



Identification of Influential Modules Considering Design Change Impacts Based on Parallel Breadth-First Search and Bat Algorithm

Xianfu Cheng*, Zhihu Guo, Xiaotian Ma and Tian Yuan

Key Laboratory of Conveyance and Equipment, East China Jiaotong University, Nanchang, China

Modular design is a widely used strategy that meets diverse customer requirements. Close relationships exist between parts inside a module and loose linkages between modules in the modular products. A change of one part or module may cause changes of other parts or modules, which in turn propagate through a product. This paper aims to present an approach to analyze the associations and change impacts between modules and identify influential modules in modular product design. The proposed framework explores all possible change propagation paths (CPPs), and measures change impact degrees between modules. In this article, a design structure matrix (DSM) is used to express dependence relationships between parts, and change propagation trees of affected parts within module are constructed. The influence of the affected part in the corresponding module is also analyzed, and a reachable matrix is employed to determine reachable parts of change propagation. The parallel breadth-first algorithm is used to search propagation paths. The influential modules are identified according to their comprehensive change impact degrees that are computed by the bat algorithm. Finally, a case study on the grab illustrates the impacts of design change in modular products.

OPEN ACCESS

Edited by:

Tinggui Chen,
Zhejiang Gongshang University, China

Reviewed by:

Yassine Meraihi,
University of Boumerdés, Algeria
Xiangtao Li,
Jilin University, China

*Correspondence:

Xianfu Cheng
1149@ecjtu.edu.cn

Specialty section:

This article was submitted to
"Bionics and Biomimetics",
a section of the journal
Frontiers in Bioengineering and
Biotechnology

Received: 08 October 2021

Accepted: 01 December 2021

Published: 05 January 2022

Citation:

Cheng X, Guo Z, Ma X and Yuan T
(2022) Identification of Influential
Modules Considering Design Change
Impacts Based on Parallel Breadth-
First Search and Bat Algorithm.
Front. Bioeng. Biotechnol. 9:791566.
doi: 10.3389/fbioe.2021.791566

Keywords: modular product, engineering change, change impact degree, influential module, parallel breadth-first algorithm

INTRODUCTION

The product modularization has a significant influence on the product development process and on the whole product life cycle (Bonvoisin et al., 2016). The modular design has become a widely used and researched product and system development strategy that meets diverse customer requirements. A module is a relatively independent chunk of a system loosely coupled to the rest of the system (Höltk-Otto et al., 2012). There are many methods for designing modular products that can be classified into two main groups: function-based and matrix-based methods (Eppinger and Browning, 2012; Kashkoush and ElMaraghy, 2017). Modular design on the basis of Design Structure Matrix (DSM) has been widely studied in the literature (AlGeddawy and ElMaraghy, 2013; Borjesson and Höltk-Otto, 2014; Cheng et al., 2018). In modular product architecture, independent complete modules are rare within a given product architecture. There often exists association relationships between modules. Due to various factors, such as changing requirements, improving the environment, technology infusion, and product upgrade, the product architecture may be modified to meet new customer requirements. A change to one module of a product may trigger a series of changes in other modules. To reduce the cost and negative impact of design

change, the designers should try to control the change of modules in a product, especially influential modules. It is crucial for designers to decrease the influence of design change by reducing the change of influential modules in product design. Further, the identification of influential modules can help prevent the products failure and distribute the development resources. Whereby, the relative importance or influence of modules is measured in terms of design change impact.

Engineering change is unavoidable during the product evolution process. Manufacturing enterprises sometimes utilize engineering change to deal with technology obsolescence or improve quality with shorter turnarounds. Design change in modular products considers not only the change of module, but also the change of characteristic parameter or assembly constraints of the components in modules. So, the problem of module change can be classified into two types: change of a part propagating to other parts within the same module, and change of a part propagating to other parts in different modules (Lee et al., 2010). Some tools and methods for assessing or predicting the influence of engineering changes on products have been proposed in extant literature (Ollinger and Stahovich, 2004; Yang and Duan, 2012; Ahmad et al., 2013; Hamraz and Clarkson, 2015). However, they tend to be confined to an integrated product architecture or a single product instance (Ollinger and Stahovich, 2004; Lee et al., 2010; Yang and Duan, 2012; Ahmad et al., 2013; Hamraz and Clarkson, 2015; Hein et al., 2021). This paper aims at presenting an approach to analyzing the associations and change impacts between modules in modular products based on design structure matrix and change propagation network. According to the relationships between parts within a module and their change propagation characteristics, the propagation trees of change impacts are constructed, and all possible propagation paths of module change are captured. To effectively manage design change and prevent product failure in modular products, the impacts of design change between modules are analyzed, and the influential modules are identified from the perspective of engineering change. For the sake of simplicity, we will use the term change impact degree to refer to the relative impacts for a part or module being changed on another part or module.

The remainder of the paper is structured as follows: the following section reviews the related research; *Analysis on Change Impact Degree Between Modules* discusses the association and change impact degrees between modules; *Change Propagation Analysis Within Affected Module* identifies reachable parts affected by change propagation and analyzes the change impact degree of the affected part within module; the change influence of affected parts within module is analyzed, and influential modules are identified in *Identification of Influential Modules Considering Design Change Impacts*; next, *Case Study* discusses a case study to illustrate the proposed approach; finally, the conclusion and discussion are remarked in *Conclusion*.

RELATED LITERATURE

Engineering change is sometimes an effective evolution mode for enhancing product quality. It enables the product to fulfill

affluent market segments, but at the same time, increase product design complexity. If properly managed, changes can provide opportunities for improvement to the product and increase its consumer value (Hein et al., 2021). A change may encompass any modification of the product as a whole or in part, and may alter the interactions and dependencies of the constituent elements of the product (Jarratt et al., 2011). Jarratt et al. (2011) summarized a categorized overview on engineering change and analyzed the nature of the engineering change process. A comprehensive review of engineering change is not repeated here. This section aims to discuss some relevant work from the association representation based on DSM and change propagation between modules and parts in the modules.

Clarkson et al. (2004) and Eckert et al. (2004) used likelihood and impact of change to predict the risk of change propagation, traced potential propagation paths among parts based on DSM, and outlined the methods to compute the risk of direct and indirect change propagation. Giffin et al. (2009) proposed a network-based analysis method using a combination of graph theory and DSM. They developed a set of indices to quantify the relative strength of each area in terms of its propensity. Koh et al. (2012) presented a modeling method matrix-based to assess the effects of engineering change propagation, which set up the house of quality and the change prediction method to model the performance of different change options. Pasqual and deWeck (2012) presented a multilayer network model of change propagation comprising product layer, change layer, and social layer. They developed the repository that includes a few novel tools and metrics for the analysis and management of change propagation. Mehta et al. (2013) proposed an approach to determine important attributes of engineering change, which employed the observed distribution and the domain knowledge to evaluate the importance of a feature set through retrieving similar past engineering changes. Maier et al. (2014) developed a discrete-event simulation model to evaluate the effects of different priority policies, which was based on a product architecture DSM and considered the combined effects of progressive iteration, rework and change propagation. Xie and Ma (2016) presented a graph-based association model to represent the network of constraints, and developed a dynamic inter-feature association map to analyze engineering change propagation. However, this previous research focused heavily on change propagation prediction and its management in an integrated or a single product.

A method based on matrix technique using engineering change forecast to prioritize product parts for modularization was introduced by (Koh et al., 2015), which can be applied to support decision makers in their modularization efforts from a change perspective. Lee et al. (2010) employed the analytic network process to measure design change impacts in modular products, and identified the final priorities of parts with their relative change impacts on the whole product. Ullah et al. (Hein et al., 2021) analyzed effective change propagation quantitative risk-based in a product family design, and adopted a seven-step mechanism comprising of a mathematical model and an algorithm. This approach takes into account direct and indirect change propagation, but does not analyze change

impact degree between modules. Cheng et al. (2018b) discussed the coupling between modules in product family design, and presented a method for coupling analysis between platform/customization modules and decoupling strategies. In this method, the coupling between platform modules, between platform module and customization module, as well as between customization modules is depicted, respectively. Cheng et al. (2020) analyzed coupling association problems in design process of modular product design, calculated association dependence degree between modules, as well as addressed corresponding decoupling strategies for coupling modules. Unfortunately, these two proposed methods only consider direct influences of change propagation between two modules, and do not analyze all possible CPPs.

Identifying influential nodes in complex networks has attracted increasing attention in recent years. It is well known that many mechanisms such as spreading, cascading, and synchronizing are highly affected by a tiny fraction of key nodes in complex networks (Liu et al., 2016). Identifying the most efficient “spreaders” in complex network is a crucial step to optimize the use of available resources and ensure the more efficient spread of information (Kitsak et al., 2010). Gao et al. (2013) proposed a method of identifying influential nodes by semi-local centrality combined with modified evidential centrality, which considered the degree distribution to build the basic probability assignment of each node. Zhang et al. (2013) utilized the information transfer probability between any pair of nodes and the k-medoid clustering algorithm to identify influential nodes. Bae and Kim (2014) proposed a ranking method to estimate the spreading influence of a node in complex network using coreness centrality. Liu et al. (2016) proposed a method to evaluate the importance of nodes in complex networks based on degree and the importance of lines, which only needs the local information of nodes. Ahmad et al. (2013) presented an improved cluster rank approach to find influential nodes, which took into account common hierarchy of nodes and their neighborhood set. All of these above focused on influential nodes in a network, and did not discuss the importance of modules as well as the impact of node’s change.

In addition, a few papers available today address influential modules. Specifically, Lee et al. (2010) measured the relative importance of module with design change impacts in modular products through pairwise comparisons. The presented method simplifies tedious works for identifying indirect influences. The main limitation is that it cannot measure the impact of changes in a part or a module on a single part or module. Li et al. (2021) presented a method for function module partition of complex products and systems through community detection using weighted and directed complex networks. Li et al. (2019) identified the influential function modules based on weighted LeaderRank algorithm, and used susceptible-infected-recovered (SIR) model to assess the influence degrees of the identified function modules. However, they thought only of the number of relation types between the modules as the relationship between the modules, and did not consider design change impact. Influential modules could be identified and regarded as design preferences. Above preferences are provided for the designers to

C	1	2	3	4	5	6	7	8
1	-	0.2			←	M_1		
2	0.7	-	0.3			0.2		
3	0.4	0.5	-	0.2				
4	0.3	0.1	0.6	-				
5			0.2		-		0.3	
6					0.6	-	0.1	
7					0.5	0.4	-	0.3
8		M_2		→		0.4	0.6	-

FIGURE 1 | Design dependency matrix with two modules.

manage design change in modular product design. So, in view of the above, this article proposes a quantitative analysis approach that can be employed to assess the importance of modules considering change propagation across different modules.

ANALYSIS ON CHANGE IMPACT DEGREE BETWEEN MODULES

DSM is a popular technique based on a square matrix with identical row and column labels to represent and analyze connections among elements within a system, process, task, or product in a compact, visual, and analytically advantageous format (Steward, 1981; Browning, 2001). DSM has been widely used to support not only modularization for a product or product family but also engineering change (Clarkson et al., 2004; Samling and de Weck, 2007; Hong and Park, 2014; Cheng et al., 2018b).

According to space, material, energy, and information link between parts, the initial DSM of the product are built. The clustering algorithm is introduced to group these parts into modules and to identify interactions between modules. Detailed discussion on DSM to facilitate module identification is not within the scope of this article. The interactions between parts in the DSM can be represented in terms of risk (Clarkson et al., 2004), difficulty (Hoelttae and Otto, 2005), probability of change (Sharman and Yassine, 2017) or dependence (Jung et al., 2018), which would result in rework. Al Handawi (Alhandawi et al., 2020) defined three more changeability aspects based on the nature of change effects, namely, robustness, scalability and modifiability. From the perspective of design change, the change considers heavily the impact of the modification to one part on other parts. This paper aims to analyze relative change impact degree between modules. So, without loss of generality the probability of change propagation is utilized in this work. The probability of change propagation refers to the probability of a redesign of the dependent part being necessary given that a change has occurred in the feeding part (Sharman and Yassine, 2017).

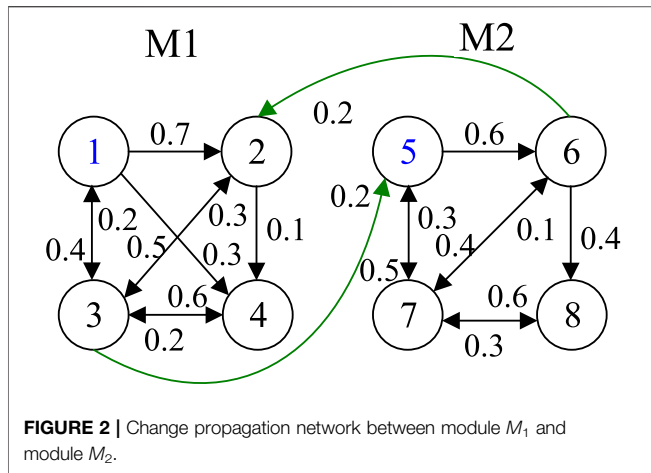


FIGURE 2 | Change propagation network between module M_1 and module M_2 .

Assume a product consists of a total of n parts and is divided into N modules, where C_i ($i = 1, 2, \dots, n$) is the i th part. These parts are arranged into DSM in the form of modules. Then the influence of design change of one part on another directly connected part, namely the probability of change propagation, is determined through analyzing relevant dependences between them. The new matrix with the probability of change propagation is called design dependency matrix (DDM) of modules.

For simplicity of explanation on design change impacts between modules, a product comprising of two modules M_1 and M_2 , are taken as an example to describe and analyze their relationship and change propagation, as shown in **Figure 1**. Here, the row and column headings represent parts. The values of off-diagonal elements denote direct dependence between parts, and the direction of change propagation is from the element in the corresponding column to the element in the corresponding row. For instance, the matrix cell “0.5” in the third row and second column, represents the probability of a redesign of part C_3 being necessary given that a change has occurred in part C_2 . The modules in DDM interact through bottleneck interactions. An interaction in DDM is considered a bottleneck when it does not allow the decomposition of DSM into mutually separable groups, and therefore appears outside the clusters. Each bottleneck in DDM represents an interaction between modules. There are two interfaces between module M_1 and module M_2 . Module M_1 influences part C_5 of module M_2 through part C_3 , and part C_6 of module M_2 influences part C_2 of module M_1 specifically. The modification of module M_1 will lead to change of module M_2 and vice versa.

Since change propagation network can capture all of the direct and indirect impacts among elements, the phenomenon of change propagation in DDM can be effectively mirrored in the network (Lee et al., 2010). The change propagation network between module M_1 and module M_2 corresponding to **Figure 1** is shown as **Figure 2**. Here, the circle represents the node that indicates the product part, the number inside the circle denotes its serial number, an arrow indicates change direction from the part the arrow leaves to that the arrow enters, and the value near the line with an arrow is the corresponding change probability propagated from predecessor to successor.

It can be seen in **Figure 2** that the change impact of one module on another module is related not only to the interfaces

between modules but also to the association among parts within downstream (affected) module. Assume there is an interaction between module M_p and module M_q , where part C_i in module M_p has effect on part C_j in module M_q , C_i and C_j are respectively called instigating part and affected part of change propagation from M_p to M_q . Since M_q is a “coupling subsystem,” C_i has influence not only on C_j but also on other parts in M_q . C_j may be regarded as a feeding part of change propagation within M_q . It can directly and indirectly affect other parts in M_q . Therefore, there is a need to analyze direct and indirect change impacts of C_i on M_q and to capture its change propagation paths. Simultaneously the change impact degree of C_j on other parts within M_q should also be computed.

Assume M_q consists of m parts, where m_q and n_q are the index of the first part and the last part of M_q in DDM, respectively. The change impact degree propagated from C_j to the k th part C_k in M_q is defined as $P(C_j, C_k)$ ($k = m_q, m_q+1, \dots, n_q, k \neq j$), then the change impact degree of C_i on M_q , denoted as $P(C_i, M_q)$, can be defined as follows.

$$P(C_i, M_q) = r_{j,i} \cdot \left[1 + \sum_{k=m_q, k \neq j}^{n_q} P(C_j, C_k) \right] \quad (1)$$

where, $r_{j,i}$ is the probability of change propagation (direct change impact) of C_i in M_p on C_j in M_q , and $r_{j,i} \cdot \sum_{k=m_q, k \neq j}^{n_q} P(C_j, C_k)$ is

indirect change impacts of C_i on other parts in M_q .

If C_i affects directly more than one part in M_q , for example, C_i affects both parts C_u and C_v in M_q , then $P(C_i, M_q)$ should be the sum of their change impact degree, expressed as in Eq. (2).

$$P(C_i, M_q) = r_{u,i} \cdot \left[1 + \sum_{\substack{k=m_q \\ k \neq u}}^{n_q} P(C_u, C_k) \right] + r_{v,i} \cdot \left[1 + \sum_{\substack{w=m_q \\ w \neq v}}^{n_q} P(C_v, C_w) \right] \quad (2)$$

where, $r_{u,i}$ and $r_{v,i}$ are respectively the probability of change propagation of C_i on C_u and C_v , $P(C_j, C_k)$ and $P(C_j, C_k)$ are respectively change impact degree of C_u and C_v on C_k in M_q .

The change impact degree propagated from upstream module to downstream module can be measured by computing the total change impact degree of all instigating parts. If M_p has t instigating parts that have influence on M_q , then the total change impact degree, $P(C_i, M_q)$ propagated from M_p to M_q , can be represented as

$$P(M_p, M_q) = \sum_{i=1}^t P(C_i, M_q) \quad (3)$$

CHANGE PROPAGATION ANALYSIS WITHIN AFFECTED MODULE

The change between modules is related to the association among parts within an affected module and the influence of the affected

A	C_1	C_2	C_3	C_4	C_5	C_6	C_7
C_1	-			0.3			
C_2	0.8	-		0.4	0.5		
C_3			-	0.5	0.7		
C_4	0.6		0.4	-		0.2	
C_5					-	0.3	
C_6						-	
C_7			0.5				-

Design dependency matrix

B	C_1	C_2	C_3	C_4	C_5	C_6	C_7
C_1	0	0	0	1	0	0	0
C_2	1	0	0	1	0	1	0
C_3	0	0	0	1	1	0	0
C_4	1	0	1	0	0	1	0
C_5	0	0	0	0	0	1	0
C_6	0	0	0	0	0	0	0
C_7	0	0	1	0	0	0	0

Boolean DSM

C	C_1	C_2	C_3	C_4	C_5	C_6	C_7
C_1	1	0	1	1	1	1	0
C_2	1	1	1	1	1	1	0
C_3	1	0	1	1	1	1	0
C_4	1	0	1	1	1	1	0
C_5	0	0	0	0	1	1	0
C_6	0	0	0	0	0	1	0
C_7	1	0	1	1	1	1	1

Reachable matrix

D	C_1	C_2	C_3	C_4	C_7
C_1	1	0	1	1	0
C_2	1	1	1	1	0
C_3	1	0	1	1	0
C_4	1	0	1	1	0
C_7	1	0	1	1	1

Reduction matrix of reachable matrix

FIGURE 3 | Identifying reachable parts of change propagation using reachable matrix.

of change propagation and the more the change propagation paths.

Identify Reachable Parts for Change Propagation

Before analyzing CPPs, it is necessary to identify which parts are influenced directly or indirectly by the affected part within the corresponding module in which it is located. DDM only represents direct dependence relationships between parts, and does not directly reflect reachable parts of change propagation of affected part. In this paper, reachable matrix is employed to achieve this task. According to DDM with clustered modules, it is possible to determine whether there are dependencies between any two modules. The modules in which affected parts are located are extracted as sub DDM. Then all of the non 0 value in the cells are modified to the value "1". Its reachable matrix is computed, and the parts influenced directly and indirectly by affected part are judged. Here the affected part is also regarded as a feeding part of change propagation in the corresponding module. These influenced parts are reachable parts of change propagation. In reachable matrix, if there is one or more cell "0" in the column in which feeding part is located, it means that the part corresponding to this element does not rely on feeding part, and then the row and column in which this part is located can be deleted. Now, the matrix after dimension reduction is called the reduction matrix of reachable matrix.

Let us consider a DDM of a module comprising seven parts, as shown in **Figure 3A**. Assume part C_4 is an affected part propagated from another module. Here, C_4 can be taken as a feeding part of change propagation. Then initial design dependency matrix should be converted to Boolean DSM and reachable matrix, as shown in **Figures 3B,C**.

It is evident from **Figures 3A,B,C**, that C_4 passes the change onto the directly connected parts C_1 , C_2 , and C_3 (they may be also indirectly influenced), and indirectly influences part C_7 through intermediate parts. Hence, C_1 , C_2 , C_3 , and C_7 are all reachable parts of change propagation from C_4 . In addition, the alteration of C_4 does not propagate to parts C_5 and C_6 . Here, reachable matrix can be reduced, as shown in **Figure 3D**. So, during analyzing all possible paths of change propagation of part C_4 , parts C_5 and C_6 can be ignored.

Search Change Propagation Paths Based on Parallel Breadth-First Algorithm

This section represents an intelligent parallel breadth-first algorithm to search change propagation paths based on the design dependency matrix and directed graph. Its basic idea is: instigating part of the module is regarded as start vertex V_0 of directed graph, the adjacent vertex (parts) directly associated with V_0 are accessed; then they will be taken as new vertices in turn, and the vertices that has not been accessed and is directly related to them are continuously accessed. Repeat the process until access traverses all the vertices.

The change propagation network can be represented as a directed graph, $G = \langle V, E \rangle$, wherein the elements of $V = (v_1,$

part in the corresponding module in which it is located. The larger the size of the module and the stronger the influence of the affected part in the corresponding module, the greater the scope

v_2, \dots, v_n) are the nodes (i.e. vertices) and elements of $E = (e_1, e_2, \dots, e_m)$ are the edges, which are used to link the nodes. Instigating part C_j and reachable part C_k are respectively considered as start vertex V_0 and target vertex V_{target} . The intelligent parallel breadth-first algorithm is utilized to search all propagation paths from V_0 to V_{target} . At the same time, the change propagation probability between parts is recorded. Then a breadth-first tree can be obtained by intelligent parallel breadth-first algorithm, where V_0 and V_{target} are respectively the root and top of the tree. This algorithm use the existing vertices and the edges between vertices in the process of searching paths from beginning to end, namely, the vertices closed to V_0 are firstly searched and then the vertices far away from V_0 are searched (Buluc and Madduri, 2011; Beamer et al., 2012; Garcia et al., 2020; Yigit et al., 2021). In search process, the vertices in current task queue form the boundary of accessed vertices, which is called active vertex set.

Assume A represents the adjacent matrix corresponding to directed graph G . If a_{ij} in A equals to 0, it means that there is no link from vertex i to vertex j . If a_{ij} equals to 1, it means that there is a link from vertex i to vertex j . We utilize k processors to parallel search paths. That is, matrix A is divided into k blocks, and every processor $K(p, q)$ ($1 \leq p \leq m/k, 1 \leq q \leq m/k$) deals with a corresponding sub-matrix A_{pq} .

The search process of change propagation path from V_0 to V_{target} based on intelligent parallel breadth-first algorithm is illustrated as follows.

Step 1: both the current boundary B_{ij} and the next boundary N_{ij} of breadth-first search algorithm are set to *null*, and the number of the lever of all the vertices is set to 0;

Step 2: all source nodes in graph G are input within the boundary corresponding to the processor where source nodes are located, which means that this vertex is reachable.

Step 3: each processor processes the assigned boundary in parallel. If the number of the lever in which a vertex of B_{ij} is located is 0, then it will be put into the temporary set H_{pq} . Meantime, the number of the lever of this vertex adds 1 (initial value is set to 1). Then, the nodes of next boundary will be calculated and saved to N_{ij} .

Step 4: After obtaining the set of the nodes of the lower boundary, they will be deemed as current boundary of the processor, and the number of the lever adds 1.

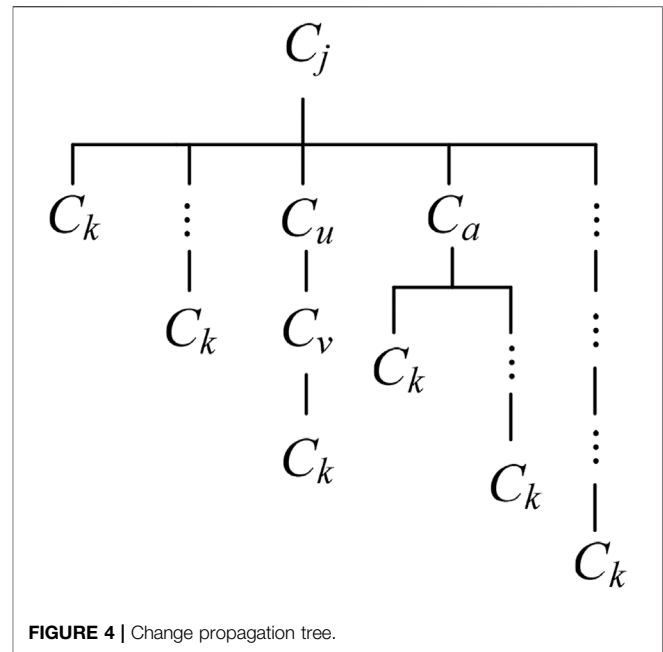
Step 5: if the number of the nodes in current boundary that need to be computed in the next time does not equal to 0, then go to step 3 and step 4. Otherwise, go to step 6.

Step 6: incorporate the lever number of all vertices in the processor, and get the information of the lever number of directed graph.

Step 7: get the information of all reachable paths of the vertices according to the layer number of the vertices.

Change Impact Degree Analysis for Feeding Part Within the Module

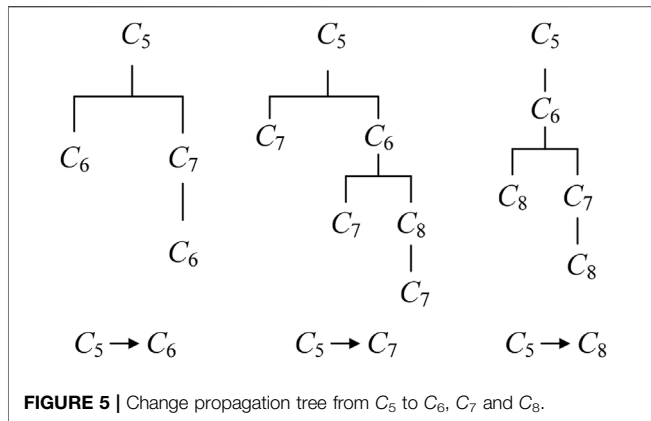
Change propagation patterns between parts have two types: 1) one part passes the change onto the directly connected parts, for



instance, from C_2 to C_3 and C_4 as shown in **Figure 2**; and 2) change propagation from one part to another part through one or more intermediate parts, for instance, from C_2 to C_4 through intermediate part C_3 , from C_2 to C_1 through intermediate parts C_4 and C_3 , both of them are indirect change propagation. There is not only direct change impact between some parts, but also indirect change impact. For some parts, there may be only indirect change impact. For example, there is no direct change impact of C_5 on C_8 , since the change of C_5 propagates to C_8 through parts C_6 and C_7 . The propagation paths of indirect change impact are all possible paths of change propagation divergence.

The complex association relationships between parts make change propagation diffuse in different directions, which will form many change propagation paths. According to the relationships between parts and change propagation patterns, all direct and indirect CPPs from feeding part to reachable part can be identified, and then the change propagation tree is constructed, as show in **Figure 4**. The change propagation tree describes propagation characteristic of feeding part propagating within the module, and its every branch represents a direct or indirect path of change propagation. The elements within any path are not repeated, so the number of the elements in the longest path will not be more than the number of the parts within corresponding module.

Generally speaking, the closer the relationships among parts within the module, the greater the divergence of change propagation, and the more the branches of the change propagation tree. For a CPP, the probability of change impact is related to the length of this propagation path, i.e., the number of relevant parts. For the sake of simplicity, the term “ $(C_j) \rightarrow (C_k)$ ” is used to refer to direct CPP. Assume there exist distinct CPPs from part C_j to part C_k , where $C_j \rightarrow C_k$ is one direct path with the propagation probability of $r_{k,j}$. Of course, there may be no direct



path, for instance, $C_7 \rightarrow C_5$, $r_{5,7} = 0$, as shown in **Figure 1**. The others $s-1$ paths are indirect. If $C_i \rightarrow C_u \rightarrow C_v \rightarrow C_k$ is one of the indirect CPPs (the r th path), as shown in **Figure 4**, then the change impact degree of this path is expressed as follows:

$$p(C_j, C_{kr}) = r_{u,j} \times r_{v,u} \times r_{k,v} \quad (4)$$

The different paths have different change impacts. All possible change impacts are reflected by corresponding propagation paths. The sum of change impacts of all possible CPPs from one part to another part are the change impact degree of the former on the latter. Hence, the change impact degree propagating from part C_j to part C_k , $P(C_j, C_k)$, is the sum of change impact degree of all of s distinct change propagation paths, namely

$$P(C_j, C_k) = \sum_{r=1}^s P(C_j, C_{kr}) \quad (5)$$

Equation 5 can be applied to compute the change impact degree of the feeding part C_j on the reachable part C_k . For instance, let us consider a case shown in **Figure 1**. The change of C_3 in module M_1 will propagate to C_5 in module M_2 , and the latter will cause the alteration of C_6 , C_7 and C_8 . Consequently, C_5 can be regarded as feeding part of design change in M_2 . The modification of C_5 will directly propagate to C_6 and C_7 , and the change of C_6 will propagate to C_8 , whereby C_6 is intercoupled with C_7 that is intercoupled with C_8 . To analyze change impact propagation from C_5 to all of other parts belong to M_2 , one should identify CPPs of C_5 within M_2 . On basis of DDM in **Figure 1** and change propagation network in **Figure 2**, the change propagation trees from C_5 to C_6 , C_7 and C_8 in module M_2 are constructed, as shown in **Figure 5**.

Then, according to change propagation trees in **Figure 5**, we can respectively compute change impact degrees of C_5 on C_6 , C_7 and C_8 , as following.

For C_5 to C_6 : $P(C_5, C_6) = r_{6,5} + r_{7,5} \times r_{6,7} = 0.6 + 0.5 \times 0.1 = 0.65$.

For C_5 to C_7 : $P(C_5, C_7) = 0.5 + 0.6 \times (0.4 + 0.4 \times 0.3) = 0.846$.

For C_5 to C_8 : $P(C_5, C_8) = 0.6 \times (0.4 + 0.4 \times 0.6) = 0.384$.

The above change impact degrees are added, and the result is comprehensive change impact degree of C_5 on all of other parts in M_2 , i.e.

$$P(C_5, M_2) = P(C_5, C_6) + P(C_5, C_7) + P(C_5, C_8) = 1.846$$

Module M_1 has only one instigating C_3 that affects M_2 , moreover, M_2 has only one affected part C_5 , so the change impact degree of M_1 on M_2 is equal to the following.

$$P(M_1, M_2) = P(C_3, M_2) = r_{5,3} \times P(C_5, M_2) = 0.2 \times [1 + 1.846] = 0.5692$$

IDENTIFICATION OF INFLUENTIAL MODULES CONSIDERING DESIGN CHANGE IMPACTS

Analysis on the Influence of Affected Part Within Module

As mentioned in *Analysis on Change Impact Degree Between Modules*, the change propagation between modules is not only dependent on the association elements between modules, but also relies on the relationships among parts within the affected module and the influence of affected parts in the corresponding module in which it is located. For instance, module M_1 affects part C_5 in module M_2 , and module M_2 affects part C_2 in module M_1 , as shown in **Figure 1**. C_5 is considered feeding part of change propagation from module M_1 to module M_2 . Similarly, C_2 is feeding part propagating from module M_2 to module M_1 . The number of the parts for the two modules is the same. Their change propagation networks within modules M_1 and M_2 are similar and direct change probabilities are also near. Since the influence of part C_2 in module M_1 and part C_5 in module M_2 is not the same, their change impact degree is also different.

For each module, we can take any part as a change source, and then calculate its change impact degree that propagates to all of the other parts within the same module in which it is located. Assume a module M is composed of m parts, the change impact degree of the p th part within a module, $P(C_p, M)$, is expressed as follows.

$$P(C_p, M) = \sum_{k=1, k \neq p}^m P(C_p, C_k) \quad (6)$$

From the perspective of engineering change, the larger the value of change impact degree, the higher the influence of the corresponding part inside the module. According to change impact degree of each part, one can judge their influence within the same module. If the change impact degree of one part is the largest in the corresponding module, it means that this part has more significant effect on other parts. Namely, its influence is the highest in this module. So, in the design process of the modular products, it should be avoided as an affected part between modules as much as possible to prevent it from propagating further. If the change impact degree of one part is the smallest in the corresponding module, it means that this part has the smallest effect on other parts. When its value is equal

TABLE 1 | The change impact degrees of parts within M_1 .

C	1	2	3	4
$P(C_i, C_1)$	-	0.104	0.2	0.04
$P(C_i, C_2)$	0.838	-	0.44	0.088
$P(C_i, C_3)$	0.817	0.502	-	0.2
$P(C_i, C_4)$	0.832	0.2	0.704	-
$P(C_i, M_1)$	2.487	0.806	1.344	0.328
Priority of the influence	1	3	2	4

TABLE 2 | The change impact degrees of parts within M_2 .

C	5	6	7	8
$P(C_i, C_5)$	-	0.156	0.3	0.09
$P(C_i, C_6)$	0.65	-	0.25	0.075
$P(C_i, C_7)$	0.846	0.52	-	0.3
$P(C_i, C_8)$	0.384	0.64	0.7	-
$P(C_i, M_2)$	1.846	1.316	1.25	0.465
Priority of the influence	1	2	3	4

to 0, it means that the corresponding part does not affect other parts and may limit the diffusion of change propagation.

For example, consider the example proposed change impacts discussed in *Analysis on Change Impact Degree Between Modules*. The change impact degrees of all parts in M_1 and M_2 are respectively shown in **Table 1, 2**.

It is evident from **Table 1, 2** that part C_1 has the largest effect on other parts ($P(C_i, M_1) = 2.487$) in M_1 . So C_1 is a dominant part in M_1 , next is C_3 , and then is C_2 and C_4 . Similarly, the priority of change impact in M_2 is C_5, C_6, C_7 , and C_8 in turn.

Identification of Influential Modules

The change impact degree between any two modules embodies their direct influence, but it does not reflect the influence of design change for a module in the whole product. According to the definition from reference (Li et al., 2019), the influential modules are the modules that once the design changes, the cascading influence will be hard to control. In other words, the influential modules are those that have a significant impact on change propagation. So it is also necessary to analyze the impact of module changes from the perspective of the whole product, and consider simultaneously indirect change impacts between modules. Here, any a module is considered change source. We can analyse the influence of its instigating parts on other modules, and calculate its comprehensive change impact degree within a given product. According to comprehensive change impact degree of each module we can identify influential modules and the redesign priority of modules in products.

In *Identify Reachable Parts for Change Ppropagation*, we determined all reachable parts to which a change source propagated with reachable matrix. Consequently, in the process of analysing comprehensive change impact degree of module, the reachable matrix can be also used to identify reachable parts of other modules to which a change has occurred in the instigating part propagates. Then the change propagation network between

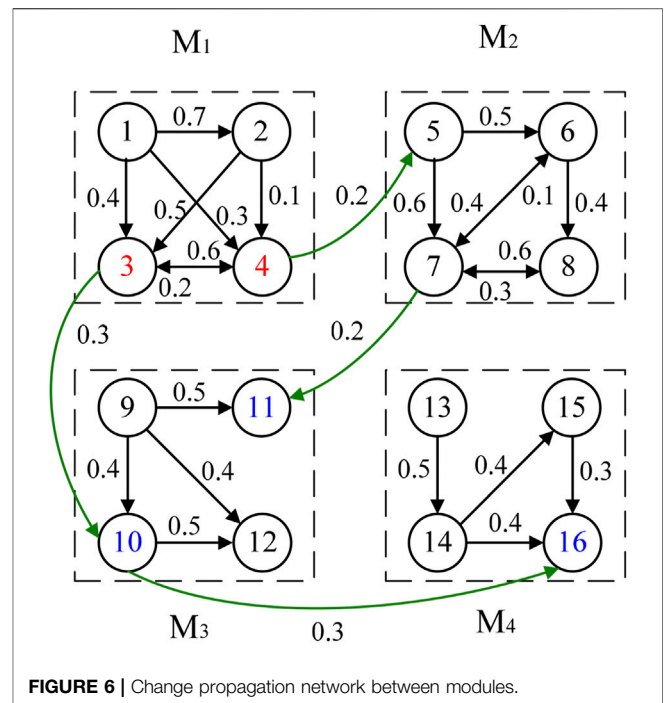


FIGURE 6 | Change propagation network between modules.

modules is constructed according to the dependence between association parts. Take any module as change source and the corresponding instigating parts that affect other modules are determined. The CPPs of each instigating part are analyzed and captured, and the influence of each propagation path is discussed. The change impact degree of each instigating part on all reachable parts in other modules is computed. Here, other affected modules can be thought as an integrated module. The comprehensive change impact degree of upstream module in which instigating parts are located will be equivalent to direct change impact degree of upstream module on this integrated module.

Let us consider a product that consists of four modules. The change propagation network between modules is represented as **Figure 6**. The change of module M_1 will directly cause alternation of modules M_2 and M_3 . The modification of M_2 will propagate to M_3 , and the latter simultaneously passes the change onto module M_4 . The measures, change impact degrees of parts within module used in *Analysis on the Influence of Affected Part within Module*, cannot be used directly to quantify comprehensive change impact degree of modules for the problem considered in this section, because their propagation ways are different. One module affects the other, but it does not mean that all parts of the former have an impact on the latter. Consequently, the module cannot be thought as node in change propagation network. For instance, M_2 influences M_3 and the latter influences M_4 in **Figure 6**, but M_2 does not affects M_4 . Since part C_{11} in M_3 influenced by M_2 is an absorption part, its change does not propagate to other parts. When the comprehensive change impact degree of M_1 is computed, all of other modules, namely, M_2, M_3 and M_4 , can be regarded as an integrated module M_1' . At this time, C_3 and C_4 are instigating parts. The change impact degree of M_1 to M_1' , is comprehensive change impact degree of M_1 within the product, $P(M_1, A)$, i.e.

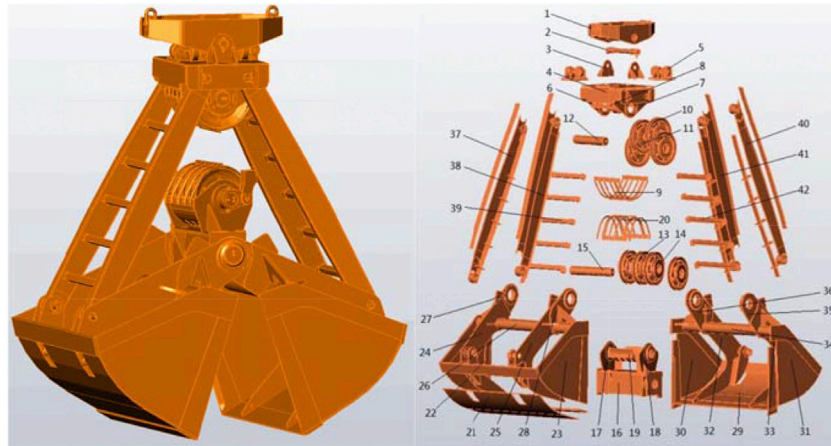


FIGURE 7 | Structure of the grab and its components explosion.

TABLE 3 | List of key components of the grab.

No	Name	No	Name	No	Name
1	Balance frame	15	Lower pulley shaft	29	Right grab bucket floor
2	Balancing shaft	16	Lower bearing beam	30	Right grab side plate 1
3	Balance frame base	17	Side plate 1 of lower bearing beam pulley	31	Right grab side plate 2
4	Upper bearing beam	18	Side plate 2 of lower bearing beam pulley	32	Right grab bent plate 1
5	Guide rope device	19	Lower bearing beam pulley cover plate	33	Right grab bent plate 2
6	Side plate 1 of upper bearing beam pulley	20	Lower protective frame	34	Right grab pipe
7	Side plate 2 of upper bearing beam pulley	21	Left grab bucket floor	35	Right grab center ear plate 1
8	Upper bearing beam pulley cover plate	22	Left grab side plate 1	36	Right grab center ear plate 2
9	Upper protective frame	23	Left grab side plate2	37	Left strut 1
10	Upper pulley group	24	Left grab bent plate 1	38	Left strut 2
11	Bearing group of upper pulley group	25	Left grab bent plate 2	39	Left strut square tube
12	Upper pulley shaft	26	Left grab pipe	40	Right strut 1
13	Lower pulley group	27	Left grab center ear plate 1	41	Right strut 2
14	Bearing group of lower pulley group	28	Left grab center ear plate 2	42	Right strut square tube

TABLE 4 | The name of each module and the serial number of the components it contains.

No	Name	The serial number of the components
M_1	Lower pulley group	(13, 15, 14)
M_2	Upper pulley group	(10, 12, 11)
M_3	Upper bearing beam	(4, 6, 7, 8, 9, 1, 2, 3)
M_4	Guide rope device	(5)
M_5	Lower bearing beam	(16, 17, 18, 19, 20)
M_6	Left grab bucket body	(21, 22, 23, 24, 25, 26, 27, 28)
M_7	Right grab bucket body	(29, 30, 31, 32, 33, 34, 35, 36)
M_8	Left strut	{37, 38, 39}
M_9	Right strut	{40, 41, 42}

$$\begin{aligned}
 P(M_1, A) &= P(M_1, M_1') \\
 &= P(M_1, M_2) + P(M_1, M_3) + P(C_4, C_7) \\
 &\quad \cdot P(C_7, M_3) + P(C_3, C_{10}) \cdot P(C_{10}, M_4) \quad (7)
 \end{aligned}$$

where “ $P(M_1, M_2) + P(M_1, M_3)$ ” is direct change impact degree of M_1 on M_2 and M_3 , “ $P(C_4, C_7) \cdot P(C_7, M_3)$ ” and “ $P(C_3, C_{10}) \cdot P(C_{10},$

$M_4)$ ” are indirect change impact degree of M_1 on M_3 and M_4 , respectively.

Assume a product is composed of N modules and n parts, where module M_i has r instigating parts that affect other modules, the comprehensive change impact degree of M_i , $P(M_i, A)$, is expressed as

$$P(M_i, A) = \sum_{j=1}^r P(M_i, M'_j) \quad (8)$$

where $P(M_i, M'_j)$ is change impact degree of one instigating part in module M_i on all of other modules.

The larger the value of $P(M_i, A)$, the larger the influence of change propagation of module M_i within product. According to the value of $P(M_i, A)$ of each module, the influential modules can be identified.

Another method computing comprehensive change impact degree is based on the level of CPPs. Since the number of modules in the longest path is not more than N , the maximum level of propagation paths does not exceed $N - 1$. Then we can calculate comprehensive change impact degree of any a module for each level. The first level is used to compute direct change impact

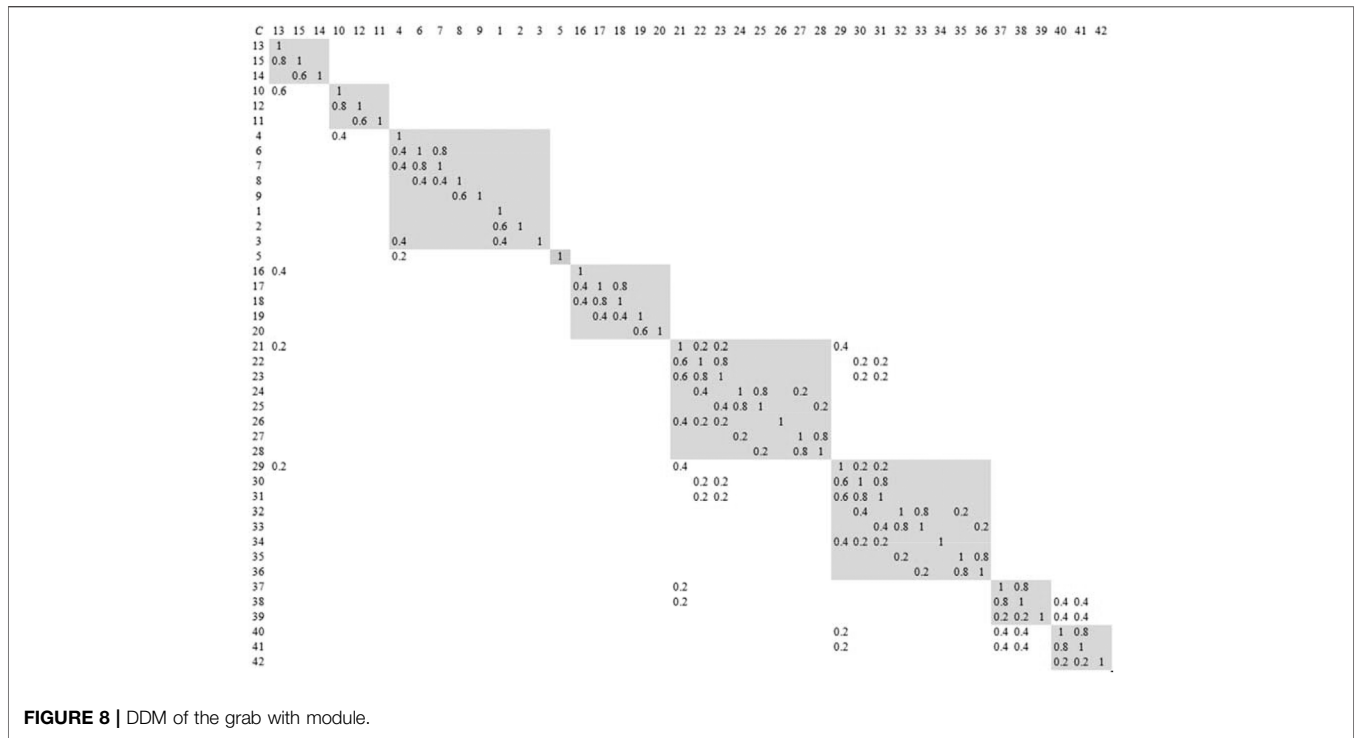


FIGURE 8 | DDM of the grab with module.

degree of initial module. The second level is applicable to compute its indirect change impact degree that propagates to another module through one intermediate module that has a direct effect on the former, and so on. The last level is for indirect change impact degree through $N - 1$ intermediate modules. The comprehensive change impact degree, $P(M_i, A)$ can be also represented as

$$\begin{aligned}
 P(M_i, A) = & \sum_{j=1, j \neq i}^N P(M_i, M_j) + \sum_{\substack{k=1 \\ k \neq j \neq i}}^N r_{b,a} \times P(C_b, C_c) \times P(C_c, M_k) + \dots \\
 & + \sum_{\substack{p=1 \\ p \neq j \neq \dots \neq i}}^N r_{b,a} \times P(C_b, C_c) \times r_{d,c} \times P(C_d, C_e) \times \dots \times r_{v,u} \times P(C_v, C_w) \times P(C_w, M_p)
 \end{aligned}
 \tag{9}$$

where $\sum_{j=1, j \neq i}^N P(M_i, M_j)$ is direct impact degree of initial module M_i on other modules in the first level of propagation paths; $\sum_{k=1, k \neq j \neq i}^N r_{b,a} \times P(C_b, C_c) \times P(C_c, M_k)$ is indirect impact

degree of M_i in the second level of the path, whereas M_i affects indirectly M_k through intermediate M_j that has a direct influence on M_k . Both parts C_b and C_c belong to M_j that is directly affected by M_i . $r_{b,a}$ is the probability of change propagation from part C_a in M_i to C_b , C_b is affected part to M_j , and C_c is an instigating part to M_k . The last item in Eq. 9 is indirect impact degree of M_i in the last level of the path, whereas $r_{d,c}, \dots, r_{v,u}$ are respectively change probability from instigating parts to affected parts between association modules, $P(C_w, M_p)$ is change impact degree of part C_w that belongs to the previous module to M_p , and $P(C_d, C_e), \dots, P(C_v, C_w)$ are respectively change impact degree within modules from feeding part to instigating part that affects the next module in the path.

Solution to Comprehensive Change Impact Degree Based on Bat Algorithm

With the increase of the number of parts, the number of propagation paths will increase exponentially. The comprehensive change impact degree involves massive calculation and is time-consuming. The biological evolution algorithms have been widely used (Li and Yin, 2011; 2011a; Li and Yin, 2011; 2011b; Chen et al., 2021a; Chen et al., 2021b). The bat algorithm has very high efficiency in optimization calculation (Mashwani et al., 2021; Saji and Barkatou, 2021). So this article uses bat algorithm to solve comprehensive change impact degree of complex network model.

The Principle of Bat Algorithm

Bat algorithm has the following three idealized assumptions (Yang and He, 2013; Xu and Zhang, 2016).

- 1) All bats use echolocation to perceive distance, and can distinguish the food or prey and obstacles in the background in the way we do not know.
- 2) At first, the bat flies at random speed and direction. They can search the prey through changing the wavelength λ and loudness S . Simultaneously, they can automatically adjust the transmitted pulse wavelength and rate r ($\in [0, 1]$) according to the distance.
- 3) Assume that the loudness varies gradually from the maximum value $S(1)$ to the lowest constant value S_{\min} .

Solution to Bat Movement

We will input the bats into n -dimension space. The frequency of the i th bat is F_i , and its position and speed are respectively,

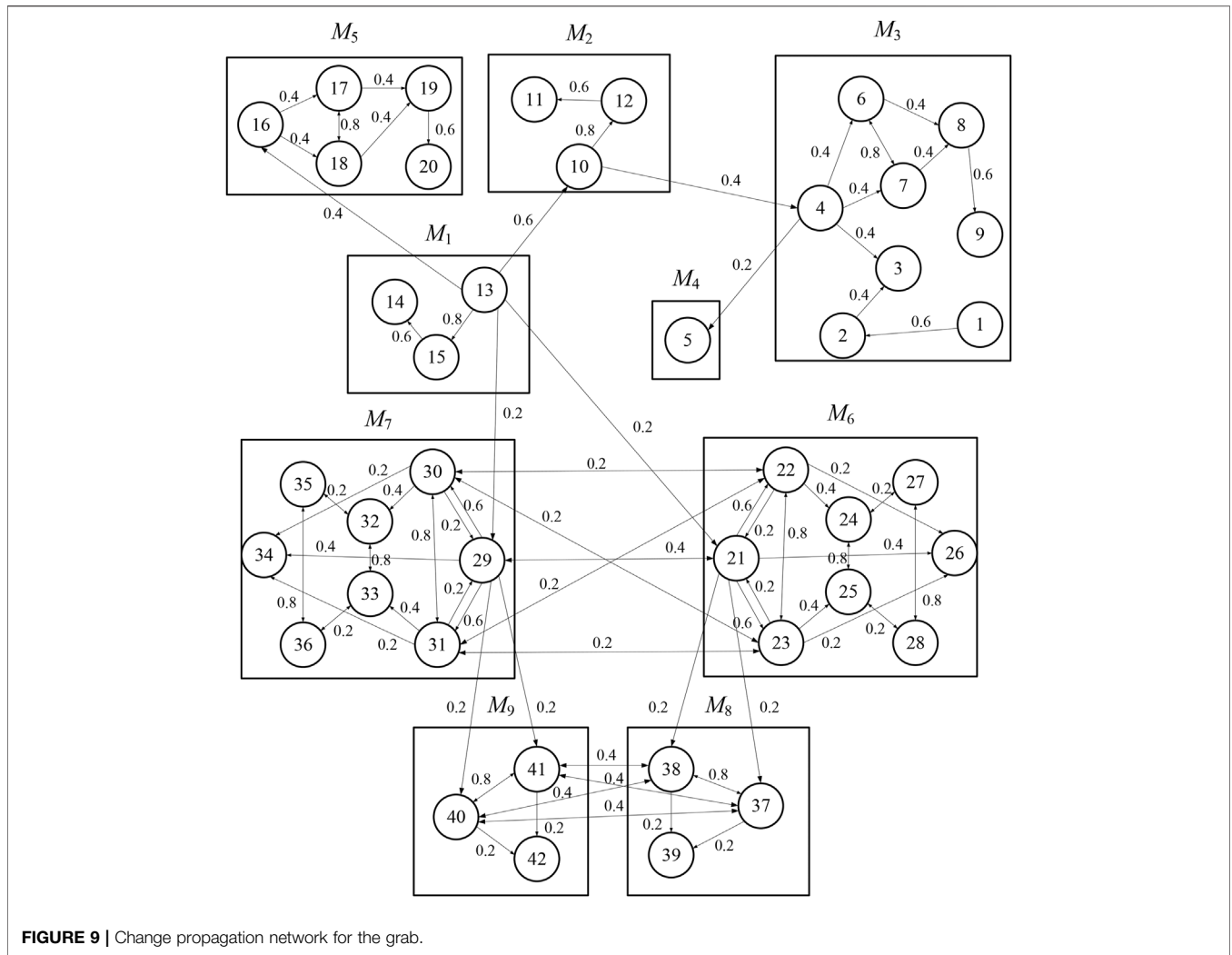


FIGURE 9 | Change propagation network for the grab.

$x_i = (x_{i1}, x_{i2}, \dots, x_{in})$ and $v_i = (v_{i1}, v_{i2}, \dots, v_{in})$. Then the equations of the t generation for the bat i on the position and speed are as follows.

$$F_i = F_{\min} + (F_{\max} - F_{\min})\beta \quad \beta \in [0, 1] \quad (10)$$

$$v_{id}(t+1) = v_{id}(t) + F_i(v_{id}(t) - p_{gd}(t)) \quad (11)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (12)$$

where, F_{\min} and F_{\max} are respectively the maximum and minimum frequencies, $\beta \in [0, 1]$ is a random vector and its elements obey uniform distribution; $x_{id}(t)$ and $x_{id}(t+1)$ are respectively the d th-dimension position of bat i in the t th and $(t+1)$ -th iteration optimization process, $v_{id}(t)$ and $v_{id}(t+1)$ are their corresponding speeds, and $p_{gd}(t)$ is the d th-dimension position of bat g with optimal global fitness in the t th and $(t+1)$ -th iteration optimization process.

When the bat performs a global search, it also conducts a local refinement search to find a better solution. Once a best solution x_{old} is chosen randomly from the current optimal solution set, the new pending position of each bat is generated nearby, as shown in type (13).

$$x_{new} = x_{old} + \epsilon S(t) \quad (13)$$

where, $\epsilon \in$ (Bonvoisin et al., 2016) is an arbitrary number, $S(t)$ is the average loudness of all bats in the t th iteration.

In addition, the loudness S_i and rate r_i of pulse emission, are required to update with iterative process, in order to achieve a good balance of the algorithm between global search and local search. Update equation is as follows.

$$S_i(t+1) = \alpha S_i(t) \quad (14)$$

$$r_i(t+1) = r_i(1)[1 - \exp(-\gamma(t+1))] \quad (15)$$

where, α is loudness attenuation factor and is a constant, $0 < \alpha < 1$; γ is pulse frequency increase coefficient and is also a constant, $\gamma > 0$; $S_i(t)$ and $S_i(t+1)$ are respectively the loudness of bat i in the t th and $(t+1)$ -th iteration optimization process; $r_i(t)$ and $r_i(t+1)$ are respectively the pulse rate of bat i in the t th and $(t+1)$ -th iteration optimization process.

Assume the size of the bat population is n , and the position of the i th bat is $x(i)$. The steps of solving the largest comprehensive change impact degree using the bat algorithm is given as follows.

	C_{16}	C_{17}	C_{18}	C_{19}	C_{20}
C_{16}	1	0	0	0	0
C_{17}	1	1	1	0	0
C_{18}	1	1	1	0	0
C_{19}	1	1	1	1	0
C_{20}	1	1	1	1	1

FIGURE 10 | Reachable matrix of M_5 for grab.

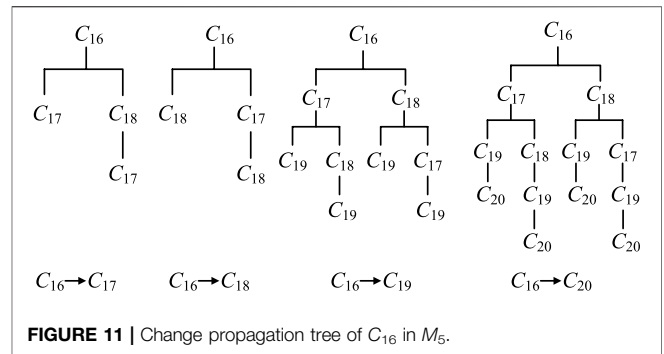


FIGURE 11 | Change propagation tree of C_{16} in M_5 .

- Step 1: Build the network model of the product parts, and set parameters' values of bat algorithm, such as the size of the bat population, the maximum number of iterations, and so on.
- Step 2: Initialize the position $x(i)$, speed $v(i)$, frequency $F(i)$, the pulse emission rate $r_i(1)$, and the pulse loudness $S_i(1)$. Apply bat algorithm to calculate the local optimal value, $P(M_i, A)$, and find out the corresponding optimal bat individual in the population.
- Step 3: According to Eqs 10–12, update the position and speed of the bat in the iteration process.
- Step 4: Produce a random number $rand$. If $rand > r_i(t)$, a new local solution $P'(M_i, A)$ is obtained by Eq. 13.
- Step 5: For each bat individual, the corresponding random number $rand$ is created. If $rand < S_i$ and $P'(M_i, A) > P(M_i, A)$, then this solution is accepted. At the same time $r_i(t)$ and $S_i(t)$ are updated by Eqs 14, 15.
- Step 6: Judge whether stop conditions are met. If they are satisfied, turn to the next step. Otherwise, go to step 3.
- Step 7: Update and output the global optimal solution.

CASE STUDY

In this section, a case study on the crane grab is provided to illustrate the proposed analysis method of change propagation between modules and to identify influential modules. The grab is

a kind of the special load handling device for the crane, which is mainly used to grab bulk cargoes. A typical crane grab contains 42 key components. Figure 7 illustrates the structure of the grab and its components explosion, and Table 3 summarizes a list of its key components. Cheng et al. (Cheng et al., 2018) clustered the components of the grab into nine modules using DBSCAN algorithm. The name of each module and the serial number of the components it contains are shown in Table 4.

According to dependence relationships between components of the grab, the probability of change propagation from one part to another part is analysed and identified, and DDM of its modules is constructed, as shown in Figure 8. The change propagation network that represents design change relationships among modules and parts in the modules is constructed, as shown in Figure 9.

Then, the change impact degree between modules can be calculated according to the proposed method. The influence of M_1 on M_5 is taken as an example to describe and analyse change propagation from the former to the latter. It can be seen from Figures 8, 9 that M_1 passes the change of part C_{13} onto the directly connected parts C_{16} in M_5 , whereby the change of C_{16} will propagate to other parts, such as parts C_{17} , C_{18} , and so on. So, C_{16} is deemed as a feeding part of change propagation in M_5 . To analyse change propagation of C_{16} , the reachable matrix of M_5 is firstly computed, and the result is shown as Figure 10. It is evident from Figure 10 that the reachable parts of change propagation for C_{16} include not only C_{17} and C_{18} (direct propagation) but also C_{19} and C_{20} (indirect propagation), i.e., the modification of C_{16} will lead to the alteration of all of other parts in M_5 .

Next, the change propagation trees from C_{16} to C_{17} , C_{18} , C_{19} and C_{20} in M_5 are constructed through parallel breadth-first search algorithm, as shown in Figure 11. All possible propagation paths of C_{16} within M_5 are captured. The change impact degree of C_{16} on each reachable part is calculated as following.

For C_{16} to C_{17} , it has two propagation paths, one is direct path from C_{16} to C_{17} , and another is indirect path from C_{16} to C_{17} through intermediate part C_{18} . So the change impact degree of C_{16} on C_{17} is.

$$P(C_{16}, C_{17}) = r_{17, 16} + r_{18, 16} \times r_{17, 18} = 0.4 + 0.4 \times 0.8 = 0.72.$$

$$\text{For } C_{16} \text{ to } C_{18}, P(C_{16}, C_{18}) = 0.4 + 0.4 \times 0.8 = 0.72.$$

$$\text{For } C_{16} \text{ to } C_{19}, P(C_{16}, C_{19}) = 0.576.$$

TABLE 5 | The results of change impact degrees of modules.

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9
$P(M_i, M_1)$	-	-	-	-	-	-	-	-	-
$P(M_i, M_2)$	1.368	-	-	-	-	-	-	-	-
$P(M_i, M_3)$	-	1.4125	-	-	-	-	-	-	-
$P(M_i, M_4)$	-	-	0.2	-	-	-	-	-	-
$P(M_i, M_5)$	1.3446	-	-	-	-	-	-	-	-
$P(M_i, M_6)$	1.1151	-	-	-	-	-	5.8281	-	-
$P(M_i, M_7)$	1.1151	-	-	-	-	5.8281	-	-	-
$P(M_i, M_8)$	-	-	-	-	-	0.864	-	-	3.456
$P(M_i, M_9)$	-	-	-	-	-	-	0.864	3.456	-
$P(M_i, A)$	9.5863	1.4925	0.2	0	0	9.0867	9.0867	3.456	3.456

For C_{16} to C_{20} , $P(C_{16}, C_{20}) = 0.3456$.

The sum of above change impact degrees propagating from C_{16} to C_{17} , C_{18} , C_{19} and C_{20} , is comprehensive change impact degree of C_{16} on all of other parts within M_5 , i.e.

$$P(C_{16}, M_5) = P(C_{16}, C_{17}) + P(C_{16}, C_{18}) + P(C_{16}, C_{19}) + P(C_{16}, C_{20}) = 2.3616.$$

M_1 has only one instigating part C_{13} that has effect on M_5 and just influence C_{16} of M_5 , so the change impact degree of M_1 on M_5 is following.

$$P(M_1, M_5) = 0.4 \times [1 + 2.3616] = 1.34464.$$

Similarly, the change impact degrees between other modules can be computed, and then the comprehensive impact degrees of all modules can be obtained by bat algorithm. The results are shown as **Table 5**.

Table 5 shows the influence of change propagation for each module. The value of $P(M_1, A)$ is largest, followed by $P(M_6, A)$ and $P(M_7, A)$. Moreover, the comprehensive change impact degrees of these three modules are much bigger than that of other modules. So M_1 , M_6 , and M_7 are influential modules in the grab. Both $P(M_4, A)$ and $P(M_5, A)$ equal to 0, which mean that M_4 and M_5 have no influence on other modules, in other words, both of them are full absorption modules. M_8 and M_9 have the same structure with synergistic functions. They just interact with each other, and have no effect on other modules. So they can be also regarded as absorption modules. From the standpoint of engineering design, design change of modules M_1 , M_6 , and M_7 should be controlled as much as possible to effectively reduce the cost of product development and prevent the product failure.

CONCLUSION

Product modularization has a significant influence on the product development process and the whole product lifecycle. There often exist association relationships between modules. A change of one part/module may cause other parts or modules to change, which in turn propagates through a product. The change between modules is not only dependent on the association element between modules, but also relies on the relationships among parts within an affected module and the influence of the affected part in the corresponding module in which it is located. The larger the size of the module and the stronger the influence of the

affected part in the corresponding module, the greater the scope of change propagation.

The proposed approach measures the relative change impacts of modules and parts in the modules. All possible CPPs are determined by a parallel breadth-first search algorithm. A reachable matrix is employed to identify which parts are influenced directly or indirectly by the affected part within the corresponding module in which it is located. A change propagation network that represents design change relationships among parts and modules is constructed. Then the direct and indirect impacts of change propagation are integrated, and the relative change impact degrees of modules and parts in the modules are computed. The influential modules are identified by the bat algorithm from the perspective of engineering change. Finally, an application of the proposed methods of association analysis and change impacts is demonstrated by an example of the crane grab. The methodology is applied to the design of other modular products. The design change impact in the modular product families can be studied in the future.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

XC built the research plan, proposed an approach to measuring the change impact degrees between modules and drafted the manuscript; ZG used parallel breadth-first algorithm to search propagation paths; XM provided help in data analysis, and TY revised the manuscript.

FUNDING

This project was supported by the National Natural Science Foundation, China (No. 51765019 and No. 71462007), and the Key Science and Technology Research Plan of Education Department of Jiangxi Province (No. GJJ200601), China.

REFERENCES

- Ahmad, N., Wynn, D. C., and Clarkson, P. J. (2013). Change Impact on a Product and its Redesign Process: A Tool for Knowledge Capture and Reuse. *Res. Eng. Des.* 24 (3), 219–244. doi:10.1007/s00163-012-0139-8
- AlGeddawy, T., and ElMaraghy, H. (2013). Optimum Granularity Level of Modular Product Design Architecture. *CIRP Annals-Manufacturing Techn.* 62 (1), 151–154. doi:10.1016/j.cirp.2013.03.118
- Alhandawi, K., Andersson, P., Panarotto, M., Isaksson, O., and Kokkolaras, M. (2020). Scalable Set-Based Design Optimization and Remanufacturing for Meeting Changing Requirements. *J. Mech. Des.*, 1–20. doi:10.1115/DETC2020-22458
- Bae, J., and Kim, S. (2014). Identifying and Ranking Influential Spreaders in Complex Networks by Neighborhood Coreness. *Physica A: Stat. Mech. Its Appl.* 395 (3), 549–559. doi:10.1016/j.physa.2013.10.047
- Beamer, S., Asanovi, K., and Patterson, D. (2012). Direction-optimizing Breadth-First Search. *entific Programming* 21 (3-4). doi:10.1109/SC.2012.50
- Bonvoisin, J., Halstenberg, F., Buchert, T., and Stark, R. (2016). A Systematic Literature Review on Product Modular Design. *J. Eng. Des.* 27 (7), 488–514. doi:10.1080/09544828.2016.1166482
- Borjesson, F., and Hölltä-Otto, K. (2014). A Module Generation Algorithm for Product Architecture Based on Component Interactions and Strategic Drivers. *Res. Eng. Des.* 25 (1), 31–51. doi:10.1007/s00163-013-0164-2
- Browning, T. R. (2001). Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions. *IEEE Trans. Eng. Manag.* 48 (3), 292–306. doi:10.1109/17.946528
- Buluc, A., and Madduri, K. (2011). Parallel Breadth-First Search on Distributed Memory Systems. *IEEE*. doi:10.1145/2063384.2063471
- Chen, T., Peng, L., Yang, J., Cong, G., and Li, G. (2021). Evolutionary Game of Multi-Subjects in Live Streaming and Governance Strategies Based on Social Preference Theory during the COVID-19 Pandemic. *Mathematics* 9 (21), 2743. doi:10.3390/math9212743
- Chen, T., Yin, X., Yang, J., Cong, G., and Li, G. (2021). Modeling Multi-Dimensional Public Opinion Process Based on Complex Network Dynamics Model in the Context of Derived Topics. *Axioms* 10 (4), 270. doi:10.3390/axioms10040270
- Cheng, X., Li, J., Wan, C., Qiu, H., and Wan, Y. (2018). DBSCAN-based Modular Design for the Crane Grab. *Int. J. Wireless Mobile Comput.* 15 (2), 157–162. doi:10.1504/IJWMC.2018.095693
- Cheng, X., Wan, C., Qiu, H., and Wan, L. (2020). Association Analysis between Modules and Decoupling Strategies in Modular Product Design. *Comp. Integrated Manufacturing Syst.* 26 (4), 1043–1051. doi:10.13196/j.cims.2020.04.018
- Cheng, X., Xiao, R., and Wang, H. (2018b). A Method for Coupling Analysis of Association Modules in Product Family Design. *J. Eng. Des.* 29 (6), 327–352. doi:10.1080/09544828.2018.1487531
- Clarkson, P. J., Simons, C., and Eckert, C. (2004). Predicting Change Propagation in Complex Design. *J. Mech. Des.* 126 (5), 788–797. doi:10.1115/1.4027495
- Eckert, C., Clarkson, P. J., and Zanker, W. (2004). Change and Customisation in Complex Engineering Domains. *Res. Eng. Des.* 15 (1), 1–21. doi:10.1007/s00163-003-0031-7
- Eppinger, S. D., and Browning, T. R. (2012). *Design Structure Matrix Methods and Applications*. Cambridge: MIT Press. doi:10.7551/mitpress/8896.001.0001
- Gao, C., Wei, D., Hu, Y., Mahadevan, S., and Deng, Y. (2013). A Modified Evidential Methodology of Identifying Influential Nodes in Weighted Networks. *Physica A: Stat. Mech. Its Appl.* 392 (21), 5490–5500. doi:10.1016/j.physa.2013.06.059
- Garcia, L. L., Arellano, A. G., and Cruz-Santos, W. (2020). A Parallel Path-Following Phase Unwrapping Algorithm Based on a Top-Down Breadth-First Search Approach. *Opt. Lasers Eng.* 124 (Jan.), 105827. doi:10.1016/j.optlaseng.2019.105827
- Giffin, M., de Weck, O., Bounova, G., Keller, R., Eckert, C., and Clarkson, P. J. (2009). Change Propagation Analysis of Complex Technical Systems. *J. Mech. Des.* 131 (8), 1–14. doi:10.1115/1.3149847
- Hamraz, B., and Clarkson, P. J. (2015). Industrial Evaluation of FBS Linkage: A Method to Support Engineering Change Management. *J. Eng. Des.* 26 (1-3), 24–47. doi:10.1080/09544828.2015.1015783
- Hein, P. H., Kames, E., Chen, C., and Morkos, B. (2021). Employing Machine Learning Techniques to Assess Requirement Change Volatility. *Res. Eng. Des.* 32 (16). doi:10.1007/s00163-020-00353-6
- Hoeltdae, K. M. M., and Otto, K. N. (2005). Incorporating Design Effort Complexity Measures in Product Architectural Design and Assessment. *Des. Stud.* 26 (5), 445–564. doi:10.1016/j.destud.2004.10.001
- Hölltä-Otto, K., Chiriac, N. A., Lysy, D., and Suh, E. S. (2012). Comparative Analysis of Coupling Modularity Metrics. *J. Eng. Des.* 23 (10-11), 790–806. doi:10.1080/09544828.2012.701728
- Hong, E. P., and Park, G. J. (2014). Modular Design Method Based on Simultaneous Consideration of Physical and Functional Relationships in the Conceptual Design Stage. *J. Mech. Sci. Techn.* 28 (1), 223–235. doi:10.1007/s12206-013-0956-3
- Jarratt, T. A. W., Eckert, C. M., Caldwell, N. H. M., and Clarkson, P. J. (2011). Engineering Change: an Overview and Perspective on the Literature. *Res. Eng. Des.* 22 (2), 103–124. doi:10.1007/s00163-010-0097-y
- Jung, S., Asikoglu, O., and Simpson, T. W. (2018). A Method to Evaluate Direct and Indirect Design Dependencies between Components in a Product Architecture. *Res. Eng. Des.* 29 (4), 507–530. doi:10.1007/s00163-018-0291-x
- Kashkoush, M., and ElMaraghy, H. (2017). Designing Modular Product Architecture for Optimal Overall Product Modularity. *J. Eng. Des.* 28 (5), 293–316. doi:10.1080/09544828.2017.1307949
- Kitsak, M., Gallos, L. K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H. E., et al. (2010). Identification of Influential Spreaders in Complex Networks. *Nat. Phys.* 6 (11), 888–893. doi:10.1038/nphys1746
- Koh, E. C., Förg, A., Kreimeyer, M., and Lienkamp, M. (2015). Using Engineering Change Forecast to Prioritise Component Modularization. *Res. Eng. Des.* 26 (4), 337–353. doi:10.1007/s00163-015-0200-5
- Koh, E. C. Y., Caldwell, N. H. M., and Clarkson, P. J. (2012). A Method to Assess the Effects of Engineering Change Propagation. *Res. Eng. Des.* 23 (4), 329–351. doi:10.1007/s00163-012-0131-3
- Lee, H., Seol, H., Sung, N., Hong, Y., and Park, Y. (2010). An Analytic Network Process Approach to Measuring Design Change Impacts in Modular Products. *J. Eng. Des.* 21 (1), 75–91. doi:10.1080/09544820802232517
- Li, X., and Yin, M. (20112011). Design of a Reconfigurable Antenna Array with Discrete Phase Shifters Using Differential. *evolution algorithm* 31, 29–43. doi:10.2528/PIERB11032902
- Li, X., and Yin, M. (20112011). Hybrid Differential Evolution with Biogeography-Based Optimization for Design of a Reconfigurable Antenna Array with Discrete Phase Shifters. *Int. J. Antennas Propagation* (1), 235–245. doi:10.1155/2011/685629
- Li, Y., Ni, Y., Zhang, N., and Liu, Z. (2021). Modularization for the Complex Product Considering the Design Change Requirements. *Res. Eng. Des.*, 1–16. doi:10.1007/S00163-021-00369-6
- Li, Y., Wang, Z., Zhong, X., and Zou, F. (2019). Identification of Influential Function Modules within Complex Products and Systems Based on Weighted and Directed Complex Networks. *J. Intell. Manufacturing* 30 (6), 2375–2390. doi:10.1007/s10845-018-1396-9
- Liu, J., Xiong, Q., Shi, W., Shi, X., and Wang, K. (2016). Evaluating the Importance of Nodes in Complex Networks. *Physica A: Stat. Mech. Its Appl.* 452, 209–219. doi:10.1016/j.physa.2016.02.049
- Maier, J. F., Wynn, D. C., Biedermann, W., Lindemann, U., and Clarkson, P. J. (2014). Simulating Progressive Iteration, Rework and Change Propagation to Prioritise Design Tasks. *Res. Eng. Des.* 25 (4), 283–307. doi:10.1007/s00163-014-0174-8
- Mashwani, W. K., Mehmood, I., Bakar, M. A., and Kocak, I. (2021). A Modified Bat Algorithm for Solving Large-Scale Bound Constrained Global Optimization Problems. *Math. Probl. Eng.* 2021 (1), 1–14. doi:10.1155/2021/6636918
- Mehta, C., Patil, L., and Dutta, D. (2013). An Approach to Determine Important Attributes for Engineering Change Evaluation. *J. Mech. Des.* 135 (4), 1–17. doi:10.1115/1.4023551
- Ollinger, G. A., and Stahovich, T. F. (2004). RedesignIT-A Model-Based Tool for Managing Design Changes. *J. Mech. Des.* 126 (2), 208–216. doi:10.1115/1.1666888
- Pasqual, M. C., and deWeck, O. L. (2012). Multilayer Network Model for Analysis and Management of Change Propagation. *Res. Eng. Des.* 23 (4), 305–328. doi:10.1007/s00163-011-0125-6

- Saji, Y., and Barkatou, M. (2021). A Discrete Bat Algorithm Based on Lévy Flights for Euclidean Traveling Salesman Problem. *Expert Syst. Appl.* 172, 114639. doi:10.1016/j.eswa.2021.114639
- Samling, R., and de Weck, O. (2007). Assessing Risks and Opportunities of Technology Infusion in System Design. *Syst. Eng.* 10 (1), 1–25. doi:10.1002/sys.20061
- Sharman, D. M., and Yassine, A. A. (2017). Architectural Valuation Using the Design Structure Matrix and Real Options Theory. *Concurrent Eng.* 15 (2), 157–173. doi:10.1177/1063293x07079320
- Steward, D. V. (1981). The Design Structure System: A Method for Managing the Design of Complex Systems. *IEEE Trans. Eng. Manag.* 28, 71–74. doi:10.1109/TEM.1981.6448589
- Xie, Y., and Ma, Y. (2016). Well-controlled Engineering Change Propagation via a Dynamic Inter-feature Association Map. *Res. Eng. Des.* 27 (4), 311–329. doi:10.1007/s00163-016-0220-9
- Xu, H., and Zhang, T. (2016). Hybrid Discrete Bat Algorithm for Solving the Multi-Objective Flexible Job Shop Scheduling Problem. *J. Mech. Eng.* 52 (18), 201–212. doi:10.3901/JME.2016.18.201
- Yang, F., and Duan, G. (2012). Developing a Parameter Linkage-Based Method for Searching Change Propagation Paths. *Res. Eng. Des.* 23 (4), 353–372. doi:10.1007/s00163-011-0124-7
- Yang, X. S., and He, X. (2013). Bat Algorithm: Literature Review and Applications. *Int. J. Bio-Inspired Comput.* 5 (3), 141–149. doi:10.1504/IJBIC.2013.055093
- Yigit, Y., Akram, V. K., and Dagdeviren, O. (2021). Breadth-first Search Tree Integrated Vertex Cover Algorithms for Link Monitoring and Routing in Wireless Sensor Networks. *Computer Networks.* doi:10.1016/J.COMNET.2021.108144
- Zhang, X., Zhu, J., Wang, Q., and Zhao, H. (2013). Identifying Influential Nodes in Complex Networks with Community Structure. *Knowledge Based Syst.* 42 (4), 74–84. doi:10.1016/j.knosys.2013.01.017

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Cheng, Guo, Ma and Yuan. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.