Concurrent Engineering
http://cer.sagepub.com/

Modeling the Product Development Process as a Dynamic System with Feedback

Hong-Zhong Huang and Ying-Kui Gu

Concurrent Engineering 2006 14: 283
DOI: 10.1177/1063293X06073134

CONCURRENT ENGINEERING: Research and Applications

Modeling the Product Development Process as a Dynamic System with Feedback

Hong-Zhong Huang[1,*] **and Ying-Kui Gu**[2]

[1]*School of Mechatronics Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, 610054, PR China*
[2]*School of Mechanical & Electronical Engineering, Jiangxi University of Science and Technology, Ganzhou, Jiangxi, 341000, PR China*

Abstract: The modeling of effective product development process can help manage the overall process efficiently and help organize a multifunctional team to develop products in a concurrent and cooperative manner. In this article, a product development process is viewed as a dynamic system with feedback based on the feedback control theory, and a dynamic system model and its design structure matrix are developed. The model and its design structure matrix can be further divided to capture the interaction and feedback of design information. A fuzzy evaluation method is presented for the evaluation of the performance of a dynamic development process; this allows a development process to be optimized based on re-organizing design constraints, re-organizing design processes, and re-organizing designer's preferences. An example is provided to illustrate the proposed model.

Key Words: process modeling, dynamic model, design structure matrix, process evaluation, information feedback.

1. Introduction

Product development is not only a process of resource allocation, but also a screwy evolution process subject to development constraints. In such a process, the design information is transformed and accumulated. Therefore, the modeling of a product development process is the key factor in determining the success of both individual product development strategies and the overall long-term industrial competitiveness [1]. Effective modeling of the product development can be used to optimize a development process and the accumulation of the design information. Effective modeling can also improve the degree of concurrency, optimize the design structure, improve the product quality, and cut the development cost and time. However, in practice, a development process has dynamic uncertainty, time overlapping, and constraint coupling, which coupled with the imperfection and fuzziness of the design information [2] lead to fluctuations in the resource allocation and the accumulation of information that hampers the process.

Moreover, the product development itself is a complex, dynamic, and uncertain system with feedback. The product development rather resembles a network (process net), where processes are highly interconnected, including feedback loops and interactions at various hierarchical levels [3]. As a feedback control system and a product development process share the common feature of information feedback, a method based on the feedback control can be very effective for modeling a product development process. In each development stage, a change in the external environment may have some influence on the whole development process. However, such influence may not bring negative impact on the product development. The influence from the external environment can prompt a system to make a fast response to an external change such that possible disturbance is minimized. The effect of the external influence can in turn either make the balance point of a development process transferring from its initial point to a new point or be optimized at the initial point. The new balance point may be better than the old one and therefore makes a product more marketable and competitive.

Traditional development process models [4,5] statically describe a process, resources, targets, and the scope with estimates of activity durations and the precedence relationships describing the network flow of activities. These methods are limited by the use of an indirect project measure and by bundling the characteristics and relationships among scope, resources, and processes in each activity into a single duration estimate. They also tend to ignore iterations or require that iterations are

---

*Author to whom correspondence should be addressed.
E-mail: hzhuang@uestc.edu.cn

1063-293X/06/04 0283–9 $10.00/0 DOI: 10.1177/1063293X06073134

implicitly incorporated into duration estimates and precedence relationships [6]. For example, if a defective product definition is released to designers who discover the defect after the design has begun, information of the defect must be fed back from the design phase to the product definition phase. The product definition must be revisited, thereby, increasing both time and cost of the development.

Iteration is a fundamental characteristic of a development process [7]. Steward [4] developed the design structure matrix to model the information flow of design tasks and to identify their iterative loops. Eppinger et al. [7] extended Steward's work by explicitly modeling the information coupling among tasks and by investigating various strategies for managing the entire development procedure. Several analytical models have been developed to represent an iterative design process. These include sequential iteration models [8], parallel iteration models [9], and overlapping models [10]. However, each model also has its own limitations. Owing to the limitation of the analytical approaches, the sequential iteration models cannot handle resource constraints or general project networks with parallel and overlapping tasks (or paths). The parallel iteration models analyze important aspects of concurrent engineering but use highly simplified assumptions. The two-task overlapping models provide an optimal way to reduce the time of two sequential tasks with the interface of unidirectional information transfer. However, the concept cannot be easily applied to multiple tasks, especially for tasks involving multiple paths with iteration. In addition, no significant work has been found on resolving issues of resource over-allocation in the overlapped and coupled project networks where tasks repeat according to a probabilistic rule [11].

A more suitable description of development dynamics must include iterative flows of work, distinct development activities, and available work constraints. Cooper [12] followed by others [6] modeled two development activities by distinguishing between the initial completion and the rework. Ford et al. [6] expanded this approach to model three development activities (the initial completion, the required rework, and the optional rework to improve quality). System dynamics models of iterative flows have been evolved from single flows of accurate work, which is slowed by implicit iteration through separating streams of correct and flawed tasks, to more realistic closed-loop flows [6].

Some of the other new theories and methods have been developed recently to emphasize the dynamic nature of a product development process, such as the dynamic feature of concurrent engineering [13], manufacturing informatics [14], information theory of engineering [14], intelligent real design [15], integrated concurrent schedule, microcosmic development process modeling, micro-design-cycle, a Petri-net based collaborative engineering process model [16], and a DSM-based dynamic process simulation model [11]. A design structure matrix [8] has been used to map development phases as well as study both time/quality tradeoff decisions and the variability in cycle times. The dynamic consequences of the iteration among development phases on cycle time have been addressed directly with the design structure matrix [7]. Although the research on design structure matrix demonstrates the results of the iteration between phases, the underlying processes that drive the cycle time have not been described or modeled [6].

This study aims to analyze the dynamic characteristics of a product development process, to model a product development process as a dynamic system with feedback, and to develop a suitable method for evaluating the performance of a product development process. Key issues related to a product development process are also discussed. Section 2 gives important definitions related to a product development process. Section 3 presents a dynamic model with feedback and a design structure matrix for a product development process. Section 4 provides a fuzzy evaluation method for a development process. An illustrative case is given in Section 5. Results are discussed in Section 6. Finally, conclusions are provided in Section 7.

2. Definitions Related to Product Development Process

Definition 2.1: Process

A process is a basic unit of activities that are carried out during a product's life cycle. It is also a set of activities whose physical meanings vary continuously over time. A process is denoted by a set P and defined as

$$P = \langle V, U, \Phi, \Delta, C, A, U_s, U_t \rangle, \tag{1}$$

where V is a set of process parameters, U is a set of process units, Φ is the process control flow of this process, Δ is the process data flow of this process, C is a constraint set that can control the flow of this process, A is a set of activities of this process, U_s is a set of initial process units, $U_s \subseteq U$, $U_s \neq \Phi$, and U_t is a set of final process units, $U_s \subseteq U$, $U_t \neq \Phi$.

Definition 2.2: Constraint evaluation point

A constraint evaluation point is a constrain set that can satisfy the proceeding of one process. Constraint variable, constraint function, and constraint value are the three basic elements of a constraint evaluation point.

In general, a constraint evaluation point is denoted by a set μ and defined as

$$\mu = \langle V, F, C, I, V_z, P \rangle, \tag{2}$$

where V is a set of constraint variables. Based on the function of variables in a constraint evaluation point, V can be classified into three different types as follows.

Decision variable V_α: V_α is the variable that each design perspective can be allowed to make independent decisions on this variable.

Intermediate variable V_β: V_β is the variable that is determined by processing decision variables.

Performance variable V_λ: V_λ is the variable that is dependent on decision variables and is used to measure the performance of a system.

F is a set of constraint functions. F can be classified into two types, i.e., the qualitative constraint function and the quantitative constraint function. C is a set of conditions, and is a sub-set of F, i.e., $C \subseteq F$. C can decide one process flow. I is a set of interface constraints, which can be used to embody the relations among constraint evaluation points. V_z is a set of constraint values. V_z consists of two elements, i.e., $V_z = \langle \text{TRUE, FALSE} \rangle$. TRUE denotes a status that constraints are satisfied. FLASE denotes that constraints are not satisfied. P is a constraint pointer, and P can be used to orient the constraints.

Definition 2.3: Process data flow [17]

Process data flow defines the data relation between two processes. Let C be a set of process behaviors, V be a set of process parameters, $v \in V$. DOM(v) denotes the type of v. Let $i(C)$ be a set of input parameter, $o(C)$ be a set of output parameter, such that C is a mapping, i.e.,

$$C : \underset{v \in i(C)}{\times} \mathrm{DOM}(v) \to \underset{v \in o(C)}{\times} \mathrm{DOM}(v) \tag{3}$$

Let P be a set of process preconditions, and it can be viewed as a mapping

$$P : \underset{V \in I(P)}{\times} \mathrm{DOM}(v) \to \{\text{TRUE, FALSE}\} \tag{4}$$

Therefore, process data flow is a mapping, and can be denoted by

$$\Delta : C \times (C \cup P) \to \bigcup_{A \in C,\, B \in C \cup P} (o(A) \times i(B)) \tag{5}$$

Definition 2.4: Process control flow

Process control flow defines the control relationship between two processes. Let C be a set of process behaviors and P be a set of process preconditions (i.e., the conditions that satisfy the constraint evaluation points and the proceeding of process). The set of process control flow can be denoted by [17]

$$\Phi \subseteq C \times C \times P. \tag{6}$$

Let projective mapping π be

$$\pi_{i_1 \ldots i_k} : M_1 \times M_2 \times \cdots \times M_n \to M_{i_1} \times \cdots \times M_{i_k}, \quad i_j \in \{1, 2, \ldots, n\} \tag{7}$$

such that $\pi_1(\phi)$ is the leader behavior of the process control flow ϕ. $\pi_2(\phi)$ is the following behavior of ϕ. $\pi_3(\phi)$ is the transform condition of the process control flow ϕ, $\phi \in \Phi$. Process control flow can be decided by its leader behavior and the following behavior only, i.e.,

$$\forall \phi, \phi' \in \Phi : \pi_1(\phi) = \pi_1(\phi') \wedge \pi_2(\phi) = \pi_2(\phi') \Rightarrow \phi = \phi' \tag{8}$$

Therefore, the set of transform conditions of entering and leaving the process behavior c can be denoted respectively by

$$\Gamma_{\Rightarrow}(c) = \pi_3\{\phi \in \Phi | \pi_2(\phi) = c\} \tag{9}$$

$$\Gamma_{\Leftarrow}(c) = \pi_3\{\phi \in \Phi | \pi_1(\phi) = c\}. \tag{10}$$

3. A Dynamic Model of the Product Development Process with Feedback

A product development process and a traditional feedback control system share one fundamental feature: information feedback. Therefore the traditional feedback control theory can be applied to model effectively a dynamic product development process. By doing so, a development team can utilize and exchange the design information effectively.

3.1 Dynamic Characteristics of the Product Development Process

Because of the limitation of people's cognition and because of the complexity of design itself, there are many uncertain factors included in each development stage. These uncertain factors bring disturbance to a development system, making the system unstable, or even stop or suspend the development process. The direct consequence of the instability of a development process is that it can affect the product quality, prolong the development time, as well as increase the development cost. From the viewpoint of the control theory, one can see that because of the information interaction and the feedback, a dynamic system can maintain its equilibrium. Therefore, in order to reduce the instability, the product development can be viewed as a dynamic system with feedback.

Based on the satisfaction of the constraint evaluation point defined in Section 2, a process has three different states as follows:

(i) *Proceeding state*. The process control flow, the process data flow, and the constraint evaluation point defined in Section 2 are satisfied at the same time. A development process can be executed.
(ii) *Stop state*. At least one of the constraint evaluation points, the process control flow, the process data flow is not satisfied. A development process is stopped or suspended.
(iii) *Feedback state*. After a process is evaluated, the evaluation information is fed back to all upstream development stages to make some adjustments.

3.2 The Structure Matrix of the Dynamic Process Model

One of the most important families of models is that based on the design structure matrix (DSM). The DSM method assumes that each design task can be modeled as an information processing task that utilizes and creates information. The output information from one task becomes the input to another task. The input–output relationships may include cycles, which indicate the need for iterations. Tasks in the matrix may be re-sequenced [18]. In this study, a new DSM is defined and used to describe a dynamic development process with feedback.

The dynamic model of a product development process can be expressed by a structure matrix as follows:

$$\begin{array}{c} \\ i-2 \\ i-1 \\ i \\ i+1 \\ i+2 \end{array}\begin{array}{c} \begin{array}{ccccc} i-2 & i-1 & i & i+1 & i+2 \end{array} \\ \left[\begin{array}{ccccc} * & a_{i-1}^{i-2} & a_{i}^{i-2} & a_{i+1}^{i-2} & a_{i+2}^{i-2} \\ b_{i-2}^{i-1} & * & a_{i}^{i-1} & a_{i+1}^{i-1} & a_{i+2}^{i-1} \\ b_{i-2}^{i} & b_{i-1}^{i} & * & a_{i+1}^{i} & a_{i+2}^{i} \\ b_{i-2}^{i+1} & b_{i-1}^{i+1} & b_{i}^{i+1} & * & a_{i+2}^{i+1} \\ b_{i-2}^{i+2} & b_{i-1}^{i+2} & b_{i}^{i+2} & b_{i+1}^{i+2} & * \end{array}\right] \end{array}. \tag{11}$$

This matrix reflects the interdependence and the feedback relationships among sub-processes in a product development process, where

$$a_j^i = \begin{cases} 1, & \text{the ending of process in } i\text{th row is the} \\ & \quad \text{input of the process in } j\text{th column } (i \neq j) \\ 0, & \text{the ending of process in } i\text{th row has no direct} \\ & \quad \text{relations with the process in } j\text{th column} \end{cases}$$

$$b_j^i = \begin{cases} 1, & \text{the information of process in } i\text{th row feeds back} \\ & \quad \text{to the process in } j\text{th column directly } (i \neq j) \\ 0, & \text{there is no information of process in } i\text{th row} \\ & \quad \text{feedback to the process in } j\text{th column directly} \end{cases}$$

The value of a_j^i affects the serial or the parallel mode of a design process. The value of b_j^i decides the feedback state of the information and the number of feedback loops. The proceeding mode of process, the number of feedback loops, the feedback stage, and the feedback location have a direct influence on the product development.

The design structure matrix can be divided into two parts according to the diagonal. The upper-right part describes the interdependence relations among processes, i.e., the correlative information among different processes. The information can be the number of parameters of process output or the communication quantity among processes. The lower-left part describes the feedback relations among processes. The feedback information can be the number of feedback loops or the number of feedback parameters.

The design structure matrix can be further divided. As shown in Figure 1, the feed-forward matrix M_v is formed by the extension of the matrix M_i and the feedback matrix M_b is formed by the extension of matrix M_k, respectively. M_i and M_k are composed of process i_1, i_2 and process k_1, k_2, respectively.

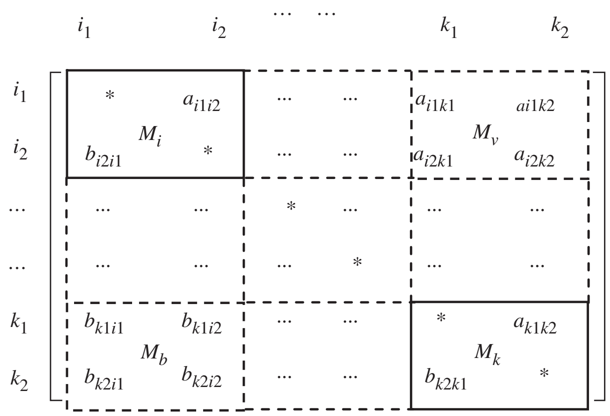


Figure 1. The division of structure matrix.

The matrix M_v includes the input information such that the matrix M_k depends on M_i. The matrix M_b includes the information that is fed back to matrix M_i from M_k. The elements of M_v and M_b can be used to determine the interdependence and the feedback relations among processes i_1, i_2, k_1, and k_2. The matrix M_i is independent of M_k if the elements of M_v and M_b are 0. Thus, processes related to these matrices can be carried out independently in time and logic.

For example, let A and B be two structure matrices of dynamic process models. The feed-forward matrix M_v and the feedback matrix M_b are formed by the extension of matrices M_i and M_k, which are composed of processes $i-1$, i and $i+1$, $i+2$, respectively.

$$A = \left[\begin{array}{cc|cc|c} * & 1 & 0 & 0 & 0 \\ \multicolumn{2}{c|}{\boldsymbol{M_i}} & \multicolumn{2}{c|}{\boldsymbol{M_v}} & \\ 1 & * & 1 & 1 & 0 \\ \hline 1 & 1 & * & 1 & 0 \\ \multicolumn{2}{c|}{\boldsymbol{M_b}} & \multicolumn{2}{c|}{\boldsymbol{M_k}} & \\ 1 & 1 & 1 & * & 1 \\ \hline 1 & 1 & 1 & 1 & * \end{array}\right], \quad B = \left[\begin{array}{cc|cc|c} * & 1 & 0 & 0 & 0 \\ \multicolumn{2}{c|}{\boldsymbol{M_i}} & \multicolumn{2}{c|}{\boldsymbol{M_v}} & \\ 1 & * & 1 & 1 & 0 \\ \hline 1 & 0 & * & 1 & 0 \\ \multicolumn{2}{c|}{\boldsymbol{M_b}} & \multicolumn{2}{c|}{\boldsymbol{M_k}} & \\ 1 & 0 & 0 & * & 1 \\ \hline 1 & 0 & 0 & 0 & * \end{array}\right]$$

$$M_i^A = \begin{bmatrix} * & 1 \\ 1 & * \end{bmatrix}, \quad M_i^B = \begin{bmatrix} * & 1 \\ 0 & * \end{bmatrix}, \quad M_k^A = \begin{bmatrix} * & 1 \\ 1 & * \end{bmatrix},$$
$$M_k^B = \begin{bmatrix} * & 1 \\ 0 & * \end{bmatrix}.$$

From the structures of matrices A and B, it is observed that the feedback loop number of the processes expressed by matrix B is less than that of the processes expressed by A. Let the elements of M_b be the number of feedback loops. The number of the feedback loops in M_b^A becomes

$$b^A = \sum_{k=i+1}^{i=i+2} \sum_{j=i-1}^{i} M_b^A(k,j) = 4. \tag{12}$$

The number of the feedback loops in M_b^B is

$$b^B = \sum_{k=i+1}^{i=i+2} \sum_{j=i-1}^{i} M_b^B(k,j) = 0. \tag{13}$$

$M_v^A = [\begin{smallmatrix} 1 & 0 \\ 1 & 0 \end{smallmatrix}]$ indicates that the output of processes $i-1$ and i is the input of process $i+1$ in matrix A, that is to say, the ending of processes $i-1$ and i is the necessary condition for carrying out the process $i+1$. The proceeding of the process $i+2$ has no direct relation with processes $i-1$ and i, and can be carried out with $i-1$ and i synchronously. If $M_v^A = [\begin{smallmatrix} 0 & 0 \\ 0 & 0 \end{smallmatrix}]$ (which can be obtained by adjusting the relationships among processes), it denotes that the correlative four processes have no forward relation such that they can be carried out synchronously in time.

$M_b^A = [\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}]$ means that there is a feedback relation between every two processes in $i+1$, $i+2$, and $i-1$, i. $M_b^B = [\begin{smallmatrix} 0 & 0 \\ 0 & 0 \end{smallmatrix}]$ indicates that there is no direct feedback relation among the four processes, which offers the necessary condition for the processes concurrent in logic.

In order to ensure the rationality of the product development in time and logic, the values of a_j^i and b_j^i should be adjusted by re-organizing constraints, feedback loops, and processes. After a design structure matrix is divided, the interdependence and feedback relationships among sub-processes can be optimized. Therefore, the resource moving, the information transforming, and the information accumulating can be completed successfully.

4. Process Evaluation

As the product development is a socio-technical system, the process evaluation should be carried out from many aspects. The basic aspects include technology, economy, and society. Owing to the subjective, fuzzy, and uncertain factors in a development process, the fuzzy evaluation can be applied to evaluate the process [19].

Evaluations can be based on factors, such as the quality, the cost, and the development time. Let $\{u_1, u_2, \ldots, u_n\}$ be a vector set of evaluation factors, associated with ranges of values representing the degree of membership of a corresponding factor's satisfaction degree. Let $\mu_i(v_i)$ be the degree of membership, and v_i be the corresponding value of u_i. A vector of weights for evaluation factors specified by experts is expressed as $\{w_1, w_2, \ldots, w_n\}$.

The evaluation method can be carried out step-by-step as follows:

(i) If $v_i \notin \mu_i(v_i)$, then the feedback information is
  - the corresponding indices of the structure matrix.
  - the difference between the index and its lowest value.

(ii) If $v_i \in \mu_i(v_i)$, giving the threshold S_m of the satisfaction degree, let $\mu_i(v_i)$ be the degree of membership of an evaluation factor, such that

$$S = \sum_{i=1}^{n} w_i \mu_i(v_i). \tag{14}$$

- If $S \geq S_m$, the process is satisfied.

- If $S < S_m$, the process has to be optimized and the information feedback has to be carried out. The optimization problem is formulated as follows:

$$\begin{aligned} \text{Min} \quad & \|A - A'\|_{\mathrm{P}}^{1} \\ \text{s.t.} \quad & \sum_{i=1}^{n} w_i \mu_i(v_i) \geq S_m \\ & \mu_v^{\min} \leq \mu_i(v_i) \leq 1 \end{aligned} \qquad (15)$$

where $A = w\mu(v)$. $A' = w\mu^{\max} \cdot \mu^{\max}$ is the biggest vector of degree of membership. $\mu_v^{\min}$ is the smallest vector of degree of membership. $\|A - A'\|_{\mathrm{P}}^{1}$ is the norm and $P > 1$.

The feedback information Δ is

(a) If $v_{i\,\text{lower limit}} > v_i$, then $\Delta_i = |v_{i\,\text{lower limit}} - v_i|$
(b) If $v_{i\,\text{lower limit}} < v_i < v_{i\text{upper limit}}$, then $\Delta_i = |v_{i\,\text{upper limit}} - v_i|$
(c) If $v_{i\,\text{upper limit}} < v_i$, then $\Delta_i = |v_i - v_{i\,\text{upper limit}}|$
(d) The corresponding indices of structure matrix.

After the information is fed back to the initial design stage, a multifunctional design team can carry out three re-organization activities to optimize the development process, based on the practical development environment:

(i) *Re-organize constraints*. Because of the identity of the design resource, constraint relations exist among design parameters. Based on the feedback information, constraint relations can be adjusted or changed to optimize the process structure.
(ii) *Re-organize a process*. The process structure can be modified by adjusting the time or the logic relation of a process. For example, a process that occurs only when both the sufficient and the necessary conditions are met can be modified such that it occurs under either the sufficient or the necessary condition.
(iii) *Re-organize the structure of designers' preference*. To a great extent, the designers' preference determines the relations among design processes. If a satisfactory design cannot be obtained under the initial preference structure and it is difficult to change the rigid constraints, designers' preference should be loosened to satisfy the requirements of design optimization by adjusting boundary values of ranges of the preference function.

5. Gear Transmission System Development

The development of a gear transmission system for the ZL50G loader is an example of the dynamic characteristics of the product development. As the working environment is an altiplano environment, there is a larger degree of complexity and uncertainty associated with the development compared to the old ZL50. In order to develop a product of high quality in a short time, dynamic characteristics of the development process should be analyzed to establish an effectual process model. At the same time, as the development of a new loader inherits features of ZL50, the interdependence and the feedback relationships are complicated.

There are seven sub-processes, listed as follows:

(i) Preparing product specifications.
(ii) Preliminary design.
(iii) Evaluating development cost.
(iv) Mechanical analysis.
(v) Designing process features.
(vi) Analyzing the product data of identical series products.
(vii) Finalizing design details.

Structure matrices under different states are shown in Figures 2–5.

From the above analysis, it can be seen that:

(i) Through the dynamic analysis of the development process, uncertain factors and some rules to be followed are found. Therefore, activities in the development process can be organized and managed effectively.
(ii) The development process can be optimized by the optimization of the interdependence and the feedback relationships among processes. As shown in Figure 5, the number of feedback loops and the complexity of the development process are reduced. The number of feedback loops is

$$b = \sum_{i=1}^{7} \sum_{j=1}^{7} M_b(i,j) = 6.$$

$$\begin{array}{c|ccccccc|} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & * & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & * & 0 & 0 & 1 & 1 & 0 \\ 3 & 0 & 0 & * & 1 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & * & 1 & 1 & 0 \\ 5 & 0 & 0 & 0 & 0 & * & 1 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & * & 0 \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & * \\ \hline \end{array}$$

Figure 2. Structure matrix of no-feedback development process.

$$\begin{array}{c|ccccccc|} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & * & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & * & 0 & 0 & 1 & 1 & 0 \\ 3 & 1 & 1 & * & 1 & 0 & 0 & 0 \\ 4 & 1 & 1 & 1 & * & 1 & 1 & 0 \\ 5 & 1 & 1 & 1 & 1 & * & 1 & 0 \\ 6 & 1 & 1 & 1 & 1 & 1 & * & 0 \\ 7 & 1 & 1 & 1 & 1 & 1 & 1 & * \end{array}$$

Figure 3. Structure matrix of full-feedback development process.

$$\begin{array}{c|ccccccc|} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 1 & * & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & * & 0 & 0 & 1 & 1 & 0 \\ 3 & 0 & 0 & * & 1 & 0 & 0 & 0 \\ 4 & 0 & 1 & 0 & * & 1 & 1 & 0 \\ 5 & 0 & 1 & 0 & 1 & * & 1 & 0 \\ 6 & 0 & 1 & 0 & 1 & 1 & * & 0 \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & * \end{array}$$

Figure 4. Structure matrix of modified development process.

(iii) From the analysis of the structure matrix, one can see that the feedback loops exist mainly in processes 2, 4, 5, and 6. It means that the feedback loops exist mainly in the detailed design stage. By substituting the big feedback cycle with microcycles, the efficiency of the development can be improved and the redundant feedback can be reduced.

(iv) By improving and optimizing the relations among processes, the degree of concurrency can be improved. The development time can be reduced. Most processes can be carried out in parallel or partly in parallel, such as the processes included in sub-matrix [1], [2 4 5 6], [3], and [7] in Figure 5.

Cost, weight, and quality of transmission are selected as the evaluation factors. They are denoted by an evaluation vector $U=(u_1, u_2, u_3)=$(cost, weight, transmission quality). The weights of evaluation factors are set by experts and expressed in a vector $W=(w_1, w_2, w_3)=(0.3, 0.3, 0.4)$. For some processes, the value of a range and the degree of membership of the corresponding range's satisfaction degree may be specified in advance. The threshold on the satisfaction degree S_m may also be given in advance as $S_m=0.90$. The process evaluation can be carried out step by step according to the evaluation method proposed in Section 4. Based on the evaluation results, one can change or adjust the helical angle, material, tooth number, modulus, tooth length, and so on. Moreover, this information is fed back to the frontward processes to modify the design processes and design constraints. The final evaluation result is $S=0.947$.

$$\begin{array}{c|ccccccc|} & 1 & 2 & 4 & 5 & 6 & 3 & 7 \\ 1 & * & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & * & 1 & 1 & 1 & 0 & 0 \\ 3 & 0 & 1 & * & 1 & 1 & 0 & 0 \\ 4 & 0 & 1 & 1 & * & 1 & 0 & 0 \\ 5 & 0 & 1 & 1 & 1 & * & 0 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & * & 0 \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & * \end{array}$$

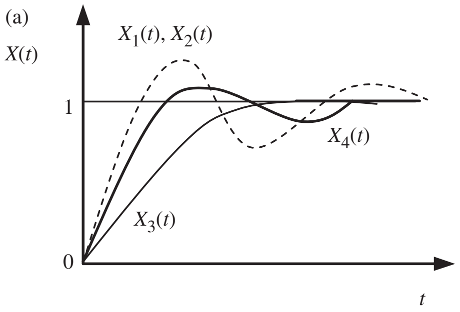
Figure 5. Structure matrix of final development process.

The implementation of the dynamic analysis brings great benefits to the product development in this example. The development time is shortened by more than 30%, and the development cost is cut down by more than 40%.

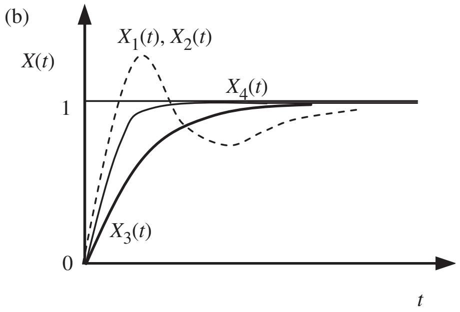
6. Discussion

(i) Owing to the flexibility of design constraints, re-organizing constraints is easier than re-organizing a process. In general, the relations between processes can be optimized through increasing or decreasing constraint variable values or feedback loops. If the development requirements cannot be satisfied under such a circumstance, re-organization of the process must be conducted to meet the development requirements.

(ii) There exists over-amplitude in the product development process, i.e., an iterative process. The number of feedback loops, the time of feedback, and the location of feedback have a significant influence on both the information interaction and the product development. In order to make a development process continue successfully, the correct design information must be fed back to the right location at the right time to reduce



Manufacturing process successes after many repetitions



Manufacturing process successes with no repetition

Figure 6. The variation of process state variables. (a) Manufacturing process successes after many repetitions (b) Manufacturing process successes with no repetition

the redundant feedback processes. Take the state variables in the development process of gear transmission system of ZL50G loader as example. Let X_1 be the state vector that designers master customers' requirements or customers understand and express their requirements. Let X_2 be the state vector that designers master the design technology related to the product in the design process. Let X_3 be the state vector of the manufacturing state in the development. Let X_4 be the state vector that the product satisfies customer's requirements after being checked and tried out. Figure 6(a) means that a manufacturing process succeeds after many repetitions, and Figure 6(b) indicates that a manufacturing process succeeds with no repetition.

(iii) A reasonable analysis of dynamic characteristics can bring many additive benefits, such as shortening development time, cutting development cost, improving product quality, etc. Failure to complete such a reasonable analysis leads to losses in a process.

(iv) The evaluation precision, the feedback information, and the re-organization activities significantly influence the convergence speed of a process optimization.

(v) From Figure 6, it can be observed that the customers' requirements and the instability of the design technology are the main contributors to the instability of a development system. Because of the increasing new requirements and the emergence of new technologies, these two state variables should be emphasized to reduce the over-amplitude of a product development process.

(vi) In view of the inheritance and the transmission of constraints, carrying out the state feedback of the entire system, especially speeding up the information interaction of the upstream processes and re-organizing the constraints can improve the entire development process.

(vii) Enhancing the evaluation precision and using the intelligent tools (such as a neural network) for the real-time evaluation of a development process can make the best of the design information and can reduce the redundancy in an interactive process.

7. Conclusions

The modeling of a product development process is the heart of the product development activities that determine the engineering productivity and the industrial competitiveness. The rapidly changing information technology and the competitive nature of the global market motivate and facilitate the fast advances in the research and application efforts on modeling a development process. Modeling a product development process as a dynamic system with feedback is a very effective method for realizing life cycle design, optimizing the whole development process, improving the degree of concurrency, capturing designers' preferences, speeding the information flow, and reducing the modification frequency.

However, due to the complexity of the product development, several issues should be taken into account in the future research.

(i) As some processes are carried out in partial parallel, there are limitations in the design structure matrix that is expressed as 0 and 1.

(ii) In order to speed up the convergence of the process optimization, the re-organization activities of a process, constraints, and the preference's structure should be implemented in parallel based on the feedback information.

(iii) The insufficiency of the process dynamic analysis as well as the impropriety of both the process evaluation and the information feedback increase the instability of a system to a great extent.

Acknowledgments

The financial support through the National Excellent Doctoral Dissertation Special Foundation under the contract number 200232, and the support from Guangxi Liu Gong Machinery Co. Ltd, P.R. of China are gratefully acknowledged.

References

1. Krause, F.L., Kimura, F., Kjellberg, T. and Lu, S.C.Y. (1993). Product Modeling, *Annals of CIRP*, **42**(2): 695–706.
2. Xie, L.W., Cheng, Y.D. and Qi, G.N. (2000). Evaluation of Complexity and Preference of Integrated Product Development Process Management, *Journal of Chinese Mechanical Engineering*, **36**(10): 37–41.
3. Negele, H., Fricke, E., Schrepfer, L. and Hartlein, N. (1999). Modeling of Integrated Product Development Processes, http://www.gfse./paper/progess modeling INCOSE 99.pdf.
4. Steward, D.V. (1981). The Design Structure System: A Method for Managing the Design of Complex Systems, *IEEE Transactions on Engineering Management*, **28**(3): 71–74.
5. Kusiak, A. and Wang, J. (1993). Decomposition of the Design Process, *Journal of Mechanical Design*, **115**(4): 687–695.
6. Ford, D.N. and Sterman, J.D. (1997). Dynamic Modeling of Product Development Process, http://web.mit.edu/jsterman/www/SDG/ford_sterman.html.
7. Eppinger, S.D., Whitney, D.E., Smith, R.P. and Gabala, D.A. (1994). A Model-based Method for Organizing Tasks in Product Development, *Research in Engineering Design*, **6**(1): 1–13.
8. Eppinger, S., Nukala, M. and Whitney, D. (1997). Generalized Models of Design Iteration Using Signal Flow Graphs, *Research in Engineering Design*, **9**(2): 112–123.
9. AitSahlia, F., Johnson, E. and Will, P. (1995). Is Concurrent Engineering Always a Sensible Proposition? *IEEE Transactions on Engineering Management*, **42**(2): 166–170.
10. Krishnan, A., Eppinger, S. and Whitney, D. (1997). A Model-based Framework to Overlap Product Development Activities, *Management Science*, **43**(4): 437– 451.
11. Cho, S.H. and Eppinger, S.D. (2001). Product Development Process Modeling Using Advanced Simulation, In: *Proceedings of DETC'01: ASME 2001 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Pittsburgh, Pennsylvania.
12. Cooper, R.G. (1986). New Product Performance and Product Innovation Strategies, *Research Management*, **29**(3): 17–25.
13. Ren, S.J. (1999). *Analysis and Design of Modern Manufacturing System*, Beijing: Science Press.
14. Zhang, B.P. (2003). *Manufacturing Informatics*, Beijing: Press of Tsinghua University.
15. Wood, W.H. and Agogino, A.M. (1996). A Case-based Conceptual Design Information Server for Concurrent Engineering, *Computer-Aided Design*, **28**(5): 361–369.
16. Li, Z. and Yan, J. (2000). Modeling Collaboration Engineering Design Process Using Petri-net, In: *Proceedings of DETC'00: 2000 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conferences*, Baltimore, Maryland.
17. Li, S.P., Liu, N.R. and Guo, M. (2002). *Data Standard of Product and PDM*, Beijing: Press of Tsinghua University.
18. Smith, R.P. and Morrow, J.A. (1999). Product Development Processes Modeling, *Design Study*, **20**(3): 237–261.
19. Huang, H.Z. (1996). Reliability Evaluation of a Hydraulic Truck Crane Using Field Data with Fuzziness, *Microelectronics and Reliability*, **36**(10): 1531–1536.

Dr Hong-Zhong Huang

Dr Hong-Zhong Huang is a full professor and dean of the School of Mechanical, Electronic, and Industrial Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, China. He has held visiting appointments at several universities in Canada, the USA, and Asia. Dr Huang received his PhD degree in Reliability Engineering from Shanghai Jiaotong University, China. Dr Huang has published 120 journal papers and five books in the fields of reliability engineering, optimization design, fuzzy sets theory, and product development. He is a (senior) member of several professional societies, and has served on the boards of professional societies. He received the Golomski Award from the Institute of Industrial Engineers. His current research interests include system reliability analysis, warranty, maintenance planning and optimization, and computational intelligence in product design.

Dr Ying-Kui Gu

Dr Ying-Kui Gu is currently an associate professor in the School of Mechanical & Electronical Engineering at Jiangxi University of Science and Technology, Ganzhou, Jiangxi, 341000, China. He received a PhD degree in Mechanical Engineering from Dalian University of Technology, China. Dr Gu has published 20 journal papers. His research interests include product development, design optimization, production, and engineering management.