. Next perform the graph traversal in the lower level to restrict the possible location of the root cause. Assume the sub-system contains m control systems, and each control system contains k variables, then the time complexity of a traversal in a single-level model is $O(m^2k^2)$, and the time complexity in a 2-level model is $O(m^2+k^2)<<O(m^2k^2)$. Thus the fault analysis in a hierarchical model has much higher efficiency.

Here the number of fault origin is assumed to be only one, that is, the reason that leads to the fault is only one (Iri et al., 1979). This is reasonable because multiple faults seldom appear at the same time (Shiozaki et al., 1985). For multiple fault origin cases, minimal cut sets diagnosis algorithm was presented (Vedam & Venkatasubramanian, 1997), where all possible combinations of overall bottom events should be input into the computer to explore and those which make the top events appear are the cut sets. This algorithm has the distinct disadvantage of low efficiency because of exponential explosion.

3.3.2 Other improved approaches

In order to utilize the system information more sufficiently, Han et al. (1994) used fuzzy set to improve the existing models and methods, but their method is not so convenient for on-line inference and is not applicable for dynamical systems. Some scholars introduced temporal evolution information such as transfer-delay (Takeda et al., 1995; Yang & Xiao, 2006a) and other kind of information into SDG for dynamic description. Probability is also proposed to model the system, which uses conditional probabilities of fault events to describe causes and effects among variables (Yang & Xiao, 2006c). Hence the inference is respect to the fault probability.