

СЦЕНАРНЫЙ ПОДХОД К ОЦЕНКЕ ЭФФЕКТИВНОСТИ ИТ-ПРОЕКТОВ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ

С.В. Разумников, ассистент, Д.А. Пранкевич, студент
Юргинский технологический институт (филиал) Национального исследовательского
Томского политехнического университета
652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26, тел. (38451)-777-64
E-mail: demolove7@inbox.ru

Введение

Вопросы о количественных методах по учету факторов риска и неопределенности является одними из самых сложных. Такое положение возникает всегда, когда речь идет о первичных, наиболее часто встречающихся в жизни понятиях, о которых у каждого есть собственное мнение. Между тем изучение первичных понятий требует сложного понятийного аппарата. Почти такое же положение имеет место для инвестиционных ИТ-проектов. Положение усложняется тем, что теорию учета факторов неопределенности в расчёте эффективности нельзя считать завершённой, так что по некоторым вопросам у разных авторов имеются разные точки зрения. Не существует общей теории измерения финансового риска [1].

Положение усложняется тем, что теорию учета факторов неопределенности в расчёте эффективности нельзя считать завершённой, так что по некоторым вопросам разные авторы имеют разные точки зрения. Не существует общей теории измерения финансового риска. В условиях неопределенности ход реализации проекта не может быть описан однозначно. Необходимо учитывать, что возможны разные сценарии реализации проекта. Поэтому имеются два подхода к оценке эффективности проектов (как в целом сам проект, так и с точки зрения оценки отдельных его участников) в условиях неопределенности [2, 3].

При первом подходе, широко используемом на практике, в основу оценки закладывается один сформированный базовый сценарий реализации проекта и отвечающий ему денежный поток. После этого расчёт производится так, как если бы денежный поток проекта на самом деле был именно таким и детерминированным. Однако, чтобы получать правильные оценки эффективности, необходимо уметь в каждом случае правильно формировать базисный денежный поток. Соответствующие рекомендации можно обосновать, только базируясь на втором подходе к учёту факторов неопределенности – сценарному.

Структура критериев ожидаемого эффекта (способ агрегирования эффектов проекта, отвечающих разным сценариям) устанавливается аксиоматически, т.е. выводится из конкретно формулируемых требований, характеризующих, рациональное экономическое поведение экономических субъектов [5]. Если при рассмотрении информации о конкретном проекте покажется, что рациональное экономическое поведение должно описываться иначе, то необходимо изменить соответствующую систему аксиом и установить, какие критерии будут ей удовлетворять. Если же в подобной ситуации производится решение воспользоваться самостоятельно придуманным критерием или критерием, относящемся к другому виду неопределенности, то в подавляющем большинстве случаев это приведет к необоснованному выводу. Самодеятельность в этой сфере весьма опасна.

Сценарный подход к оценке эффективности ИТ-проектов

Принятый подход к оценке эффективности ИТ-проекта при наличии неопределенности требует детальной информации о сценариях реализации ИТ-проекта, а также степени возможности их осуществления и о значениях основных технико-экономических показателей ИТ-проекта при каждом из сценариев. Как необходимо понимать термин «степень возможности»? Это может быть вероятность сценария, а также что-то иное [4, 6]. Конкретизируем этот нечеткий термин и приведем примеры, которые показывают, что речь здесь идет не только о вероятностях, но и о широком круге количественных измерителей неопределенности.

Общую схему оценки ИТ-проектов в условиях неопределенности можно представить так:

1. Описывают все возможные сценарии реализации ИТ-проекта (или путем их перечисления, или в виде системы ограничений на значения параметров ИТ-проекта и внешней среды).
2. Исследуют, как происходит функционирование организационно-экономического механизма реализации ИТ-проекта при каждом сценарии, определяют (рассчитывают или задают аналитическими выражениями) моменты прекращения ИТ-проекта, которые отвечают этим сценариям, денежные потоки и обобщающие показатели эффективности. При этом в сценариях, в которых предусматриваются нештатные ситуации (стихийные бедствия, аварии, изменения рыночной конъюнктуры)

- юктуры и т. д.), учитывают соответствующие дополнительные затраты. Например, если при определенном сценарии не удастся погасить кредит, который был взят под государственную гарантию, то предусматривают возмещение таких потерь кредитора из бюджета государства [7].
3. Проверяют финансовую реализуемость ИТ-проекта [8]. При возникновении условий, которые нарушают реализуемость данного сценария, то его рассматривают как сопряженный с финансовым риском.
 4. Исходную информацию о факторах неопределенности представляют в количественной форме, которая выражает степень возможности проявления того или иного сценария. Например, ее можно выразить вероятностями таких сценариев или интервалами изменения этих вероятностей. Следует учитывать, что разные участники могут по-разному оценивать степень возможности проявления одного и того же сценария.
 5. Оценивают риск нереализуемости ИТ-проекта, который отражает степень возможности сценариев, при которых нарушаются условия для финансовой реализуемости ИТ-проекта. Наличие такого риска будет свидетельствовать о плохой проработке организационно-экономического механизма для реализации ИТ-проекта. Такой риск может быть устранен, для этого необходимо подробно описать условия для прекращения проекта, а также предусмотреть меры по обеспечению его реализуемости при временной нехватке денежных средств.
 6. По каждому сценарию определяют интегральный эффект (ЧДД) [9]. Премию за риск в ставке дисконта не учитывают! Это вытекает из того, что риск проявляется только в возможности осуществления неблагоприятного сценария, но если он уже осуществился, то соответствующие чистые доходы с учётом возникших потерь оценены точно и риск в них уже отражен. Таким образом, при таком подходе под ставкой дисконтирования следует понимать максимальную годовую доходность доступных и альтернативных безрисковых направлений инвестирования, а ставки дисконтирования при оценке реализуемости и эффективности ИТ-проекта становятся одинаковыми.
 7. Оценивают риск неэффективности ИТ-проекта, который отражает степень возможности сценариев, при которых интегральный эффект становится отрицательным, а также средний ущерб от реализации ИТ-проекта в случае его неэффективности. Здесь важно учитывать, что одной из причин неэффективности ИТ-проекта может быть его досрочное прекращение по финансовым причинам, которое сопряжено с убытками для инвесторов, кредиторов и государства.
 8. На основе показателей каждого сценария определяют обобщающие показатели ожидаемого эффекта. Основными показателями, которые используют для сравнения ИТ-проектов (или вариантов проекта) и выбора наилучшего из них, являются показатели ожидаемого интегрального эффекта. Эти же показатели можно использовать для обоснования рациональных размеров и форм страхования и резервирования.

Заключение

Показатель интегрального эффекта рассчитывается на основе всех возможных значений интегрального эффекта ИТ-проекта (отвечающих всем сценариям). В литературе есть рекомендации по расчету данного показателя. Однако часть из них относится к частным видам неопределенности (это, как правило, вероятностная неопределенность), другие же не обоснованы с точки зрения математики и таким образом могут привести к неправильным результатам. Для того чтобы используемые расчетные формулы привели к рациональным результатам, которые будут отвечать здравому смыслу, важно чтобы они соответствовали определенным требованиям, а именно аксиомам рационального поведения в экономике.

Литература

1. Виленский П.Л., Лившиц В.Н., Смоляк С.А. Оценка эффективности инвестиционных проектов: Теория и практика: учеб. пособие. – 4-е изд., перераб. и доп. – М.: Издательство «Дело» АНХ. – 1104 с.
2. Лившиц С.В. О методологии оценки эффективности производственных инвестиционных проектов в российской переходной экономике // Экономика и экономические методы. 2004. Т. 40. № 3.
3. Балдин К.В., Управленческие решения: Учебник для вузов / Балдин К.В., Воробьев С.Н., Уткин В.Б.-7-е изд. – М.: Дашков и К, 2010. – 496 с.
4. Разумников С.В. Моделирование оценки рисков при использовании облачных ИТ-сервисов // Фундаментальные исследования. - 2014 - №. 5-1. - С. 39-43.
5. Разумников С.В. Использование метода линейного программирования для оценки эффективности применения облачных ИТ-сервисов // Приволжский научный вестник. - 2013 - №. 7(23). - С. 43-45.
6. Разумников С.В. Интегральная модель оценки эффективности и рисков облачных ИТ-сервисов для внедрения на предприятии // Фундаментальные исследования. - 2015 - №. 2-24. - С. 5362-5366.

7. Замков О.О., Толстопятенко А.В., Черемных Ю.Н. Математические методы в экономике: Учебник. 2-е изд. — М: МГУ им. М.В. Ломоносова, Издательство «Дело и сервис», 1999. — 368 с.
8. Колчанов В.Д., Кобко Л.И. Экономическая эффективность внедрения информационных технологий // Учеб.пособие, Москва 2006, с. 177.
9. Маслов А.В., Григорьева А.А. Математическое моделирование в экономике и управлении: Учебное пособие – Юрга: Изд-во Юргинского технологического института (филиала) Томского политехнического университета, 2007. – 264 с.

ОСОБЕННОСТИ НАПИСАНИЯ ПОСТАНОВКИ ЗАДАЧИ НА РАЗРАБОТКУ ЭМУЛЯТОРА

У.В. Лаптева, ст. преподаватель, А.В. Романенко, студент гр. УТСб-12

Тюменский государственный нефтегазовый университет

625000, г. Тюмень ул. Володарского, 38, тел. (3452)28-3022

Как известно разработка любой автоматизированной системы управления начинается с процесса сбора и анализа информации в рассматриваемой предметной области, а на основе собранной информации формулируется задача на разработку. При реализации автоматизированных систем управления (АСУ) любого класса для постановки задачи на разработку из имеющейся нормативно-технической документации (НТД) рекомендуется пользоваться либо ГОСТ 34.603 – 92, либо ГОСТ 34.698 – 90. Применение этих нормативных документов проблематично при написании задания на разработку такого класса систем, как эмулятор, так как:

- 1) Эмуляторы – это не системы автоматизации производственных (технических) процессов, для которых техническое задание предусматривает некоторые пункты раздела «виды обеспечения» (н-р, техническое, метрологическое и т.д.), так как рекомендации, указанные стандартом, требуют большой переработки, так как они не применимы для написания задания на разработку эмулятора.
- 2) Эмуляторы – это и не информационные системы, так как их реализация не всегда предусматривает наличие базы данных и больших объемов обработки информации. Иногда в качестве входных и выходных данных эмуляторов будет унарный символ, поэтому некоторые пункты разделов ГОСТ 34.698 – 90 описания входных/выходные данные невозможно учесть.

Таким образом, при написании задачи на разработку эмулятора целесообразнее выполнить симбиоз ГОСТ 34.603 -92 и ГОСТ 34.698 – 90.

В данной работе рассмотрены проблемы и пути их решения при написании задания на разработку эмулятора фазы лексического анализа процесса компиляции. За основу взят документ ГОСТ 34.698 – 90. Первый пункт связан с написанием характеристики комплекса задач. Здесь рекомендуется описывать: наименование программы, сокращенное название, целевая задача и функции, необходимые для решения поставленной задачи.

В работе данный пункт прописывается следующим образом:

Наименование программы: Эмулятор фазы лексического анализа процесса компиляции

Сокращенное название: ЛексАн

Целевая задача: формирование дескрипторного текста

Далее дается только перечисление функций необходимых для решения поставленной задачи, а пояснения каждой функции приводится ниже по тексту и раскрывает семантику функции.

Например:

Для решения поставленной цели программа должна выполнять следующие функции:

- чтение кода программы из файла;
 - форматирование кода исходной программы;
 - заполнение таблиц лексем;
 - составление дескрипторного текста;
 - сохранение дескрипторного текста и таблиц лексем в файл.
- а) Чтение кода программы из файла. Программа читает код исходной программы, разбивая на лексемы и записывает каждую лексему в массив;
 - б) Форматирование кода исходной программы. Удаляет в исходном коде программы все комментарии пользователя, если они есть;
 - в) Заполнение таблиц лексем. Текст исходной программы разбивается на лексемы, которые записываются в соответствующие таблицы;