

## **МЕТОД НЕПОДВИЖНОЙ ТОЧКИ В ДИНАМИЧЕСКОЙ ТЕОРИИ НЕЛИНЕЙНЫХ СИСТЕМ**

*Рассмотрена проблема управления проектами разработки программного обеспечения, которые являются динамическими итерационными системами и применение теории неподвижных точек к процессам управления проектами.*

**Ключевые слова:** динамические системы, проект, неподвижная точка.

**Постановка проблемы и цель исследования.** Анализ теории трех миров К. Поппера подтверждает справедливость утверждения, согласно которому все исследуемые нелинейные динамические процессы протекают либо в одном из них, либо некоторой их комбинации. До недавнего времени основным объектом исследований являлись динамические процессы физической природы. В последние десятилетия значительные усилия исследователей направлены на исследования нелинейных динамических процессов в предметной области самой математики. Как результат, сформировались такие направления обширных исследований как динамические системы, динамическая теория чисел, теория нелинейных систем и ряд других направлений [1].

Динамические системы в ментальном мире, в комбинации ментального мира с миром математики вызвали значительный интерес исследователей в последние два десятилетия. К классу таких динамических систем относятся временные одномерные и многомерные ряды на валютных, фондовых и товарных рынках. Однако следует выделить рост интереса исследователей к динамическим системам, которые имеют содержательное представление. Одним из таких направлений является динамическая теория управления проектами. В работе Грея и Ларсена [2] приводится фундаментальное исследование этого класса динамических систем на содержательном уровне.

Анализ всех направлений управления проектами не может оставить за пределами внимания управление проектами в области программного обеспечения [3]. Данное направление с одной стороны возникло из современной компьютерной науки, а с другой стороны оно в не меньшей степени основано на современной математике включающей теорию рекурсивных функций и эффективность вычислений, теорию чисел, теорию сложных систем.

Опыт работы компаний Microsoft, Google, Apple, IBM свидетельствует о том, что сформировалось особое направление управления проектам, в котором разработчик ПО является и заказчиком и финансирующей стороной №1, а заказчиком и финансирующей стороной №2 является общество потребителей. Разработка ПО имеет непрерывный динамический итерационный характер. Процесс итераций сводится к разработке все более развитых версий ПО, в максимальной степени удовлетворяющих требованиям общества потребителей. Предположим, что  $X$  – множество свойств создаваемого ПО в соответствии с представлениями создателей ПО, а  $Y$  – множество требований к ПО со стороны потребителей. Процесс создания ПО сводится к построению весьма общей функции  $\mathcal{F}(X, t)$ , отображающей  $X$  на/в  $Y$ , которая является мерой соответствия представлений производителя к ПО. А функция  $\Psi(Y, t)$ , отображающая  $Y$  на/в  $X$ , определяет меру соответствия требований потребителя в приобретаемом ПО в момент времени  $t$ . Тогда выражение

$$Q(X_t, Y_t, t) = \int_{t_0}^T (X_t - \Psi(Y, t))^2 dt + \int_{t_0}^T (Y_t - \mathcal{F}(X, t))^2 dt$$

можно рассматривать как меру соответствия  $X, Y$  на интервале времени  $[t_0, T]$ . Интересы сторон учитываются при минимизации  $Q(X_t, Y_t, t)$ . Однако минимизация этого функционала связана еще с одним итерационным процессом, относящимся к реализации проекта исполнителем. При реализации проекта в момент времени  $t$  модель может предполагать, что имеет место равенство  $X_t = Y_t$ , тогда функционал  $Q(X_t, Y_t, t)$  можно заменить функцией  $g(x, t)$ . Таким образом в момент времени  $t$  есть проект  $X_t$  и его необходимо реализовать идентично  $Y_t$  с оптимальными затратами ресурсов всех видов. В работе Уокера [3] детально

анализируется итерационный процесс создания ПО. Итерационные модели создания ПО позволяют разработчику систематически устранять ошибки, неточности, тестировать фрагменты ПО, корректировать версии ПО и устранять все варианты несоответствия уже созданных программных продуктов с проектными требованиями.

Функционал  $Q(X_t, Y_t, t)$  и в частности функция  $g(x, t)$  могут иметь неподвижные точки. Согласно теории динамических систем возможны неподвижные точки следующих типов: простая, итерационная. Так как рассматриваемые итерационные процессы имеют стохастический характер, то введено понятие стохастической неподвижной точки. Будем полагать, что в момент  $t_i$  имеет простая неподвижная точка, если

$$Q(X_{t_i}, Y_{t_i}, t_i) = (X_{t_i}, Y_{t_i}); g(x_{t_i}, t_i) = x_{t_i}$$

и итерационная точка с длиной цикла  $q$ , если

$$Q^{(q)}(X_{t_i}, Y_{t_i}, t_i) = (X_{t_i}, Y_{t_i}); g^{(q)}(x_{t_i}, t_i) = x_{t_i}.$$

В стохастическом случае идеальное равенство маловероятно, но оно может выполняться с заданной вероятностью  $p$  и заданным  $\varepsilon$  таким, что

$$P(Q^{(q)}(X_{t_i}, Y_{t_i}, t_i) = (\tilde{x}_{t_i}, \tilde{y}_{t_i}) \& (\tilde{x}_{t_i}, \tilde{y}_{t_i}) \in ((x_{t_i} - \varepsilon, y_{t_i} - \varepsilon), (x_{t_i} + \varepsilon, y_{t_i} + \varepsilon))) = p$$

**Выводы.** Доказано, что существование стохастических неподвижных точек свидетельствует, что устранение одних недостатков создаваемого ПО влечет появление новых ошибок, неточностей, несоответствий ПО требованиям потребителей.

*Руководитель к.т.н., доцент кафедры ПМ Востров Г.Н.*

## Литература

1. Каток А.Б., Хассельблат Б., Введение в современную теорию динамических систем. – М.: Факториал, 1999, – 768с.
2. Грей Ф.К., Ларсен Э.У., Управление проектами: практическое руководство. – М.: Дело, 2004, – 561с.
1. Уокер Р., Управление проектами по созданию программного обеспечения. Унифицированный подход, – М.: Лори, 2004, – 453с.