A design exploration method for resolving parameter coupling in engineering change propagation

Yuliang Li*

Department of Mechanical Engineering,
Zhejiang University,
38 Zheda Road, Hangzhou,
Zhejiang Province, 310027, China
E-mail: lyl_zw@zju.edu.cn
*Corresponding author

Wei Zhao

Zhejiang University of Finance and Economics,
18 Xueyuan Street, Hangzhou,
Zhejiang Province, 310018, China
E-mail: joycezhaowei@126.com

Y-S. Ma

Department of Mechanical Engineering,
University of Alberta,
Edmonton, AB T6G 2G8, Canada
E-mail: yongshen@ualberta.ca

Abstract: Product design tasks in the upstream and downstream stages are often interdependent in engineering design processes. When design changes propagate from the upstream to the downstream, or vice versa, design tasks in different stages affect each other. Then solving the relevant design problems has to be repeated if the designer cannot find an acceptable solution to satisfy both downstream and upstream design requirements. In this paper, those design task connections with the interdependent nature or phenomena are referred to as design change *propagation couplings* and the analysis of the coupling is presented. Two types of coupling morphology named *concurrent coupling* and *sequential coupling* are identified. A theoretical method as well as a software system to solve such propagation couplings is developed. A design case of the feeding servo system on a numerical controlled machine tool is used to demonstrate the application of the software.

Keywords: propagation coupling; engineering analysis; evolutionary design.

Reference to this paper should be made as follows: Li, Y., Zhao, W. and Ma, Y-S. (2011) 'A design exploration method for resolving parameter coupling in engineering change propagation', *Int. J. Design Engineering*, Vol. 4, No. 3, pp.220–243.

Biographical notes: Yuliang Li received his PhD in Mechanical Engineering from Huazhong University of Science and Technology in 2004. He worked as a Post-doc at Zhejiang University from 2004 to 2006. He is currently an Associate Professor at Zhejiang University. His current work is focused on product conceptual design, design evolution and vehicle performance analysis.

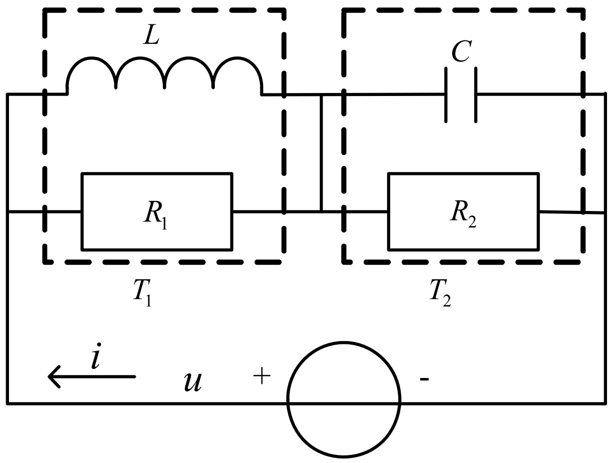
Wei Zhao is a Lecturer at Zhejiang University of Finance and Economics. She received her MSc in 2004.

Y-S. Ma is an Associate Professor at University of Alberta. He received his PhD from the University of Manchester (UMIST), UK in 1993. His research interests include collaborative engineering, product lifecycle management (PLM), knowledge-based design and manufacturing, CAD/CAM/CAPP, etc.

1 Introduction

In this paper propagation coupling refers to the mutual impacts between design tasks at different stages that are caused by design parameter and interval changes through their propagations in design iteration cycles. In most of the design cases, such couplings can be represented with design parameter associations. Propagation couplings can be resulted when, firstly, the design problem is inherently coupled; secondly, the consequent design variable changes, introduced by change propagations, generate the new values that exceed allowable tolerance margins.

Figure 1 A simple design case of concurrent coupling



For inherently coupled design problems, coupling strength can be evaluated where the coupling can be partial or full when variable values and intervals are taken into account. Figure 1 is a simple electric circuit design case to demonstrate the partial and full couplings. Suppose the two connected units represented in dashed blocks belong to two design tasks respectively. The first task of design has resistance R_1 and inductance L, and the second task contains resistance R_2 and capacitance C. The design requirement is to make the electric current and voltage have the same phase. After solving this problem, we can get the equation: $R_1 = R_2 = \sqrt{\frac{L}{C}}$. It is evident that any parameter change in one design task (for instance, L in task 1) will affect at least one parameter of the other task (e.g., C in task 2), and the two tasks are then called partially coupled. However in some cases, if the electric current and voltage have large changes while the resistances, inductance and capacitance have limited change spaces, then all design variables must be recalculated to achieve the required electric current and voltage values; then in such cases, these tasks are fully coupled. These two different coupling cases, partial or full, can be resulted from changes in the static configuration, the structure of the to-be-designed system and dynamic parameter evolutions in different design scenarios.

The second group of the propagation couplings are usually caused by strong variable constraints imposed in those interdependent design tasks. Typically, shared design variables are commonly used in designing mechanical products. Such shared variables transfer design information. When one design task is completed, another design task, which shares design variables with the former one, can get initial values for these design variables. In certain cases, these shared variables introduce design couplings when:

- A downstream design task get the transferred design information from an associated upstream design task through shared design variables, but the corresponding design problem cannot be solved or no appropriate values can be assigned to the output variables of the downstream task
- Two design tasks are supposed to generate the similar output values for the shared variables, but they in fact do not match each other, so conflicts occur; and one or both tasks should be solved again to eliminate conflicts. This kind of coupling appears dynamically in the design process.

Due to the intricate interdependencies among product components, propagation couplings can have a huge impact on the product design process. Thus, a lot of design efforts are required to reach satisfactory design results, especially when avalanches caused by design changes occur in complex products (Eckert et al., 2004). So it is important to figure out an appropriate solving strategy or a method for propagation couplings. This paper provides a solution for those non-hierarchical coupling problems, and introduces a sensitivity analysis-based method to resolve propagation parameter couplings.

The following parts of the paper are arranged as follows: Section 2 is the literature review relating to the methods for solving coupled design problems, and describes the research scope of this paper. Section 3 presents the mathematical models of two basic parameter coupling forms, i.e., concurrent coupling and sequential coupling, and proposes a sensitivity-based method for solving propagation coupling problems. The software architecture for the design exploration method is given in Section 4. Section 5 details the application of the design method and system by a case study. Conclusions and future work are presented in Section 6.

2 Literature review

Considering that coupled design tasks may spend up to 51% of the total iteration time spent in the whole design process (Boudouh et al., 2006), researchers made a lot of efforts on how to solve them in the past years and many experts presented insightful methods or strategies from the aspects of optimisation and sensitivity analysis. There are largely two approaches reported in the literature, optimisation-based and sensitivity-based.

2.1 Optimisation-based methods

Coupled design tasks usually involve multi-disciplinary design problems, so multi-disciplinary optimisation is one of effective methods for solving this kind of tasks. Kroo and Sobieski (Kroo et al., 1994; Sobieski and Kroo, 1995) proposed a collaborative optimisation (CO) method for coupled design problems with single objective and multi-objectives. Balling and Sobieski (1996) identified six fundamental CO approaches for coupled hierarchic or non-hierarchic design systems according to the criteria of whether the systems are decomposed into different levels and how the state variables of the systems are treated. Tappeta and Renaud (1997, 1999) and Tappeta et al. (2000) compared different multi-objective and CO formulations and developed an interactive multi-objective CO procedure and strategy. Concurrent Subspace Optimisation (CSSO) method (Sobieski, 1988; Renaud and Gabriele, 1993; Parashar and Bloebaum, 2006) and bilevel integrated system synthesis (BLISS) (Sobieski et al., 2000; Kim et al., 2004) were proposed to decompose hierarchically coupled systems into non-hierarchical subspace or bi-level subsystems to solve large scale complex multi-disciplinary design problems. Nair and Keane (2002) developed a co-evolutionary architecture for distributed optimisation of complex coupled systems by modelling the optimisation procedure as the process of co-adaptations between sympatric species in an ecosystem. Chamis (1999) described the modelling of inherent multidisciplinary interactions that govern the accurate response of propulsion structure systems by using disciplinary performance tailoring and simulation. In order to propagate the desirable top level design specifications to appropriate specifications for the various subsystems and components in a consistent and efficient manner, Kim et al. (2003) developed a hierarchical formulation of analytical target cascading by defining one or more pairs of target and response couplings between any two adjacent levels. Tosserams et al. (2010) present a non-hierarchical ATC formulation that allows target cascading couplings between sub-problems. Optimisation-based methods are mainly used for solving tightly coupled design problems, but for design problems that propagate through the design process, it may not be easy to turn them into standard optimisation models and then find an optimal solution for each of them.

2.2 Sensitivity analysis-based methods

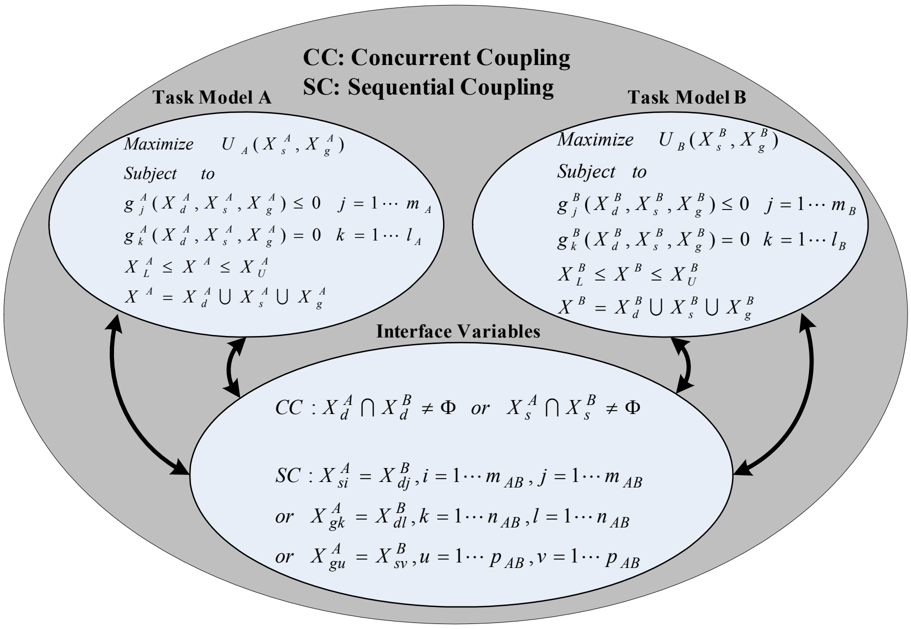
Sobieski (1990) presented two alternative algorithms for computing sensitivity derivatives with respect to independent variables for internally coupled systems. The sensitivity derivatives are useful for decision making since they can indicate how the coupling outputs of the system will change following the infinitesimal variations of the input and independent parameters, but Sobieski did not give the computing method of sensitivity derivatives with regard to the output variables, which will be more convenient for designers to decide the extent to which they can change the value of the independent input variables. English and Bloebaum (1998, 2000, 2001) and English et al. (1996) developed a sensitivity-based coupling strength analysis method to totally or temporally eliminate weak subsystem coupling factors in order to reduce computation time for solving complex coupled problems. Wujek and Renaud (1996) reported the application of automatic differentiation technology to the multidisciplinary design analysis, which illustrated that efficient technique, such as Newton's method, can be used to solve coupled system analysis problems at a fraction of the cost for forward differentiation. Chen et al. (2001) identified three classes of coupling factors in multi-disciplinary optimisation problems, and presented a strategy to handle them respectively.

2.3 Research work of this paper

So far, most references as summarised above are related to tightly coupled design problems, named as *concurrent coupling* in this paper. Few authors dealt with loosely coupled design problems – *sequential coupling*, which are caused by change propagation in the design process and decreasing intervals for design variables. It should be noted that propagation coupling can also happen in the form of concurrent coupling. While this paper mainly focuses on the sequential coupling since it is a weak point that needs to be further addressed according to the above references analysis. Chanron and Lewis (2006) gave a game theory-based approach for managing the dynamics of decentralised design processes. Three steps were presented to unify the decision-making process based on the mathematical representation of the objective functions of all involved designers. However they assumed that design problems are static, and did not take the design evolutions such as changes of design space into consideration. So the coupling issue resulted from design process evolution or change propagation has not been fully addressed. As pointed out by Eckert et al. (2004), whether a design change can be accepted depends on two factors: the initial specification of the product and the margins of design parameters that are allowed in the product design model. And they further described that margins themselves are not static but may change over the history of the design. In our opinion, this observation is also applicable to the propagation coupling problems. In addition, the third factor is also important, i.e., the customer expectation or utilisation performance objectives. In terms of propagation coupling, sensitivity and interval-based design analysis can generate a lot of predictable design scenarios of sensitive change propagation and of limited design spaces, and such information is very useful for designers to make the necessary decisions. This approach can be fully brought into play when sensitivity, interval, utility and visualisation techniques are synthesised to facilitate the analysis of interdependent design objectives for designers.

Therefore, this paper report the investigation on how to effectively manage the above three factors, i.e., the initial specification, the margin and the customer expectation, and to find appropriate solutions for propagation coupled design problems. A systematic method considering sensitivity, interval and utility for handling propagation coupling problems is proposed and a case related to the electric and mechanical design of a numerical control machine tool is studied in details to illustrate the application of the developed prototype software.

Figure 2 Mathematic model for the concurrent and sequential couplings



3 Coupled design task model and a solving method

3.1 Coupling model

For a complex design problem, decomposition is always used to transform the design problem into some simpler ones. Each resultant design task contains several or many design variables that need to be solved, and these design variables, which can be related to structure sizes, detailed geometry or performance attributes that are across the product lifecycle with the necessary reliability. More specifically, product design variables can be categorised into four groups, i.e., specification variables x_s, decision variables x_d, goal variables x_g and intermediate variables x_i (Kusiak and Wang, 1995). If these sub-domain variables are not independent, usually strong or weak dependencies exist among them through the functional or non-functional relationships. Naturally, the design tasks determining the above variables are also interdependent. In this paper, only the functional relationships are taken into consideration, and it is assumed that different sub-tasks do not seek the same goal variables. If two design tasks are mutually dependent or more than three tasks are sequentially dependent, design coupling occurs. When design changes, which need above design variables to change their values to satisfy customer requirements, propagate among these tasks, two coupling forms can be identified, i.e., *concurrent coupling* and *sequential coupling* (Figure 2). If task A and task B have a concurrent coupling relationship, then an intersection set of decision variables or specification variables between tasks A and B exists, so when the values of these design variables change, goal variables in the design tasks will be affected concurrently. While if they have a sequential coupling relationship, the intersection set of decision or specification variables can be empty, but the values of specification or goal variables in task A are associated and shared with those in task B, then designers may not find satisfactory values for later affected design variables based on the design results in the former design tasks. Design work has to be repeated in those tasks to resolve the conflicts. However, it should be pointed out that the coupling relationship among design tasks is a sufficient but not a necessary condition for building the mathematical coupling model as shown in Figure 2. That is to say, if the intersection set between two tasks' decision or specification variables are not empty, tasks A and task B may not definitely have a concurrent coupling relationship. Similarly if the specification or goal variables in task A are also used in task B, tasks A and B do not definitely have a sequential coupling relationship. This is because that they are also decided by the dependence strength among design variables, variable intervals and customer expectations.

3.2 Sensitivity-based solving method

In a design task, the relationships among specification, goal, decision and intermediate variables, as described by the inequality and equality constraints in Figure 2, can be rewritten as the following implicit or explicit equations:

$$X_{\mathrm{s}} = G_{\mathrm{s}}\left(X_{\mathrm{d}}\right) \text{ or } G_{\mathrm{s}}\left(X_{\mathrm{d}}, X_{\mathrm{s}}\right) = 0 \tag{1}$$

$$X_{\mathrm{g}} = G_{\mathrm{g}}\left(X_{\mathrm{d}}\right) \text{ or } G_{\mathrm{g}}\left(X_{\mathrm{d}}, X_{\mathrm{g}}\right) = 0 \tag{2}$$

$$X_{\mathrm{g}} = G_{\mathrm{gs}}\left(X_{\mathrm{d}}, X_{\mathrm{s}}\right) \text{ or } G_{\mathrm{gs}}\left(X_{\mathrm{d}}, X_{\mathrm{s}}, X_{\mathrm{g}}\right) = 0 \tag{3}$$

in which X_{d}, X_{s}, X_{g} are decision variable vector, specification variable vector and goal variable vector respectively. G_{s}, G_{g}, G_{gs} are function vectors among those sets of design variables. For the above models, we further emphasise the following two points:

- since computationally expensive models are generally not appropriate for direct local sensitivity analysis, development of a low-fidelity model by experiment, simulation and/or response surface method is necessary if the above functions cannot be obtained
- it can be seen that the equations are easy to be transformed into an adequate optimisation problem.

But for the modular-based product development, a module may be used in different products, which means it must meet different design requirements. An optimal design result may not be robust enough to satisfy all the design requirements and design changes. So we adopt a utility-based method to find the most suitable design solutions, and the utility model can be a straight line, broken line or exponent curve model.

In the above function vectors, one specification variable or goal variable can be affected by one or several decision variables and one decision variable may influence several specification or goal variables. Certainly it is possible that one specification variable may affect a few goal variables, and one goal variable may change with the variation of several specification variables. Taking the relationship between a decision variable and specification variables as an example to analyse, the dependence can be divided into '*and*' and '*or*' types. If the dependence relationship between a decision variable and the specification variable is '*and*', then all the specification variables must change their values when the value of the decision variable is updated. While if it is '*or*' dependence relation, when one decision variable changes, one or several specification variables can change, but usually not all of them should change. In the '*or*' case, designers should be careful to choose which decision variable to change and how much its value should be changed. Similarly, when the change is propagated to the downstream design tasks, tight constraints may also be imposed on design variables in the tasks. Therefore propagation coupling (sequential coupling) can occur when these variables cannot be assigned with appropriate values to satisfy the constraints simultaneously.

Figure 3 Decision and dependence model for coupled design tasks caused by parameter propagation

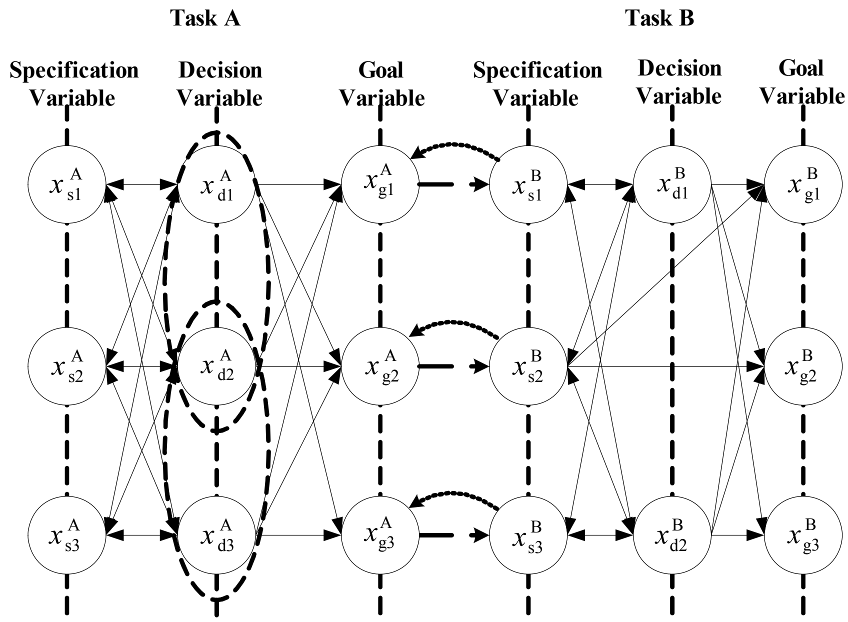


Figure 3 shows a propagation coupling case, in which two design tasks are involved. The dotted arcs in the figure represent design feedback or counteractions that task B transfers to task A. To avoid this kind of coupling, it is necessary not only to analyse the internal relationship among design variables in task A, but to find out the change impacts of task A's variables on task B's variables (especially goal variables in task B). Three types of sensitivity analysis were given by Li and Pan (2006) in order to realise collaborative design. In this paper these three types of sensitivity analysis are applied to solve propagation coupling design problems, and they are: analysis of sensitivity between decision variable and specification variable [equation (4)], decision variable and goal variable [equation (5)] within one task (the first type of sensitivity analysis), analysis of sensitivity between decision variables [equation (6)] within one task (the second type of sensitivity analysis), and analysis of sensitivity between design tasks [equations (4) and (8)] are used in cases when specification and goal variables in different tasks have and do not have direct functional relationships respectively, (the third type of sensitivity analysis).

$$\begin{bmatrix} \frac{\partial X_{\mathrm{s}1}}{\partial X_{\mathrm{d}1}} & \cdots & \frac{\partial X_{\mathrm{s}1}}{\partial X_{\mathrm{d}m}} \\ \vdots & & \vdots \\ \frac{\partial X_{\mathrm{s}k}}{\partial X_{\mathrm{d}1}} & \cdots & \frac{\partial X_{\mathrm{s}k}}{\partial X_{\mathrm{d}m}} \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} \frac{\partial X_{\mathrm{g}1}}{\partial X_{\mathrm{d}1}} & \cdots & \frac{\partial X_{\mathrm{g}1}}{\partial X_{\mathrm{d}m}} \\ \vdots & & \vdots \\ \frac{\partial X_{\mathrm{g}l}}{\partial X_{\mathrm{d}1}} & \cdots & \frac{\partial X_{\mathrm{g}l}}{\partial X_{\mathrm{d}m}} \end{bmatrix} \tag{5}$$

$$\begin{bmatrix} \frac{\partial X_{\mathrm{d},u-m_2+1}}{\partial X_{\mathrm{d},1}} & \frac{\partial X_{\mathrm{d},u-m_2+2}}{\partial X_{\mathrm{d},1}} & \cdots & \frac{\partial X_{\mathrm{d},u}}{\partial X_{\mathrm{d},1}} \\ \frac{\partial X_{\mathrm{d},u-m_2+1}}{\partial X_{\mathrm{d},2}} & \frac{\partial X_{\mathrm{d},u-m_2+2}}{\partial X_{\mathrm{d},2}} & \cdots & \frac{\partial X_{\mathrm{d},u}}{\partial X_{\mathrm{d},2}} \\ \vdots & \vdots & \frac{\partial X_{\mathrm{d},u-m_2+j}}{\partial X_{\mathrm{d},i}} & \vdots \\ \frac{\partial X_{\mathrm{d},u-m_2+1}}{\partial X_{\mathrm{d},m_2}} & \frac{\partial X_{\mathrm{d},u-m_2+2}}{\partial X_{\mathrm{d},m_2}} & \cdots & \frac{\partial X_{\mathrm{d},u}}{\partial X_{\mathrm{d},m_2}} \end{bmatrix} i = 1, 2, \cdots, m_2,\ \ j = 1, 2, \cdots, m_1 \tag{6}$$

$$\begin{bmatrix} \frac{\partial X_{\mathrm{s}i}}{\partial X_{\mathrm{s}1}^{\mathrm{A}}} & \cdots & \frac{\partial X_{\mathrm{g}j}}{\partial X_{\mathrm{s}1}^{\mathrm{A}}} & \cdots \\ \frac{\partial X_{\mathrm{s}i}}{\partial X_{\mathrm{s}2}^{\mathrm{A}}} & \cdots & \frac{\partial X_{\mathrm{g}j}}{\partial X_{\mathrm{s}2}^{\mathrm{A}}} & \cdots \\ \vdots & & \vdots & \cdots \\ \frac{\partial X_{\mathrm{s}i}}{\partial X_{\mathrm{s}k}^{\mathrm{A}}} & \cdots & \frac{\partial X_{\mathrm{g}j}}{\partial X_{\mathrm{s}k}^{\mathrm{A}}} & \cdots \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} \frac{\mathrm{d}X_{\mathrm{s}i}}{\mathrm{d}X_{\mathrm{s}1}^{\mathrm{A}}} & \cdots & \frac{\mathrm{d}X_{\mathrm{g}j}}{\mathrm{d}X_{\mathrm{s}1}^{\mathrm{A}}} & \cdots \\ \frac{\mathrm{d}X_{\mathrm{s}i}}{\mathrm{d}X_{\mathrm{s}2}^{\mathrm{A}}} & \cdots & \frac{\mathrm{d}X_{\mathrm{g}j}}{\mathrm{d}X_{\mathrm{s}2}^{\mathrm{A}}} & \cdots \\ \vdots & & \vdots & \cdots \\ \frac{\mathrm{d}X_{\mathrm{s}i}}{\mathrm{d}X_{\mathrm{s}w}^{\mathrm{A}}} & \cdots & \frac{\mathrm{d}X_{\mathrm{g}j}}{\mathrm{d}X_{\mathrm{s}w}^{\mathrm{A}}} & \cdots \end{bmatrix} i = 1, \cdots, k_{\mathrm{B}} - w,\ \ j = 1, \cdots, l_{\mathrm{B}} \tag{8}$$

among which:

k is the number of specification variables in the task, and k_B is the number of specification variables in task B

m is the number of decision variables in the task

l is the number of goal variables in the task, and I_B is the number of goal variables in task B

u is the number of specification and decision variables in the task, i.e., $u = m + k$

m_1 is the number of goal or specification variables whose values do not need to be changed

m_2 is the number of goal or specification variables whose values need to be changed, and $m_1 + m_2 = l + k$. If there are n specification variables that are the independent variables of goal variables in equation (3), then $m_2 = l + k - n$

w is the number of design variables in the task A that is output to task B, and they are part of the k_B specification variables in task B

X_{si}^{A} is the specification variable in task B, but its values is given by task A

X_{si} is the specification variable in task B except those w specification variables

X_{gj} is the goal variable in task B.

Each element in matrices (4) to (7) can be computed through the direct derivation of the dependent variables with respect to the corresponding design variables. Bu for the element $\frac{\mathrm{d}X_{si}}{\mathrm{d}X_{sq}^{\mathrm{A}}}$ or $\frac{\mathrm{d}X_{gj}}{\mathrm{d}X_{sq}^{\mathrm{A}}}(i = 1\cdots S,\ j = 1\cdots g,\ q = 1\cdots k)$ in matrix (8) represents the sensitivity of specification variable X_{si} or goal variable X_{gj} with respect to the specification variable X_{sq}^{A}. Here the superscript A represents the values of the k specification variables in task B come from task A. For elements in matrix (8), there is no direct functional relationship between the denominator and the numerator, so the direct derivation is not feasible. But the generalised matrix inverse theory can be used to derive the sensitivity relationship. According to the equations (1) and (2), they can be expressed as the following equations:

$$X_{si}^{\mathrm{A}} = g_{si}\left(\vec{X}_d^{\mathrm{B}}\right) \tag{9}$$

$$X_{gj} = g_{gj}\left(\vec{X}_d^{\mathrm{B}}\right) \tag{10}$$

Total differentiation can be performed on both sides of equations (9) and (10), and then we have

$$\mathrm{d}X_{si}^{\mathrm{A}} = \nabla g_{si} \cdot \left(\mathrm{d}\vec{X}_d^{\mathrm{B}}\right)^{\mathrm{T}} \tag{11}$$

$$\mathrm{d}X_{gj} = \nabla g_{gj} \cdot \left(\mathrm{d}\vec{X}_d^{\mathrm{B}}\right)^{\mathrm{T}} \tag{12}$$

among which, $\nabla g_{si} = \left(\frac{\partial g_{si}}{\partial x_{\mathrm{d}1}^{B}}, \frac{\partial g_{si}}{\partial x_{\mathrm{d}2}^{B}}, \cdots \frac{\partial g_{si}}{\partial x_{\mathrm{d}m_{\mathrm{B}}}^{B}}\right)$, $\mathrm{d}\vec{X}_d^{\mathrm{B}} = \left(\mathrm{d}X_{\mathrm{d}1}, \mathrm{d}X_{\mathrm{d}2}, \cdots, \mathrm{d}X_{\mathrm{d}m_{\mathrm{B}}}\right)$.

According to the generalised matrix inverse theory, equation (11) can be transformed into

$$\mathrm{d}\vec{X}_{\mathrm{d}}^{\mathrm{BT}} = \left[\nabla g_{si} \cdot \nabla g_{si}{}^{\mathrm{T}}\right]^{-1} \cdot \nabla g_{si}{}^{\mathrm{T}} \mathrm{d}X_{si}^{\mathrm{A}} \tag{13}$$

Then substitute the equation (13) for the corresponding part in equation (11), and we obtain

$$\frac{\mathrm{d}X_{si}}{\mathrm{d}X_{sj}^{\mathrm{A}}} = \nabla g_{si} \cdot \left(\nabla g_{sj}^{\mathrm{A}} \cdot \nabla g_{sj}^{A\mathrm{T}}\right)^{-1} \cdot \nabla g_{sj}^{\mathrm{AT}} \tag{14}$$

among which, $\nabla g_{sj}^{\mathrm{A}} = \left(\frac{\partial g_{sj}^{\mathrm{A}}}{\partial X_{\mathrm{d}1}^{\mathrm{B}}}, \frac{\partial g_{sj}^{\mathrm{A}}}{\partial X_{\mathrm{d}2}^{\mathrm{B}}}, \cdots, \frac{\partial g_{sj}^{\mathrm{A}}}{\partial X_{\mathrm{d}m_{\mathrm{B}}}^{\mathrm{B}}}\right).$

The meaning of the above equation is the change that will happen on the specification variable X_{si} in task B when the specification variable X_{sj}^{A} has a small change at the current design point. Similarly we can substitute the equation (13) for the corresponding part in equation (12), and we have

$$\frac{\mathrm{d}X_{gk}}{\mathrm{d}X_{sj}^{\mathrm{A}}} = \nabla g_{gk} \cdot \left(\nabla g_{sj}^{\mathrm{A}} \cdot \nabla g_{sj}^{\mathrm{A}T}\right)^{-1} \cdot \nabla g_{sj}^{\mathrm{AT}} \tag{15}$$

among which, $\nabla g_{gk} = \left(\frac{\partial g_{gk}}{\partial X_{\mathrm{d}1}^{\mathrm{B}}}, \frac{\partial g_{gk}}{\partial X_{\mathrm{d}2}^{\mathrm{B}}}, \cdots, \frac{\partial g_{gk}}{\partial X_{\mathrm{d}m_{\mathrm{B}}}^{\mathrm{B}}}\right).$

The meaning of the above equation is the change that will happen on the goal variable X_{gk} in task B when the specification variable X_{sj}^{A} has a small change at the current design point.

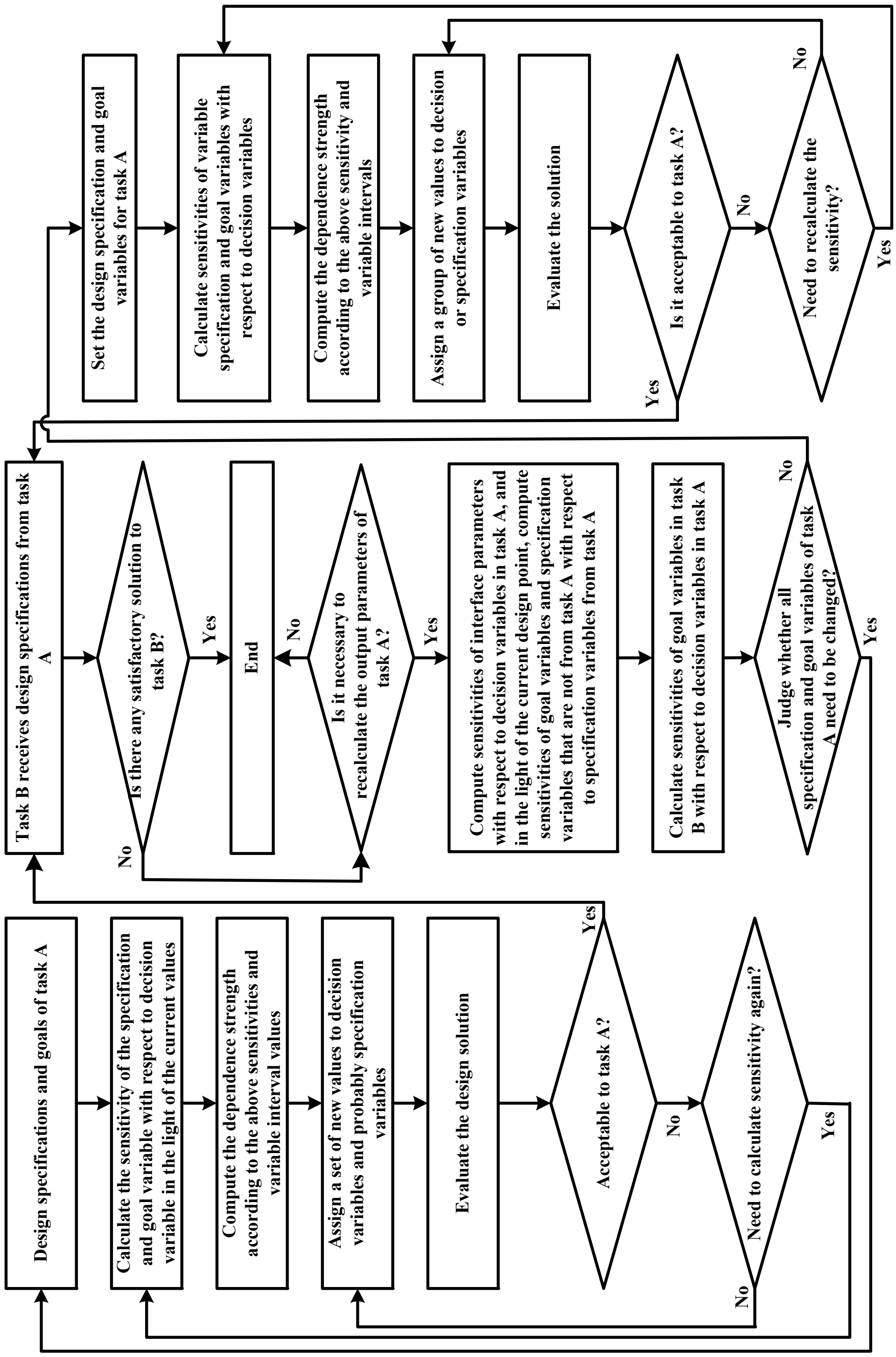
The detailed decision making process based on the above sensitivity analysis is shown in Figure 4. Generally, three stages are identified to accomplish the solving of coupled tasks by the least design iterations:

1 When new design specifications are assigned to task A, designers should calculate the first type of sensitivity matrix according to the current design information, and find a satisfactory solution.

2 After the solution is found, the agent transfers dependent design information (design parameter) to downstream task B. If no right solution can be found for task B according to the current design result of task A, then it is necessary for task A to compute the third type of sensitivity matrix, and then adjust variable values to loosen the constraint imposed on the task B, which may lead to further design couplings.

3 After the third sensitivity analysis is finished, designers determine which goal and specification variable values should be changed, if only some of variable values can be tuned, and not-necessarily-changeable goal and specification variables impose an effective constraint on task A at the current status, then it needs to carry out the second sensitivity analysis. In the above three design phases, the variable value is assigned according to current sensitivity and variable interval, so propagational coupling can be avoided, and design iteration is reduced correspondingly.

Figure 4 Decision making process for solving coupled design tasks



4 Design and implementation of system architecture

In order to reduce the workload of designers to calculate three types of sensitivities, a software prototype system has developed. It eliminates the tedious sensitivity computations and graphically displays the intervals, sensitivities and utilities of goal or specification variables (see Figure 5). The system is built on top of the foundation of a process template-based integration framework developed by the authors (Li and Zhao, 2011). Four modules are developed to fulfil different functions. The model-driven engine is the core to implement the sensitivity computation and assign values to the goal or specification variables according to the specific design models. Decision variables are independent variables, while goal variables usually depend on the decision or specification variables and specification variables can be independent or dependent variables. For the examples of decision, specification and goal variables, readers can refer to Table 1. Utility (satisfaction degree)-based design goal and specification evaluation method is adopted to guide the designer interactively to make the right trade-off decisions for any design scenario. Considering that most design tasks have more than one design specifications and goals, the geometric mean of total utility product is taken as the synthetic evaluation index for the solution. The interval management module has been implemented with the user interface of the software, i.e., the lower part of each variable panel. A rectangular bar in each variable small panel represents the whole variation range for each design, while colour portions show the used and the remaining intervals. Designers decide the interval change direction based on the satisfaction degree and the sensitivities of goal or specification variables with respect to each independent variable.

Figure 5 Architecture based on interval and sensitivity analysis for propagation parameter design

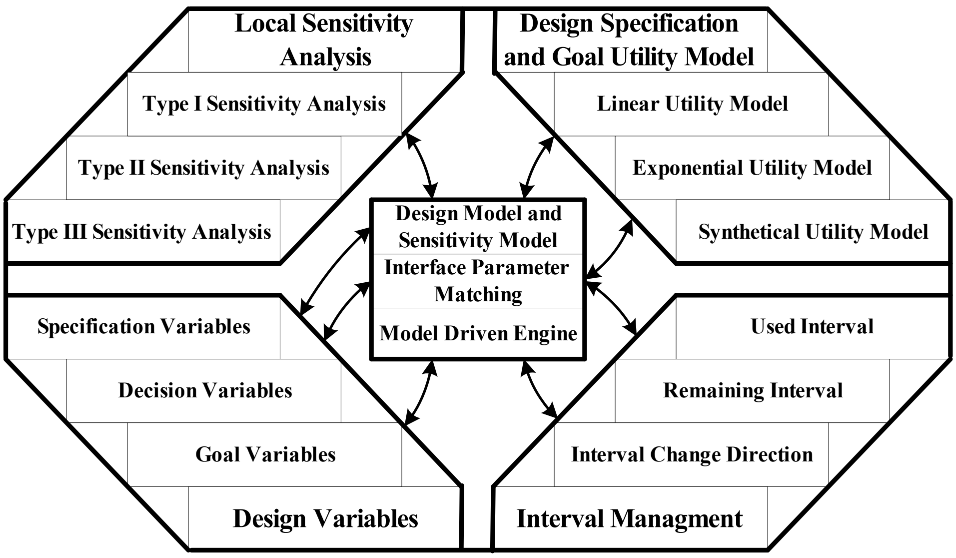


Table 1 Design variables of the example feeding servo system

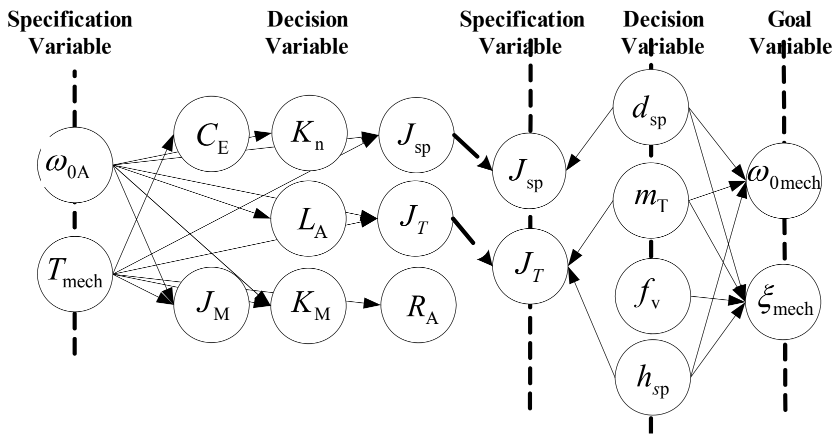
Variable name	*Design task*	*Variable type*	*Constant value or range*
Resonance frequency ω_{0A} (rad·s^{-1})	A	Specification variable	-
Amplification factor of velocity loop K_n	A	Decision variable	[450, 2 100]
Mechanical characteristic time T_{mech} (s)	A	Specification variable	-
Total inductance of circuit L_A (H)	A	Decision variable	[8.5×10^{-4}, 2.7×10^{-3}]
Total resistance of circuit R_A (Ω)	A	Decision variable	[0.22, 0.7]
Total inertia reduced to the axis of motor shaft J_{gen} (kg·m^2)	A	Intermediate variable	[1.9×10^{-2}, 4.9×10^{-2}]
Viscous damping coefficient of motor f_M (s·N·m·rad^{-1})	A	Constant	0.023 5
Torque moment of motor K_M (N·m·A^{-1})	A	Decision variable	[0.242, 1.15]
Coefficient of counter electromotive force of motor C_E (s·V·rad^{-1})	A	Decision variable	[0.24, 1.15]
Rotary inertia of motor J_M (kg·m^2)	A	Decision variable	[0.019, 0.044]
Feedback coefficient of velocity C_n (s·V·rad^{-1})	A	Constant	0.028
Rotary inertia of lead screw J_{sp} (kg·m^2)	A/B	Decision variable/ specification variable	[3×10^{-7}, 1.4×10^{-3}]
Rotary inertia of working platform reduced to the axis of lead screw J_T (kg·m^2)	A/B	Decision variable/ specification variable	[5.06×10^{-5}, 3.56×10^{-3}]
Mass of lead screw m_{sp} (kg)	B	Intermediate variable	[0.066 6, 4.624]
Bottom diameter of lead screw d_{sp} (m)	B	Decision variable	[0.006, 0.05]
Total length of lead screw including journal $L_{sp,gen}$ (m)	B	Constant	0.3
Mass of working platform m_T (kg)	B	Decision variable	[500, 550]
Lead of screw h_{sp} (m)	B	Decision variable	[0.002, 0.016]
Resonance Frequency ω_{0mech} (rad·s^{-1})	B	Goal variable	-
Damping ratio generated by the friction of working platform ξ_{mech}	B	Decision variable	-
Total stiffness of the system k_{gen} (N·m^{-1})	B	Intermediate variable	[4.833×10^6, 2.78×10^8]
Axial stiffness of bearing k_L (N·m^{-1})	B	Constant	8×10^8
Tensile stiffness of lead screw k_s (N·m^{-1})	B	Intermediate variable	[1.98×10^7, 1.375×10^9]
Stiffness of screw pair k_{LM} (N·m^{-1})	B	Constant	8×10^8
Tractional stiffness of screw pair reduced to the straight movement of the working platform k_{Ts} (N·m^{-1})	B	Intermediate variable	[5.223×10^6, 1.612×10^{12}]
Stiffness of nut bracket k_{TM} (N·m^{-1})	B	Constant	1×10^9
Kinetic viscous stamping ratio of working platform f_v (s·N·m^{-1})	B	Decision variable	[1.1×10^5, 7.9×10^5]

5 Case study

5.1 Description of electro-mechanic system design for the numerical control machine tool

For a numerical control machine tool, among many design specifications and performance requirements to meet customer's demands, this paper focuses on the speed performance of the feeding servo subsystem. Three requirements for the dynamic performance of the servo system are studied in details. Firstly, on the condition that enough system stability and servo precision are guaranteed, the system gain should be increased as much as possible to obtain a quick response. However, it should be noted that with the increase of the system gain, self-induced oscillation of the closed loop control of the system occurs. It can be concluded from the model analysis and physical experiments that there are two second-order oscillation elements in the servo system (Wang and Bai, 2003), i.e., motor-driven module and transmission module, and the oscillation frequency of the system is determined by the lowest one of the two modules. Secondly, the resonance frequency of the two oscillation elements must not be the same, it would be better that the principal resonance frequency of the transmission module is at least twice of the frequency of the motor-driven module. Thirdly, the transmission module should have an appropriate damping ratio in order to absorb the oscillation caused by the alternative cutting force. The design problem can be described as Figure 6. Design variables and their ranges are listed in Table 1.

Figure 6 Decomposed design tasks for speed design of feeding servo system, (a) design of motor-driven module (b) design of transmission module



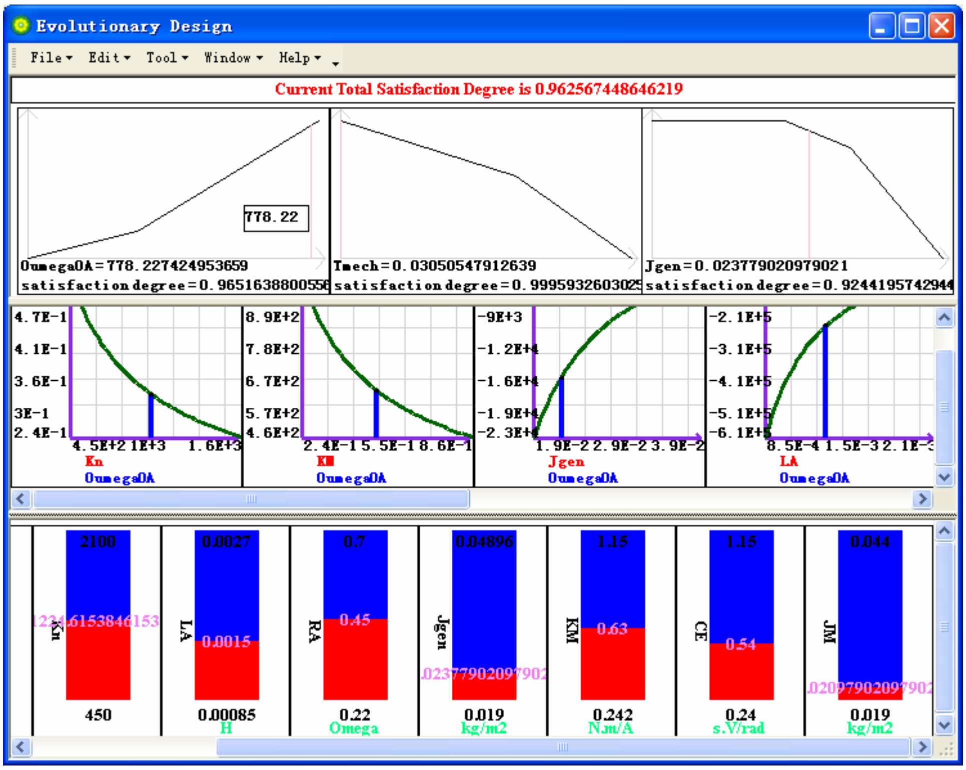
5.2 Solving process with the software

According to the design requirements and the actual design process, the design of motor-driven module and mechanical transmission module can be implemented in the following procedures to avoid the propagation coupling.

5.2.1 Activity 1: determination of the resonance frequency ω_{0A} and mechanical characteristic time T_{mech} of the motor-driven module.

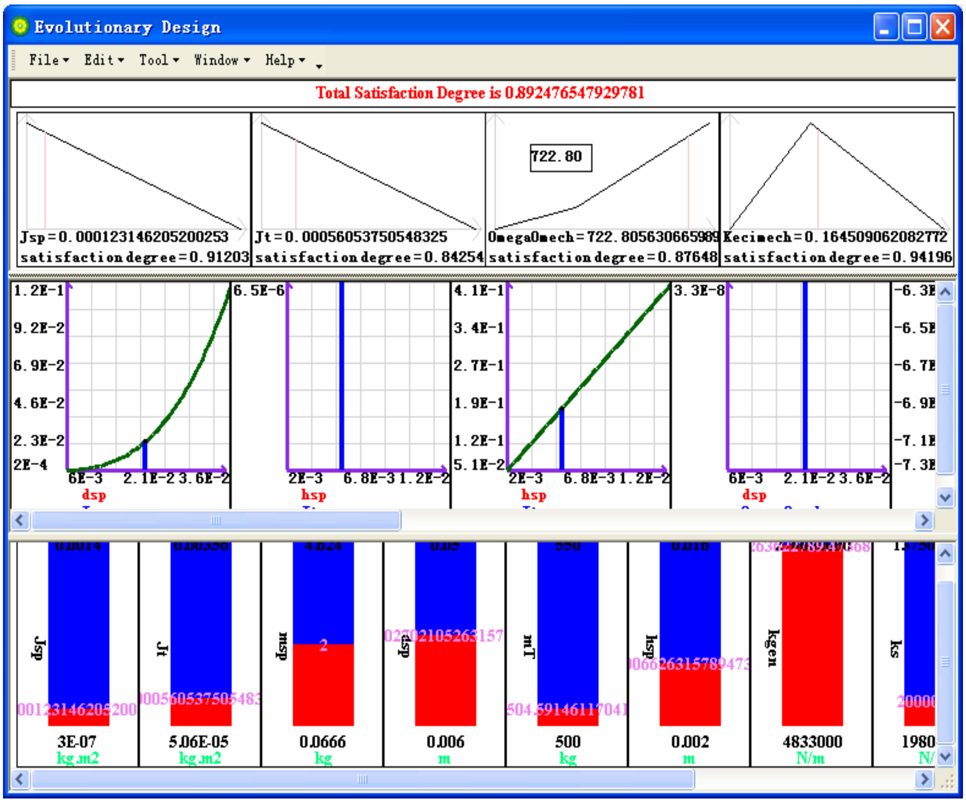
In order to maximise the acceleration capacity of the feeding servo system, the resonance frequency ω_{0A} should be as high as possible, and the mechanical characteristic time T_{mech} as short as possible. After checking the values and variation intervals of decision variables of the motor-driven module, it can be concluded that sensitivity of resonance frequency ω_{0A} with regard to the total inductance of driving circuit L_A is high. But the variation interval of the total inductance L_A is small, so only limited adjustment is feasible. While the above two specification variables also have high sensitivities with respect to the rotary inertia of motor J_M, and this variable's variation interval is a little wider, so a potentially greater value change is allowable. Although the two variables are less sensitive to the amplification factor of velocity loop K_n, the variation range of the amplification factor is rather wide, so relatively bigger range of tuning is possible. According to the above analysis, the adopted design results are shown in Figure 8 which also shows the design activity user interface of the software.

Figure 7 Dynamic design of motor-driven module (see online version for colours)



In Figure 7, the upper section is the utility panel displaying the specification or goal variable values and their utilities, the middle is the sensitivity panel and the lower is the design variable and their interval panel. This display arrangement is convenient for designers to visualise the effect of the changes. When designers click at the red or blue rectangles of each variable's sub-panel (for the variables shown in the screenshots, please refer to Table 1), the variable value, the corresponding sensitivity sub-panels and the affected specification or goal variable utility sub-panels will display different information too. The sensitivities displayed in the middle panel are computed by using the formulas derived from the equations in Wang and Bai (2003) according to the three types of sensitivity formulas. The sensitivity of the two variables can be either a non-linear or linear slope. The target variables can also be either goal or specification variables whose values are determined by several other decision variables (type 3 sensitivity). Only one sensitivity point (a vertical line) is given in the screenshot. This is because the variation range of a specification or goal variable can only be exactly determined after its values are computed throughout all the decision variables' variation ranges. Hence, the computing time would be too long to be realistically applicable. At this moment, designers are satisfied with the values of resonance frequency ω_{0A} and mechanical characteristic time T_{mech}, so these parameters can be transferred to the downstream task.

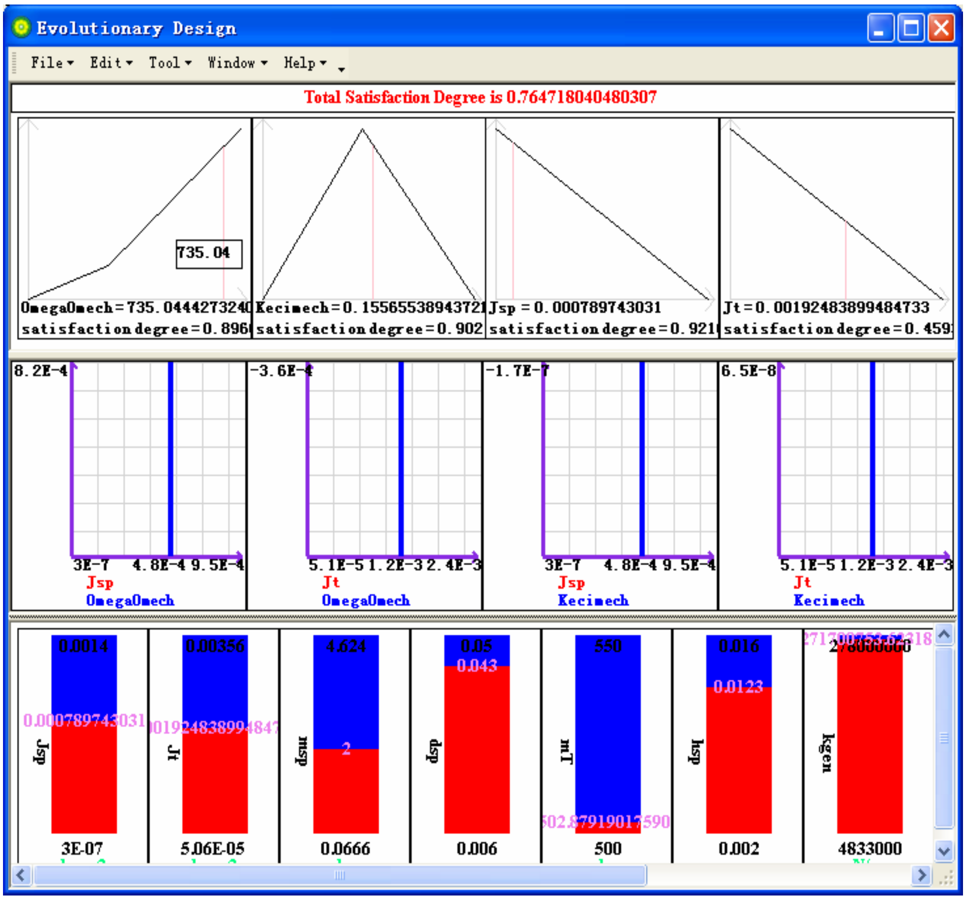
Figure 8 Dynamic design of transmission module (see online version for colours)



5.2.2 *Activity 2: feedback design of the transmission module*

The transmission module is designed based on the specification parameters. The designer executes the first path of solving the transmission module procedure by associating specification parameters with design goals of the task (Figure 8). Constrained by the upstream design task, the resulted principal resonance frequency of the transmission module $\omega_{0\text{mech}}$ at this stage is very close to the value of the motor-driven module ω_{0A}. According to the requirement for distant resonance frequencies (refer to Section 5.1) between the motor-driven module and this transmission module, this design result is not acceptable and has to be reconsidered. To solve the identified frequency conflict, the current design task, i.e., the design of the transmission module is revised first. As discussed in Section 3.1, if this conflict cannot be resolved within this current task, then a propagation coupling is identified since the designer must readjust the value of the resonance frequency of motor-driven module ω_{0A}. To check the possibility of avoiding the coupling and to further raise the damping ratio of the module ξ_{mech}, i.e., to decrease the stiffness of the system or reduce the mass of working platform, it is necessary to perform sensitivity analysis between goal variables and specification variables contained in this design task.

Figure 9 Analysis of sensitivities between goal variables and specification variables in transmission module (see online version for colours)

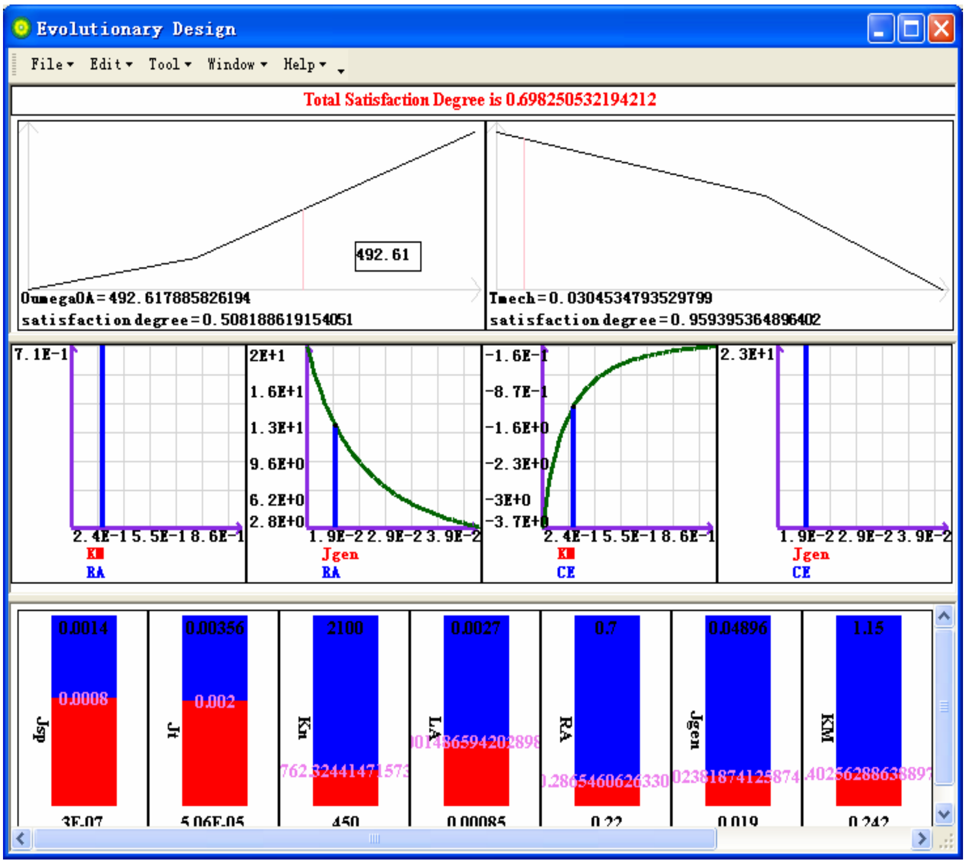


5.2.3 Activity 3: collaborative decision making of motor-driven module and transmission module

Sensitivities of goal variables with regard to specification variables in transmission module are shown in Figure 9. It can be seen that the value of the rotary inertia J_{sp} is small and has a rather limited interval. So after considering the sensitivity information from Figure 9, the frequency of the motor-driven module must be decreased because the principal resonance frequency of transmission module ω_{0mech} must be at least twice as high as the motor-driven module's resonance frequency ω_{0A}.

Meanwhile, after the first adjustment of variable values in motor-driven module, the value of mechanical characteristic time has been relatively short (for this goal variable, the smaller its value is, the more satisfactory it will be), so tuning of the decision variables' (K_n, L_A, R_A, etc., as shown in Figure 6) values of motor-driven module should not affect the mechanical characteristic time T_{mech} to keep this goal variable's value at a low level. In this case, the sensitivity analysis of mechanical characteristic time with regard to decision parameters is performed to determine these decision variables' values.

Figure 10 Analysis of sensitivities between decision parameters in motor-driven module (see online version for colours)



5.2.4 Activity 4: analysis of sensitivities between decision variables in motor-driven module

In order to keep the value of mechanical characteristic time of the motor-driven module T_{mech} fixed, the analysis of sensitivities between decision variables that affect mechanical characteristic time are performed, the results are listed in Figure 10. When the rotary inertia of lead screw J_{sp} and the rotary inertia of working platform reduced to the axis of this screw J_T decreases, values of total inductance of circuit L_A and coefficient of counter electromotive force of motor C_E should be reduced too. Since the total inductance of circuit L_A has a pretty small interval, values of amplification factor of velocity loop K_n should be greatly adjusted if the resonance frequency of electric driving module is to be decreased.

Table 2 The final parameter values of feeding servo system

Variable	*Value*	*Variable*	*Value*
Resonance frequency $\omega_{0A}/(\text{rad}\cdot\text{s}^{-1})$	490	Rotary inertia of lead screw $J_{sp}/(\text{kg}\cdot\text{m}^2)$	2.4×10^{-4}
Mechanical characteristic time T_{mech}/ms	16.1	Lead of screw h_{sp}/m	0.01
Amplification factor of velocity loop K_n	480	Resonance frequency $\omega_{0mech}/(\text{rad}\cdot\text{s}^{-1})$	765
Moment of torque of motor $K_M/(\text{N}\cdot\text{m}\cdot\text{A}^{-1})$	0.57	Damping ratio generated by the friction of working platform ξ_{mech}	0.156 7
Rotary inertia of motor $J_M/(\text{kg}\cdot\text{m}^2)$	0.019	Bottom diameter of lead screw d_{sp}/m	0.032
Total inductance of driving circuit L_A/H	1.5×10^{-3}	Mass of working platform m_T/kg	500
Total resistance of driving circuit R_A/Ω	0.26	Kinetic viscous stamping ratio of working platform f_v	1.2×10^5
Coefficient of counter electromotive force of motor $C_E/(\text{s}\cdot\text{V}\cdot\text{rad}^{-1})$	0.57	Rotary inertia of working platform reduced to the axis of lead screw $J_T/(\text{kg}\cdot\text{m}^2)$	1.3×10^{-3}

5.2.5 Design results: determination of final variable values

Based on the above analysis of sensitivity and design space variation, the propagation coupling can be predicted, and guidable information is presented to decision makers to attenuate the coming coupling, which can speed up design process greatly. After a few iterations of design collaboration, the final values for decision variables, specification variables and goal variables are given in Table 2 with the help of the design standard of motor and lead screw. The results are obtained by maximising the satisfaction degree for each design task. Table 2 shows that the speed of the feeding servo system is improved greatly. But the rotary inertia of the motor is relatively big because fairly high values are assigned to rotary inertia of lead screw J_{sp} and rotary inertia of working platform J_T in order to make the transmission module stiff enough. Compared to the design requirements, the above design results are satisfactory and can best meet the design expectation of the customer. However, designers may obtain different design results if different utility models are used. The final choice will be determined according to the designer's as well as the customer's preferences.

6 Conclusions

In this paper, a sensitivity analysis method to handle propagation couplings is proposed, mainly involving design parameters. The system introduced in this paper has the scalability to be applied to much larger coupled systems. The merits of this proposed method can be justified in three folds. Firstly, usually not all elements of a complex system are critical to the holistic performance of the system, so analysis of sensitivity and design space variation can be conducted in order to identify the key design control links in the system. Secondly, although the case study in this paper only includes two tasks, the analysis method can be extended to the whole design process according to the specific design propagation route. This is because parameter dependencies are always there in those coupled tasks and the proposed method will be useful to identify the effective design parameters to be changed when propagating design evolvement changes. Thirdly, with the development of automatic differentiation techniques (Bücker et al., 2006), the workload for modelling complex systems can be further reduced if they are combined with our analysis method and system.

The contributions of the paper are:

- Concurrent and sequential couplings are introduced to represent different coupling scenarios caused by design propagations.
- A general method to deal with design couplings in engineering design lifecycle is proposed based on three kinds of sensitivity analyses. The method can orientate the designer by identifying those design change parameters and hence reduce the designer's cyclic revision time in solving propagation parameter coupling problems.
- A prototype system has been developed to realise the analysis methods and a dynamically updated user interface is designed which can graphically display the computation results for designers.

Compared with the former methods for solving coupled design problems, the proposed method is more effective because the analysis is based on the situated design scenarios and can guide designers in their dynamic design decision-making process. Thus, overwhelming re-design effort avalanche caused by design change propagation can be avoided.

Acknowledgements

This research work is funded by the National Natural Science Foundation of China with the grant number of 50705083 and 51175457.

References

Balling, R. and Sobieski, S.J. (1996) 'Optimization of coupled systems: a critical overview of approaches', *AIAA Journal*, Vol. 34, No. 1, pp.6–17.

Boudouh, T., Anghel, D., Garro, O. (2006) 'Design iterations in a geographically distributed design process', *Advances in Design*, Springer, London, pp.377–385.

Bücker, M., Corliss, G., Hovland, P., Naumann, U. and Norris, B. (Eds.) (2006) *Automatic Differentiation: Applications, Theory, and Implementations*, Springer, The Netherlands.

Chamis, C. (1999) 'Coupled multidisciplinary optimization of engine structural performance', *Journal of Aircraft*, Vol. 36, No. 1, pp.190–199.

Chanron, V. and Lewis, K.E. (2006) 'The dynamics of decentralized design processes: the issue of convergence and its impact on decision-making', in Lewis, K., Chen, W. and Schmidt, L.C. (Eds.): *Decision Making in Engineering Design*, pp.281-290, ASME Press, New York.

Chen, B., Xiao, R., Liu, J. and Zhong, Y. (2001) 'Research on the coupled factors in multi-disciplinary optimization for complex products', *Chinese Journal of Mechanical Engineering*, Vol. 37, No. 1, pp.19–23.

Eckert, C., Clarkson, P.J. and Zanker, W. (2004) 'Change and customization in complex engineering domains', *Research in Engineering Design*, Vol. 15, No. 1, pp.1–21.

English, K. and Bloebaum, C.L. (1998) 'Development of multiple cycle coupling suspension in complex system optimization', *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2–4 September, St. Louis, MO.

English, K. and Bloebaum, C.L. (2000) 'A comparison of optimization techniques for solving the coupling suspension problem', *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, 6–8 September.

English, K. and Bloebaum, C.L. (2001) 'Enhanced total derivative based coupling suspension in complex system optimization', *39th Aerospace Sciences Meeting and Exhibit*, Reno, NV, 8–11 January.

English, K., Miller, E. and Bloebaum, C.L. (1996) 'Total derivative-based coupling suspension for system reduction in complex design', *6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Belleuve WA, 4–6 September.

Kim, H., Michelena, N., Papalambros, P. and Jiang, T. (2003) 'Target cascading in optimal system design', *Transactions of ASME Journal of Mechanical Design*, Vol. 125, No. 3, pp.474–480.

Kim, H., Ragon, S., Soremekun, G., Malone, B. and Sobieszczanski-Sobieski, J. (2004) 'Flexible approximation model approach for bi-level integrated system synthesis', *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, 30 August–1 September.

Kroo, I., Altus, S., Braun, R., Gage, P. and Sobieski, I. (1994) 'Multidisciplinary optimization methods for aircraft preliminary design, AIAA-94-4325-CP', *Proceedings of the 5th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Panama City, Florida, 7–9 September, pp.697–707.

Kusiak, A. and Wang, J. (1995) 'Dependency analysis in constraint negotiation', *IEEE Transactions on Systems, Man and Cybernetics (Part B)*, Vol. 25, No. 9, pp.1301–1313.

Li, Y.L. and Pan, S.X. (2006) 'Sensitivity and knowledge based collaborative product decision making', *Chinese Journal of Mechanical Engineering*, Vol. 42, No. 9, pp.33–42.

Li, Y.L. and Zhao, W. (2011) 'Flexible design process and resource integration framework for agile and innovative reuse design of complex products', *International Journal of Computer Integrated Manufacturing*, Vol. 24, No. 1, pp.53–73.

Nair, P. and Keane, A. (2002) 'Coevolutionary architecture for distributed optimization of complex coupled systems', *AIAA Journal*, Vol. 40, No. 7, pp.1434–1443.

Parashar, S. and Bloebaum, C.L. (2006) 'Multi-objective genetic algorithm concurrent subspace optimization (MOGACSSO) for multidisciplinary design', *47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 1–4 May, Newport, Rhode Island.

Renaud, J. and Gabriele, G. (1993) 'Improved Coordination in non-hierarchic system optimization', *AIAA Journal*, Vol. 31, No. 12, pp.2367–2373.

Sobieski, I. and Kroo, I. (1995) 'Aircraft design using collaborative optimization', AIAA-96-07I5, *34th Aerospace Sciences Meeting and Exhibit*, 15–18 January, Reno, Nevada.

Sobieski, S.J. (1988) 'Optimization by decomposition: a step from hierarchic to non-hierarchic systems', *Proceedings of 2nd NASA/Air Force Symposium on Recent Advances in Multidisciplinary Analysis and Optimization*, Hampton, VA, September, pp.51–78.

Sobieski, S.J. (1990) 'Sensitivity of Complex, Internally Coupled Systems', *AIAA Journal*, Vol. 28, No. 1, pp.153–160.

Sobieski, S.J., Agte, J. and Sandusky, R. (2000) 'Bilevel integrated system synthesis', *AIAA Journal*, Vol. 38, No. 1, pp.164–172.

Tappeta, R. and Renaud, J. (1997) 'Multiobjective collaborative optimization', *Transactions of ASME Journal of Mechanical Design*, Vol. 119, No. 3, pp.403–411.

Tappeta, R. and Renaud, J. (1999) 'Interactive multiobjective optimization procedure', *AIAA Journal*, Vol. 37, No. 7, pp.881–889.

Tappeta, R., Renaud, J. and Rodriguez, J. (2000) 'An interactive multiobjective optimization design strategy for multidisciplinary systems', *4th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Material Conference and Exhibit*, Atlanta GA, 3–8 April.

Tosserams, S., Kokkolaras, M., Etman, L.F.P. and Rooda, J.E. (2010) 'A nonhierarchical formulation of analytical target cascading', *Transactions of ASME Journal of Mechanical Design*, Vol. 132, No. 5, 051002.

Wang, A. and Bai, E. (2003) *Modern Numerical Control Machine Tool*, National Defense Industry Press, Beijing.

Wujek, B.A. and Renaud, J.E. (1996) 'Automatic differentiation for more efficient multidisciplinary design analysis and optimization', *6th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 4–6 September, Belleuve WA.

Nomenclature

U_A	utility models for the specification and goal variables in task A
U_B	utility models for the specification and goal variables in task B
X_s^A	the specification variables in task A
X_s^B	the specification variables in task B
X_d^A	the decision variables in task A
X_d^B	the decision variables in task B
X_g^A	the goal variables in task A
X_g^B	the goal variables in task B

X^A	the design variables in task A, it is the union set of X_s^A, X_d^A and X_g^A
X^B	the design variables in task B, it is the union set of X_s^B, X_d^B and X_g^B
X_L^A	the lower bounds for the design variables of task A
X_L^B	the lower bounds for the design variables of task B
g_j^A	the inequality constraints in task A, and m_A is the number of inequality constraints
g_k^A	the equality constraints in task A, and l_A is the number of equality constraints
g_j^B	the inequality constraints in task B, and m_B is the number of inequality constraints
g_k^B	the equality constraints in task B, and l_B is the number of equality constraints
m_{AB}	the number of intersection set members for specification variables in task A and decision variables in task B
n_{AB}	the number of intersection set members for goal variables in task A and decision variables in task B
p_{AB}	the number of intersection set members for goal variables in task A and specification variables in task B.