Research on Schedule Management of Information System Development Projects Based on Improved Critical Chain Management Method

Yuhao Wang[1,a], Wenqi Jiang*[,2,b], Shuqun Wang[3,c], Nina Gao[4,d]

{youhasir@163.com[a], 595125353@qq.com[b], wangshuqun_301@163.com[c], 33741395@qq.com[d]}

Nanjing University of Science and Technology, Nanjing , China[1,2]
Shanghai Aerospace Equipments Manufacturer Co., Ltd. , Shanghai , China[3,4]

Abstract. Previous studies on project schedule management were mainly concentrated on construction or manufacturing projects. Information system development projects, given their differences and uniqueness, do not show significant outcomes when conventional project management techniques are directly applied. Therefore, this study aims to improve and optimize the Critical Chain Project Management (CCPM), providing reference cases and methodological insights for schedule management of information system development projects. The study proposes using Company Y as the subject, applying a workload estimation model based on functional size for regression analysis of Company Y's information system development projects, and then, using a dynamic weight model to investigate the optimal allocation scheme for constrained resources. By studying the performance of the CCPM before and after improvement in dealing with resource conflict issues, it is first concluded that the improved critical chain management method can quickly complete resource allocation; secondly, it results in the shortest planned project duration; thirdly, the improved method can dynamically adjust the resource allocation plan according to the project execution situation. The results indicate that the CCPM is insufficient for the needs of information system development projects, while the improved CCPM method is more in line with the characteristics of such projects.

Keywords: Project schedule management; Critical chain project management method; Workload estimation; Resource scheduling;

1 Introduction

In recent years, project schedule management, as a critical component of project management, has attracted widespread attention from all sectors of society and various enterprises, and has become a focal point for numerous scholars. The Project Management Institute (PMI) defines project schedule management in the PMBOK guidelines as the management of the progress of various stages and the final completion deadline throughout the project implementation process. According to literature [1], the objective of project schedule management is to accomplish engineering tasks with the highest quality, in the shortest duration, and with minimal capital investment to secure the maximum economic return. It is evident that effective project schedule management can directly improve the quality of project and reduce project engineering costs for enterprises. Past research on schedule management has been mainly concentrated on manufacturing or construction projects. However, with the advent of the information age, the

PMIS 2024, March 15-17, Changsha, People's Republic of China

DOI 10.4108/eai.15-3-2024.2346384

number of information technology companies in our country has increased exponentially. According to the fourth national economic census by the National Bureau of Statistics, the internet and related service industries are among the fastest-growing sectors in national economic development, with the number of corporate entities increasing by 3.8 times compared to five years ago. This has also driven the continuous enhancement of informatization in the production and operation of various enterprises, with 84.4% achieving informatization of financial management, 43.8% of procurement, sales, and inventory management, and 32.8% of human resource management. Although information technology companies have provided strong momentum for high-quality development in our country, the majority of these companies are newly established and have not yet formed effective management systems, relying too heavily on the experience of management personnel, which has led to relatively inefficient project schedule management in information technology enterprises. To better promote the vigorous development of the information technology industry, it is particularly important to do well in the schedule management of information system construction projects, and clarifying the current situation and characteristics of schedule management in these projects is an important prerequisite for achieving this goal.

An increasing number of scholars are beginning to explore issues surrounding schedule management in information system construction projects, covering various aspects. The early research primarily focused on the characteristics inherent to information system construction projects themselves, Literature [2] notes that the production factors required in information system development projects include: (1) knowledge-based employees, such as programmers; (2) tools, such as desktop or laptop computers; (3) a workspace with minimal space requirements; (4) a certain amount of capital for employee remuneration. Compared with construction and manufacturing projects, information system development projects require very little in terms of capital and raw materials, but have a substantial demand for human resources. Literature [3] focuses on refining the current practices of schedule management in software projects, optimizing project plans while bolstering employee management to significantly enhance the execution efficiency of project schedules.

With the deepening of research, scholars are increasingly concentrating on enhancing project timelines using diverse project management tools. Literature [4] has improved the estimation method for the duration of IT project activities, allowing for more precise and rational predictions of activity durations. Literature[5] focuses on agile methodologies to construct a software project schedule management system, which has been proven to be more efficient in corporate project examples. Meanwhile, although many scholars have explored schedule management of information system development projects based on the critical chain method, their focus has primarily been on applying traditional critical chain management techniques to optimize major client projects within corporations [6], constructing schedule management models for road and bridge construction projects by improving buffer settings of the gray critical chain [7], and several other areas such as the construction of 5G base stations, where buffers are established based on critical paths with resource constraints [8].

According to existing literature, research on project schedule management and its control methods is mainly focused on the construction and engineering manufacturing industries, often using construction or manufacturing projects as subjects of examination, however, there is a scarcity of systematic research on the application and enhancement of the critical chain method in actual project schedule management within IT enterprises. The critical chain management method, formulated by Dr. Eliyahu M. Goldratt [9], merges the theory of constraints with the critical path method, incorporating the interdependencies of tasks and resource constraints within project schedule management. Literature [10] also outlines the definition of the critical chain management method, suggesting that it can address the adverse effects of multitasking, “student syndrome,” “Parkinson’s Law,” and ineffective organizational incentives.

The study uses Y Company, a domestic information technology firm, as the research subject, building on the principles of Critical Chain Project Management (CCPM), initially using historical project data from Y Company to develop a workload estimation model, and proceeds to estimate the duration of activities within information system development projects. Subsequently, it employs a dynamic weight model to identify key indicators for resource allocation amidst project resource conflicts. The research findings obtained offer essential guidance for Y Company and similar businesses in managing project schedules.

2 Research methods

2.1 Workload estimation model based on project functional size and variable selection

The subject of this paper is information system development projects, which differ from projects in traditional industries such as construction and manufacturing, in that the functional size of each stage and module in software projects is a quantifiable indicator. The functional size can be measured based on requirements using the Full Function Point measurement method introduced by the Common Software Measurement International Consortium (COSMIC), and the measured functional size is an important factor affecting project workload. Traditional methods for estimating the duration of project activities, such as expert judgment and PERT, typically do not record which indicators were used during the estimation process, hence it is impossible to trace the historical data behind the expert experience models, and it is also impossible to evaluate the performance of these expert experience models, because the key project indicators cannot be quantified and standardized objectively. Therefore, in order to make the estimation results for the duration of project activities more precise, this paper adopts a workload estimation model based on project functional size to estimate the man-hour for information system development projects.

In the detailed analysis, the software functional size is considered the independent variable, with the project workload serving as the dependent variable, and the least squares method is applied to estimate the constant term β_0 and regression coefficient β_1, resulting in the estimated values ^β_0 and ^β_1. Thereby constructing a regression model for project workload in relation to the functional size of the information system and defining the coefficient of correlation for the linear regression:

$$\hat{Y} = \widehat{\beta_0} + \widehat{\beta_1} * x \ . \tag{1}$$

$$\widehat{\beta_0} = \frac{\sum(x_i - \bar{x})y_i}{\sum(x_i - \bar{x})^2} \ . \tag{2}$$

$$\widehat{\beta_0} = \bar{y} - \widehat{\beta_1}\bar{x} \ . \tag{3}$$

$$R^2 = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2} \ . \tag{4}$$

In equation (1), the independent variable is represented by x, and for each specific value of x, there is a corresponding dependent variable ^Y, while the estimated values of β_0 and β_1 are derived from a historical project dataset. In equations (2) and (3), x represents the functional size of the information system, which is measured using the COSMIC method, with the unit of functional size being CFP. y represents the project workload, that is, the total number of man-hours spent to complete the project, with the unit being man-hours. In equation (4), R^2 indicates the linear relationship between the two variables in the regression equation, and when $R^2 > 0.7$, it suggests a strong correlation between the functional size of the information system and the project workload, otherwise, the correlation is weak.

2.2 Dynamic weight model

After clarifying the methods for estimating workload in information system development projects, in order to ensure that the project progresses as planned, it is necessary to discuss resource management surrounding information system development projects. Resource management is the core of Critical Chain Project Management (CCPM), and if multiple project activities simultaneously require the same resource, but the quantity of available resources is less than needed, a resource conflict arises. To design a scientifically sound algorithm that, in the event of resource conflicts, can quickly determine which project activity has a greater impact on project progress, thus allowing the allocation of limited resources to the more urgent project activities. Specifically, this study uses the weight values of project activities as the dependent variable, with the main influencing factors being the degree to which project activities affect the schedule, whereby a delay in project activities on the critical path directly affects project advancement, while activities not on the critical path have a certain degree of buffer time, thus the weight values are contingent upon the activity's position within the entire project pathway. The basic form of the model is as follows:

$$\begin{cases} Q = \frac{F_i + H_i}{G_i}, \text{If the project activity i is on a non critical chain} \\ Q = 1, \text{others} \end{cases} \quad . \tag{5}$$

In equation (5), F represents the total remaining man-hours of unfinished project activities on the non-critical chain path where the project activity i is located, H represents the feeding buffer time at the end of the non-critical chain path where the project activity i is located, G represents the total execution time of all project activities on the critical chain from the currently executing project activity to the convergence point where the non-critical chain merges with the critical chain at the project activity i. The larger the weight value Q obtained in the end, the higher the priority of that path, and the greater the likelihood that it will impact the overall progress of the project when delays occur, thus necessitating the prioritization of resource allocation.

2.3 Data acquisition and cleaning

Original data on the functional size and workload of information system development projects are not currently tracked by official statistics and have been minimally covered in existing research, and if the data source is from an external dataset, there is a possibility that it does not align with the actual situation of the enterprise. For this reason, this paper will collect historical project data from Company Y, and in order to improve the accuracy of model estimation, this paper will also clean and preprocess the collected data.

This paper randomly selects 20 historical project datasets from Company Y's software engineering database as samples, and initially calculates the functional size and actual workload of each project based on the sample project datasets, as shown in Table 1. By graphically analyzing these data, a scatter plot of Company Y's project size versus actual workload can be obtained, as depicted in Figure 1.

Table 1. Data set of actual project workload and functional size

Number	Functional Size(CFP)	Workload(h)	Number	Functional Size(CFP)	Workload(h)
1	215	625	11	141	316
2	88	151	12	131	295
3	122	289	13	206	451
4	683	1390	14	101	336
5	160	453	15	618	1356
6	170	533	16	172	447
7	191	498	17	213	513
8	201	546	18	237	669
9	222	988	19	189	520
10	85	385	20	150	404

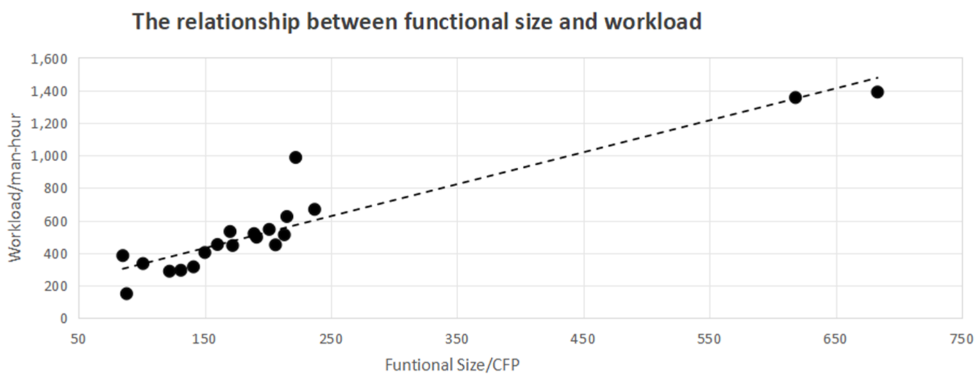

Figure 1. Scatter plot of functional size and workload

Since software engineering databases often contain a large number of routine projects, along with a few large and very large projects, the randomly drawn sample sets generally do not follow a normal distribution, and a non-normal distribution implies that the model is less representative in areas with fewer data points. Therefore, points in the input sample that significantly deviate from the overall mean of this data set are identified as outliers, and the Grubbs test is used to analyze which data points are outliers. When the P-value of the Grubbs test is less than 0.05, the sample point is defined as an outlier. Grubbs test applied to the data in Table 1 and Figure 1 reveals that the first variable, being the functional size for Project 4 and Project 15, at 683 CFP and 618 CFP, respectively, are noticeably higher than the rest, which is over 3 standard deviations above the mean of 214 CFP, suggesting that these two projects can be treated as outliers within the scope of this research data. For the second variable, the actual workload, Grubbs test indicates that Project 4 (1390 man-hours), Project 9 (988 man-hours), and Project 15 (1356 man-hours) are quite distant from the other sample points, exceeding the average of 558 man-hours by more than 2 standard deviations, are also treated as outliers in the data for this study.

After excluding these three sample points, the average values of functional size and workload significantly decreased, with the average workload dropping from 558 man-hours to 437 man-hours, and the average functional size decreasing from 214 CFP to 163 CFP. After the removal of these three outliers, the distribution of both variables became closer to a normal distribution, as shown in Table 2. The Shapiro-Wilk test conducted using SPSS software yielded a P-value above the threshold of 0.05, therefore, it can be assumed that the variables conform to a normal distribution.

Table 2. Shapiro Wilk test for normal distribution

Variable	N	W	P
Functional size	17	0.957	0.581
workload	17	0.980	0.954

Furthermore, through the review of relevant literature and the investigation of the project execution at Company Y, it is found that software projects require the coordination and allocation of substantial resources during the development phase, which is most prone to issues of resource conflict[11]. Therefore, this study focuses on the research of the software development phase of Company Y's M Information System development project, collecting data on the dependencies between project activities and the resources required for the execution of these activities, with the specific project activities involved shown in Table 3.

Table 3. Information related to Y Company's M Information System Development Project

No.	Project Activities	Immediate Predecessor	Functional Size(CFP)	Required resources
1	Software design document writing	Null	6	Project manager
2	Software design review and modification	1	5	Project manager
3	Development of smart contract module	2	16	Developer
4	Development of internal and external data channel module	2	14	Developer
5	Development of internal data upload module	2	9	Developer
6	Server underlying environment preparation	2	14	Operation Engineer
7	Test Case Writing and Execution	3	12	Test engineer

8	Module security testing	4	12	Information security engineer
9	Fix testing issues and write testing reports	7	6	Test engineer
10	Online document writing and review	5，8，9	8	Project manager
11	Hardware security testing	6	12	Information security engineer
12	Deployment system	10，11	4	Operation Engineer

3 Application of Improving Critical Chain Management Method

3.1 Project workload estimation

Based on the equation constructed in the previous text, using the data set of Company Y’s project functional size and workload after the exclusion of outliers, a workload estimation model based on project functional size can be constructed using SPSS software, as shown in equation (6). The coefficient of determination $R^2 = 0.743 > 0.7$, indicating a strong correlation between functional size and project workload.

$$\hat{Y} = 2.481 * x + 32.63 \quad . \tag{6}$$

Specifically for Company Y’s M project, the workload can be calculated by inputting the functional size of each project activity into the model. Furthermore, by applying the critical chain management method to set project buffers and feeding buffers, the project’s single-number network diagram is as shown in Figure 2, with feeding buffer FB1 being 16 man-hours, feeding buffer FB2 being 8 man-hours, feeding buffer FB3 being 16 man-hours, and the project buffer PB being 20 man-hours.

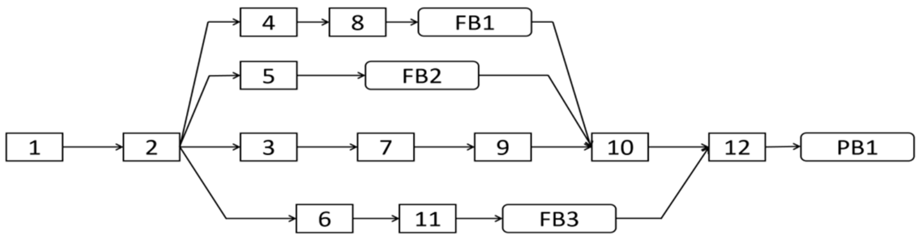

Figure 2. Project single code network diagram

3.2 Determine the weight values of project activities

The dynamic weight model is capable of dynamically tracking the weight values of each project activity during the project’s execution, in accordance with the actual progress of the project, and it can also make preliminary estimates of the weight values of each project activity during the planning phase, based on the test results of the workload estimation model, thus identifying potential resource conflicts within the project in advance.

The dynamic weight model has been employed in Company Y's development project for the M information system, involving the initial step of applying the dynamic weight model in the project planning phase to perform the first assessment of the weight values for each activity, with the outcomes of this assessment displayed in Table 4. These calculated weight values may change during the actual execution of the project, and are used only to address resource conflicts that arise during the project planning phase. Secondly, during the project execution, there is the possibility that project activities may finish ahead of schedule or be delayed. Therefore, when unplanned resource contention issues arise from the planning stage, the dynamic weight model can immediately compute the most recent weight values, effectuating a dynamic reallocation of resources.

Table 4. Corresponding Table of Weight Values for Project Activities

project activity number	1、2、3、7、9、10、12	4	8	6	11	5
weight values	1	0.94	0.89	0.76	0.64	0.27

4 Result analysis

The previous chapter applied an improved critical chain management method to estimate the workload and determine the weight values for Company Y's M information system development project. On this basis, this chapter compares the effects of schedule control between the traditional and the improved critical chain management methods, as indicated in Table 5. As known from the previous text, project activities 3, 4, and 5 all belong to the development of functional modules and must be completed by developers, project activities 7 and 9 are all related to testing work, which needs to be done by test engineers, and project activities 8 and 11 pertain to security checks, which need to be carried out by information security engineers. Given the dependency between project activities 7 and 9, they cannot be carried out at the same time, eliminating the possibility of resource conflicts, other project activities may concurrently utilize project resources. If there are no constraints on project resources, no resource conflict issues will occur, nor would it be easy to reflect the advantages and disadvantages between different schedule management methods. Therefore, this study imposes reasonable constraints on the resources for Company Y's M information system development project, limiting the number of developers within the project team to 2, and the number of information security engineers to 1, which also conforms to the actual situation of Company Y's project team's human resources.

From the perspectives of project planning and execution phases, in the planning phase, resource conflicts occur in project activities 3, 4, 5, 8, and 11. The traditional critical chain project management method would prioritize resource allocation to project activity 3, which is on the critical chain. For activities 4, 5, 8, and 11, which are not on the critical chain, traditional CCPM can only list all possibilities through exhaustive enumeration, and The computational load for resolving resource conflicts increases exponentially as the frequency of conflicts rises. If resources for activities not on the critical chain are allocated randomly to reduce computation, the project duration could potentially extend to a maximum of 194 hours. Meanwhile, critical chain management based on a dynamic weight model accomplishes resource allocation for activities 4, 5, 8, and 11 by directly comparing their activity weight values. Through comparison with unconstrained resource scenarios, it is evident that the improved critical chain management method can maintain the project duration at 180 hours under resource constraints, which is at the same level as the unconstrained situation of project resources. Therefore, in the project planning stage, the improved CCPM is clearly superior to the traditional method in both efficiency and effectiveness

In the execution phase of the project, deviations from the planned schedule, either delays or accelerations, lead to unforeseen resource conflicts. The improved critical chain management method can use a dynamic weight model to calculate in real time the weight values of project activities competing for resources. Based on the weight values, the resource allocation plan is dynamically adjusted to allocate resources to the project activities that have a greater impact on the overall project plan.

Table 5. Comparison of project schedule management methods and their effectiveness

Project schedule control methods	Project resource allocation plan	Project Constraint Resources	Priority Project Activity Number	Project planning cycle
Comparison data	Unrestricted resources	3 Developers	3,4,5	158hours
		2 Information security engineers	8,11	
Critical chain management method	Priority for critical chain	2 Developers	3,5	172hours
		1 Information security engineers	8	
	Priority for critical chain	2 Developers	3,5	169hours
		1 Information security engineers	11	
	Priority for critical chain	2 Developers	3,4	169hours
		1 Information security engineers	11	
	Priority for critical chain	2 Developers	3,4	158hours
		1 Information security engineers	8	
Improved critical chain management method	Priority for high weight values	2 Developers	3,4	158hours
		1 Information security engineers	8	

5 Conclusions

This study takes the project schedule management of Company Y as the research subject and investigates improvements in Critical Chain Project Management (CCPM) with respect to workload estimation and resource management. This serves as a supplement to the research system on schedule management issues in information system development projects, and it also provides a reference for Company Y to achieve more effective project schedule control. The research results show that the improved CCPM can allocate project resources more rapidly and accurately than the traditional CCPM under the same resource constraints, and it enables the formulation of the optimal project plan and the realization of project schedule control.

Compared to traditional critical chain management, this study improves the project workload estimation method by estimating the duration of project activities based on historical project data of the enterprise, which is more scientific and objective than the traditional methods like expert judgment and PERT that rely on the expert's experience. On the other hand, in terms of resource management, a dynamic weight model has been constructed that calculates the weight values of project activities in real-time based on the situation of resource conflicts, thus achieving dynamic adjustment of resource allocation during the project execution phase. This study also has certain limitations, mainly reflected in the following: firstly, due to the limited retention of historical project data by Company Y, workload estimation is solely based on the index of project feature size, which may lead to poor correlation between the project's functional size and workload, thereby causing deviations between the planned project cycle and actual circumstances. Secondly, the absence of effective simulation tools makes it challenging to replicate potential resource conflicts that could emerge in actual projects. Consequently, the capability of the dynamic weight model to allocate resources dynamically can only be demonstrated through scenario simulations. There is an urgent need to broaden the sources of data to guarantee more objective and reasoned analysis outcomes. Furthermore, there could be efforts to implement the refined critical chain management approach to the schedule management of multiple projects, particularly those with resource constraints.

References

[1] M. Wang, "Application of BIM Technology to project management objectives," 2021 2nd International Conference on Information Science and Education (ICISE-IE), Chongqing, China, 2021, pp. 1236-1240, doi: 10.1109/ICISE-IE53922.2021.00277.

[2] XueMei Xiang. Research on Multi project Human Resource Scheduling in Chinese Software Development Enterprises[D].Shandong University,2006.

[3] Wei Wenwen Research on Optimization of Software Project Schedule Management in L Company [D]. Beijing University of Chemical Technology, 2023. DOI: 10.26939/dcnki.gbhgu.2023.001692

[4] Przemyslaw Korytkowski, Bartlomiej Malachowski,Competence-based estimation of activity duration in IT projects,European Journal of Operational Research, Volume 275, Issue 2,2019,Pages 708-720

[5] Guo Shuai Research on Optimization of Software Development Project Management in Company H Based on Agile Methods [D]. China University of Mining and Technology, 2023. DOI: 10.27623/d.cnki. gzkyu. 2023.002896

[6] Eliyahu M Goldratt. Critical Chain[M]. Taylor and Francis:2017-10-03

[7] George Ellis. Chapter 6 - Critical Chain Project Management(CCPM)[M].Elsevier Inc.:2016-06-15.

[8] DeFen Yuan. Research on Progress Management of A Company's Construction Bank Key Customer Dedicated Line Project Based on Critical Chain Technology [D]. Beijing University of Chemical Technology,2022.DOI:10.26939/d.cnki.gbhgu.2022.002174.

[9] Zhang Luping Research on Progress Management of Cross Railway Bridge Projects Based on Improved Critical Chain Method [D]. Lanzhou Jiaotong University, 2023. DOI: 10.27205/d. cnki. gltec.2023.000773
[10] Jie He. Research on Progress Management of 5G Base Station Projects Based on Critical Chain Technology [D]. Henan University,2021.DOI:10.27114/d.cnki.ghnau.2021.002005.
[11] Souza A F ,Robson L ,Victória L J B .A risk prediction model for software project management based on similarity analysis of context histories[J].Information and Software Technology,2021,131106497-.