23rd INTERNATIONAL DEPENDENCY AND STRUCTURE MODELING CONFERENCE, DSM 2021
MONTRÉAL, CANADA, 12 – 14 October, 2021

Multi-domain Knowledge Integration and Organizational Clustering in product development project

Mingxing Chang[1], Qing Yang*[1], Qinru Wang[1],Tao Yao[2]

[1]School of Economics and Management, University of Science & Technology Beijing, China

[2]Alibaba US, 500 108th Ave NE #800, Bellevue, WA 98004, USA

Abstract: Product development project is a complex system involving customer needs, product functions and components, organization and other different knowledge fields. In order to design a suitable product, this article constructs a function-product DMM, "Team-Function-Product" EDMM further analyzes the dependency relationship between the three, and further combines functional DSM and product DSM to establish an extended multi-domain matrix, and constructs a derivation model of organizational DSM. Then, in order to reduce the complexity of organization management, an improved two-stage organization clustering model based on information entropy is constructed, and the criteria for maximizing the "increased average intra-external dependence intensity ratio" and minimizing the "organization weighted information entropy" are proposed to optimize the organizational structure.

Keywords: project management; dependency structure matrix (DSM); multi-domain Matrix (MDM); clustering

1 Introduction

With the rapid development of big data information, the market demand changes rapidly and the cycle of product upgrading is greatly shortened, which poses a huge challenge to the research and development of new products (Wu et al.,2020).In order to gain competitive advantages, enterprises must constantly carry out a large number of new product research and development, and orderly management of product research and development organizations, reduce the complexity of management, so as to shorten the cycle of product research and development (Eppinger et al.,2012). R&D project is a complex system involving customer demand, product, process and organization and other fields. The coupling relationship between elements in different fields is crucial for effective management of organizational complexity and improvement of success rate of product research and development (Teodoridis,2017). Close technical communication between teams can better promote the completion of the project. Previous studies have neglected to quantify the dependency between functions and products.

This paper first constructs a “function-product” domain mapping matrix (DMM), and then constructs a “product-function-team” extended multi-domain matrix (EMDM), and thus establishes a model for deriving the organizational dependency structure matrix (DSM). Finally, a two-stage entropy clustering model is used to optimize the organization, so as to reduce the complexity of organization management, enhance the orderliness of the organization, and improve the success rate of product development.

2 Literature Review

This research mainly explores the interaction and mutual influence between different knowledge domain elements in product development projects.

Dependency Structure Matrix (DSM) is a visualization tool used to describe the dependencies between elements in a certain field of a project. At present, DSM is widely used in both static (product, function, organization) and dynamic (process) fields in the research and development of complex projects. It has become an important tool for analyzing the relationship between elements and performing visual modeling. Regarding the modeling of organizational product architecture DSM, Baldwin C et al. (2014) used graph theory, Wang R et al. (2014) used social network analysis to determine the strength of dependencies between components; Yang et al. (2020) determined the dependencies between components based on key design parameter information. On the basis of DSM and DMM, Maurer (2010) first proposed the Multi-Domain Matrix (MDM) to systematically analyze the dependency between the two domains. On this basis, this paper adopts a visual method to analyze the dependencies among the three domain elements of "team-product-function" at the same time.

Clustering is the most commonly used optimization method for or DSM. It is a method used to mine element classes with strong dependencies or small interactions in DSM, and optimize the organization into different classes according to the construction criteria (Browning,2016). In recent years, scholars have proposed many clustering methods. Qiao et al. (2019) constructed a clustering criterion based on the similarity between items; Li et al. (2012) proposed a spectral clustering algorithm based on neighborhood propagation. Yang et al. (2014) proposed a clustering criterion based on maximizing the ratio of inner and outer dependency strength of the class and an entropy-based clustering method (Yang et al.,2018). The main shortcomings of the existing clustering algorithms are that the cluster size is too small and the results are unstable. For this reason, this paper improves the two-stage entropy clustering criterion, so that the result is more stable and the scale is suitable for the actual situation.

3 The Model Formulation

3.1 Building "product-function" domain mapping matrix

In R&D projects, products are designed to achieve desired functions through integrated or modular design. Generally, each function is implemented by one or more components, and each component may also support one or more functions. Therefore, in order to describe the relationship between the two knowledge domains of product components and functions, this paper builds a "product-function" domain mapping matrix (DMM), as shown in Figure 1(a), which is an $M \times N$ matrix. The value in the matrix represents the contribution of different product components to the realization of a specific function, expressed as a percentage. The rows of the matrix indicate which functions of the product are supported by a component and the degree of influence; the columns indicate that a certain function of the product will be affected by which components and the degree of influence respectively, the sum of the columns of each function is 100%. For example, the first column of Figure 1(a) shows that the function F_1 is realized by the components C_1 and C_4, in which the influence of C_1 on F_1 accounts for 60%, and the influence of C_4 on F_1 accounts for 40%.

	F_1	F_2	F_3
C_1	0.6		0.2
C_2		0.7	
C_3		0.3	
C_4	0.4		
C_5			0.8

(a) Product-Function DMM

	F_1	F_2	F_3
F_1	F_1	2	2
F_2	1	F_2	
F_3	4		F_3

(b)Function DSM

	C_1	C_2	C_3	C_4	C_5
C_1	C_1	1		2	
C_2	1	C_2	1		5
C_3		1	C_3	3	
C_4	2		3	C_4	
C_5		5			C_5

(c)Product DSM

	F_1	F_2	F_3
C_1	a,1		e,1
C_2		c,2	
C_3		d,2	
C_4	b,2		
C_5			f,3

(d) Team-Product-Function EDMM

Figure 1. DSM and domain mapping matrix DMM

As shown in Figure 1(d), "Team-Product-Function" EDMM represents the dependency between the team and each component and function. In practice, the team is directly involved in the design of the components, so this article only considers the strength of dependence between the team and the components.

The meaning of the element (Ti,α) in EDMM: T_i-a design team; α-the strength of the dependency relationship between the team and the component.For example, the element (d, 2) of EDMM(3,2) means that team *d* is responsible for completing the design of component C_3 in implementing function F_3, and the dependency between team *d* and component C_3 is 2.

This article uses 1~3 to quantify the dependence of the team and the component, the larger the value, the stronger the dependence.

3.2 Building "team-product-function" EMDM

In R&D projects, the design team implements specific functions through the design of product components. Therefore, the intensity of technical communication between the teams is closely related to the product components and functions they are responsible for, and the dependencies between these components and functions. Previous studies constructed a multi-domain matrix to derive the organization DSM, but did not quantify the dependency between functions and components. Based on the DSM of each knowledge domain and the domain mapping matrix DMM obtained above, this paper establishes an EMDM (extended multi-domain matrix) describing the mapping relationship between the three knowledge domains of "team-product-function", and then derives the dependency relationship between teams and establishes an organizational DSM.

As shown in Figure 2(a), the "team-product-function" EMDM is composed of four parts: product DSM, function DSM, "team-product-function" EDMM and "product-function" DMM. Product DSM (Fig.1(c)) and function DSM (Fig.1(b)) respectively represent the dependencies between components and functions; "Team-Product-Function" EDMM represents the dependency relationship between the team and components and functions; "Product-Function" DMM represents the component's contribution to the realization of specific functions.

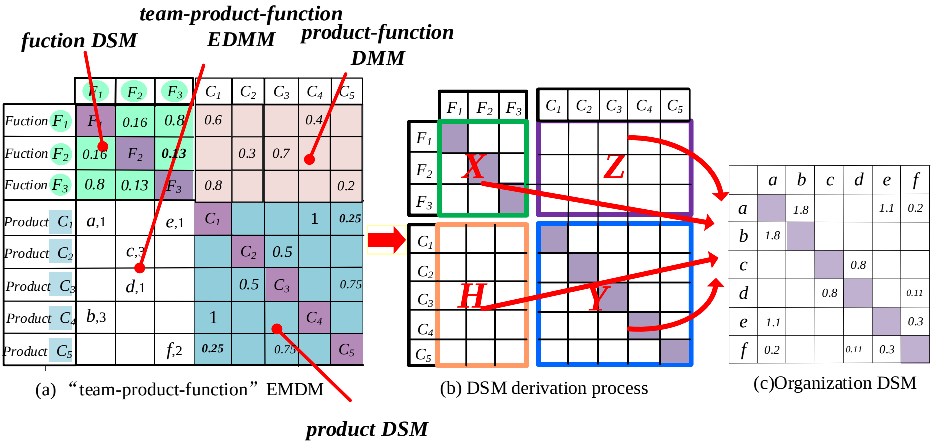

Figure 2. Deriving the organization DSM from the "team-product-function" EMDM

3.3 Measuring organization DSM by extended multi-domain matrix (EMDM)

For the purposes of this article, the organizational DSM represents the technical communication dependencies between teams/people. Figure 2(b) shows the principal process of deriving the organization DSM from EMDM. According to the literature (Yang et al., 2014), if the relationship between two knowledge domains is known and the multi-domain matrix is adopted, we can deduce the dependency relationship between the third knowledge domain. Therefore, this paper adds a "product-function" DMM to the traditional model, and then derives the basic principle model of the organizational DSM from the "team-product-function" EMDM:

$$O = H \bullet Z \bullet Y + H \bullet X \bullet Z \qquad (1)$$

Where X and Y represent DSM of two known knowledge domains (product and function), Z is "product-function" DMM, and H is "team-product-function" EDMM.

As shown in Figure 3, the dependency relationship between teams derived from the relationship between "team-product-function" can be divided into two cases, the first case (Figure 3(a)): If team i and j respectively design component C_I and C_J, there is a dependency/coupling relationship between these two components, and they implement the same function F_I, then there is a technical dependency relationship between the two teams;

the second case (Figure 3(b)): If both teams i and j participate in the design of the component C_I, which is to implement two functions F_I and F_J, and there is a dependency/coupling relationship between these two functions, and there is a technical dependency between the two teams.

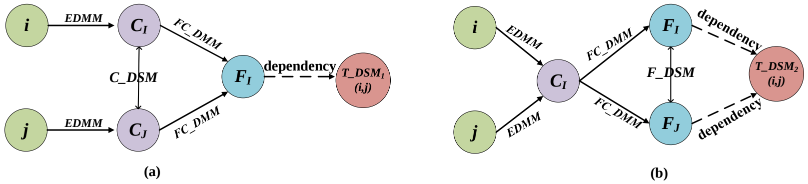

Figure 3. Deriving the inter-team dependency relationship based on the relationship between "team-product-function"

The above two situations can be derived from the "team-product-function" EMDM:

1) Observe the elements (T_i,α) in each column of "Team-Product-Function" EDMM. They indicate that in order to achieve the same function, different teams need to design different components. If there is a dependency between these components, then there is a technical dependency between these teams. For example, in Figure 5(a), in the second column of "Team-Product-Function" EDMM, teams c and d design components C_2 and C_3 respectively. It can be seen from the product DSM and product-function DMM that components C_2 and C_3 have a dependency relationship, and contributions to F_3 have different proportions. Therefore, teams c and d are dependent on each other.

Therefore, the formula for calculating the dependency between team i and j in the organizational DSM is as follows:

$$T\_DSM_1(i,j) = (EDMM(i,\alpha_i) \times FC\_DMM(F_I,C_I) + EDMM(j,\alpha_j) \times FC\_DMM(F_I,C_J)) \times P\_DSM(I,J) \quad (2)$$

Where $EDMM(i,\alpha_i)$ and $EDMM(j,\alpha_j)$ respectively represent the dependence of team i and j on component C_i and C_j; $FC\_DMM(F_I,C_I)$ and $FC\_DMM(F_I,C_J)$ respectively represent the contribution ratio of component C_i and C_j to the realization of function F_i.

2) Observe the elements (T_i,α) in each line of the "team-product-function" EDMM. If two teams participate in the design of a common component, the component will have an impact on different functions, and there is a dependency between the two functions, then there is a technical dependency between the two teams. Therefore, in the second case, the two teams design common components to achieve different functions, and build the formula for calculating the dependency between team i and j in the organizational DSM:

$$\begin{aligned} T\_DSM_2(i,j) = (&EDMM(i,\alpha_i) \times FC\_DMM\left(F_I,C_J\right) \\ &+ EDMM\left(j,\alpha_j\right) \times FC\_DMM(F_J,C_J)) \times F\_DSM(I,J) \end{aligned} \quad (3)$$

Therefore, the total strength of dependence between teams *i* and *j* is:

$$T\_DSM(i,j) = T\_DSM_1(i,j) + T\_DSM_2(i,j) \qquad (4)$$

According to the above formula, the dependence of technical information communication between teams due to the dependence of functions and components is obtained, and then the organizational DSM is finally derived.

4 Two-stage DSM clustering based on entropy

After determining the strength of dependence between teams, the clustering algorithm is further used to optimize the organizational DSM. The goal of DSM clustering is to maximize the interaction of elements in a cluster (or "group", "module"), and minimize the interaction of elements between clusters. In order to solve the complexity management problem of the organization in the context of big data, this paper adopts the two-stage clustering method based on information entropy to construct the organization DSM clustering criterion: maximizing the “increased average intra-class dependency intensity ratio”; minimizing organizational information entropy.

1) First-stage clustering criterion

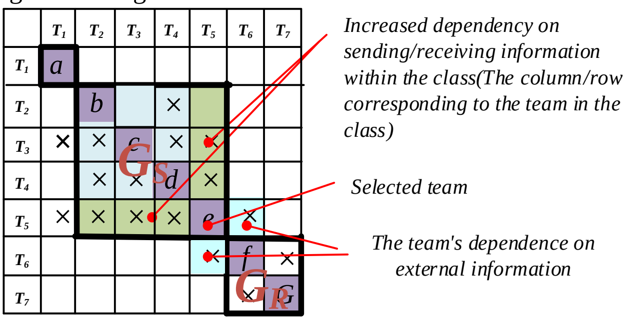

Figure 4. AAIDER clustering criterion legend

When considering the organizational DSM clustering criteria, not only must the intra-class dependency be maximized, but also the degree of influence on out-of-class factors must be considered, thereby reducing the complexity of coordination between teams. Therefore, According to this literature (Yang et al.,2018) this article proposes maximizing “Added Average Internal Dependency and External dependency Ration” (AAIDER), as shown in Figure 4, It refers to the ratio of the weighted sum of the reliance on information sent and received by the selected team to other teams in this category, and the weighted sum of the total reliance on the information sent and received by this team on all other teams in the entire organization. The larger AAIDER represents the information technology communication mainly occurs within the cluster.

Compared with the previous clustering goals, the AAIDER not only considers the influence of newly added elements within the class on the strength of the organizational dependency relationship, but also considers the degree of influence on the outside of the class, and regards the elements inside and outside the class as receivers and senders to assign weights separately. You can choose to receive information or send information, so that the organization in the clustering result is more focused on receiving or sending information, and make the dependence of the organization closer.

Therefore, the first-stage clustering criterion formula is as follows:

$$max \quad AIDER(cluster_k) = \frac{\frac{\left(\omega_1 \cdot \sum_{I=n_k}^{m_k} O\_DSM(I,m_k) + \omega_2 \cdot \sum_{J=n_k}^{m_k} O\_DSM(m_k,J)\right)^{\alpha}}{2(cl_k-1)}}{\frac{\left(\omega_1 \cdot \sum_{I=1}^{N} O\_DSM(I,m_k) + \omega_2 \cdot \sum_{J=1}^{N} O\_DSM(m_k,J)\right)^{\beta}}{2(N-1)}} \tag{5}$$

Where cl_k is the scale of the k-th class, n_k is the label of the first element of the k-th class, m_k is the last element of the k-th class, N is the total number of teams, α, β is the adjustment coefficient. ω_1, ω_2 is the weight of sending and receiving, and satisfies $\omega_1 + \omega_2 = 1$.

2) Second-stage clustering criterion

The second-stage clustering criterion is to minimize the total weighted entropy (TWE) of the organization, which is composed of external cluster entropy (ECE) and internal cluster entropy (ICE) (Yang et al.,2018). Organizational information entropy indicates the orderly degree of coordination and communication of teams within a class (outside the class), which is mainly related to the probability of information exchange in the class team. First of all, this paper constructs a probability model for information exchange between groups (within groups). The inter-group probability model mainly considers the inter-class dependencies and reduces the impact of intra-class dependencies, so that there are as few out-of-class points of the clustering results as possible, and information communication between groups is reduced.

The inter-group probability $p(G_S, G_R)$ refers to the ratio of the strength of the coordination dependence $TS(G_S, G_R)$ between the group G_S and the group G_R to the strength of the dependence relationship between the related groups in the entire system (G_S and other groups G_K in the system):

$$p(G_S, G_R) = \frac{TS(G_S, G_R)}{\sum_{K=1}^{M} TS(G_S, G_K)} = \frac{\sum_{i=n_R}^{m_R} \sum_{j=n_S}^{m_S} DSM(i,j)}{\sum_{j=n_S}^{m_S} \left(\sum_{i=1}^{n_S} DSM(i,j) + \sum_{i=m_S}^{n} DSM(i,j)\right)} \tag{6}$$

Where $DSM(i,j)$ is the strength of the dependency relationship between team i and team j, $n_s$and m_s are the number of tags from the first element to the last element in the group, $n_R$and m_R are the number of tags from the first element to the last element in the group, and n is the total number of teams.

In the same way, $p(G_S)$ represents the information exchange probability of the team in the cluster G_S, and the calculation formula is as follows:

$$p(G_S) = \frac{TS(G_S,G_S)}{\sum_{K=1}^{N} TS(G_S,G_K)} = \frac{\sum_{i=n_R}^{m_R} \sum_{j=n_S}^{m_S} DSM(i,j)}{\sum_{i=1}^{n} \sum_{j=n_S}^{m_S} DSM(i,j)} \tag{7}$$

Furthermore, the information entropy between group G_S and group G_R is solved by equation (8):

$$\begin{aligned} E_{external}(G_S,G_R) = -\Big(p(G_S,G_R)\log p(G_S,G_R) + \big(1-p(G_S,G_R)\big)\log\big(1-p(G_S,G_R)\big)\Big) \\ \times Z_i\left(G_S,G_R\right)^{\lambda} \end{aligned} \tag{8}$$

Where $Z_i(G_S, G_R)$ represents the number of non-zero elements of class G_S and G_R,M is the number of classes and λ is a penalty coefficient, which is used to adjust the degree of influence of class size on information entropy.

Further, the information entropy of the out-of-class organization can be calculated by Equation (11):

$$ECE = -\sum_{S=1}^{M}\sum_{R=1}^{M} E_{external}(G_S,G_R) \qquad (S \neq R) \tag{9}$$

Similarly, the organizational information entropy ICE within the class refers to the order degree of team coordination and communication within the class G_S, which can be calculated by Equations (13) and (14) :

$$E_{internal}(G_S) = -\sum_{S=1}^{M}\Big(p(G_S)\log p(G_S) + \big(1-p(G_S)\big)\log\big(1-p(G_S)\big)\Big) \times Z_e\left(G_S\right)^{\mu} \tag{10}$$

$$ICE = \sum_{S=1}^{M} E_{internal}(G_S) \tag{11}$$

Where, $Z_e(G_S)$ represents the number of zero elements in class G_S, and μ is the adjustment coefficient.

Finally, this paper establishes the second-stage clustering criterion goal of organizational DSM based on information entropy: to minimize organizational information entropy.

$$minE = \varphi_1 ECE + \varphi_2 ICE \tag{12}$$

Where φ_1 and φ_2 are weighting coefficients, and $\varphi_1+\varphi_2=1$.

5 Conclusion and Outlook

The constantly changing environment faced by current R&D projects is embodied in a complex system composed of different fields such as products, functions, and organizational processes. Therefore, this paper constructs the product-component correlation matrix DMM and adds it to the traditional multi-domain matrix MDM, and finally builds a model that calculates the strength of the organizational dependency relationship from the dependency relationship between team-function-component.

Furthermore, in order to reduce the complexity of the product R&D project system structure, an optimized two-stage clustering criterion based on entropy is proposed to make the clustering effect more stable. In project management, the use of the clustering method in this article can better deal with complex and changeable organization management, the clustering effect is more stable, and it provides a more direct method for reducing organization management.

In the future, in-depth research can be carried out from the following aspects: combining big data to further establish a more complete calculation model that depends on the intensity value in the context of big data. In addition, the complexity measure of entropy organization and the clustering method are worth further study.

Acknowledgments

This study was supported by the National Natural Science Foundation of China(No.71872011 and No.71929101)

References

Wu L, Hitt L, Lou B.,2020. Data Analytics, Innovation, and Firm Productivity. Management Science, 2020, 66(5), 1783-1785.

Eppinger, S. D. , & Browning, T. R., 2012. Design structure matrix methods and applications. Mit Press Books, 1.

Teodoridis F.,2017.Understanding Team Knowledge Production: The Interrelated Roles of Technology and Expertise. Management Science, 2017, 64(8).

Baldwin, C. , Maccormack, A. , & Rusnak, J., 2014. Hidden structure: using network methods to map system architecture. Research Policy, 43(8), 1381-1397.

Wang, R. , Huang, R. , & Qu, B., 2014. Network-based analysis of software change propagation. The Scientific World Journal, 2014, 237243.

Yang, Q., Yang, N., Browning, T. R., Jiang, B., Yao, T., 2020. Clustering product development project organization from the perspective of social network analysis. IEEE Transactions on Engineering Management, DOI: 10.1109/TEM.2019.2939398.

Maurer, M. , & Kesper, H., 2010. Knowledge transfer applying the structural Complexity Management approach. International Conference on Information Retrieval & Knowledge Management. IEEE.

Browning, T. R., 2016. Design structure matrix extensions and innovations: a survey and new opportunities. IEEE Transactions on Engineering Management, 63(1), 1-26.

Yu, Qiao, Jon, D., Fricker, & Sa mu el, et al., 2019. Quantifying the similarity between different project types based on their pay item compositions: application to bundling. Journal of Construction Engineering and Management, 145(9).

Li, X. Y., & Guo, L. J., 2012. Constructing affinity matrix in spectral clustering based on neighbor propagation. Neurocomputing, 97, 125-130.

Yang, Q., Yao, T., Lu, T., Zhang, B., 2014. An overlapping-based design structure matrix for measuring interaction strength and clustering analysis in product development project. IEEE Transactions on Engineering Management, 61(1):159–170.

Yang, Q. , Shan, C. , Jiang, B. , Yang, N. , & Yao, T., 2018. Managing the complexity of new product development project from the perspectives of customer needs and entropy. Concurrent Engineering, 26(4), 1063293X1879800.

Yang Q, Lü J R, KherbachI S. Progress in modeling and optimization for complex R&D project based on design structure matrix[J]. Systems Engineering — Theory & Practice, 2016, 36(4): 989–1002.

Contact: Qing Yang, University of Science & Technology Beijing, School of Economics and Management, No.30 Xueyuan Road, Haidian District, Beijing 100083, China, 86+13911463353, yangqing@manage.ustb.edu.cn. http://dsem.ustb.edu.cn/jgen/a/Faculfy_List/20141009/124.html