Journal of Engineering Design
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/cjen20

Adaptable design of open architecture products with robust performance

Jian Zhang[ab], Deyi Xue[b] & Peihua Gu[a]

[a] Department of Mechatronics Engineering, Shantou University, Shantou, Guangdong, People's Republic of China

[b] Department of Mechanical and Manufacturing Engineering, University of Calgary, Calgary, Alberta, Canada

Published online: 18 Feb 2015.

Journal of Engineering Design, 2015
http://dx.doi.org/10.1080/09544828.2015.1012055

Taylor & Francis
Taylor & Francis Group

Adaptable design of open architecture products with robust performance

Jian Zhang[a,b], Deyi Xue[b] and Peihua Gu[a*]

[a]*Department of Mechatronics Engineering, Shantou University, Shantou, Guangdong, People's Republic of China;* [b]*Department of Mechanical and Manufacturing Engineering, University of Calgary, Calgary, Alberta, Canada*

(*Received 17 July 2014; accepted 22 January 2015*)

Adaptable design is an approach to design adaptable products whose modules/configurations and parameter values can be changed during an operation stage to satisfy different customer requirements. An open architecture product is an adaptable product with open interfaces to allow the third-party vendors to develop new add-on modules and connect these add-on modules through the open interfaces. In this work, an open architecture adaptable product is modelled by a platform, alternative add-on modules, and open interfaces to connect the add-on modules with the platform. Both the specific add-on modules that need to be designed at the product development stage and the unknown add-on modules that could be added in the future are considered. In this research, a novel robust design approach is introduced to identify the optimal design of an open architecture adaptable product whose functional performance measures are the least sensitive to variations of the product and operating parameters due to uncertainties. First, characteristics of open architecture adaptable products are discussed. Methods for modelling of platform, add-on modules, and open interfaces are then introduced. A multi-level optimisation method is subsequently explained to identify the optimal design configuration and parameters considering product performance measures and their variations.

Keywords: adaptable products; open architecture products; robust design; uncertainties

1. Introduction

Increasing competition in the global marketplace demands products be adaptable to the changes of functional requirements, operation environments, and technology advancement. Adaptable product is the one that can be changed/adapted, such as reconfigured and upgraded, during a product operation stage to satisfy different requirements of customers (Gu, Hashemina, and Nee 2004). Compared with the traditional products whose functional performance and working conditions specified in the design stage cannot be changed after the products are manufactured, an adaptable product can be easily modified in the product operation stage to satisfy the changed requirements.

Adaptable design is the design approach for developing adaptable products (Gu, Hashemina, and Nee 2004). Since introduction of the adaptable design concept, many adaptable design methods have been developed in the past decade (Gu, Xue, and Nee 2009). Li, Xue, and Gu (2008) introduced new adaptability evaluation measures considering the extendibility of functions,

*Corresponding author. Email: peihuagu@stu.edu.cn

upgradeability of modules, and customisability of components for identification of the optimal adaptable design based on evaluation to different design candidates. By comparing the actual structure of the product with its ideal structure that can be easily changed, Fletcher, Brennan, and Gu (2009) developed a method to evaluate adaptability of an adaptable product. Cheng et al. (2011) developed a structure-based approach to evaluate design by measuring essential adaptability and behavioural adaptability. Xue et al. (2012) extended the adaptable design method through modelling of the changeable requirements and product descriptions as numerical functions of the product life-cycle time parameter. The four key issues in research of adaptable design, including function modelling, design modelling, design evaluation and design process, requirements and methods in the key issues of adaptable design as well as applications of the adaptable design methods were summarised in a literature review by Gu, Xue, and Nee (2009).

Product architecture also provides influence on an adaptable product. In general, both closed architecture and open architecture can be used in adaptable design (Koren et al. 2013; Peng et al. 2013). For an adaptable product with closed architecture, although interfaces among various changeable modules of the product are designed and these modules are connected through these interfaces to achieve different functions, these functional requirements are specified by the original equipment manufacturer (OEM). These modules are produced by the OEM and/or its suppliers according to specifications defined by the OEM. Most of the traditional reconfigurable products belong to this category. An open architecture product (OAP) can be considered as the product with a platform such that add-on modules developed by different vendors can be connected with the platform through interfaces of the platform (Koren et al. 2013). For an adaptable product with open architecture, specifications of the interfaces are open to the public to allow the third-party vendors to develop new modules with new functions as they like. A personal computer is a typical OAP that allows different devices developed by the third-party vendors to be connected with the motherboard through open USB interfaces. For the OAP, since different manufacturers/customers can participate the design of add-on modules, product variety can be easily achieved. Product sustainability, adaptability, upgradeability, and extendibility can also be achieved by designing an OAP (Koren et al. 2013).

Despite the progress in research on adaptable design, the influence of uncertainties on performances of an OAP has never been investigated. Robustness is considered as the product's capability to resist the influence of uncertainties on product performances. Product robustness is usually measured by the sensitivity of functional performance to parameter variations caused by uncertainties. Many robust design methods have been developed in the past decades. In this research area, Taguchi (1978, 1993) developed a robust design method by using the signal-to-noise ratio (SNR) to measure the product robustness. Parkinson, Sorensen, and Pourhassan (1993) developed the robust design methods based on the analysis of the relations among product performance measures, product/operating parameters, and variations of these parameters. Du and Chen (2000) developed a robust design method to maintain the robustness of design feasibility by considering the influence of uncertainties on design constraints. Fonseca, Friswell, and Lees (2007) and Kumar et al. (2008) developed the randomised robust design methods by employing the Monte-Carlo method to simulate the influence of uncertainties on product functional performance. Samadiani et al. (2009) improved the robustness of a system considering different operation conditions. Zhang et al. (2010) and Hu, Azarm, and Almansoori (2013) developed the robust design methods considering multiple design objective functions.

For the design of an adaptable product, robustness needs to be considered to improve the product quality (Zhang, Xue, and Gu 2012). Since modules/configurations and parameters of an adaptable product are changeable during the product operation stage, the existing robust design methods cannot be used directly for developing a robust adaptable product. Therefore, a robust adaptable design method is required to create an adaptable product whose functional performances are insensitive to uncertainties in parameters. In the research by Zhang et al.

(2014), a robust adaptable design method considering changes of both configurations and parameter values was introduced. However, this robust adaptable design method cannot be used directly to design an OAP. The research presented in this paper aims at introducing a robust adaptable design approach for developing OAPs such that an OAP is adaptable to various changes in requirements during the product operation stage, meanwhile the functional performance measures are the least sensitive to parameter variations.

2. Open architecture products

In this research, an OAP is considered as the one with a platform and open interfaces through which different add-on modules from different sources can be connected to satisfy the requirements of customers. Characteristics of OAPs include the following:

(1) An OAP is composed of a platform, add-on modules, and open interfaces to connect the platform and the add-on modules.
(2) Specifications and constraints of the open interface parameters are open to the public.
(3) The platform and add-on modules are connected by open interfaces through relationships defined by input and output parameters.
(4) Add-on modules can be specific ones that need to be designed during the product development stage and unknown ones that could be designed and added in the future.
(5) Add-on modules can be provided by both the OEM and the third-party vendors.

OAPs provide variety and flexibilities to customers. At the product purchasing stage, customers can select the required add-on modules from all available ones based on their needs. At the product operation stage, customers can change, upgrade, and extend functions by using different add-on modules.

Research on design of OAPs is based on the results achieved on design of mass produced products, mass customised products, reconfigurable products, and upgradeable products (Gu, Hashemina, and Nee 2004). Comparison between OAP with other types of products is provided in Table 1. For the mass produced products, functions are determined by the manufacturer and cannot be changed. For the mass customised products, different options are available for customers to select at the product purchasing stage. The mass customised products, however, usually cannot be changed in the product operation stage. Both reconfigurable products and upgradeable products enable customers to change product configurations during the product operation stage to satisfy different needs of customers. A reconfigurable product is usually used to replace multiple products with a single one. New components/modules are not added to the reconfigurable product during the product operation stage. For an upgradable product, new components/modules are usually used to replace the old ones for improving functions of the product.

Table 1. Comparison among different types of products.

Type of product	Achievement of different needs: Different options at product purchasing stage	Achievement of different needs: Different options at product operation stage	New components and/or modules added at operation stage
Mass produced product	No	No	No
Mass customised product	Yes	No	No
Reconfigurable product	No	Yes	No
Upgradeable product	No	Yes	Yes
OAP	Yes	Yes	Yes

2.1. *Platform and add-on modules in an OAP*

In this research, an OAP as shown in Figure 1 is composed of a platform, different add-on modules, and interfaces to connect different add-on modules with the platform.

Generally, the platform (M^{P}) is designed and manufactured by the OEM. The platform of an OAP can have several open interfaces to connect different add-on modules. In robust design of the open architecture adaptable product, both the specific add-on modules designed at the product development stage and the unknown add-on modules which could be added later on during the product operation stage should be considered.

The specific add-on modules, $M^{\mathrm{S}}_{im_i}(i = 1, 2, \ldots, l)$, are the ones whose configurations and parameters need to be determined at the product design stage. For each interface of the platform in an OAP, usually several add-on modules need to be designed to achieve different functions. Since all the design details of the specific add-on modules are determined at the design stage, performance measures and their variations of an OAP with specific add-on modules can be calculated for evaluation of the product robustness. The unknown add-on modules, $M^{\mathrm{U}}_i(i = 1, 2, \ldots, l)$, are those that could be added later on in the product operation stage. Since configurations and parameters of these unknown add-on modules cannot be decided at the design stage, performance measures of the OAP with unknown add-on modules are not described by specific values and large variations of these performance measures are also expected. In this research, an unknown add-on module is defined by constraints on interface parameters such as the available value sets for discrete interface parameters and the upper/lower boundaries for continuous interface parameters.

2.2. *Open interfaces in an OAP*

The open interfaces in an OAP are used to connect the various add-on modules to the platform. Compared with a closed architecture product where interfaces are used to connect predefined modules, the open interfaces in an OAP allow new add-on modules developed by the third-party vendors to be connected with the platform to achieve additional functions.

In this research, interactions between platform and add-on modules in an OAP are defined by input and output parameters of the open interfaces. Since both the specific add-on modules and the unknown add-on modules are considered in this work, the input/output parameters of interfaces for interactions between the platform and the add-on modules should be defined differently:

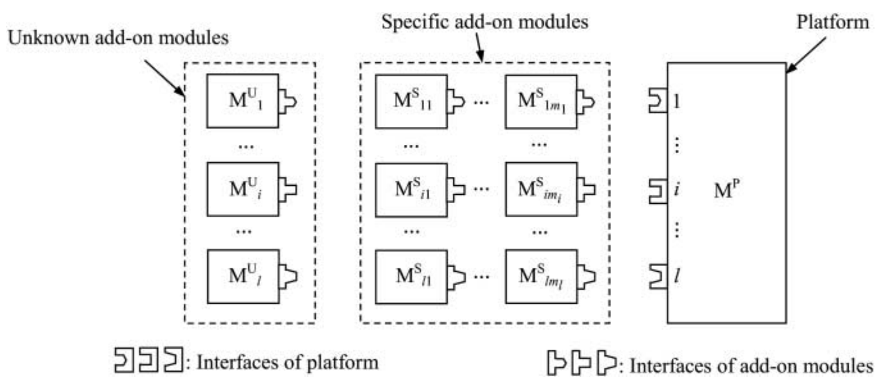

Figure 1. Platform and add-on modules of an OAP.

- *Input/output parameters of interfaces for specific add-on modules*
  Since design configurations and parameters of the add-on modules are determined at the product development stage, the values of the input/output interface parameters can be calculated from the parameters of the platform and add-on modules.
- *Input/output parameters of interfaces for unknown add-on modules*
  Since design configurations and parameters of the add-on modules are not determined at the product development stage, the values of the input/output interface parameters have to be defined by constraints. In this research, constraints for discrete and continuous parameters are considered.

2.3. *Robustness of an OAP*

An adaptable design approach can be used for the design of OAPs. Generally, an OAP can be evaluated by different measures such as performance, product adaptability, cost, and so on. In this research, robustness is selected as the evaluation measure to identify the optimal design of OAP considering uncertainties in product/operating parameters.

Adaptable products can be evaluated by their capabilities of change in the operation stage such as adaptabilities (or flexibilities) (de Neufville and Scholtes 2011; Inkermann, Stechert, and Vietor 2013). Adaptability is used to evaluate the capability of a product to be changed in the operation stage to satisfy different requirements (Gu, Hashemina, and Nee 2004; Olewnik and Lewis 2006). For an OAP, both the specific add-on modules with specific functions and the unknown add-on modules without specified functions are considered. Product adaptability measure can be achieved through connections of different add-on modules, including the specific add-on modules and the unknown add-on modules, with the platform of an OAP. In the research area of adaptable design, different methods have been developed for the evaluation of product adaptability (Suh, de Weck, and Chang 2007; Gu, Xue, and Nee 2009; Inkermann, Stechert, and Vietor 2013). Despite the progress, robustness, which can significantly influence the quality of an OAP, has never been considered in the past.

Robustness is a measure to evaluate a design considering both the product performances and the variations of these performances. Generally, product robustness can be improved through minimising the sensitivity of product performance to parameter variations caused by uncertainties (Taguchi 1993). For an OAP, robustness is influenced by (1) add-on modules including specific and unknown add-on modules, (2) configurations of platform and add-on modules, and (3) parameter values associated with configurations. Since robustness of OAP has not been considered in adaptable design, the achieved OAPs based on the existing adaptable design methods are vulnerable to the variations of parameters due to uncertainties.

In this research, robustness is selected as the evaluation measure by assuming the adaptabilities and costs of all feasible design candidates are comparable.

3. An adaptable design approach for developing robust OAPs

An OAP is one type of adaptable products. In this research, an adaptable design approach is introduced to identify the robust OAP whose functional performance measures are the least sensitive to the variation of parameters due to uncertainties.

3.1. *Modelling of an OAP*

An OAP is composed of a platform, add-on modules, and open interfaces. During the operation stage, different add-on modules are connected to the platform through the open interfaces to form different *operation configuration states* for achieving different functions. At the design stage, different *design configuration candidates* are provided such that the best one can be selected considering product robustness. Both the operation configuration states and the design configuration candidates are further modelled by parameters. The platform and add-on modules are connected through open interfaces.

3.1.1. *Modelling of product operation configuration states*

An OAP usually has several open interfaces. For each interface, different add-on modules can be connected with the platform. For the OAP shown in Figure 1, the platform (M^{P}) has l interfaces. For the i-th ($i = 1, \ldots, l$) interface, m_i specific add-on modules $M_{i1}^{\mathrm{S}}, \ldots, M_{im_i}^{\mathrm{S}}$ and the unknown add-on module M_i^{U} are considered.

During the product operation stage, each interface of the platform can be used to connect with different add-on modules. In this work, each combination of the add-on modules and the platform for achieving a certain function is called an *operation configuration state*. When n operation configuration states of an OAP are considered, collection of these operation configuration states, S, is defined by

$$S = \{S_1, \ldots, S_n\}. \tag{1}$$

Each operation configuration state is associated with a probability representing the percentage of time this operation configuration state is used considering all operation configuration states. Collection of the probabilities for the n operation configuration states, P, is defined by

$$P = \{P_1, \ldots, P_n\}. \tag{2}$$

The i-th operation configuration state, S_i ($i = 1, \ldots, n$), is modelled by its platform and add-on modules:

$$S_i = \{M_1, \ldots, M_l, M^{\mathrm{P}}\}, \tag{3}$$

where M_j represents the add-on module connected through the j-th open interface of the platform M^{P} ($j = 1, \ldots, l$).

3.1.2. *Modelling of product design configuration candidates*

For the design of an OAP, different feasible design configuration candidates can be created from the same design requirements. In this work, the different alternative configurations for selection in the design stage are called *design configuration candidates*. The platform and each of the specific add-on modules can be modelled by an AND–OR tree (Gu, Xue, and Nee 2009). In an AND–OR tree, when all the sub-nodes need to be selected to support a super-node, all these sub-nodes are associated with an AND relation. As shown in Figure 2(a), for example, a personal computer is composed of a motherboard, an internal data drive, a monitor, a mouse, etc. When the super-node is supported by one of its sub-nodes in the process of design configuration candidate selection, all these sub-nodes are associated with an OR relation. For example, either the DVD drive or the Blu-ray drive needs to be selected for the internal data drive. For each AND–OR tree, a configuration candidate with only AND relations can be created based on the following rules:

- The root node should be selected first.
- When the sub-nodes of a selected node are associated with an AND relation, all these sub-nodes should be selected.

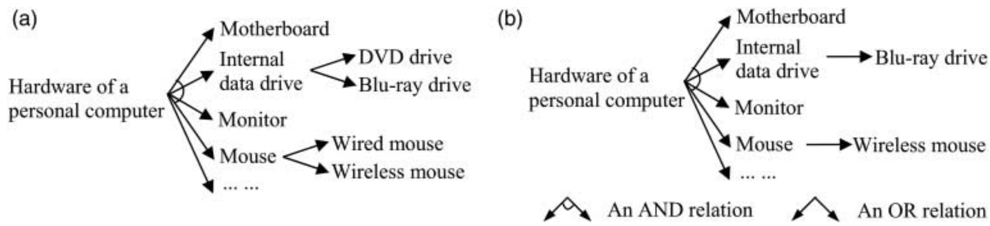

Figure 2. Modelling of design configuration candidates. (a) Modelling of feasible design configuration candidates. (b) A design configuration candidate generated from the AND–OR tree.

- When the sub-nodes of a selected node are associated with an OR relation, only one of these sub-nodes should be selected.

Figure 2(a) shows an AND–OR tree and Figure 2(b) shows a created feasible design configuration candidate with only AND relations. A complete design configuration candidate is modelled by a design configuration candidate for the platform and the design configuration candidates for all the add-on modules.

3.1.3. *Modelling of parameters*

A product operation configuration state or a product design configuration candidate is further modelled by parameters. In this work, the parameters of an OAP are classified into two categories: design parameters and non-design parameters (Zhang, Xue, and Gu 2012):

- *Design parameters* are those whose values need to be determined at the design stage. In this work, the design parameters are further classified into un-adaptable design parameters and adaptable design parameters.
  - *Un-adaptable design parameters* are the parameters whose values are not changed in the product operation stage. For example, the width of an office chair is an un-adaptable design parameter. In this work, the values of un-adaptable design parameters are achieved through optimisation.
  - *Adaptable design parameters* are the parameters whose values need to be adapted at the product operation stage when requirements are changed. For example, the height of an office chair is an adaptable design parameter which can be adjusted for different persons. In this work, the values of adaptable design parameters are calculated from requirements, working conditions and other product parameters based on design rules.
- *Non-design parameters* are those whose values are provided as given conditions in design. For example, the mechanical properties of materials selected for the office chair are non-design parameter.

Design parameters (i.e. un-adaptable design parameters and adaptable design parameters) used in this research are different from the design parameters used in the axiomatic design. In the axiomatic design, functional requirements belong to functional domain and design parameters belong to physical domain. Functional requirements can be satisfied by the selection of physical design parameters (Suh 1999). Product performance is used to measure the degree that a functional requirement is satisfied by design parameters. In the axiomatic design, the physical design parameters can be used to represent all physical entities including modules, configurations, and parameters created in design (Suh 2001). The design parameters in our work are used to model parameters of product modules and configurations.

For an OAP, the values of the input/output parameters of the open interfaces are calculated from the un-adaptable design parameters, adaptable design parameters, and non-design parameters of the platform and add-on modules. In addition, constraints can also be applied to these parameters.

For an OAP, since both the specific add-on modules and the unknown add-on modules need to be considered, the values of the adaptable design parameters are changed in two different ways in the product operation stage.

- Change of adaptable design parameter values with specific add-on modules
  When all the add-on modules are specific add-on modules, the relation among product functional performance F, un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$, adaptable design parameters $\boldsymbol{X}^{\mathrm{A}}$, and non-design parameters $\boldsymbol{X}^{\mathrm{N}}$ can be described by $F = f(\boldsymbol{X}^{\mathrm{D}},\boldsymbol{X}^{\mathrm{A}},\boldsymbol{X}^{\mathrm{N}})$. In this case, the adaptable design parameters are calculated from the target functional performance, un-adaptable design parameters, and non-design parameters. In other words, the values of $\boldsymbol{X}^{\mathrm{A}}$ are calculated from F, $\boldsymbol{X}^{\mathrm{D}}$, and $\boldsymbol{X}^{\mathrm{N}}$. In this work, the unique values of adaptable design parameters are obtained based on the experience of design engineer to avoid the problem of multiple solutions.
- Change of adaptable design parameter values with unknown add-on modules
  When some add-on modules are unknown add-on modules, since these unknown add-on modules are only defined by the constraints of interface parameters, the relation among product functional performance F, un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$, adaptable design parameters $\boldsymbol{X}^{\mathrm{A}}$, non-design parameters $\boldsymbol{X}^{\mathrm{N}}$, input parameters of unknown add-on modules $\boldsymbol{I}^{\mathrm{U}}$, and output parameters of unknown add-on modules $\boldsymbol{O}^{\mathrm{U}}$ can be described by $F = f(\boldsymbol{X}^{\mathrm{D}},\boldsymbol{X}^{\mathrm{A}},\boldsymbol{X}^{\mathrm{N}},\boldsymbol{I}^{\mathrm{U}},\boldsymbol{O}^{\mathrm{U}})$. In this case, the adaptable design parameters are calculated from the target functional performance, un-adaptable design parameters, non-design parameters, and input/output interface parameters of the unknown add-on modules. In other words, the values of $\boldsymbol{X}^{\mathrm{A}}$ are calculated from F, $\boldsymbol{X}^{\mathrm{D}}$, $\boldsymbol{X}^{\mathrm{N}}$, $\boldsymbol{I}^{\mathrm{U}}$, and $\boldsymbol{O}^{\mathrm{U}}$. Since the input/output interface parameters are usually defined by discrete and continuous parameters, all these possible input/output interface parameter values have to be considered. In addition, the experience of design engineer is used in this work to obtain the unique values of adaptable design parameters to avoid the problem of multiple solutions.

3.1.4. *Modelling of interfaces for the interactions between platform and add-on modules*

For an OAP, the add-on modules are connected with the platform through the open interfaces. In this work, the interactions between the platform and the add-on modules are defined by the input and output parameters of the open interfaces. Figure 3 shows interactions between a platform and add-on modules through interfaces. The values of the input parameters of an interface for an add-on module are determined by the values of the corresponding output parameters of the interface for the platform, while the values of the input parameters of an interface for the platform are determined by the values of the corresponding output parameters of the interface for the add-on module.

When a specific add-on module is connected with the platform, such as the M_i^{S} for the *i*-th interface shown in Figure 3, the values of the input and output parameters of the interfaces for the platform and the specific add-on module are calculated from the parameters of the platform and the add-on module.

When an unknown module is considered, such as the M_j^{U} for the *j*-th interface shown in Figure 3, constraints are then used to define the input and output parameters of the interface for the unknown add-on module. In this work, the input and output parameters of the interface for the unknown add-on module are defined by continuous and discrete parameters.

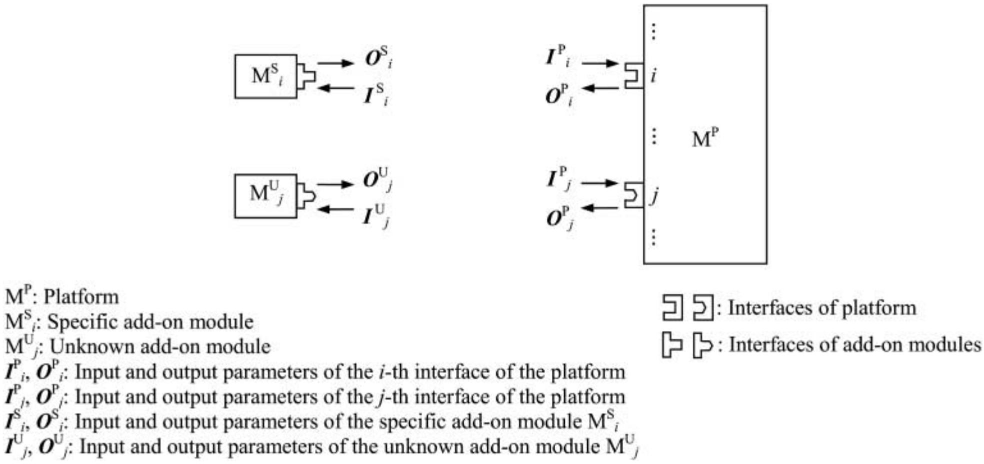

Figure 3. Interactions between platform and add-on modules.

When the interface input/output parameters are continuous parameters, they are defined by

$$I_j^{\mathrm{U}} \in [L_j^{\mathrm{I}}, U_j^{\mathrm{I}}] \quad \text{and} \quad O_j^{\mathrm{U}} \in [L_j^{\mathrm{O}}, U_j^{\mathrm{O}}], \tag{4}$$

where I_j^{U} and O_j^{U} are the input and output parameters of the unknown add-on module, and $[L_j^{\mathrm{I}}, U_j^{\mathrm{I}}]$ and $[L_j^{\mathrm{O}}, U_j^{\mathrm{O}}]$ are the lower/upper boundaries of the I_j^{U} and O_j^{U}, respectively.

When the interface input/output parameters are discrete parameters, they are defined by

$$I_j^{\mathrm{U}} \in \{A_j^{\mathrm{I}}, B_j^{\mathrm{I}}, \ldots\} \quad \text{and} \quad \in O_j^{\mathrm{U}} \in \{A_j^{\mathrm{O}}, B_j^{\mathrm{O}}, \ldots\}, \tag{5}$$

where $\{A_j^{\mathrm{I}}, B_j^{\mathrm{I}}, \ldots\}$ and $\{A_j^{\mathrm{O}}, B_j^{\mathrm{O}}, \ldots\}$ are the feasible values of I_j^{U} and O_j^{U}, respectively.

3.2. *Evaluation of robustness for an OAP*

3.2.1. *Robustness with specific add-on modules*

In the product operation stage, an OAP is changed to different operation configuration states to achieve different required functions. In this work, suppose that $F_{s,1}, \ldots, F_{s,n_s}$ are functional performance measures of the s-th operation configuration state, and $\boldsymbol{X}^{\mathrm{D}}$, $\boldsymbol{X}^{\mathrm{A}}$, and $\boldsymbol{X}^{\mathrm{N}}$ are the un-adaptable design parameters, adaptable design parameters, and non-design parameters, the functional performance $F_{s,k}$, can be calculated by

$$F_{s,k} = \Phi_{s,k}(\boldsymbol{X}^{\mathrm{D}}, \boldsymbol{X}^{\mathrm{A}}, \boldsymbol{X}^{\mathrm{N}}), \quad k = 1, \ldots, n_s,\ s = 1, \ldots, n, \tag{6}$$

where $\Phi_{s,k}()$ represents the relation among the parameters $\boldsymbol{X}^{\mathrm{D}}$, $\boldsymbol{X}^{\mathrm{A}}$, $\boldsymbol{X}^{\mathrm{N}}$, and $F_{s,k}$, n_s is the number of functional performance measures for the s-th operation configuration state, and n is the total number of operation configuration states.

In this work, robustness is used to evaluate the quality of design considering both the performance measures and the variations of these performance measures due to uncertainties. Many measures, such as the SNR and the variance of functional performance, can be used to evaluate the robustness of a product design (Taguchi 1993). Since both functional performance and variation of the performance are considered in the SNR, the SNR has been selected in this research as the robustness evaluation measure. Robustness of the s-th operation configuration state for the i-th design configuration candidate considering the parameters $\boldsymbol{X}^{\mathrm{D}}$ can be calculated by

$$R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}) = \sum_{k=1}^{n_s}\left[\omega_{s,k} 10 \log_{10}\left(\frac{\mu_{s,k}^2}{\sigma_{s,k}^2}\right)\right], \tag{7}$$

where $\mu_{s,k}$ and σ_{s,k^2} represent the expected value and the variance of $F_{s,k}$ which can be calculated using Equation (6), respectively, $\omega_{s,k}$ represents the weight for the performance measure $F_{s,k}$, and n_s is the number of functional performance measures for the s-th operation configuration state.

The overall robustness of the i-th design configuration candidate considering all the n operation configuration states with the un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$ can be calculated by

$$R^{(i)}(\boldsymbol{X}^{\mathrm{D}}) = \sum_{s=1}^{n}[P_s \cdot R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}})], \tag{8}$$

where P_s represents the probability of use for the s-th operation configuration state.

From Equation (8), the optimal parameters of $\boldsymbol{X}^{\mathrm{D}}$ to achieve the best overall robustness for the i-th design configuration candidate can be obtained through optimisation:

$$R^{(i)} = \max_{\text{w.r.t.}\boldsymbol{X}^{\mathrm{D}}} : R^{(i)}(\boldsymbol{X}^{\mathrm{D}}). \tag{9}$$

3.2.2. *Robustness with unknown add-on modules*

For an unknown add-on module, since values of the interface input/output parameters can be varied as continuous values and discrete values, the product performance measures are also influenced by the changes of these interface input/output parameters. In this work, suppose that $\boldsymbol{I}^{\mathrm{U}}$ and $\boldsymbol{O}^{\mathrm{U}}$ represent the input and output parameters of the unknown add-on modules, respectively, and the functional performance $F_{s,k}$ can be calculated by

$$F_{s,k} = \Phi_{s,k}(\boldsymbol{X}^{\mathrm{D}}, \boldsymbol{X}^{\mathrm{A}}, \boldsymbol{X}^{\mathrm{N}}, \boldsymbol{I}^{\mathrm{U}}, \boldsymbol{O}^{\mathrm{U}}), \quad k = 1, \ldots, n_s,\ s = 1, \ldots, n, \tag{10}$$

where $\Phi_{s,k}()$ represents the relation among the parameters $\boldsymbol{X}^{\mathrm{D}}$, $\boldsymbol{X}^{\mathrm{A}}$, $\boldsymbol{X}^{\mathrm{N}}$, $\boldsymbol{I}^{\mathrm{U}}$, $\boldsymbol{O}^{\mathrm{U}}$, and $F_{s,k}$, n_s is the number of functional performance measures for the s-th operation configuration state, and n is the total number of operation configuration states.

Robustness of the s-th operation configuration state for the i-th design configuration candidate considering the parameters $\boldsymbol{X}^{\mathrm{D}}$, $\boldsymbol{I}^{\mathrm{U}}$, and $\boldsymbol{O}^{\mathrm{U}}$ can be calculated by

$$R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}, \boldsymbol{I}^{\mathrm{U}}, \boldsymbol{O}^{\mathrm{U}}) = \sum_{k=1}^{n_s}\left[\omega_{s,k} 10 \log_{10}\left(\frac{\mu_{s,k}^2}{\sigma_{s,k}^2}\right)\right], \tag{11}$$

where $\mu_{s,k}$ and σ_{s,k^2} represent the expected value and the variance of $F_{s,k}$ which can be calculated using Equation (10), $\omega_{s,k}$ represents the weight for the performance measure $F_{s,k}$, and n_s is the number of functional performance measures for the s-th operation configuration state.

Since the values of the interface input/output parameters for the unknown add-on modules are varied as continuous and discrete parameters, the influence of the different values of interface input/output parameters for the unknown add-on modules on product performance measures has to be considered. In this research, a statistical method and a worst-case method are developed considering possible changes of interface parameter values for the unknown add-on modules.

- The statistical method for calculation of robustness

Changes of input/output interface parameters can be continuous and discrete. If $\boldsymbol{I}^{\mathrm{U}}$ and $\boldsymbol{O}^{\mathrm{U}}$ are continuous parameters, $P(\boldsymbol{I}^{\mathrm{U}})$ and $P(\boldsymbol{O}^{\mathrm{U}})$ are the probability density functions of $\boldsymbol{I}^{\mathrm{U}}$ and $\boldsymbol{O}^{\mathrm{U}}$ for the s-th operation configuration state, and $\boldsymbol{I}_{\mathrm{L}}^{\mathrm{U}}, \boldsymbol{I}_{\mathrm{U}}^{\mathrm{U}}, \boldsymbol{O}_{\mathrm{L}}^{\mathrm{U}},$ and $\boldsymbol{O}_{\mathrm{U}}^{\mathrm{U}}$ are the lower/upper boundaries of the $\boldsymbol{I}^{\mathrm{U}}$ and $\boldsymbol{O}^{\mathrm{U}}$, the robustness of the s-th operation configuration state for the i-th design configuration candidate considering the un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$ is calculated by

$$R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}) = \int_{\mathbf{I}_{\mathrm{L}}^{\mathrm{U}}}^{\mathbf{I}_{\mathrm{U}}^{\mathrm{U}}} \int_{\mathbf{O}_{\mathrm{L}}^{\mathrm{U}}}^{\mathrm{U}_{\mathrm{U}}} P(\boldsymbol{I}^{\mathrm{U}})P(\boldsymbol{O}^{\mathrm{U}})R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}, \boldsymbol{O}^{\mathrm{U}}, \boldsymbol{I}^{\mathrm{U}})\mathrm{d}\boldsymbol{O}^{\mathrm{U}}\mathrm{d}\boldsymbol{I}^{\mathrm{U}}. \tag{12}$$

If $\boldsymbol{I}^{\mathrm{U}}$ and $\boldsymbol{O}^{\mathrm{U}}$ are discrete for the s-th operation configuration state, the robustness of the s-th operation configuration state for the i-th design configuration candidate considering the un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$ is calculated by

$$R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}) = \sum\sum P(\boldsymbol{I}^{\mathrm{U}})P(\boldsymbol{O}^{\mathrm{U}})R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}, \boldsymbol{I}^{\mathrm{U}}, \boldsymbol{O}^{\mathrm{U}}). \tag{13}$$

- The worst-case method for calculation of robustness
  In this method, the worst-case robustness considering changes of input/output interface parameters is calculated. The robustness of the s-th operation configuration state for the i-th design configuration candidate considering the un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$ is calculated by

$$R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}) = \min_{\mathrm{w.r.t.}\boldsymbol{I}^{\mathrm{U}},\boldsymbol{O}^{\mathrm{U}}} : R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}}, \boldsymbol{I}^{\mathrm{U}}, \boldsymbol{O}^{\mathrm{U}}). \tag{14}$$

The best robustness for the i-th design configuration candidate, $R^{(i)}$, considering un-adaptable design parameters $\boldsymbol{X}^{\mathrm{D}}$ can be calculated using Equation (9) in which the $R_s^{(i)}$ is calculated from Equation (12), (13), or (14).

3.3. *A multi-level optimisation model to identify the optimal design of OAP*

A multi-level optimisation model is developed in this work to identify the best design configuration candidate and its un-adaptable design parameter values considering robustness. In this optimisation model, first the optimal un-adaptable design parameter values for the i-th design configuration candidate are achieved through parameter optimisation.

$$\begin{aligned} &\mathit{Find}: \text{the un-adaptable design parameters } \boldsymbol{X}^{D} \\ &\mathit{Maximise}: R^{(i)} = \sum_{s=1}^{n} [P_s \cdot R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}})] \\ &\mathit{Subject\ to}: \boldsymbol{X}_{\mathrm{L}}^{\mathrm{D}} \leqslant \boldsymbol{X}^{\mathrm{D}} \leqslant \boldsymbol{X}_{\mathrm{U}}^{\mathrm{D}} \end{aligned} \tag{15}$$

where $\boldsymbol{X}_{\mathrm{L}}^{\mathrm{D}}$ and $\boldsymbol{X}_{\mathrm{U}}^{\mathrm{D}}$ represent the lower boundaries and upper boundaries of $\boldsymbol{X}^{\mathrm{D}}$, respectively. The robustness $R_s^{(i)}(\boldsymbol{X}^{\mathrm{D}})$ in Equation (15) is obtained from Equation (8), (12), (13), or (14).

Among all the p feasible product design configuration candidates, the optimal design configuration is obtained through configuration optimisation.

$$\begin{aligned} &\mathit{Find}: \text{the } i\text{-th design configuration candidate} \\ &\mathit{Maximise}: R = R^{(i)} \\ &\mathit{Subject\ to}: 1 \leqslant i \leqslant p \end{aligned} \tag{16}$$

where i represents the i-th design configuration candidate and p is the number of all feasible design configuration candidates.

To consider the influence of interface parameters of unknown add-on modules on the robustness, the statistical method and the worst-case method are employed. A third-level optimisation, modelled by Equation (14), is needed to calculate the worst-case influence of interface parameters of unknown add-on modules on the robustness of OAP.

In this research, parameter optimisation is conducted through numerical search (Arora 1989), while configuration optimisation is conducted by genetic programming (Koza 1992). Genetic programming is an evolutionary method to solve an optimisation problem when solutions to the optimisation problem can be modelled by tree data structures. In genetic programming, individuals (also called chromosomes) are used to describe the multiple solutions in the population of a generation. Reproduction, crossover, and mutation are the three operations used to evolve individuals from a generation to the next one with a better average evaluation measure (Hong et al. 2008). In this research, robustness is used as the evaluation measure.

4. Adaptable design of an open architecture linear labelling machine with robust performance

In this research, the newly developed design method has been applied to the design of a linear labelling machine with an open architecture.

4.1. *Design requirements*

A linear labelling machine is the equipment for applying self-adhesive labels on the containers such as bottles and boxes (Xiao 2009). Generally, a linear labelling machine is composed of a working table with power systems, a linear conveyor, an applicator, a pressing unit, and other auxiliary modules. At the operation stage, containers that need to be labelled are transported at a specific speed on the linear conveyor, the applicator is used to dispense self-adhesive labels from the supporting web at the required speed on containers, and the pressing unit is then used to apply forces to stick the labels with the containers (Huang 2007). The labelling machines are usually classified into two categories: vertical labelling machines and horizontal labelling machines. For both the vertical and horizontal labelling machines, as shown in Figure 4, the driving roller in the applicator module is rotated to pull the Web from the storage roller at a certain speed, and the labels at the edge of the plate are stripped off from the Web and moved on the tops of the containers. The driving roller in the pressing unit is rotated at a certain speed to apply the forces on the labels to stick the labels with the containers smoothly. Both the driving roller of the applicator and the driving roller of the pressing unit are powered through the power systems inside the working table. The power transmission system for the driving roller of the applicator in a labelling machine is shown in Figure 5. The power transmission system for the driving roller of the pressing unit works in a similar way.

For the linear labelling machine, the applicator and the pressing unit can be either vertical or horizontal to carry out different labelling tasks, the width of the web with attached labels can vary within a certain range, shapes of the container can be different, and the applicator and the pressing unit can have different functions (e.g. the pressing unit can also be used to count the number of containers that have been labelled). To satisfy these different labelling requirements, the applicator and the pressing unit need to be changed in the product operation stage. The current linear labelling machines are designed using closed product architecture with fixed applicators and fixed pressing units. When requirements of customers are changed, the currently designed labelling machines are hard to be changed to satisfy the changed requirements. In this research, design of a linear labelling machine with an open architecture is conducted such that

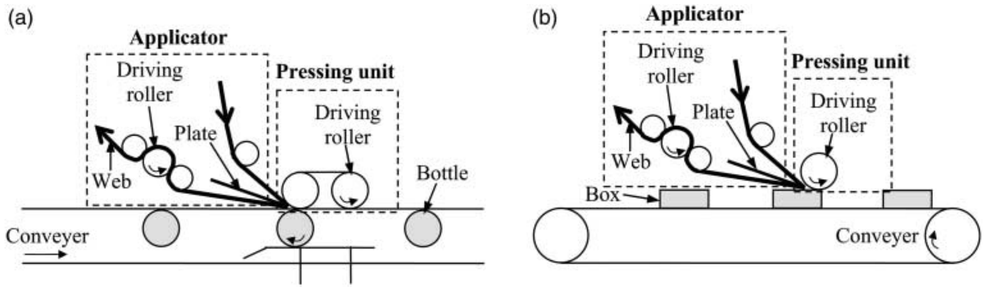

Figure 4. Schematic diagrams of linear labelling machines. (a) Diagram of vertical labelling machine (top view). (b) Diagram of horizontal labelling machine (front view).

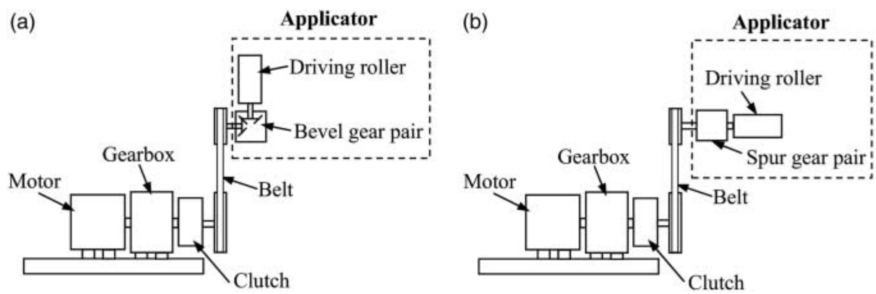

Figure 5. Power transmissions for the driving rollers of applicators in labelling machines. (a) Power transmission for the driving roller of applicator in a vertical labelling machine (right side view). (b) Power transmission for the driving roller of applicator in a horizontal labelling machine (right side view).

the applicator and the pressing unit can be changed in the operation stage to satisfy different requirements.

In addition, since the labels should be applied on the containers at the expected positions without wrinkles and bubbles, the variations of both labelling speed and pressing speed from the target values should be minimised, that is, both the labelling speed and the pressing speed should be the least insensitive to the parameter variations caused by uncertainties. In order to satisfy the requirements of customers considering both the product adaptation and the product robustness, the newly developed robust design approach is used in this work for the design of the open architecture adaptable linear labelling machine.

4.2. *Modelling of the open architecture linear labelling machine*

The platform and add-on modules of the open architecture labelling machine developed in this work are shown in Figure 6. The platform (M^{P}) has two open interfaces to connect with the applicator and the pressing unit. For the first interface, two specific add-on modules (i.e. the specific vertical applicator M_{11}^{S} and the specific horizontal applicator M_{12}^{S}) and one unknown applicator (M_1^{U}) are considered. For the second interface, two specific add-on modules (i.e. the specific vertical pressing unit M_{21}^{S} and the specific horizontal pressing unit M_{22}^{S}) and one unknown pressing unit (M_2^{U}) are considered. In the product operation stage, the two open interfaces are used to connect with different add-on modules. The operation configuration states and their probabilities in terms of percentages of time usage are summarised in Table 2.

The feasible design configurations of the platform and the specific add-on modules are shown in Figures 7 and 8, respectively. The design parameters associated with the design configurations

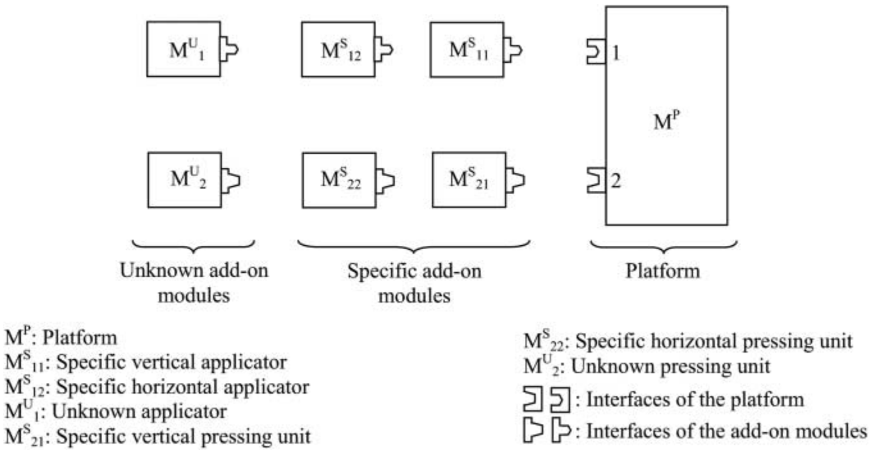

Figure 6. The platform and add-on modules.

Table 2. Different operation configuration states.

No.	Modules in an operation configuration state	Probability (time percentage)
1	$S_1 = \{M_{11}^S, M_{21}^S, M^P\}$	$P_1 = 20$
2	$S_2 = \{M_{12}^S, M_{22}^S, M^P\}$	$P_2 = 20$
3	$S_3 = \{M_{11}^S, M_2^U, M^P\}$	$P_3 = 10$
4	$S_4 = \{M_{12}^S, M_2^U, M^P\}$	$P_4 = 10$
5	$S_5 = \{M_1^U, M_{21}^S, M^P\}$	$P_5 = 10$
6	$S_6 = \{M_1^U, M_{22}^S, M^P\}$	$P_6 = 10$
7	$S_7 = \{M_1^U, M_2^U, M^P\}$	$P_7 = 20$

are summarised in Table 3. The non-design parameters associated with the different configurations are summarised in Table 4. The variations of parameters caused by uncertainties are summarised in Table 5.

The interactions between the platform and the unknown add-on modules are described in Figure 9. For the open architecture labelling machine, the input and output parameters of unknown add-on modules, M_1^U and M_2^U, are summarised in Table 6. The input parameters, N_1^U and N_2^U, are subjected to the following relations:

$$N_1^U = N_1^P = N_a \cdot R_a \cdot I_a, \tag{17}$$

$$N_2^U = N_2^P = N_p \cdot R_p \cdot I_p. \tag{18}$$

4.3. *Evaluation of robustness*

In this work, for operation configuration states 1, 3, and 5, the labelling machine is used as a vertical one and the target values of speeds for the applicator and pressing unit, V_a^T and V_p^T, were set to 20,000 mm/min. For operation configuration states 2, 4, and 6, the labelling machine is used as a horizontal one and the target values of speeds for the applicator and pressing unit, V_a^T and V_p^T, were set to 10,000 mm/min. For the operation configuration state 7, the unknown applicator and unknown pressing unit are connected with the platform. In this work, only horizontal labelling machine is considered under operation configuration state 7, and the target values of functional performance, V_a^T and V_p^T, were set to 10,000 mm/min.

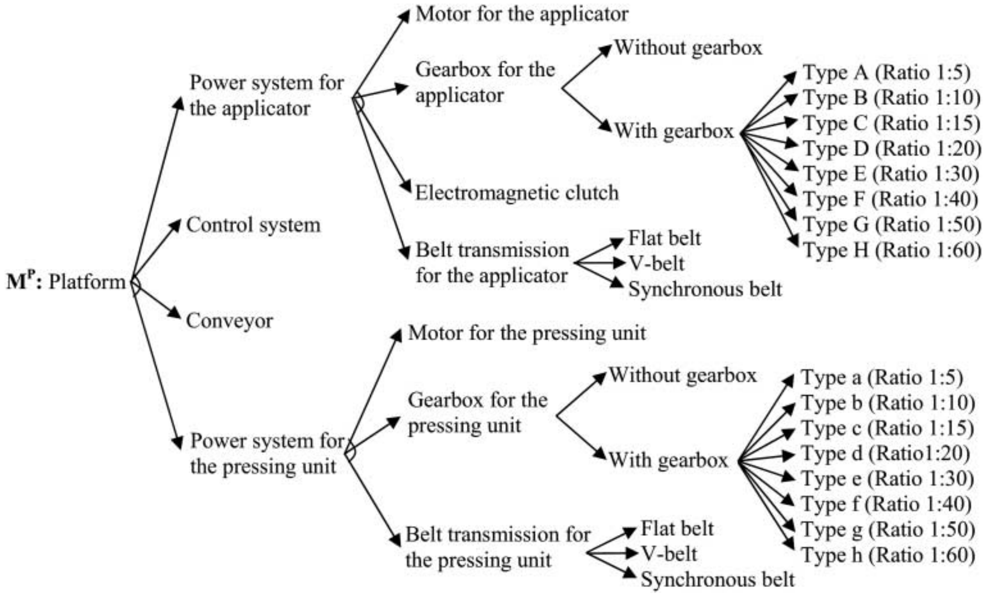

Figure 7. Modelling of different design configurations of the platform.

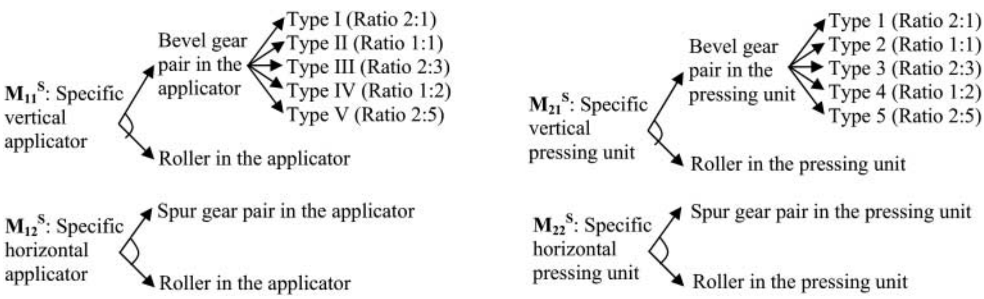

Figure 8. Modelling of design configurations of add-on modules.

Table 3. Design parameters.

Type of parameters	Name	Symbol	Boundary	Unit
Un-adaptable design parameters	Belt transmission ratio for the applicator	I_a	[0.1, 10]	–
	Belt transmission ratio for the pressing unit	I_p	[0.1, 10]	–
Adaptable design parameters	Rotation speed of the motor for the applicator	N_a	[0, 1500]	rpm
	Rotation speed of the motor for the pressing unit	N_p	[0, 1500]	rpm

(1) When the open interfaces are connected with specific add-on modules, the labelling speed V_a and the pressing speed V_p can be calculated by

$$V_a = \pi \cdot N_a \cdot R_a \cdot I_a \cdot I_1 \cdot D_1, \tag{19}$$

$$V_p = \pi \cdot N_p \cdot R_p \cdot I_p \cdot I_2 \cdot D_2, \tag{20}$$

Table 4. Non-design parameters.

Name	Symbol	Value	Unit
Transmission ratio of a gearbox for the specific applicator	R_a	TBD	–
Transmission ratio of a gearbox for the specific pressing unit	R_p	TBD	–
Transmission ratio of a bevel gear pair for the specific applicator	I_{11}	TBD	–
Transmission ratio of a bevel gear pair for the specific pressing unit	I_{21}	TBD	–
Transmission ratio of a spur gear pair for the specific applicator	I_{12}	1	–
Transmission ratio of a spur gear pair for the specific pressing unit	I_{22}	1	–
Diameter of the driving roller for the specific applicator	D_1	60	mm
Diameter of the driving roller for the specific pressing unit	D_2	60	mm

Note: TBD: To be determined depending on the selected configuration.

Table 5. Parameter variations caused by uncertainties.

Name	Symbol	Standard deviation	Unit
Variation of rotation speed of motor for the applicator	ΔN_a	0.5	rpm
Variation of rotation speed of motor for the pressing unit	ΔN_p	0.3	rpm
Variation of gearbox ratio for the applicator	ΔR_a	2×10^{-6}	–
Variation of gearbox ratio for the pressing unit	ΔR_p	2×10^{-6}	–
Variation of transmission ratio of flat belt for the applicator	$\Delta I_{a(1)}$	0.03	–
Variation of transmission ratio of V-belt for the applicator	$\Delta I_{a(2)}$	0.02	–
Variation of transmission ratio of synchronous belt for the applicator	$\Delta I_{a(3)}$	3×10^{-6}	–
Variation of transmission ratio of flat belt for the pressing unit	$\Delta I_{p(1)}$	0.02	–
Variation of transmission ratio of V-belt for the pressing unit	$\Delta I_{p(2)}$	0.01	–
Variation of transmission ratio of synchronous belt for the pressing unit	$\Delta I_{p(3)}$	2×10^{-3}	–
Variation of transmission ratio of a bevel gear pair for the applicator	ΔI_{11}	2×10^{-6}	–
Variation of transmission ratio of a bevel gear pair for the pressing unit	ΔI_{21}	2×10^{-6}	–
Variation of transmission ratio of a spur gear pair for the applicator	ΔI_{12}	2×10^{-6}	–
Variation of transmission ratio of a spur gear pair for the pressing unit	ΔI_{22}	2×10^{-6}	–
Variation of diameter of the driving roller for the applicator	ΔD_1	8×10^{-3}	mm
Variation of diameter of the driving roller for the pressing unit	ΔD_2	8×10^{-3}	mm

where I_1 is selected either as I_{11} or I_{12} and I_2 is selected either as I_{21} or I_{22}.

(2) When the open interfaces are connected with unknown add-on modules, the labelling speed V_a and the pressing speed V_p can be calculated by

$$V_a = \pi \cdot N_a \cdot R_a \cdot I_a \cdot I_1^U \cdot D_1^U, \tag{21}$$

$$V_p = \pi \cdot N_p \cdot R_p \cdot I_p \cdot I_2^U \cdot D_2^U. \tag{22}$$

By considering the importance factors of the labelling speed V_a and the pressing speed V_p, the weighting factors of the robustness measures for the labelling speed V_a and the pressing speed V_p at each operation configuration state were selected as 0.5 and 0.5, respectively. The overall robustness of the *i*-th design configuration candidate for the *s*-th operation configuration state considering the un-adaptable design parameters, I_a and I_p, and the interface parameters,

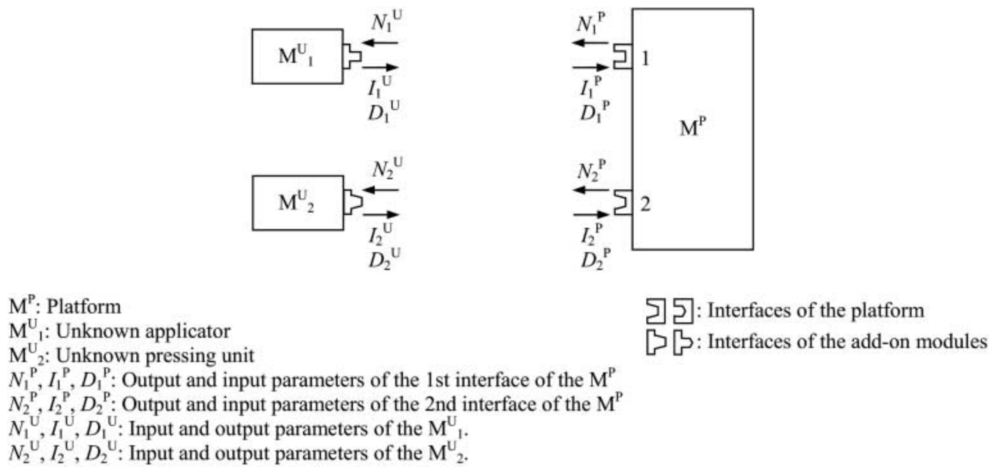

Figure 9. Interactions between platform and add-on modules under operation configuration 7.

Table 6. Input and output parameters of add-on modules M_1^{U} and M_2^{U}.

Add-on module	Input/output parameter	Symbol	Scope	Unit
M_1^{U}	Rotation speed of the pulley	N_1^{U}	[0, 1500]	rpm
	Diameter of the driving roller	D_1^{U}	[30, 150]	mm
	Gear ratio of the gear pair	I_1^{U}	[2, 1, 2/3, 0.5, 0.4]	–
M_2^{U}	Rotation speed of the pulley	N_2^{U}	[0, 1500]	rpm
	Diameter of the driving roller	D_2^{U}	[30, 150]	mm
	Gear ratio of the gear pair	I_2^{U}	[2, 1, 2/3, 0.5, 0.4]	–

$I_1^{\mathrm{U}}, D_1^{\mathrm{U}}, I_2^{\mathrm{U}}$ and D_2^{U}, can be calculated by

$$R_s^{(i)}(I_{\mathrm{a}}, I_{\mathrm{p}}, I_1^{\mathrm{U}}, D_1^{\mathrm{U}}, I_2^{\mathrm{U}}, D_2^{\mathrm{U}}) = 0.5 \times 10\ \log_{10}\left(\frac{\mu_{\mathrm{a}}^2}{\sigma_{\mathrm{a}}^2}\right) + 0.5 \times 10\ \log_{10}\left(\frac{\mu_{\mathrm{p}}^2}{\sigma_{\mathrm{p}}^2}\right)$$

$$= 5\ \log_{10}\left(\frac{\mu_{\mathrm{a}}^2}{\sigma_{\mathrm{a}}^2}\right) + 5\ \log_{10}\left(\frac{\mu_{\mathrm{p}}^2}{\sigma_{\mathrm{p}}^2}\right), \tag{23}$$

where μ_{a} and μ_{p} are the nominal values of V_{a} and V_{p}, respectively. In this work, $\mu_{\mathrm{a}} = V_{\mathrm{a}}^{\mathrm{T}}$, $\mu_{\mathrm{p}} = V_{\mathrm{p}}^{\mathrm{T}}$, and σ_{a} and σ_{p} are the standard deviations of V_{a} and V_{p}, respectively.

In this case study, the overall robustness of the *i*-th design configuration candidate, $R^{(i)}$, can be calculated using Equation (9).

4.4. *A multi-level optimisation model to identify the optimal design considering robustness*

In order to improve the robustness of the OAP, the overall performance of labelling machine under seven different operation configuration states should be insensitive to the parameter variations caused by uncertainties. The optimal product design configuration and its associated product/operating parameter values need to be identified to achieve the best overall robustness. Optimisation was employed to identify the best product design configuration candidate and its un-adaptable design parameter values.

In this case study, the configuration optimisation was formulated as

$$\begin{aligned}&\textit{Find}: \text{the } i\text{-th design configuration candidate}\\&\textit{Maximise}: R = R^{(i)}\\&\textit{Subject to}: 1 \leqslant i \leqslant 18,225\end{aligned} \tag{24}$$

where i represents the i-th design configuration candidate and 18,225 is the number of all feasible design configuration candidates.

Parameter optimisation to obtain the un-adaptable design parameters was formulated a:

$$\begin{aligned}&\textit{Find}: \text{the un-adaptable design parameter values } I_{\mathrm{a}}, \text{ and } I_{\mathrm{p}}\\&\textit{Maximise}: R^{(i)} = \sum_{s=1}^{n}[P_s \cdot R_s^{(i)}(I_{\mathrm{a}}, I_{\mathrm{p}})]\\&\textit{Subject to}: 0.1 \leqslant I_{\mathrm{a}} \leqslant 10;\ 0.1 \leqslant I_{\mathrm{p}} \leqslant 10\end{aligned} \tag{25}$$

In this work, both the statistical method and the worst-case method were used to calculate the robustness by considering the changes of input/output interface parameters of unknown add-on modules.

4.4.1. *The statistical method*

In the statistical method, the probabilities of changes of interface parameters are required. In this work, it was assumed that the values of interface parameters D_1^{U} and D_2^{U} follow even distribution between 30 and 150 mm for the s-th operation configuration state, that is, $P(D_1^{\mathrm{U}}) = P(D_2^{\mathrm{U}}) = 1/120$. It was also assumed that the values of interface parameter I_1^{U} and I_2^{U} are discrete ones defined by [2, 1, 2/3, 0.5, 0.4] and follow probabilities of $P(2) = P(0.4) = 15\%$, $P(1) = P(0.5) = 20\%$, and $P(2/3) = 30\%$ for the s-th operation configuration state. The robustness $R_{\mathrm{s}}^{(i)}(I_{\mathrm{a}}, I_{\mathrm{p}})$ can be calculated using Equations (12) and (13). For the design configuration candidate optimisation, the population size and the maximal generation were selected as 12 and 80, respectively. The predefined threshold crossover probability and mutation probability were selected as 0.65 and 0.7, respectively. The stopping criterion of the genetic programming was predefined as the maximum change of the average overall robustness of the last five generations be less than 1×10^{-6}.

In this case study, 26 generations of individuals were generated to achieve the optimal product configuration and its parameter values. The average overall robustness measures of these 26 generations are shown in Figure 10. The un-adaptable design parameter optimisation is also shown in Figure 10. The optimal configuration, the optimal parameter values for this configuration, and the overall robustness measure were obtained as shown in Table 7. In this work, the ratio of the belt transmission for the applicator, I_{a}, was optimised as 2.5 (i.e. the numbers of teeth of the two pulleys for belt transmission mechanism were selected as 25 and 10, and the diameters of the two pulleys were selected as 100 and 40 mm). The ratio of the belt transmission for the pressing unit, I_{p}, was optimised as 4 (i.e. the numbers of teeth of the two pulleys for the belt transmission mechanism were selected as 40 and 10, and the diameters of the two pulleys were selected as 160 and 40 mm).

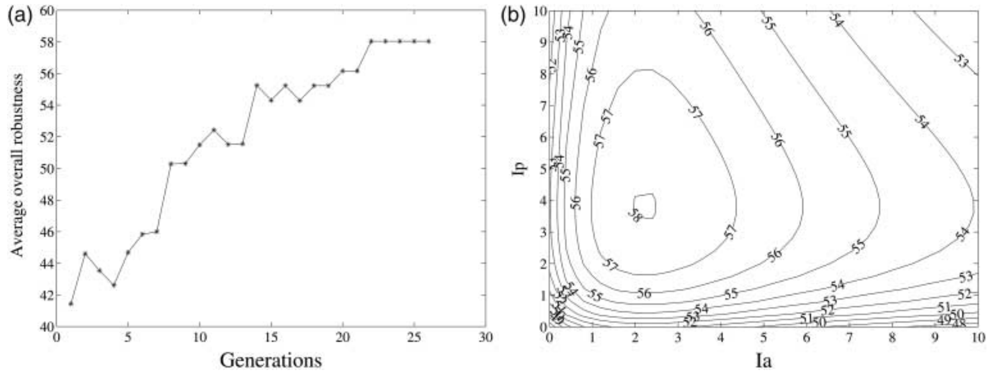

Figure 10. Optimisation results by using the statistical method. (a) Configuration optimisation based on generic programming. (b) Un-adaptable design parameter optimisation.

Table 7. The optimal design results based on the statistical method.

Item	The optimal configuration, parameters, and robustness
Configuration	Gearbox for the applicator: type F; belt transmission for the applicator: synchronous belt; gearbox for the pressing unit: type e; belt transmission for the pressing unit: synchronous belt; bevel gear pair in the applicator: type IV; bevel gear pair in the pressing unit: type 1
Parameters	Belt transmission ratio for the applicator, $I_a = 2.5$ (–)
	Belt transmission ratio for the pressing unit, $I_p = 4.0$ (–)
Robustness	The overall robustness, $R = 58.0$ (–)

4.4.2. *The worst-case method*

In the worst-case method, optimisation models for different operation configuration states can be established separately according to Table 2. In this case study, 47 generations of individuals were generated to achieve the optimal product configuration and its parameter values. The average overall robustness measures of these 47 generations are shown in Figure 11. The un-adaptable design parameter optimisation is also shown in Figure 11. The optimal configurations, the optimal parameter values associated with the configurations, and the overall robustness measures were obtained as shown in Table 8. In this work, the ratio of the belt transmission for the applicator, I_a, was optimised as 2 (i.e. the numbers of teeth of the pulleys for belt transmission were selected as 20 and 10, and the diameters of the pulleys for belt transmission were selected as 120 and 60 mm). The ratio of the belt transmission for the applicator, I_p, was optimised as 3.6 (i.e. the numbers of teeth of the pulleys for belt transmission were selected as 36 and 10, and the diameters of the pulleys for belt transmission were selected as 180 mm and 50 mm).

Compared with the design results using the worst-case method and the statistical method, we can see that the overall robustness with the worst-case method can maximise the minimal robustness considering all possible values of the input/output interface parameters. The average overall robustness obtained using the worst-case method, however, is lower than the overall robustness achieved through the statistical method.

4.5. *Comparative study*

In the traditional design of a reconfigurable or an adaptable product with closed architecture, only platform and specific add-on modules are considered. Since the different possible values of the

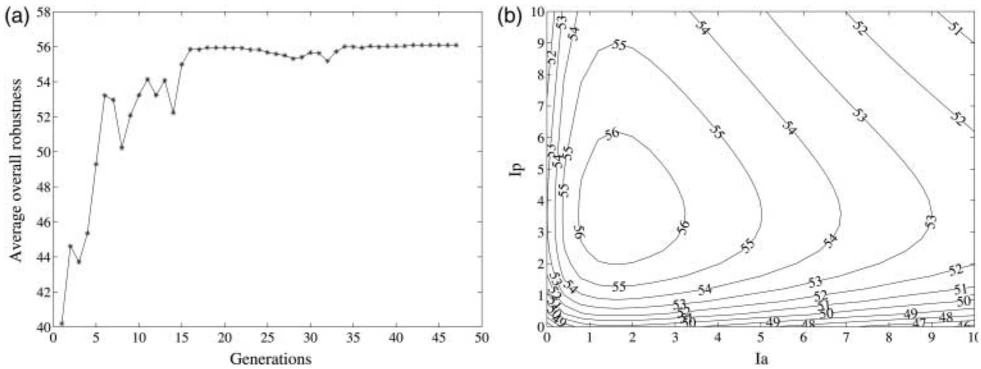

Figure 11. Optimisation results by using the worst-case method. (a) Configuration optimisation based on generic programming. (b) Un-adaptable design parameter optimisation.

Table 8. The optimal design for the open architecture labelling machine with the worst-case method.

Item	The optimal configuration, parameters, and robustness
Configuration	Gearbox for the applicator: type C; belt transmission for the applicator: synchronous belt; gearbox for the pressing unit: type g; belt transmission for the pressing unit: synchronous belt; bevel gear pair in the applicator: type II; bevel gear pair in the pressing unit: type 3
Parameter	Belt transmission ratio for the applicator, $I_a = 2.0$ (–) Belt transmission ratio for the pressing unit, $I_p = 3.6$ (–)
Robustness	The overall robustness, $R = 56.0$ (–)

Table 9. The optimal design results obtained using the traditional design method.

Item	The optimal configuration, parameters, and robustness
Configuration	Gearbox for the applicator: type H; belt transmission for the applicator: synchronous belt; gearbox for the pressing unit: type g; belt transmission for the pressing unit: synchronous belt; bevel gear pair in the applicator: type IV; bevel gear pair in the pressing unit: type 3
Parameter	Belt transmission ratio for the applicator, $I_a = 6.0$ (–) Belt transmission ratio for the pressing unit, $I_p = 5.4$ (–)
Robustness	$R = 63.3$ (–)

input/output interface parameters for unknown add-on modules are not considered at the product design stage, when new add-on modules are designed and added in the product operation stage, the original design created using the traditional design method is no longer optimal considering both the add-on modules created at the product design stage and the add-on modules created later on at the product operation stage.

Among all product operation configuration states shown in Table 2, only the operation configuration states 1 and 2 are not composed of any unknown add-on modules. In other words, the traditional design can be used to identify the optimal design with the best robustness considering the operation configuration states 1 and 2. The optimal design is shown in Table 9.

In this work, suppose that a new horizontal applicator add-on module and a new horizontal pressing unit add-on module need to be added and the labelling machine will be operated with

Table 10. Input and output interface parameters of new add-on modules M_{13} and M_{23}.

Add-on module	Input/output parameter	Symbol	Value	Unit
M_{13}	Rotation speed of the pulley	N_{13}	26.5	rpm
	Diameter of the driving roller	D_{13}	120	mm
	Gear ratio of the spur gear pair	I_{13}	1	–
M_{23}	Rotation speed of the pulley	N_{23}	26.5	rpm
	Diameter of the driving roller	D_{23}	120	mm
	Gear ratio of the spur gear pair	I_{23}	1	–

Table 11. Results in the comparative study.

Design method	Robustness of product for operation configuration states 1 and 2	Robustness of product for operation configuration states 1, 2 and N
The traditional design	$R_T = 63.3$ ($\sigma_T = 10.2$ mm/min)	$R_T = 58.2$ ($\sigma_T = 16.4$ mm/min)
The statistical method	$R_S = 59.0$ ($\sigma_S = 16.8$ mm/min)	$R_S = 59.4$ ($\sigma_S = 14.3$ mm/min)
The worst-case method	$R_W = 58.1$ ($\sigma_W = 18.6$ mm/min)	$R_W = 58.6$ ($\sigma_W = 15.6$ mm/min)

a new operation configuration state, the robustness of the design considering all three operation configuration states is no longer optimal. In this case study, this new operation configuration state is called operation configuration state N. Interface parameters of the two new add-on modules are shown in Table 10. In this table, the values of D_{13}, D_{23}, I_{13}, and I_{23} were selected based on design experience, and values of N_{13} and N_{23} were calculated from the parameters of the platform and add-on modules.

When the two new add-on modules were added, the overall robustness considering all three operation configuration states was lowered. In this work, the same probability values (i.e. 33.33%) were assigned to the three operation configuration states 1, 2, and N for the ease of explanation. The overall robustness is changed from 63.3 to 58.2 as shown in Table 11. To better demonstrate the physical meaning of robustness, the average values of the standard deviations of the labelling speed V_a and the pressing speed V_p for the different designs were also calculated and summarised in Table 11.

The newly developed statistical method and worst-case method were also employed in this comparative study. First, the statistical method and worst-case method were used to obtain the optimal designs as shown in Tables 7 and 8. Then the overall robustness measures considering only operation configuration states 1 and 2 were calculated as shown in Table 11. From this table, we can see that the robustness measures, 59.0 and 58.1, using the statistical method and the worst-case method are lower than the robustness obtained using the tradition method. When the new operation configuration state N with the two new add-on modules is considered, the robustness measures, 59.4 and 58.6, using the statistical method and the worst-case method are higher than the robustness obtained using the tradition design method.

5. Conclusions

Adaptable design is a new design paradigm that aims to create adaptable products. Among various adaptable products, the OAPs allow new add-on modules with new functions to be designed by third-party vendors and added to the existing products. In this research, an adaptable design method is introduced to design the OAP with robust performance. An OAP is modelled by a platform, add-on modules and interfaces to connect the platform with different add-on modules.

In this work, in addition to the specific add-on modules that need to be designed at the product development stage, the unknown add-on modules that could be added in the future are also considered. The optimal design with the best robustness is achieved through optimisation.

Characteristics of this research are summarised as follows:

(1) Open architecture is a good architecture to design adaptable products. The method to use a platform, add-on modules and open interfaces is effective to model OAPs.
(2) Robustness is a good measure to evaluate the quality of an OAP considering both functional performance measures and their variations.
(3) By considering both specific add-on modules that need to be designed at the product development stage and unknown add-on modules that could be added in the future, the overall robustness of OAP can be improved. In addition, the statistical method and the worst-case method are effective to identify the optimal design.
(4) The multi-level optimisation method is effective to identify the optimal design configuration and its parameter values of an open architecture adaptable product.

Despite the progress, a number of issues need to be further addressed to improve the currently developed method. In this research, only aleatory uncertainties in parameters (i.e. parameter variations caused by uncertainties) are considered. New design methods considering other types of uncertainties such as epistemic uncertainties modelled by fuzzy membership functions need to be developed. In addition, only embodiment design stage is considered in the newly developed method. Since product robustness can also be influenced by selection of design concepts, an integrated approach to design OAPs considering both conceptual design and embodiment design stages needs to be developed.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

The authors wish to thank the Leading Talent Project of Guangdong Province, China, the Natural Sciences and Engineering Research Council (NSERC) of Canada, and the National Natural Science Foundation of China [grant number 51375287] for providing financial supports to this research.

References

Arora, J. S. 1989. *Introduction to Optimum Design*. New York: McGraw-Hill.

Cheng, Q., G. Zhang, Z. Liu, P. Gu, and L. Cai. 2011. "A Structure-based Approach to Evaluation Product Adaptability in Adaptable Design." *Journal of Mechanical Science and Technology* 25 (5): 1081–1094. doi:10.1007/s12206-011-0224-3.

Du, X., and W. Chen. 2000. "Towards a Better Understanding of Modeling Feasibility Robustness in Engineering Design." *Journal of Mechanical Design, Transactions of the ASME* 122 (4): 385–394. doi:10.1115/1.1290247.

Fletcher, D., R. W. Brennan, and P. Gu. 2009. "A Method for Quantifying Adaptability in Engineering Design." *Concurrent Engineering: Research and Applications* 17 (4): 279–289. doi:10.1177/1063293X09352123.

Fonseca, J. R., M. I. Friswell, and A. W. Lees. 2007. "Efficient Robust Design via Monte Carlo Sample Reweighting." *International Journal for Numerical Methods in Engineering* 69: 2279–2301. doi:10.1002/nme.1850.

Gu, P., M. Hashemina, and A. Y. C. Nee. 2004. "Adaptable Design." *CIRP Annals – Manufacturing Technology* 53 (2): 539–557. doi:10.1016/S0007-8506(07)60028-6.

Gu, P., D. Xue, and A. Y. C. Nee. 2009. "Adaptable Design: Concepts, Methods and Applications." *Proceedings of the Institution of Mechanical Engineers, Part B, Journal of Engineering Manufacture* 223 (11): 1367–1387. doi:10.1243/09544054JEM1387

Hong, G., L. Hu, D. Xue, Y. L. Tu, and Y. L. Xiong. 2008. "Identification of the Optimal Product Configuration and Parameters Based on Individual Customer Requirements on Performance and Costs in One-of-a-kind Production." *International Journal of Production Research* 46 (12): 3297–3326. doi:10.1080/00207540601099274.

Hu, W., S. Azarm, and A. Almansoori. 2013. "New Approximation Assisted Multi-objective Collaborative Robust Optimization (New AA-McRO) under Interval Uncertainty." *Structural and Multidisciplinary Optimization* 47 (1): 19–35. doi:10.1007/s00158-012-0816-2.

Huang, Y. 2007. *Structure and Design of Packaging Machine*. Beijing: Chemical Industry Press.

Inkermann, D., C. Stechert, and T. Vietor. 2013. "A Framework for the Application of Adaptronic Solution Principles." Proceedings of the ASME 2013 international design engineering technical conferences and computers and information in engineering conference, Portland, OR, August 4–7.

Koren, Y., S. J. Hu, P. Gu, and M. Shpitalni. 2013. "Open-architecture Products." *CIRP Annals – Manufacturing Technology* 62 (2): 719–729. doi:10.1016/j.cirp.2013.06.001.

Koza, J. R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA: The MIT Press.

Kumar, A., P. Nair, A. Keane, and S. Shahpar. 2008. "Robust Design Using Bayesian Monte Carlo." *International Journal for Numerical Method in Engineering* 73 (11): 1497–1517. doi:10.1002/nme.2126.

Li, Y., D. Xue, and P. Gu. 2008. "Design for Product Adaptability." *Concurrent Engineering: Research and Applications* 16 (3): 221–232. doi:10.1177/1063293X08096178.

de Neufville, R., and S. Scholtes. 2011. *Flexibility in Engineering Design*. Cambridge: MIT Press.

Olewnik, A., and K. Lewis. 2006. "A Decision Support Framework for Flexible System Design." *Journal of Engineering Design* 17 (1): 75–97. doi:10.1080/09544820500274019.

Parkinson, A., C. Sorensen, and N. Pourhassan. 1993. "A General Approach for Robust Optimal Design." *Journal of Mechanical Design, Transactions of the ASME* 115 (1): 74–80. doi:10.1115/1.2919328.

Peng, Q., Y. Liu, P. Gu, and Z. Fan. 2013. "Development of an Open-architecture Electric Vehicle Using Adaptable Design." In *Advances in Sustainable and Competitive Manufacturing Systems*, edited by A. Azevedo, 79–90. Cham: Springer.

Samadiani, E., Y. Joshi, J. K. Allen, and F. Mistree. 2009. "Adaptable Robust Design of Multi-scale Convective Systems Applied to Energy Efficient Data Centers." *Numerical Heat Transfer, Part A, Application: An International Journal of Computation and Methodology* 57 (2): 69–100. doi:10.1080/10407780903582901.

Suh, E. S., O. L. de Weck, and D. Chang. 2007. "Flexible Product Platforms: Framework and Case Study." *Research in Engineering Design* 18 (2): 67–89. doi:10.1007/s00163-007-0032-z.

Suh, N. P. 1999. "A Theory of Complexity, Periodicity and the Design Axioms." *Research in Engineering Design* 11 (2): 116–132. doi:10.1007/PL00003883.

Suh, N. P. 2001. *Axiomatic Design: Advances and Applications*. New York: Oxford University Press.

Taguchi, G. 1978. "Performance Analysis Design." *International Journal of Production Research* 16 (6): 521–530. doi:10.1080/00207547808930043.

Taguchi, G. 1993. *Taguchi on Robust Technology Development: Bringing Quality Engineering Upstream*. New York: ASME Press.

Xiao, S. 2009. *Modern Logistics Equipment*. Beijing: Science Press.

Xue, D., G. Hua, V. Mehrad, and P. Gu. 2012. "Optimal Adaptable Design for Creating the Changeable Product Based on Changeable Requirements Considering the Whole Product Life-cycle." *Journal of Manufacturing Systems* 31 (1): 59–68. doi:10.1016/j.jmsy.2011.04.003.

Zhang, J., Y. Chen, D. Xue, and P. Gu. 2014. "Robust Design of Configurations and Parameters of Adaptable Products." *Frontiers of Mechanical Engineering* 9 (1): 1–14. doi:10.1007/s11465-014-0296-8.

Zhang, J., S. Li, N. Bao, G. Zhang, D. Xue, and P. Gu. 2010. "A Robust Design Approach to Determination of Tolerance of Mechanical Products." *CIRP Annals – Manufacturing Technology* 59 (1): 195–198. doi:10.1016/j.cirp.2010.03.099.

Zhang, J., D. Xue, and P. Gu. 2012. "Robust Adaptable Design Considering Changes of Parameter Values in Product Operation Stage." Proceedings of the 22nd CIRP design conference, Bangalore, March 28–30.