# Developing and Applying

# Management Project Payment Schedule Algorithm using Genetic Algorithm

# and Nash Equilibrium

Huỳnh Quyết Thắng, Trịnh Bảo Ngọc, Bùi Đức Hưng and Lê Tuấn Dũng

2nd EAI International Conference on Nature of Computation and Communication

March 17–18, 2016, Rach Gia, Vietnam

thanghq@soict@hust.edu.vn, ngoctb@hanu.edu.vn, buiduchung.bkhn@gmail.com,

**Abstract** – To schedule a suitable payment list for each phase of ​​work in the project, including the satisfactory of both the investors and the developers, payment and payee receives money on time that both side found the moment where both sides have seen fit to payments. The project was described by the method of networking projects with nodes with direction, the model of network project will propose a best strategy is a list with the order of payment determined. Using genetic algorithms into this network project, the strategy of payments will be gradually improved through the new generation , and after the specific number of times running, the ultimate solution , which be regarded as an equilibrium of problem by using the Nash equilibrium to find the balance point for an overall payment schedule on time , is a reasonable plan . To find the Nash equilibrium, this article propose using game theory to solve the problem of the two players have complete information and the points of conflict.

**Keywords** – the Nash equilibrium, game theory, payment schedule, project network, genetic algorithm.

#### Introduction

The process of decision - making in the scheduling project payment is a difficult problem in project management, even for financial experts with a lot of experience almost do not have an appropriate solution to compute accurately cost and profit of the project. There have been several studies on the issue of payment of the project but the study was execute based on the conditions relating to the payment in order to find the optimal scheme. But in reality, the payer is the owner and the beneficiary is the developer or project team is always conflict with each other in terms of time to pay, the pay tends to pay as late as possible and with the beneficiary, in contrast, want to receive as soon as possible. Of course, in a professional project, when people have a late payment, they must pay additional interest rates for late payments. The model of a project payment will normally be stipulated by investor, some specified milestone in projects must be completed with payment conditions in a number of milestones, and the agreement is generally agreed in the contract. Based on these scheduling project payment, the developer needs to propose an activities schedule for their projects.

Models project billing which have a conflict agree with the definition of game theory, and may see this problem with two players related to many of the tasks in the project. The purpose of this study aims at modeling problems in project payments towards settlement of the genetic algorithm to find a Nash equilibrium for the payment of both parties, who pays and who received money is satisfied.

#### Methodology

Network project is an application method used game theory to determine the network path in the task direction from the inception of the project until the end of the project, through the work and relationships between this works. A network project must clearly identify the priority of all operations involved in the project. Obviously a project network needs to be expressed in the form of directed graph without cycles (DAG-Direct Cyclic Graph). There are two types of common networking project is AON (Activity-on-Node) and AoA (Activity-on-Arc).

A network AoA G = (E, A) is a graph which set of nodes E = {, ... } corresponds to the set of events in the project, and set A = {, ... } corresponds to a set of activities. Where m is the number of events and n is the number of tasks in the project. Based on the binding AoA network's priority activities , we number for activities with different priority levels consistent with project , activities with higher priority( or activity with the smaller indexes) , will be executed as soon as possible . These activities will be performed in the nodes of graph, and only if developer team completed the task of a node then they are able to move to the next node in the network

The problem in coordinating the project payment schedule is represented by the state node in the network project. Assuming the payment terms of the contract of the project is implemented in different stages and the payment of the project is implemented in a network node of an operation which will be completed from one or more activities. The project will be started from the first node status and will end at the final status button. Each task involved in the project needs the participation of one or more resources. So in making decisions project payment schedule, the protocol ideal for investors - as well as the payment - is the amount payable only once for all projects as projects finish. In this case, the net present value of cash flow is minimal.

However that the best protocol for developers - payee - the developers get all the payments at the start of the project. In this case, the net present value of cash flows of the payee is maximized. Clearly, in most cases, any profit plan also difficult to meet the requirements of both parties to accept and the compromises needs to be considered. A more realistic scenario is in the process of this agreement, the investors make a schedule of payments to the initial payment to the developer, and then the developer should consider it and make the corresponding activities schedule appropriate with payment schedule. So the problem of optimizing the project payment schedule can be modeled into a form of games with complete information is represented by the form of the strategy are as follows:

*G* = *{,* , *,*  *}*

Where:

* : Set of strategies of investor
* : The difference in costs compared with the optimal investment case of investor
* : Set of strategies of client or developer
* : Net benefit of the client

To build a project progress payments, the payer must specify a list of payment:

**Payment list:** (*,, ...,*) **(1)**

Where is the percentage of budget that the owner intends to pay to the client in the event i. List of payment must satisfy constraints:

* *= 1* **(2)**

Before we determine the costs and benefits function of investors and developers, we need to define the concept of discount rate. As follows, after a certain time period t that the value of the cash will be changed, including inflation, exchange rate fluctuations and opportunity costs. Thus, it is clear to see that the value of the funds will be reduced after a certain period of time because the money should be invested and yield a profit rather than taking it away and not used. So the discount rate would help us calculate the attenuation value for money due to the opportunity cost caused as follows:

at

Where is the net benefit, is the sales and is the cost for the stage of project in a period of time . So to compute the net benefit in , the start time of project, we need to use discount rate to specify a reduction in value of cash flow as follows:

at

Where r is discount rate and is the net benefit in compared to the benefit in . And we generally choose = 0, so we have:

Thus, if the whole project is divided into several stages, we have the net benefit as follows:

Because r << 1 so we can use Taylor series of to approximate this series:

Where is the benefit function and is the cost function

So the payoff – function is:

F =

**With investor:**

According to the formula above, we see that for investors, the best method of pay is paid after the project has finished. Because the value of is biggest, where is time to complete the entire project. So the minimum value of project with investor at =0 is:

Thus, we have the difference in costs compared with the optimal investment case of investor or as follows:

Hence, we need to minimize to reduce the cost that investor must paid.

**With client:**

As for the developers, from the formula above, it is easy to see that they want to be paid as soon as possible to maximize the net profit which they can be obtained. So we have the formula for calculating the net profit of the developer as follows:

For this reason, to meet the needs of developers, we need to maximize the value of the function .

So to find an optimal plan satisfies the requirements of both sides we use genetic algorithms to perform the search for optimal results, thereby achieving Nash equilibrium for the game including 2 people investors and developers.

#### Application: Optimization using GA

**General Model:**

In optimizing the payment of the project, both progress payments and project progress also needs to be optimized. To perform the optimal decision variables, we designed a solution to consider this issue:

* Payment list: (,, ..,)
* Activity list: ()

To achieve simultaneous desired requirements above, we use the following cost function:

Because so achieve a maximum value when reach its maximum value and reach its minimum value. I multiply by 2 to ensure that are not suppressed when compute the adaptive function. Thus, we have:

=

Hence, we need to find the schedule project payment and activities to maximum , and it is our fitness function.

**Model chromosomes:**

For simplicity, we consider the project includes 10 nodes for example, so to be able simultaneously to find an optimal result for both of two side, I will build a chromosome consists of 20 genes, including 10 genes first percentage of budget for each phase of the project, 10 next gen is the amount of work to do at each stage of the project for the development team, of course, the work to do here was to ensure order priorities, through numbering priority for the activity. But one principle always must be guaranteed for the chromosomes, that the total value of the first 10 genes equals 100 and the total value of last 10 genes equals the total number of tasks to be done for development team. For example:

It means, the development team have 20 tasks (total of last 10 genes) to complete this project with the payment schedule as array above. Concretely, we have:

* Owner: = 5% of budget, = 5% of budget, = 15% of budget, = 10% of budget and so on.
* Activity list: There are 20 tasks. In stage 0 (or node 0 in node graph), the development team have to do set of activities which have only 1 task, thus they have to complete task1. In stage 1 ( or node 1 in node graph ) , the development team have to do set of activities which have 3 task , so they have to complete task 2 , 3 and 4 . And so on, till they complete all the task.

#### Initialization:

To build the population with a given size, we build a method that allows randomly generate an array of n elements, for example n = 20, and it is very important that the generated array must satisfy the conditions that we have mentioned in constructing chromosomes above.

#### Selection:

In order to obtain better plans after each generation, we have to choose the best chromosomes to perform crossover. So clearly that we need to make arrangements chromosomes descending order of magnitude of fitness function. Thus, the chromosome zero in populations is the chromosome which has greatest value of fitness function and also the best solution that we have after each generation.

#### Crossover:

To implement a hybrid crossover chromosome 2, the most important thing is to ensure the following conditions:

After performing the stages of arranging chromosomes according to magnitude of fitness function, then the next to do is choose a certain number of best chromosomes to perform crossover. To ensure the diversity of the population while maintaining limited the best solution , I will crossover the first half chromosomes in population (with 2n chromosomes) and call set of chromosomes will be mated as C (with n chromosomes). Afterwards, we will select half of the first chromosome of C, known as C1 (with n1 = n / 2 chromosomes), and performing crossover each chromosome in C1, called chromosomes of mothers, with one chromosome is randomly selected in C, called chromosomes of father. Each time two chromosome are mated, we will have two off-spring. So after performing crossover in populations, we obtained 2.n1 = n new chromosome. And we perform replace the other half of the chromosomes of the initial population by n new chromosomes. Then we again repeat the arrangement of chromosomes of the population according to decreasing magnitude of fitness function (population now covers half of the chromosomes of the population older and the other half is the new chromosomes). Thus we have obtained a new generation with new chromosome 0 is the best solution of this generation.

The next important issue is that after we have pairs of chromosomes to perform crossover, we should build the manner to crossover to ensure the constraints which I have mentioned above is satisfied. As follows , we will simply consider that chromosome has 20 gene, and to ensure that required , divide the chromosomes into two sections , the first of 10 genes (the payment schedule ) and the other 10 genes ( the task list or activities schedule ). Before the implementation of crossover, I still need to make a selection on the chromosomal crossover-point, then to ensure that the plan is feasible and ensuring the diversity of genetic resources, so I choose 4-point crossovers, 2-points for the top 10 genes and 2-points for the last 10 genes. Since the conditions that forced the first 10 genes and last 10 genes were similar, so we will just mentioned how crossovers 2-points with 10 first gene and will likewise with last 10 genes. With cutting point position, we will randomly choose 2 points (of course we must ensure 2 positions is different), and called 2 point cut is cut-1 and cut-2 and cut-1 < cut-2. With a father-mother pairs of genes, we obtained two children called off-spring1 and off-spring2 with equal number of genes with the genes of the parents, the next we will copy the genes in the genome segments from cut-1 to cut-2 of father’s and mother’s genes, transcription and turn it into place from cut1 to cut2 of off-spring1 and off-spring2. With off-spring1, after inherited chromosomal genetic element from cut-1 to cut2 position of chromosomes of the father, the next step we need to take to copy two chromosomal genetic segment of the mother, including gene segments from the beginning to cut1 position and from cut-2 position to the end of chromosomes’ mother, but as mentioned above due to ensure that force the constraint (specifically here is that total of the percentage of the payment stage must equal 100 – that is the constraint with first 10 genes), so we will create a variable call $sum ( obviously $sum must be less than or equal 100) and initialize the value of $sum equals the total value of the genes located in the section from cut1 to cut2 position , the next each time we copy of a gene of chromosomes’ mother, we will recalculate the sum value by adding the value of the gene was added. So there will be some possibilities as follow, the first possibility is the sum will exceed 100 after a certain gene is added, if the gene is the last gene, simple that we just assign a value of that gene by 100 minus $sum, if the gene is not the last gen, we will randomly assign values ​​to the genes from the current gene to the last gene, which ensures total value of these genes equals 100 minus $sum. The second possibility is the sum all of gene was less than 100, then we just leave the final value of the added genes and recalculate the $sum, and then assign values ​​to the final gene equals 100 minus the $sum. Thus we will ensure that the constraint is satisfied. Similarly with the remaining 10 genes and offspring2.

E.g.:

* Father :
* Mother:
* Off-spring 1 :
* Off-spring 2 :

**Mutation:**

The mutation will take place along with the process of crossover. Each off-spring will be mutated with a mutation probability is . Like when we perform the crossover, we will separate the chromosomes into 2 segment, the first 10 genes and the last 10 genes, and performing mutations on two sections. Each section when performing mutations, we just swap the value of the two randomly genes in chromosome, of course two genes must be different.

**Termination:**

Conditions of the problem to terminating is proposed here that the time that the best result of the generations then not change anymore or after a certain number of generations that the best results do not differ over of the budget. This value will be adjusted depending on the problem to achieve an acceptable solution with appropriate calculation time.

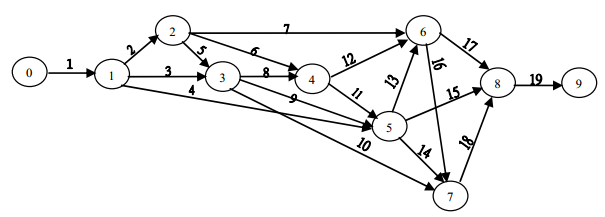
**Selection of parameters:**

The parameters used in the algorithm will be chosen precisely to enhance the effectiveness of the genetic algorithm .And the program also provide the function to allow customize the parameters depending on the number of phases of the project as well as the timing requirements of the program to achieve an acceptable result in the time allowed.

#### Investment Problem

**Problem definition**

An activity network of a software development project is shown in Fig. 1. The parameters of activity network are shown in Fig. 2. The resource limit for resource 1 and resource 2 is 6 and 2 units, respectively. The cost per unit for resource 1 is 1 million dollar and the cost per unit for resource 2 is 2 million dollar. The monthly discount rate for both the owner and the client is estimated to be 0.7%.



**Figure 1: Project Network**

|  |  |  |  |
| --- | --- | --- | --- |
| Activity | Time | Resource 1 | Resource 2 |
| 1 | 10 | 6 | 2 |
| 2 | 30 | 6 | 1 |
| 3 | 50 | 3 | 0 |
| 4 | 50 | 3 | 0 |
| 5 | 10 | 3 | 1 |
| 6 | 60 | 6 | 2 |
| 1 | 60 | 6 | 2 |
| 8 | 50 | 3 | 1 |
| 9 | 40 | 6 | 0 |
| 10 | 90 | 3 | 2 |
| 11 | 80 | 2 | 2 |
| 12 | 30 | 3 | 0 |
| 13 | 30 | 3 | 0 |
| 14 | 50 | 3 | 2 |
| 15 | 150 | 3 | 2 |
| 16 | 50 | 3 | 2 |
| 17 | 40 | 0 | 1 |
| 18 | 110 | 4 | 1 |
| 19 | 90 | 6 | 2 |
| 20 | 50 | 5 | 1 |

Table 12: The parameters of activity network

Where:

* Time (day): time to complete task.
* Resource 1, resource 2: the number of units to complete this task.

**Results**

When the project started, there are two schedules are given, as follows:

* Schedule 1 :

|  |  |  |
| --- | --- | --- |
| Node | Activitiy | Payment( % ) |
| 1 | 1 | 15 |
| 2 | 2,3,4 | 5 |
| 3 | 5,6,7 | 5 |
| 4 | 8,9,10 | 25 |
| 5 | 11,12 | 30 |
| 6 | 13,14,15 | 5 |
| 7 | 16,17 | 5 |
| 8 | 18 | 5 |
| 9 | 19,20 | 5 |

**Figure 3: Schedule 1**

So we have the net profit for client is 31.11 million dollar, whereas the amount paid by the owner is 83.52 million dollar.

* Scheme 2 :

|  |  |  |
| --- | --- | --- |
| Node | Activity | Payment( % ) |
| 1 | 1,2 | 25 |
| 2 | 3,4,5 | 5 |
| 3 | 6,7 | 5 |
| 4 | 8,9 | 15 |
| 5 | 10,11,12 | 10 |
| 6 | 13,14,15 | 10 |
| 7 | 16,17 | 15 |
| 8 | 18 | 5 |
| 9 | 19,20 | 10 |

**Figure 4: Schedule 2**

So we have the net profit for client is 35.311 million dollar, whereas the amount paid by the owner is 85.33 million dollar.

By computer-based calculation, we obtain optimal results for both investors and developers and results are given in the following table:

|  |  |  |
| --- | --- | --- |
| Node | Activity | Payment( % ) |
| 1 | 1,2 | 15 |
| 2 | 3 | 15 |
| 3 | 4 | 15 |
| 4 | 5,6 | 5 |
| 5 | 7,8 | 3 |
| 6 | 9,10,11,12 | 5 |
| 7 | 13,14 | 5 |
| 8 | 15,16,17,18,19 | 10 |
| 9 | 20 | 27 |

**Figure 5: The optimal schedule**

With the both payment schedule and activities schedule is optimal, both parties of the game meet at the equilibrium. The amount of money must to pay by the investors is 82.27 million dollars, saving 1.26 million dollars compared to 83.52 million dollars of the first plan and 3.07 million dollars compared to 85.33 of the second plan, whereas with the developers, their net profit is 43.12 million dollars, more than 12.01 million dollars compared to 31.11 million dollars of the first plan and 7.81 million dollars compared to 35.31 million dollars of the second plan.

**Assessment of the algorithm**

A computers are used to solve this problem with a CPU core i7 3630QM - 2.4Ghz - 8 cores and 8GB of ram can calculate the problem with 9 node in a period of approximately one minute , however there are still many limitations , especially construct a fitness function and method to create automatically the arrays which satisfy the given constraint . With a method to generate randomly arrays, there are some cases arrays which be generated can become an unfeasible solution in reality, and populations which be initialize at the beginning of program depends directly on this method, so the obtained results can be changed when we change in population size and perform at different times, so we have to build this method with a reasonable probability density to avoid the problems are outlined above.

#### Conclusions

A game model with complete information was build to solve the problem of the optimal project progress payments which interest in both the developer and investment or the player. The goal of the model is to minimize the deviation of net present value of cash flows in the project payment between two players is the payer and payee. The theory of Nash equilibrium is a reasonable solution to find the optimal schedule and use the genetic algorithm is a positive schemes and consistent with the pattern of this problems. Some problems derive from this study are as follows:

* Using Nash equilibrium in a problem that the both players want to become a winner such as this problem is reasonable
* We can research other conflict issues of the project with the model similar to the model of this problem
* The application of genetic algorithms to other complex problems which similar to this problem is entirely feasible

#### References

1. Yanf KK, Talbot FB, Patterson JH. "Scheduling of activities to maximize the net present value of projects".

2. Sung CS, Lim SK. A project activity scheduling problem with net present value measurement.

3. DENG Ze-min , GAO Chun-ping and LI Zhong-xue. "Optimization of project payment schedules with Nash equilibrium model"

4. Wei-neng Chen and Jun Zhang. "A Preference-Based Bi-Objective Approach to the Payment Scheduling Negotiation Problem with the Extended r-Dominance and NSGAII".

5. Massimo Orazio Spata and Salvatore Rinaudo. "Merging Nash equilibrium solution with Genetic algorithm: the game genetic algorithm".

6. Grefenstette JJ. "Optimization of control parameters for genetic algorithms".

7. G, Ulusoy, S.Cebelli "An equitable approach to the payment scheduling problem in project management".

8. Sivrikaya-Serfoglu "A new uniform order-based crossover operator for genetic algorithm applications to multi-component combinatorial optimization problems".

9. Sebt, Fazel Zarandi, Alipouri "Genetic algorithms to solve resource-constrained project scheduling problems with variable activity durations".

10. Meredith JR, Mantel SJ Jr. "Project management: a managerial approach."

11. Elmaghraby SE, Herroelen WS. "The scheduling of activities to maximize the net present value of projects".