

СРЕДСТВА УПРАВЛЕНИЯ ПРОГРАММНЫМ ПРОЕКТОМ

Наумов Владимир Николаевич, доктор военных наук, профессор, окончил Высшее военно-морское училище радиоэлектроники им. А.С. Попова. Профессор Военно-морского института радиоэлектроники им. А.С. Попова. Имеет монографии, статьи, учебники и учебные пособия в области автоматизации систем управления ВМФ. [E-mail: naumov122@list.ru].

Аннотация

В статье рассмотрены оптимизационные модели планирования работ по разработке программного проекта на основе различных моделей жизненного цикла программных продуктов. Для определения параметров сетевых графиков предложено использовать методы инженерии программного обеспечения.

Ключевые слова: программный проект, модели планирования работ, методы инженерии программного обеспечения.

Abstract

The article deals with optimization models of planning of software project development on basis of different life-cycle models of software products. In order to determine parameters of road-maps, software engineering methods are proposed to be used.

Key words:

software project, planning models, software engineering methods.

ВВЕДЕНИЕ

Одной из наиболее динамично развивающихся информационных технологий является инженерия программного обеспечения. В ее состав входят: инженерия программного продукта, инженерия программного проекта и инженерия персонала. К инженерии программного проекта можно отнести вопросы: создания структуры пооперационного перечня работ; документирования планов; оценки стоимости; управления требованиями; менеджмента рисков; составления графика проектов и др. В настоящей статье рассматриваются математические модели разработки плана проекта, основанные на использовании методов сетевого планирования и управления.

Построение сетевого графика можно рассмотреть как решение задачи оптимизации. Предлагается два варианта:

1. Минимизация по длительности выполнения всего комплекса работ по созданию программных средств при выделенных ресурсах.

2. Минимизация числа затрачиваемых ресурсов при заданной длительности работ.

Для определения параметров сетевых графиков использована модель СОСОМО [1]. Ее применение позволяет связать трудоемкость отдельных работ с размером проектируемого программного продукта.

1 Минимизация по длительности выполнения всего комплекса работ по созданию программных средств

Пусть задан сетевой график $G(E, O)$ выполнения комплекса работ проектирования, где E – множество событий начала или завершения отдельных этапов проектирования, O – множество работ (этапов, фаз) проектирования. Предположим, что продолжительность каждой работы (i, j) – детерминированная величина. Известно, что число исполнителей каждого этапа определяет время его выполнения,

$$t_{ij} = f(x_{ij}) = (K_{ij}E / x_{ij}),$$

где E – общие трудовые затраты на выполнение всего проекта;

K_{ij} – доля трудозатрат на выполнение (i, j) -й работы (этапа, фазы проекта);

x_{ij} – число исполнителей, участвующих в данной работе.

Однако, во-первых, есть какое-то минимальное значение данного времени, во-вторых, продолжительность работ напрямую не связана с числом исполнителей. Зависимость продолжительности от числа исполнителей нелинейна. С ростом сложности проекта степень нелинейности возрастает [2]. Для каждой работы существует минимально необходимое время ее выполнения

$t_{\min}(i, j)$. Сетевой график позволяет

определить параметры отдельных работ и отдельных событий. Определим следующие параметры: $t(i, j)_n$ – наиболее раннее время начала работы (i, j) ; $t(i, j)_o$ – наиболее раннее время завершения работы (i, j) ; $t(i, j)_o = t(i, j)_n + t_{ij}$.

С учетом дополнительных ресурсов можно изменить значение t_{ij} . Однако для всех работ комплекса должно выполняться неравенство $t_{ij} \geq t_{\min}(i, j)$. Кроме того, при построении сетевого графика должны быть выполнены отношения предшествования, которые определяют непосредственных предшественников для каждой из работ, что определяет множество ограничений вида $t(j, m)_n \geq t(i, j)_o$.

В задачах на нахождение минимума времени должны быть определены выпуклые ограничения. Поэтому следует задать ограничения на размер выделяемых ресурсов. В этом случае данное ограничение должно иметь вид

$$\sum_{\forall(i,j)} x_{ij} \leq X,$$

где X – максимальный размер выделяемых ресурсов.

Если в качестве ресурсов определить число исполнителей проекта, то данное ограничение определяет ограничения на человеческие ресурсы.

С учетом требования неотрицательности всех переменных оптимизационных задач ее математическая постановка будет иметь вид:

$$T_{кр}(X) \rightarrow \min;$$

$$t(j, m)_n \geq t(i, j)_o;$$

$$t_{ij} \geq t_{\min}(i, j);$$

$$\sum_{\forall(i,j)} x_{ij} \leq X; \tag{1}$$

$$\forall(i, j): x_{ij} \geq 1;$$

$$\forall(i, j): t(i, j)_n \geq 0; t(i, j)_o \geq 0.$$

Ограничение $x_{ij} \geq 1$ предполагает, что если имеется какая-то работа, которую необходимо выполнить в процессе проектирования, то на нее должен быть назначен хотя бы один исполнитель. В частных случаях постановки данной задачи могут быть введены ограничения не только на общее число исполнителей, но и на число

исполнителей на каждый этап (фазу) проекта. Например, возможно отсутствие квалифицированных системных аналитиков, в данный момент ограничено число программистов, которых можно привлечь к проектированию и др. Также следует дополнительно ввести ограничение на то, что некоторые работы начинаются сразу же с началом проектирования, и поэтому для них $t(i, j)_n = 0$.

В том случае, если исполнители заняты на нескольких фазах проекта и на каждой фазе участвует свое число исполнителей, то вместо

ограничения $\sum_{\forall(i,j)} x_{ij} \leq X$ следует использовать

ограничения вида $\forall j: \sum_{i=1}^{n_j} x_{ij} \leq X_j$.

Сформулируем данную задачу при условии, что существуют четыре этапа проектирования [2]:

- этап планирования и анализа требований с продолжительностью t_1 ;
- этап проектирования с продолжительностью t_2 ;
- этап программирования с продолжительностью t_3 ;
- этап комплексирования и отладки с продолжительностью t_4 .

Это соответствует водопадной модели жизненного цикла, предложенным этапам проектирования в инженерии программного обеспечения [1]. Соответствие этапов проектирования для различных моделей жизненного цикла, в том числе определенном существующим российским ГОСТ, приведено в таблице 1. При использовании другой модели названия этапов и их содержание будут отличаться, однако, подход к расчету параметров сетевого графика не изменится.

Общий объем работы в человекомесяцах для разработки программных средств встроенного типа, которые присущи автоматизированным системам реального времени, на основании модели СОСОМО определяется с помощью соотношения

$$E = 3,6Size^{1,2}.$$

Значения коэффициентов $K_i, i = \overline{1,4}$, предназначенных для определения

t_{ij} , приведены в таблице 2 [1].

Таблица 1

Этапы жизненного цикла

ГОСТ 34	Бозм	Технологические стандарты Oracle CDM	Rational Unified Process
Формирование требований к АС. Разработка концепции, ТЗ	Анализ осуществимости. Планирование и анализ требований к ПО	Стратегия. Анализ	Начальная стадия (Inception)
Эскизный и технический проект	Проектирование изделия. Детальное проектирование	Проектирование	Разработка (Elaboration)
Рабочая документация	Кодирование	Реализация	Конструирование (Construction)

Таблица 2

Затраты труда, в %

Фаза	Размер проекта, К строк (<i>Size</i>)					
	Малый, 2	Промежуточный, 8	Средний, 32	Большой, 128	Очень большой, 512	
Планирование и анализ требований	8	8	8	8	8	
Проектирование	10	10	10	10	10	
Программирование	60	57	54	51	48	
	Детальное проектирование	28	27	26	25	24
	Кодирование и автономная отладка	32	30	28	26	24
Комплексование и испытания	22	25	28	31	34	

В том случае, если нужное значение *Size* в таблице отсутствует, то следует использовать правила линейной интерполяции:

$$E = E_i + (Size - Size_i) / (Size_{i+1} - Size_i) (Size_{i+1} - E_i). \quad (2)$$

Ограничения на число исполнителей *X* определены с помощью оценочного соотношения

$$X = E / T = E / 2,5E^{0,32} = 4E^{0,68}.$$

Данное ограничение может быть задано реальным наличием исполнителей проекта. Результаты решения задачи оптимизации для программных проектов различного размера приведены на рисунке 1.

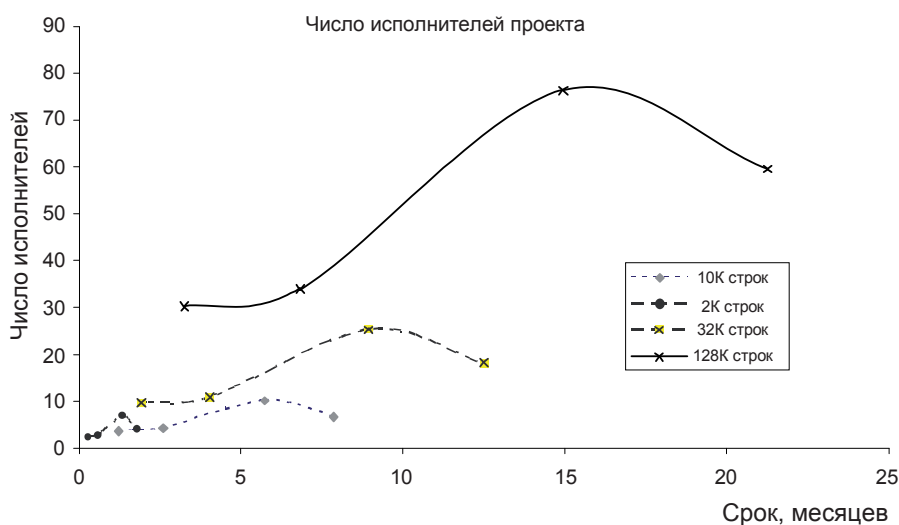


Рис. 1. Результаты решения задач оптимизации по длительности процесса проектирования для различных размеров программных проектов

Приведенные результаты показывают, что наибольшее число исполнителей требуется на этапе непосредственного программирования. Следовательно, наибольшая потребность возникает в аналитических программистах. Данную категорию исполнителей можно нанимать по договору или перераспределять между различными проектами. Таким образом, можно использовать принцип постепенного увеличения вознаграждения исполнителей проекта, так как общее число исполнителей на конечных этапах уменьшается. Реализация такого принципа позволяет обеспечить: поощрение команды программистов в целом за успешное выполнение проекта; поощрение отдельных групп пропорционально их вкладу в общее дело; поощрение ведущих исполнителей в каждой отдельной группе; минимизацию случаев увольнения хороших сотрудников.

При этом, судя по графику, рост числа программистов от размера проекта нелинейный. Данный вывод согласуется с результатами других исследований, что подтверждает достоверность разработанной оптимизационной модели.

В случае использования спиральной модели

Таблица 3

Трудозатраты на выполнение различных процессов на разных стадиях жизненного цикла, в %

Процессы	Стадии жизненного цикла			
	Начальная 5%	Уточнение 20%	Конструирование 65%	Ввод в действие 10%
Управление проектом	50	10	10	10
Проектирование архитектуры	20	50	10	5
Разработка ПО	20	20	50	35
Оценка ПО	10	20	30	50

жизненного цикла [2] число основных процессов увеличивается, следовательно, увеличивается число переменных задачи оптимизации, повышается сложность решаемой задачи. Для решения задачи необходимо определить долю трудозатрат для каждого из определенных в спиральной модели процессов. В таблице 3 [1] приведены данные по трудозатратам на выполнение различных процессов.

С учетом данной таблицы получены коэффициенты доли трудозатрат для всех работ, приведенные в таблице 4.

На данном рисунке также показаны результаты расчета, проведенного с использованием распределения Рэля при значении параметра распределения (времени директивного срока) $t_d = 4$ месяца. Видно, что в этом распределении предполагается, что на начальных этапах проектирования выделяется большее число исполнителей. Это не в полной мере соответствует сложившейся практике проектирования, когда на начальных этапах имеется состояние «раскачки», в проектировании наблюдается «студенческий синдром». Тем не менее, похожесть результа-

Таблица 4
Трудозатраты на выполнение различных процессов в общем жизненном цикле, в %

Процессы	Стадии жизненного цикла			
	Начальная	Уточнение	Конструирование	Ввод в действие
Управление проектом	2,5	2	6,5	1
Проектирование архитектуры	1	10	6,5	0,5
Разработка ПО	1	4	32,5	3,5
Оценка ПО	0,5	4	19,5	5

2 Минимизация числа затрачиваемых ресурсов при заданной длительности работ

В данной задаче предполагается, что необходимо минимизировать число исполнителей проекта. Целевая функция принимает следующий вид: $\sum_{(i,j)} x_{ij} \rightarrow \min$.

При этом накладываются ограничения на общую продолжительность всего комплекса работ — длину критического пути.

$$T_{кр}(x) \leq T_{пред}'$$

где $T_{пред}$ — предельный срок, к которому необходимо завершить процесс проектирования.

Так как длина критического пути равна времени завершения последней работы критического пути, $T_{кр} = t(i, j)_o$. В том случае, если не указан директивный срок завершения всего комплекса работ проектирования или нет жестких требований на это значение, а указан возможный интервал значений, то возможны два варианта постановки задачи:

1. Не накладывается ограничение на $T_{кр}$.

2. Накладывается ограничение

$$T_{пред\ min} \leq T_{кр} \leq T_{пред\ max}$$

В обоих случаях множество альтернатив является компактным (на основе других, ранее указанных ограничений) и в соответствии с теоремой Вейерштрасса существует глобальный максимум (минимум) внутри или на границах данного множества. Результаты решения задачи для размера программного комплекса 10К строк приведены на рисунке 2.

тов, приведенных на рисунке 2, подтверждает достоверность решения задач оптимизации.

В случае различной стоимости работ на разных этапах C_i целевая функция усложняется

$$\sum_{(i,j)} C_i x_{ij} \rightarrow \min$$

3 ПЕРЕРАСПРЕДЕЛЕНИЕ РЕСУРСОВ МЕЖДУ ФАЗАМИ (ЭТАПАМИ) ПРОЦЕССА ПРОЕКТИРОВАНИЯ ПРОГРАММНЫХ СРЕДСТВ

Здесь также возможно решение двух видов задач:

1. Произвести назначение средств на этапы (фазы) проектирования при условии, что общее число исполнителей проекта постоянно и равно X , а длина критического пути минимальна.

2. Произвести перераспределение исполнителей между фазами проекта, если известно ис-

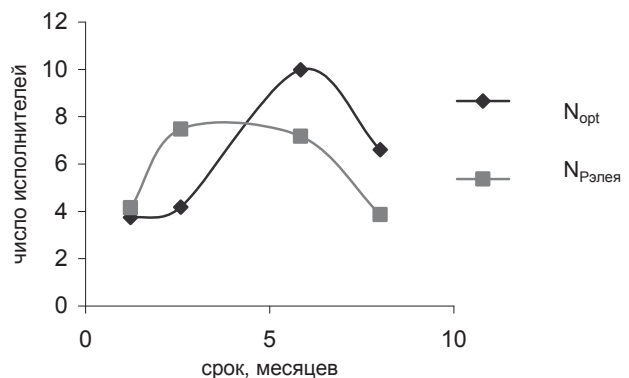


Рис. 2. Результаты решения задачи оптимизации для определения числа исполнителей программного проекта

ходное распределение исполнителей по стадиям проекта.

Рассмотрим формальную постановку задачи первого вида:

$$\begin{aligned}
 & T_{кр}(X) \rightarrow \min; \\
 & t(j, m)_n \geq t(i, j)_o; \\
 & t_{ij} \geq t_{\min}(i, j); \\
 & \sum_{\forall(i,j)} x_{ij} = X; \\
 & \forall(i, j): x_{ij} \geq 1; \\
 & \forall(i, j): t(i, j)_n \geq 0; t(i, j)_o \geq 0.
 \end{aligned}
 \tag{3}$$

При условии, что существует не один, а несколько критических путей, задача оптимизации усложнится. Целевая функция примет вид:

$$\{T_{кр_j}\} \xrightarrow{j} \min.$$

При решении задачи второго типа предполагается, что x_i — величины постоянные. Их значения определены или по результатам решения задачи планирования, или исходя из реальной численности коллектива, выполняющего программный проект. В этом случае в качестве

$$\begin{aligned}
 & t(j, m)_n \geq t(i, j)_o; \\
 & t_{ij} \geq t_{\min}(i, j); \\
 & \forall(i, j): (x_{ij} - \Delta x_{ij}) \geq 1; \\
 & \forall(i, j): t(i, j)_n \geq 0; t(i, j)_o \geq 0.
 \end{aligned}$$

Рассмотрим пример реализации данной задачи для сетевого графика, перечень работ которого для проектирования программного средства размером 32К строк приведен в таблице 5.

Множество ограничений приведено в таблице 6.

Пусть определено число исполнителей программного проекта 16. Для данного числа исполнителей определено следующее распределение работ (фаз, этапов проектирования): $X = (2, 3, 4, 3, 4)$. При таком распределении общая продолжительность работ составляет 54,5 месяца. Полученный вектор распределения персонала имеет вид: $X = (2, 3, 3, 3, 5)$. Таким образом, определена необходимость сократить число исполнителей фазы «детальное проектирование» на одного участника и увеличить число исполнителей процесса комплексирования и испытаний также на одного участника. Это позво-

Таблица 5

Перечень работ сетевого графика для проектирования программного средства размером 32К строк

Название работы	Идентификатор	Продолжительность	Непосредственные предшественники
Планирование и анализ требований	x_1	$0,1E/(x_1 - \Delta x_1)$	-
Проектирование	x_2	$0,8E/(x_2 - \Delta x_2)$	x_1
Детальное проектирование	x_3	$0,26E/(x_3 - \Delta x_3)$	x_2
Кодирование и автономная отладка	x_4	$0,28E/(x_4 - \Delta x_4)$	x_2
Комплексирование и испытания	x_5	$0,28E/(x_5 - \Delta x_5)$	x_3, x_4

переменных выступают величины Δx_i — изменение числа исполнителей i -й фазы программного проекта. В общем случае должно выполняться ограничение:

$$\sum_{(i,j)} (x_{ij} - \Delta x_{ij}) = X,
 \tag{4}$$

где Δx_{ij} — целое (положительное или отрицательное).

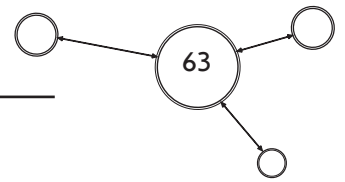
Общая постановка задачи оптимизации будет иметь вид:

$$\begin{aligned}
 & T_{кр_k} \xrightarrow{k} \min; \\
 & \Delta x_{ij} \geq 0, \text{ целое}; \\
 & \sum_{(i,j)} (x_{ij} - \Delta x_{ij}) = X; \\
 & \forall(i, j): \Delta x_{ij} < X;
 \end{aligned}
 \tag{5}$$

Таблица 6

Множество ограничений оптимизационной задачи

ОГРАНИЧЕНИЯ
$t_{1n} = 0$
$t_{1o} = 0,1E/(x_1 - \Delta x_1)$
$t_{2o} = t_{2n} + 0,8E/(x_2 - \Delta x_2)$
$t_{2n} \geq t_{1o}; t_{3n} \geq t_{2o}; t_{4n} \geq t_{2o}; t_{5n} \geq t_{3o}; t_{5n} \geq t_{4o}$
$t_{3o} = t_{3n} + 0,26E/(x_3 - \Delta x_3)$
$t_{4o} = t_{4n} + 0,28E/(x_4 - \Delta x_4)$
$t_{5o} = t_{5n} + 0,28E/(x_5 - \Delta x_5)$
$\forall i: x_i - \text{целое}; t_{in} \geq 0; \Delta x_i < X;$
$(x_i - \Delta x_i) \geq 1$



ляет сократить общую длину критического пути (продолжительность процесса проектирования) до 51,3 месяца.

В случае снятия ограничений на целочисленность решения задачи, можно определить пределы устойчивости задачи оптимизации. В этом случае множители Лагранжа позволяют определить влияние каждой из оптимизируемых переменных на значение целевой функции. В случае, если множители Лагранжа = 0, то соответствующие ресурсы не являются дефицитными, и они израсходованы не полностью. В противном случае, ресурсы израсходованы полностью и с их изменением на единицу значение целевой функции увеличится (уменьшится) на значение данного множителя. Для приведенного примера все значения Δx_i являются не критическими. Критическими становятся X . С увеличением числа исполнителей на единицу значение целевой функции уменьшится на 7.

Также критическими становятся значения временных характеристик. При этом наиболее критичной является продолжительность времени $t_{2н}$. Таким образом, данные выводы согласуются со здравым смыслом. С ростом числа исполнителей уменьшается общее время выполнения проекта. Чем раньше закончен этап анализа и планирования проекта, тем быстрее можно начать другие работы и тем быстрее завершится весь процесс разработки программного продукта.

Задача усложнится, если для каждого этапа проектирования следует выделять различные категории исполнителей. В этом случае должны накладываться дополнительные ограничения на значения $\Delta x_{(i,j)(s,v)}$, где данная величина указывает, сколько может быть перераспределено работников между работами (i, j) и (s, v) .

В этом случае задача может быть сформулирована следующим образом.

$$\begin{aligned}
 & T_{кр_k} \xrightarrow{k} \min; \\
 & \Delta x_{(i,j)(s,v)} \geq 0, \text{ целое}; \\
 & \forall (i, j) : \Delta x_{(i,j)(s,v)} \leq \Delta_{(i,j)(s,v)}; \\
 & \Delta x_{(i,j)(s,v)} = -\Delta x_{(s,v)(i,j)}; \\
 & \sum_{(s,v)} (x_{(i,j)} - \Delta x_{(i,j)(s,v)}) \leq X_{(i,j)}; \quad (6)
 \end{aligned}$$

$$\begin{aligned}
 & \forall (i, j) : \Delta x_{ij} < X; \\
 & t(j, m)_н \geq t(i, j)_о; \\
 & t_{ij} \geq t_{\min}(i, j); \\
 & \forall (i, j) : (x_{ij} - \Delta x_{ij}) \geq 1; \\
 & \forall (i, j) : t(i, j)_н \geq 0; t(i, j)_о \geq 0.
 \end{aligned}$$

Такая постановка задачи показывает, что существует комплекс ограничений на каждое перераспределение исполнителей между работами (i, j) и (s, v) . Частным случаем такой задачи являются фиксированные значения числа исполнителей для каждой из фаз проекта. Такая ситуация возможна, если существует штатное число сотрудников, которые могут участвовать в исполнении программного проекта.

В качестве переменных задачи оптимизации выступают элементы обратно симметричной матрицы $\Delta x_{(i,j)(s,v)} = -\Delta x_{(s,v)(i,j)}$. Поэтому можно сократить количество ограничений задачи, уменьшив ее размерность. Тем не менее, задача имеет большую вычислительную сложность. Число шагов оптимизации зависит от размерности задачи, вида ограничений. Однако даже для задач небольшой размерности число таких шагов исчисляется сотнями. Это позволяет сделать вывод, что решение задачи целесообразно использовать, когда число различных комбинаций велико и задача не может быть решена простым перебором вариантов.

ЗАКЛЮЧЕНИЕ

Предложенные модели позволяют распределить исполнителей по отдельным этапам проекта разработки программного продукта. Предложенные модели позволяют решать задачи оптимизации традиционными методами линейного программирования, в том числе с использованием нашего широкого применение пакета Excel. Решение таких задач позволяет улучшить организацию планирования и управления программными проектами.

СПИСОК ЛИТЕРАТУРЫ

- 1 Бозм Б.У. Инженерное проектирование программного обеспечения/ пер. с англ. — М.: Радио и связь, 1985.
- 2 Шафер Ф., Фатрелл Р., Шафер Л. Управление программными проектами: достижение оптимального качества при минимуме затрат. — М.: Вильямс, 2003.