Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«Московский физико-технический институт
(государственный университет)»

Факультет управления и прикладной математики

Кафедра информатики

МАТЕМАТИЧЕСКИЕ МЕТОДЫ И СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ НА ОСНОВЕ НЕФОРМАЛИЗОВАННЫХ ЭКСПЕРТНЫХ ДАННЫХ

Выпускная квалификационная работа
(магистерская диссертация)

Направление подготовки: 03.04.01 Прикладные математика и физика

Выполнил:
студент 973 группы _______________ Смаль Сергей Александрович

Научный руководитель:
к.ф.-м.н., доцент __________________ Кадощук Игорь Тарасович

Москва 2015

Оглавление

1. Введение

В современном мире очень часто встречается понятие «проект», «проектная деятельность». Эти термины характеризуют некоторую человеческую деятельность, направленную на создание нового, уникального продукта. Примером могут служить организованные группой студентов start-up-ы или создание информационных систем «под заказ» или вывод нового продукта либо услуги на рынок. Любой проект имеет цели, временные рамки, ограничения по ресурсам, необходимым для реализации проекта, большую долю неопределенности и, соответственно, неизбежной частью проекта являются всевозможные риски. Поэтому, учитывая новизну, риски и ограничения, для того, чтобы успешно завершить проект его необходимо предварительно тщательно спланировать и оценить. От качества планирования проекта зависит его финансирование, развитие, и, в конечном счете, сможет ли проект быть завершенным или будет провален. Ошибка в планировании времени выполнения той или иной фазы проекта может повлечь за собой сдвиг окончания сроков всего проекта, что в свою очередь может привести к финансовым потерям (если есть обязательства по окончанию какому-либо стороннему лицу или организации) или даже остановке и закрытию проекта в целом.

Производить подобное планирование вручную достаточно сложно даже для небольших проектов и практически невозможно для больших проектов, длящихся месяцами и более. Поэтому для таких целей используют специальные программные комплексы – информационные системы управления проектами (Project Management Information System – PMIS). Наиболее известные из них: MS Project и Oracle Primavera (Primavera P6). Однако эти программы не позволяют точно оценивать время окончания нашего проекта и необходимые для этого ресурсы. Здесь следует уточнить, что мы подразумеваем под словом «точно». Конечно, мы не можем видеть будущее и предсказывать за месяцы вперед точную конкретную дату окончания той или иной фазы проекта. Учитывая присущую проектам неопределенность, мы можем указывать лишь предполагаемый диапазон дат, в рамках которого (по нашему мнению) должна завершиться данная стадия проекта. Таким образом, указав для всех работ проекта предполагаемые сроки окончания, мы можем рассчитать предполагаемые даты окончания всего проекта. Но как это сделать правильно? Как узнать с какой достоверностью мы получили результат? Если мы хотим уточнить результат и добавить оценки нескольких экспертов по каждой из работ, то как это сделать? Существующие программные решения предоставляют очень ограниченную и упрощенную функциональность для решения этих задач, что приводит к неточным результатам и может повлиять на успешность окончания проекта.

Целью данной работы будет разработка методики оценки временных рамок стадий проекта и всего проекта в целом, оценки необходимых для данного проекта ресурсов, а так же оценки рисков. Данные вопросы будут рассмотрены с точки зрения определения достоверности получаемых результатов, вычисления указанных параметров проекта с заданной достоверностью, а так же будет написана практическая реализация описанных подходов для проверки на реальных данных. Для простоты последующей работы программа будет взаимодействовать с MS Project (с экспортированными данными) и строить на их основе свои прогнозы.

2. Постановка задачи

Наиболее распространенной информационной системой управления проектами является MS Project. Поэтому мы будем проводить исследование интересующих нас вопросов на основе проекта созданного с помощью данной программы.

Основная цель: используя входные данные проекта оценить с заранее заданной точностью следующие параметры:

- интервал дат, в которые может закончиться каждая из задач
- интервал дат окончания проекта в целом

В качестве входных данных для проведения расчетов будем брать экспертные оценки, указанные при составлении плана в MS Project, а так же дополнительные экспертные оценки по каждой стадии проекта, ресурсам и рискам для увеличения точности получаемых результатов.

Предположим, что каждую работу будут оценивать несколько экспертов в данной области. Они будут предоставлять свои экспертные оценки (ЭО), на основании которых будет рассчитываться итоговая длительность работы. Под принятием решения группой экспертов относительно длительности работы обычно подразумевают агрегацию результатов решений каждого из экспертов, который решал задачу с теми же альтернативами, ограничениями, критериями и целями. Поскольку в принятии решения участвует группа экспертов, у каждого из которых есть свой опыт и знания, то финальная альтернатива с большей достоверностью будет верной, чем при решении той же задачи одним человеком.

Поскольку в каждой оценке содержится неопределённость, то будем рассматривать длительность работы как некоторую случайную величину. Эксперты не могут оценить параметров распределения этой случайной величины и, исходя из личного опыта, оценивают длительность той или иной работы. В зависимости от числа и характера выданных параметров оценки, мы можем сделать заключение: какой тип распределения (треугольное, равномерное, нормальное, трапециевидное) предполагает эксперт, и какие параметры имеет данная случайная величина. Здесь рассматривается только 4 вида распределений, как наиболее простых для оперирования и оценки СВ экспертом и, соответственно, наиболее правдоподобных. Построенная программная модель же не налагает никаких ограничений на использование других типов распределений.

Таким образом, в качестве входных данных мы имеем набор распределений с параметрами. В качестве выходных данных мы хотим видеть интервал сроков, в которые эта работа будет выполнена с наперед заданной вероятностью. Проектный менеджер задает параметр характеризующий уровень достоверности полученного интервала. Например, если задать уровень доверия $\gamma = 0.8$, это будет означать, что с вероятностью 80% срок выполнения работы будет лежать в интервале, полученном после оценивания. Следует отметить, что чем выше уровень доверия, тем шире будет оцениваемый интервал, но поскольку нас интересуют более точные сроки, то нам бы хотелось получать как можно более узкие интервалы. Поэтому к выбору данного параметра стоит подходить ответственно.

3. Основная идея и структура программы

Основная идея – оперировать с оценками длительности задачи (и проекта в целом) как со случайными величинами. Вычислив их распределения, мы можем с заданной точностью посчитать интервал дат, в которые завершится задача (или проект).

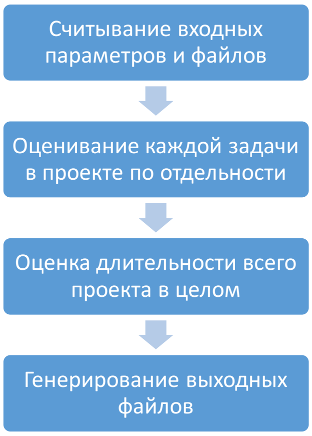
Программа на вход принимает оценки экспертов, а также входные параметры (уровень доверия, использовать ли CPM или Monte-Carlo Analysis и другие). Входные данные являются xls файлами (программа поддерживает версии файлов excel 97 и более поздние) с определенной структурой, которая описана в последующих пунктах. Подобных подход облегчает ввод данных в программу:

- Из MS Project достаточно сделать экспорт в excel
- Файлы можно легко создать руками (простая структура)
- Файлы могут быть сгенерированы из стороннего ПО

Проанализировав входные файлы, узнаем оценки и взаимосвязи между задачами. После этого выполняется оценка каждой задачи (каждую из задач могут оценить как один, так и несколько экспертов, поэтому нужно правильно обработать эти оценки). Зная оценки каждой из задач, выполняется оценка длительности всего проекта (здесь возможны 2 варианта: быстрый, но менее точный и медленный, но более точный[1], они будут описаны далее). После этого создаются 2 xls файла – один файл является копией входного с той лишь разницей, что у него стоят новые посчитанные оценки длительностей, а второй содержит график с финальным распределением длительности всего проекта и предполагаемые интервалы дат, отмеченные на графике. Естественно, что нет необходимости создавать файл с распределением под каждую задачу, так как в проекте их могут быть сотни. Но если интересует оценивание именно одной

[1] «Точность» здесь понимается в смысле правильности оценок, сам интервал дат будет вероятнее всего шире, чем «менее точный» вариант.

задачи, то достаточно создать проект с этой одной задачей и полученное распределение будет именно для нее одной.



Блок-схема работы программы

Если не углубляться в подробности реализации и функционала программы, то можно сказать, что она, на основании входных экспертных оценок о подзадачах проекта, принимает решение о результирующих параметрах. Таким образом, в основе логики программы используются некоторые подходы Теории принятия решений. Поэтому прежде чем приступать к реализации программы необходимо рассмотреть существующие подходы к решению подобных проблем и принятию решений.

Забегая вперед, стоит отметить, что ни один из рассмотренных существующих подходов не подошел полностью для разрабатываемой программы по разным причинам, однако, рассмотренные методы сами по себе очень полезны и их краткий обзор представляет определенную ценность при разработке подобного рода программ. Кроме того некоторые полезные идеи были позаимствованы из разных методов, что дало в совокупности с выбранной математической моделью программы уникальный подход для подобного рода задач.

4. Принятие решений. Что необходимо знать?

Принимать решение - значит определить и выбрать одну из альтернатив, основываясь на каких-либо критериях и/или предпочтениях выбирающего человека. Данный процесс подразумевает, что могут быть рассмотрены несколько альтернатив, но из всех мы выберем наилучшую. Согласно Бейкеру (Baker, et al., 2001) принятия решения должно начинаться с определения лиц, принимающих решение, устранение неясностей в постановке задаче, целей, ограничений, критериев. Затем процесс принятия решения может быть разделен на следующие шаги:

1. *Определение задачи*

   Задача должна быть поставлена четко и ясно. Не должно быть двусмысленности в формулировке. В идеальном случае цель должна быть сформулирована одним предложением и записана.

2. *Определение требований*

   Решение задачи должно обязательно удовлетворять поставленным ограничениям. В математических постановках задач такие требования присутствуют в виде уравнений, неравенств, которым должно удовлетворять выбранное решение.

3. *Определение целей*

   Цели в отличие от требований не обязательно должны быть выполнены. Это то, к чему мы стремимся и что хотим получить. Более того, существуют ситуации, когда мы можем не достигать целей, но выбранная альтернатива будет наилучшей, поскольку наиболее близка к цели.

4. *Определение альтернатив*

   Альтернативы должны удовлетворять требованиям. Если их конечное число, то мы можем проверить их по очереди. Если нет, то рассматривается только подмножество альтернатив, которое дает множество решений, удовлетворяющее требованиям с математической точки зрения.

5. *Определение критериев*

Критерии, по которым различаются альтернативы, должны основываться на целях. Критерии должны показывать, насколько хорошо данная альтернатива соответствует цели. Согласно Бейкеру (Baker, и др., 2001) критерии должны обладать следующими свойствами:

- Возможность сравнивать альтернативы по данному критерию и определять какая из них лучше
- Критериев должно быть достаточно, чтобы достичь всех поставленных целей
- Критерий должен быть вычислимым и значимым
- Критерий не должен быть излишним
- Критериев не должно быть слишком много

В некоторых методах свойство «не быть излишним» заменяется на независимость критериев.

6. *Определение метода принятия решения*

Этот шаг является достаточно важным, поскольку от правильности выбора метода зависит качество принимаемого решения.

7. *Проверка альтернатив согласно выбранному методу и критериям*

После выполнения этого шага все альтернативы могут быть отсортированы в том порядке, в котором они наиболее близки к целям.

8. *Проверка получившихся решений*

Получившиеся решения должны всегда проверяться на правильность. Они должны удовлетворять поставленным критериям, а так же быть наиболее близкими к целям или удовлетворять цели полностью.

Если решение принимает человек или группа людей (экспертов), то очень важным фактором в принятии решений является правильная постановка целей. Как показывают исследования ((Simon, 1986)) разные формулировки одного и того же вопроса приводят к разным решениям, принимаемыми людьми. Например, был проведен опрос с целью узнать, согласится ли опрашиваемый человек на медицинскую операцию при указанных шансах на ее исход. Были опрошены 2 группы людей, первой группе людей формулировка ставилась в следующей форме: «Согласитесь ли вы на операцию, если шансы на то, что вы не выживете, составляют 20%?». Второй группе людей задавали вопрос: «Согласитесь ли вы на операцию, если шансы на успешный исход составляют 80%?». Как показали результаты, количество человек согласившихся на операцию во второй группе опрашиваемых было намного больше чем в первой. Таким образом, нельзя недооценивать то, как именно поставлен вопрос.

5. Обзор методов постановки задачи и принятия решений

5.1. Теория субъективной ожидаемой полезности (Subjective Expected Utility, SEU)

Широкое распространение получила теория ожидаемой полезности. Данная теория была разработана Леонардом Сэвиджем в 1954 (Savage, 1954). Она является логическим следствием и во многом опирается на теорию ожидаемой полезности, разработанную Джоном фон Нейманом и Оскаром Моргенштерном. Данная теория предполагает абсолютную рациональность каждого индивида. Основа теории заключается во введении индивидуальной функции полезности альтернативы для каждого участника, а так же персональные распределения вероятностей по каждой альтернативе.

Таким образом, если индивид оценивает предполагаемый результат как x_i и имеет собственную функцию полезности от получаемого результата $U(x_i)$, а также вероятность данного исхода x_i как $P(x_i)$, то результирующая полезность рассчитывается как

$$U = \sum_{i=1}^{n} P(x_i)U(x_i)$$

Для принятия решения необходимо найти тот результат x_i, который даст максимальное значение результирующей функции полезности.

Преимущества и недостатки данной теории.

Наибольшим преимуществом данной теории является то, что разные индивиды могут делать разные выборы среди одинаковых альтернатив. Например, они могут более правильно (с их точки зрения) оценивать альтернативы, опираясь на собственный опыт.

Минусом данной теории является ее предположение о рациональности индивида. Далеко не всегда (особенно это заметно в экономике) индивиды ведут себя рационально. Например, выбирая кондиционер, покупатели предпочитают модели, которые стоят изначально на 25% меньше, но потребляют больше энергии, хотя экономически выгоднее брать более дорогие модели и экономить на электроэнергии.

Так же существует другой существенный минус. Как показали эмпирические данные, полученные М. Алле и мысленный эксперимент Д.Эллсберга (позже получившие названия парадокса Алле и парадокса Эллсберга), большинство индивидов действуют вопреки положениям теории субъективной ожидаемой полезности. Парадокс Алле (Allais, 1953), в частности, демонстрирует, что индивид, ведущий себя рационально, предпочитает не максимизировать ожидаемую функцию полезности, а ведет себя таким образом, чтобы достичь абсолютной надежности.

5.2. Эмпирические подходы при наличии неопределенностей

Данная группа подходов полагается на разные эмпирические техники и приемы для получения более правдоподобных результатов в задаче принятия решения при наличии неопределенности, несогласованности или неполноты информации.

Рассмотрим пример реального принятия решения. Пусть задана выборка людей с двумя профессиями: адвокаты и инженеры, а так же заданы их численности. После этого требуется на основании некоторой дополнительной информации, которая описывает человека, присутствующего в выборке, но не имеющее никаких данных относительно его профессии определить профессию выбранного человека. Оказывается, что человек, принимающий решение склонен игнорировать априорные вероятности в пользу неполной или даже не относящейся к делу информации. Таким образом, даже не смотря на то, что опрашиваемые знали, что 70% в выборке имеют профессию адвоката, в половине случаев говорили, что выбранный человек с данным описанием имеют профессию адвоката или инженера.

Основываясь на подобных исследованиях можно улучшать техники, применяющиеся в эмпирических подходах. Например, опрашивать более специфичные ситуации, вместо обобщений; учитывать ответы в зависимости от формы заданного вопроса; стараться приближать рассматриваемую ситуацию максимально близко к жизненной; использовать метод «размышлений вслух» (think aloud method) для наблюдения за ходом мыслей опрашиваемого.

Существуют различные эмпирические методы для принятия решений.

Первый подход для принятия решения это рассматривать каждую альтернативу, однако, поскольку число альтернатив может быть очень большим, следует рассматривать только некоторые из них (гроссмейстер никогда не рассматривает все возможные ходы и последствия в шахматах для того, чтобы сделать очередной ход).

Второй подход – «восхождение на холм» (hill climbing) заключается в том, что рассмотрев произвольно начальную альтернативу, мы выбираем следующую альтернативу близкую к рассмотренной (несильно изменив какой либо параметр) и проверяем, приблизила ли она нас к решению (получили ли мы более хороший результат). Если это так, то продолжаем двигаться в том же направлении, если нет – то меняем другой параметр.

Третий подход «анализ цели-средства» (means-ends analysis) заключается в том, что рассмотрев текущую альтернативу и определив «расстояние» до желаемой цели мы выбираем средство для ее достижения. Например, узнав, что находясь в некоторой точке, нам необходимо преодолеть 50 километров до цели, мы исключим из рассмотрения воздушный транспорт и переход пешком, т.к. эти виды передвижения не подходят для данной дистанции.

Четвертый подход подходит для принятия решения экспертами, поскольку основывается на хранящихся в памяти данных и различных ситуация, личном опыте индивида. Как только мы узнаем, что проблема имеет отношение к этим данным, мы добавляем эту информацию, как информацию полезную для решения. Собранная таким образом информация может помочь найти недостающие части для принятия правильного решения. Например, существует база симптомов и болезней, и как только появляются какие-либо симптомы у больного, мы извлекаем все возможные болезни с данными симптомами. Чтобы уточнить или окончательно принять решение относительно диагноза мы анализируем список возможных болезней и добавляем необходимую информацию для сужения этого списка (делаем анализы, осмотры и т.п.).

5.3. Многокритериальные методы принятия решений

Очень часто для того чтобы выбрать какую либо альтернативу необходимо рассмотреть ее с разных точек зрения. То есть необходимо сравнивать альтернативы по различным критериям. Для этого используются *таблицы принятия решений* (decision table).

Пусть мы рассматриваем m критериев: $C_1, \ldots, C_m$ и n альтернатив: $A_1, \ldots, A_m$. В таблице принятия решений строки соответствуют критериям, а столбцы – альтернативам. Значения a_{ij} показывают ценность альтернативы A_i при рассматривании критерия C_j, при чём, чем ближе данная альтернатива по этому критерию к цели, тем большее значение a_{ij}.

Как видно в таблице ниже каждому критерию соответствуют веса w_i, характеризующие важность i-го критерия для сравнения. Эти веса чаще всего подбираются экспертами исходя из субъективных представлений о важности этого критерия.

Сверху таблицы присутствуют значения x_i, которые являются финальными характеристиками альтернатив при выборе. Основываясь на этих значениях можно сравнивать альтернативы друг с другом и выбрать ту, которая имеет наибольшее значение, а значит достигает цели или находится ближе всех остальных к цели.

$$\begin{array}{cc|cccc|}
 & & x_1 & \cdot & \cdot & x_n \\
 & & A_1 & \cdot & \cdot & A_n \\
\hline
w_1 & C_1 & a_{11} & \cdot & \cdot & a_{1n} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
w_m & C_m & a_{m1} & \cdot & \cdot & a_{mn} \\
\hline
\end{array}$$

Таблица принятия решений

Многокритериальные методы принятия решений могут частично или полностью отсортировать альтернативы по их близости к цели. В данной группе методов выделяют два семейства: Multi-attribute Utility Theory (MAUT) и Outranking методы.

Семейство методов MAUT основано на том, что они агрегируют все критерии в одну некоторую функцию, результат которой показывает значимость альтернативы, учитывая все критерии (поля x_i в таблице принятия решений). Таким образом, процесс принятия сводится к задаче оптимизации (максимизации) данной функции.

Семейство Outranking методов сравнивает альтернативы друг с другом и определяет какая из альтернатив более предпочтительна. Более подробно: альтернатива A_i предпочтительнее альтернативы A_j ($A_i \succ A_j$), если на большем числе критериев A_i имеет значения a_{ix} не меньшие, чем a_{jx}, в то время как ее худшее значение не сильно хуже соответствующего значения для A_j. После попарного сравнения всех альтернатив они могут быть полностью или частично отсортированы по предпочтительности. В отличие от методов MAUT, которые явно определяют наилучшую альтернативу, данное семейство методов выделяет подмножество альтернатив, которые являются лучшими по отношению к другим (альтернативы, не вошедшие в это множество хуже как минимум одной альтернативы из множества). Таким образом, целью данного семейства методов является предоставить небольшой список наиболее значимых альтернатив. Методы данного семейства должны стараться сократить список наилучших альтернатив на столько, насколько это возможно.

5.3.1. Simple Multi-Attribute Rating Technique (SMART)

SMART – один из самых простых многокритериальных методов. Финальные значения x_i определяются как взвешенное среднее значений для каждого критерия:

$$x_j = \sum_{i=1}^{m} w_i a_{ij} / \sum_{i=1}^{m} w_i, \qquad j = 1, \dots, n$$

Этот простой метод позволяет рассчитать важность альтернатив, однако для этого необходимо знать значения весов критериев. Эдвардс (1977 (Edwards, 1977)) предложил присвоить критериям значения от 10 до 1 от самого значимого до наименее значимого и после этого выполнить нормировку всех значения на единицу. Однако позже было замечено, что данных подход не имеет смысла, если не учитывать разброс значений каждой из альтернатив по критериям. Например, по одному критерию значения альтернатив могут быть в пределах [2-4], а для другого - [10-100], поэтому нормировка для каждого веса должна учитывать ширину такого диапазона. В связи с этим был предложен метод SMARTS (SMART using Swings), который решал эту проблему.

5.3.2. Обобщенные средние (Generalized means)

В теории принятия решений вектор $\boldsymbol{x} = (x_1, \dots, x_n)$ играет роль агрегационного вектора, который отражает значения альтернатив на каждом критерии с учетом их весов. Это значит, что данный вектор должен отражать строки матрицы наилучшим образом. Cs. Mészáros и T. Rapcsák (Meszaros & Rapcsak, 1996) предложили метод оптимизации энтропии, который и находил вектор **x**:

$$w = \sum_{i=1}^{m} w_i$$

$$x_j = \prod_{i=1}^{m} a_{ij}^{w_i/w}, \qquad i = 1, \dots, n$$

5.3.3. Analytic Hierarchy Process

Данный метод был предложен Саати (Saaty, The Analytic Hierarchy Process, 1980) в 1980 году. Основная идея этого метода заключается в переводе оценок важности одного критерия относительно другого в результирующий набор весов критериев. В данном подходе необходимо попарно сравнить каждые критерии и присвоить оценку каждому из них:

1 = Одинаковая важность или предпочтительность.

3 = Первый критерий немного важнее второго.

5 = Первый критерий значительно важнее второго.

7 = Первый критерий намного более важный, чем второй.

9 = Наибольшее предпочтение отдается первому из критериев.

Пусть значение C_{ij} отражает относительную важность критериев C_i и C_j соответственно. Поскольку суждения об относительной важности присуждает разумный эксперт, то выполнены так же следующие предположения: $C_{ij} = 1/C_{ji}$ и $C_{ii} = 1$. Таким образом, достаточно выполнить всего $\frac{1}{2}m(m-1)$ сравнений чтобы получить всевозможные сравнения критериев. Из получившихся значений составляется матрица $m \times m$. Следующим шагом будет нахождение m-вектора весов, такого, что матрица W со значениями $w_{ij} = w_i/w_j$ будет наиболее близка к матрице попарных сравнений критериев C. Было предложено много методов по нахождению данного вектора, который минимизирует расстояние между матрицами. Мы не будем описывать эти методы, поскольку они не имеют отношения к данной работе, подробнее о них можно найти в (Saaty & Vargas, Comparison of eigenvalue, logarithmic least squares and least squares methods in estimating ratios, 1984) и (Gass & Rapcsak, 2004).

На практике часто используют группировку критериев в иерархию (дерево). Таким образом, сначала производится попарное сравнение критериев внутри более низких уровней дерева, затем внутри более высоких.

После того, как найден и нормализован вектор весов критериев, производится сравнение альтернатив MAUT методом (например, одним из вышеописанных).

5.3.4. ELECTRE methods

Данные методы относятся к Outranking-методам. Простейшим из этой группы методов является метод ELECTRE I. Данное семейство методов основано на использовании двух индексов: согласованность и разногласие (concordance и discordance). Для расчета этих индексов также используется матрица принятия решений и предполагается, что веса критериев нормированы на единицу. Индекс согласованности (concordance) альтернатив $(A_j,\ A_k)$ определяется так:

$$C_{jk} = \sum_{i:a_{ij} \geq a_{ik}} w_i\,, \qquad j,k = 1,\dots,n,\ \ j \neq k$$

Очевидно, что индекс согласованности лежит в пределах [0, 1].

Индекс разногласия (discordance) альтернатив $(A_j,\ A_k)$ определяется так:

$$d_{jk} = \max_{i=1,\dots,m} \frac{a_{ik} - a_{ij}}{\max\limits_{j=1,\dots,n} a_{ij} - \min\limits_{j=1,\dots,n} a_{ij}}, \quad j,k = 1,\dots,n, \qquad j \neq k$$

Вводятся допустимые уровни согласия c^* и разногласия d^* так, что выполняется $0 < d^* < c^* < 1$. Считаем, что $A_j \succ A_k$, если $c_{jk} > c^*$ и $d_{jk} < d^*$, т.е. индекс согласованности выше заранее выбранного уровня, в уровень разногласия – ниже.

5.3.5. PROMETHEE methods

Таблица принятия решений также является основой для этого метода. Так же накладываются ограничения на нормализацию весов на единицу. Однако в отличие от ELECTRE-методов, данная группа методов учитывает особенности каждого критерия в отдельности. Вводятся функции предпочтения $P_i(A_j, A_k)$, которая показывает уровень предпочтения альтернативы A_j относительно альтернативы A_k при сравнении по критерию C_i. Мы рассматриваем только нормализованные на единицу функции, таким образом, выполняется $0 \leq P_i(A_j, A_k) \leq 1$ и верно:

$P_i(A_j, A_k) = 0$ означает отсутствие предпочтений,

$P_i(A_j, A_k) \approx 0$ означает слабое предпочтение,

$P_i(A_j, A_k) \approx 1$ означает сильное предпочтение,

$P_i(A_j, A_k) = 1$ означает строгое предпочтение.

В большинстве случаев функция $P_i(A_j, A_k)$ является функцией разности $d = a_{ij} - a_{ik}$, т.е. $P_i(A_j, A_k) = p_i(a_{ij} - a_{ik})$, где p_i – неубывающая функция, $p_i(d) = 0, d < 0$ и $0 \leq p_i(d) \leq 1, d > 0$.

Далее вводится многокритериальная функция предпочтения $\pi(A_j, A_k)$:

$$\pi(A_j, A_k) = \sum_{i=1}^{m} w_i P_i(A_j, A_k)$$

Эта функция так же принимает значения от 0 до 1 и представляет глобальное предпочтение альтернатив A_j и A_k по всем критериям.

Для того чтобы выявить какая из альтернатив предпочтительнее вводятся следующие функции:

Positive outranking flow:

$$\phi^{+}(A_j) = \frac{1}{n-1}\sum_{k=1}^{n} \pi(A_j, A_k)$$

Negative outranking flow:

$$\phi^{-}(A_j) = \frac{1}{n-1}\sum_{k=1}^{n} \pi(A_k, A_j)$$

Первая функция отражает силу данной альтернативы, то есть на сколько данная альтернатива превосходит все остальные (предпочитаема всем остальным). Чем больше значение этой функции, тем лучше альтернатива.

Вторая функция отражает слабость данной альтернативы, то есть сколько альтернатив предпочтительнее, чем эта. Чем меньше значение этой функции, тем лучше альтернатива.

PROMETHEE I – метод

$A_j \succ A_k$ если $\phi^{+}(A_j) \geq \phi^{+}(A_k)$ и $\phi^{-}(A_j) \leq \phi^{-}(A_k)$

A_j и A_k не различимы, если $\phi^{+}(A_j) = \phi^{+}(A_k)$ и $\phi^{-}(A_j) = \phi^{-}(A_k)$

Иначе A_j и A_k несравнимы.

PROMETHEE II – метод

Если требуется всегда иметь сравнение альтернатив (исключить случаи, когда альтернативы несравнимы), то рассматривается следующая функция:

$$\phi(A_j) = \phi^{+}(A_j) - \phi^{-}(A_j)$$

Таким образом, альтернативы:

$A_j \succ A_k$, если $\phi(A_j) > \phi(A_i)$

A_j и A_k не различимы, если $\phi(A_j) = \phi(A_k)$

Теперь все альтернативы сравнимы и альтернатива с наибольшим $\phi(A_i)$ может считаться наилучшей.

5.4. Принятие решения группой экспертов

Под принятием решения группой экспертов обычно подразумевают агрегацию результатов решений каждого из экспертов, который решал задачу с теми же альтернативами, ограничениями, критериями и целями. Поскольку в принятии решения участвует группа экспертов, у каждого из которых есть свой опыт и знания, то финальная альтернатива с большей достоверностью будет верной, чем при решении той же задачи одним человеком.

Зачастую в данной группе подходов участвует руководящий процессом эксперт (Supra Decision Maker, SDM). Этот человек устанавливает правила и приоритеты оценивающих экспертов.

Рассмотрим метод, основанный на вышеописанном MAUT-подходе (Fulop). Пусть в принятии решения участвует l экспертов $D_1, \dots, D_l$, n альтернатив $A_1, \dots, A_n$, и m критериев $C_1, \dots, C_m$. Обозначим оценку эксперта D_k альтернативы A_j по критерию C_k как a_{ij}^{k}. Также каждый участник устанавливает свои веса для критериев: $w_i^k \geq 0$ – вес критерия C_i, присвоенный ему экспертом D_k, $i = 1, \dots, m; k = 1, \dots, l$. Однако все эксперты обладают разным опытом и разными знаниями, поэтому руководящий эксперт (SDM) устанавливает соответствующие веса экспертов.

Обозначим $V(w)_i^k$ – вес голосующего D_k при оценке веса критерия C_i и $V(q)_i^k$ – вес голосующего D_k при выставлении оценки альтернативе по критерию C_i. Тогда метод, вычисляющий финальную оценку альтернативы A_j следующий:

Для каждого критерия C_i индивидуальные оценки экспертов веса критерия будут агрегированы в групповой вес W_i:

$$W_i = \frac{\sum_{k=1}^{l} V(w)_i^k w_i^k}{\sum_{k=1}^{l} V(w)_i^k}, \quad i = 1, \dots, m.$$

Групповая оценка Q_{ij} альтернативы A_j по критерию C_i тогда будет вычисляться так:

$$Q_{ij} = \frac{\sum_{k=1}^{l} V(q)_i^k a_{ij}^k}{\sum_{k=1}^{l} V(q)_i^k}, \quad i = 1, \dots, m, j = 1, \dots, n.$$

Финальная групповая оценка U_j альтернативы A_j будет считаться как взвешенное среднее арифметическое агрегированных оценок по агрегированным весам:

$$U_j = \frac{\sum_{i=1}^{m} W_i Q_{ij}}{\sum_{i=1}^{m} W_i}, \quad j = 1, \dots, n.$$

5.5. Использование теории нечетких множеств(Fuzzy Sets)

Под нечётким множеством (Fuzzy Set) A понимается совокупность упорядоченных пар, составленных из элементов x универсального множества X и соответствующих степеней принадлежности $\mu_A(x)$:

$$A = \{(x, \mu_A(x)) | x \in X\},$$

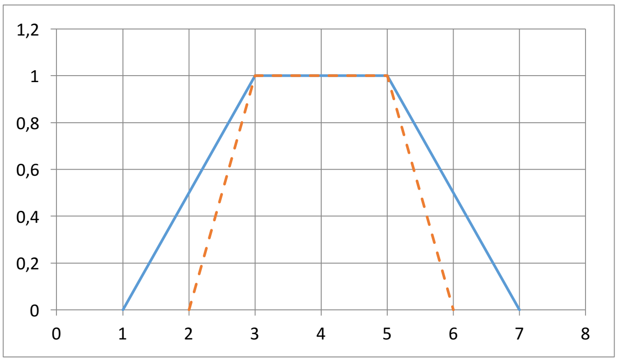
Причем $\mu_A(x)$ — функция принадлежности (характеристическая функция), указывающая в какой степени (мере) элемент x принадлежит нечёткому множеству A.

Для вычислительных целей на нечеткие числа накладывают дополнительные ограничения нормальности и выпуклости:

Нормальность: $\sup\{\mu_x\} = 1 \; \forall \, x \in R$. Это требование означает, что найдется хотя бы один носитель, такой, что его степень принадлежности множеству равняется 1

Выпуклость: $\mu\{\lambda x_1 + (1-\lambda)x_2\} \geq \min\{\mu(x_1), \mu(x_2)\}$, $\forall \, x_1, \, x_2 \in R, \; \lambda \in [0, 1]$. Это требование означает, что носители с наибольшими значениями функции принадлежности сгруппированы вместе.

В данной группе методов используются нечеткие числа (fuzzy numbers) для характеристики оценки, которую дает каждый эксперт. Для удобства нечеткие числа представимы в виде трапециевидных нечетких чисел. Такие числа задаются набором 4 чисел: (a_1, a_2, a_3, a_4), где $a_1 \leq a_2 \leq a_3 \leq a_4$. При $a_2 = a_3$ число называет треугольным. При $a_1 = a_2$ и $a_3 = a_4$ получаем обычный замкнутый интервал, а при $a_1 = a_2 = a_3 = a_4$ получаем обычное действительное число.



Таким образом, чтобы задать свою оценку эксперту необходимо определить 4 числа – контрольные точки, которые задают нечеткое число. Участок [3, 5] соответствующий значению функции принадлежности равной 1 означает уверенность эксперта в том, что задача примет значения из интервала

[3, 5]. В то время как наклонные участки означают неуверенность эксперта, однако, не исключают возможности данного исхода.

Пусть $R_i = (a_1, a_2, a_3, a_4)$ – положительное трапециевидное нечеткое число, отражающее оценку i-го эксперта по данной проблеме. Задача состоит в нахождении агрегационной функции F, для получения итоговой оценки: $R = F(R_1, R_2, \dots, R_n)$.

Основная проблема подхода с использованием нечетких чисел состоит в вопросе правильной агрегации оценок предоставленной группой экспертов. Как правильно сравнивать и выявлять похожие мнения экспертов и как находить различия?

В данной группе есть 3 подхода:

- Агрегация на основе подобия (похожести) нечетких чисел
- Агрегация на основе расстояния между нечеткими числами
- Агрегация с учетом похожести и расстояния между нечеткими числами

В первом подходе, предложенном Hsu и Chen (Similarity Aggregation Method, SAM) (Hsu & Chen, 1996), предполагается, что нечеткие числа обязаны иметь ненулевое пересечение, т.е. принадлежать некоторому срезу нечеткого множества. В этом состоит первый минус данного подхода. Рассмотрим для примера нечеткие числа (для упрощения указываем только концевые точки, предполагая, что промежуточные равны концевым, задавая тем самым прямоугольную область): A=[2, 3], B=[3, 4], C=[5,6]. С точки зрения подхода, описанного Hsu и Chen степень похожести между A и B равно 0, так же как и степень похожести B и C равна нулю. Однако, это очевидно не так, поскольку A и B расположены ближе друг к другу и больше похоже друг на друга, чем B и C. Второй минус описанного подхода заключается в том, что степень похожести чисел определяется отношением пересекающихся областей к области определения в целом и не учитывается отношение области носителей нечетких чисел к области определения. Таким образом, теряется часть информация о входных нечетких числах.

В SAM подсчитывался средний уровень согласия каждого эксперта E_i $(i = 1, 2, \dots, n)$ путем усреднения схожести оценок разных экспертов:

$$A(E_i) = \frac{1}{(n-1)} \sum_{j=1, j\neq i}^{n} S(\tilde{R}_i, \tilde{R}_j) \,,$$

Где

$$S(\tilde{R}_i, \tilde{R}_j) = \frac{\int_x \left(\min\left\{\mu_{R_i}(x),\ \mu_{R_j}(x)\right\}\right) dx}{\int_x \left(\max\left\{\mu_{R_i}(x),\ \mu_{R_j}(x)\right\}\right) dx}$$

Это функция похожести построена на отношении участков пересечения к общей области. Далее рассчитывается агрегационный вес мнения каждого эксперта:

$$RAD_i = \frac{A(E_i)}{\sum_{j=1}^{n} A(E_j)}$$

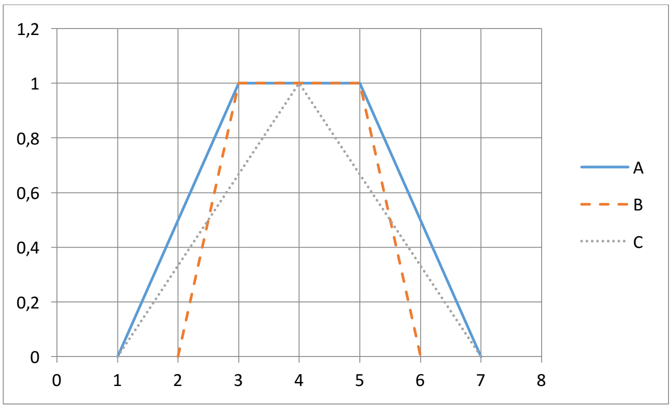
Без учета авторитетности каждого эксперта (считая мнения экспертов одинаково важными), получаем результирующую функцию:

$$\tilde{R} = F(\tilde{R}_1, \tilde{R}_2, \dots, \tilde{R}_n) = \sum_{i=1}^{n} RAD_i \odot \tilde{R}_i \,,$$

где $\odot$ - оператор умножения нечетких чисел.

Работа Hsu и Chen основаны на идее, что вес мнения эксперта тем больше, чем более похоже его мнение на мнения других экспертов. Но у такого подхода есть недостатки, которые были описаны ранее. Чтобы решить эти проблемы Lee предложил Optimal Aggregation Method (OAM), который базируется на идее, что вес мнения эксперта должен быть тем больше, чем меньше *расстояние* между его мнением и мнением других экспертов. Однако оба эти подхода не могут быть достаточно эффективными, поскольку каждый из подходов теряет часть информации либо о расположении нечетких чисел на оси (первый подход), либо структуру самих нечетких чисел (второй подход). По факту эти два подхода являются взаимодополняющими друг друга и их нужно использовать вместе. Данную идею воплотили Chengguo LU, Jibin LAN, Zhongxing WANG предложив новый метод агрегации Consistency Aggregation Method (CAD) основывающийся на сравнении нечетких чисел между собой на похожесть и учитывая расстояния между ними. (Lu, Lan, & Wang, 2006)

Рассмотрим следующий пример:



Здесь взяты нечеткие числа $\tilde{A} = (1, 3, 5, 7), \tilde{B} = (2, 3, 5, 6), \tilde{C} = (1, 4, 4, 7)$. Используя Hsu и Chen меру измерения похожести этих чисел получим, что $S(\tilde{A}, \tilde{B}) = S(\tilde{A}, \tilde{C}) = \frac{3}{4}$. Однако из графика видно, что нечеткие числа $\tilde{A}$ и $\tilde{B}$ имеют большую пересекающуюся область носителя и значит больше похожи друг на друга чем $\tilde{A}$ и $\tilde{C}$. Исправить это можно, если брать взвешенные носители. В качестве веса w_i можно брать значение функции принадлежности в этой точке, таким образом, получим:

$$S_w(\tilde{R}_i, \tilde{R}_j) = \frac{\int_x \left(\min\left\{\mu_{R_i}(x),\ \mu_{R_j}(x)\right\}\right)^2 dx}{\int_x \left(\max\left\{\mu_{R_i}(x),\ \mu_{R_j}(x)\right\}\right)^2 dx}$$

Пересчитывая наш пример по новой формуле получим:

$$S_w(\tilde{A}, \tilde{B}) = \frac{4}{5} \geq S_w(\tilde{A}, \tilde{C}) = \frac{3}{5}$$

Этот результат лучше отражает зависимости между нечеткими числами.

Введем теперь функции расстояния между нечеткими числами. Для этого будем использовать расстояние Хэмминга $d_H(\tilde{A}, \tilde{B})$:

$$d_H(\tilde{A}, \tilde{B}) = \int_x |\mu_{\tilde{A}}(x) - \mu_{\tilde{B}}(x)| dx$$

И расстояние между множествами $d_{inf}(\tilde{A},\ \tilde{B})$:

$$d_{inf}(\tilde{A}, \tilde{B}) = \inf\{d(a, b), a \in A, b \in B\}$$

Где d – это обычная метрика. Для трапециевидных нечетких чисел $\tilde{A} = (a_1, a_2, a_3, a_4)$ и $\tilde{B} = (b_1, b_2, b_3, b_4)$ имеем:

$$d_{inf}(\tilde{A}, \tilde{B}) = \inf\{d(a, b), a \in [a_1, a_4], b \in [b_1, b_4]\}$$

Таким образом, вводим расстояние между нечеткими числами (почему именно так можно посмотреть в (Lu, Lan, & Wang, 2006)):

$$D(\tilde{A}, \tilde{B}) = \frac{1}{2}\left(d_H(\tilde{A}, \tilde{B}) + d_{inf}(\tilde{A}, \tilde{B})\right) = \frac{1}{2}\left(\int_x |\mu_{\tilde{A}}(x) - \mu_{\tilde{B}}(x)| dx + d_{inf}(\tilde{A}, \tilde{B})\right)$$

После этого нормализуем расстояния:

$$d(\tilde{R}_i, \tilde{R}_j) = \frac{D(\tilde{R}_i, \tilde{R}_j)}{\max_{i,j} D(\tilde{R}_i, \tilde{R}_j)}$$

Пусть $\tilde{R}_1, \tilde{R}_2, \dots, \tilde{R}_n$ – n нечетких чисел, представляющих мнения экспертов. Имея выше описанные функции похожести нечетких чисел и расстояния между ними, можно ввести функцию, показывающую на сколько действительно два нечетких числа близки и похожи друг на друга (consistency measure):

$$r\left(\tilde{R}_i, \tilde{R}_j\right) = \beta(S_w\left(\tilde{R}_i, \tilde{R}_j\right) + (1-\beta)\left(1 - d\left(\tilde{R}_i, \tilde{R}_j\right)\right),$$

где $\beta \in [0, 1]$ вес функции похожести $S_w\left(\tilde{R}_i, \tilde{R}_j\right)$, который показывает относительную важность параметра похожести к параметру расстояния между двумя нечеткими числами. Основная идея данного подхода заключается в том, что чем более похожи (similarity) два нечетких числа (чем больше $S_w\left(\tilde{R}_i, \tilde{R}_j\right)$) и чем меньше расстояние между ними ($d\left(\tilde{R}_i, \tilde{R}_j\right)$), тем эти два числа считаются более близкими (в общем смысле слова), более «одинаковыми» (consistency between fuzzy numbers).

Выводы: методы с применением нечетких множеств хорошо подходят для поставленной задачи с точки зрения выполнения расчетов, однако, мы не можем ничего сказать о том, насколько хорошим получился результат. В зависимости от метода мы можем получать более хорошие или более плохие результаты, но эта оценка является лишь субъективным, логичным суждением с точки зрения человека и не имеет никакого математического основания. Иными словами мы не можем строго сравнивать методы и объективно рассуждать о том, насколько хорошие получены результаты. Мы не можем сказать, с какой достоверностью результат агрегации будет получен в реальности. А так же остается открытым вопрос об интерпретации нечеткого числа, полученного в результате.

6. Существующее программное обеспечение

На данный момент программы, предоставляющие удобные средства для управления проектами (Project Management) практически не используют методы по поддержке и принятию решений для оценивания длительности работ. Наиболее распространена следующая функциональность по оцениванию длительности работ:

- Задать длительность одной работы (число)
- Задать ограничения на сроки начала и окончания работы
- Задать зависимости между подзадачами
- Посчитать общие сроки выполнения проекта, используя **метод критического пути**[2]

Однако не уделяется должного внимания непосредственно оценке длительности работы, а так же групповой оценке экспертов.

Вышеописанные методы, тем не менее, являются широко распространенными и общепринятыми при решении задач принятия решений. Данные алгоритмы были заложены во многие программы. Так, например, Analytic Hierarchy Process (AHP) используется в программах Expert Choice и Criterium DecisionPlus. Многокритериальные методы принятия решений используются в D-Sight (методы группы PROMETHEE) и в Criterium DecisionPlus (методы группы SMART). Также развивается ПО, специализирующееся на методах групповой оценки - Windows based Group Decision Support System (WINGDSS).

[2] См. PMBOK [14] страницы 176-177

7. Математическая модель

Как уже было сказано ранее входные данные представляют собой набор экспертных оценок по каждой задаче, а также уровень доверия, с которым мы хоти получить результирующую оценку. Она отражает мнение экспертов о продолжительности всего объема работ (последовательное выполнение всех задач). Итак, более подробно процесс составления результирующей оценки будет следующий:

- Задать уровень доверия и описать каждую задачу, которую необходимо оценить и выполнить
- Собрать экспертные данные по каждой задаче, для этого каждый эксперт
  - Указывает тип распределения СВ (выбирает один из 4х возможных)
  - Указывает параметры распределения
  - Повторяет действия для оставшихся задач
- Для каждой задачи на основе полученных данных сделать оценку длительности с учетом уровня доверия
- Выполнить агрегацию всех оцененных интервалов для получения суммарной оценки работ

Таким образом, процесс получения результирующей оценки логически распался на 3 шага:

- Инициализация задачи и сбор данных
- Получение оценки по каждой задаче
- Получение общей оценки длительности

Первый шаг является чисто техническим и не содержит в себе никакой математики. Стоит отметить только, что к выбору доверительного интервала нужно подходить ответственно. Слишком большой уровень (более 95%) может привести к слишком размытой оценке, поскольку более высокие гарантии требуют согласия большего количества экспертов, а значит, что интервал оценки становится шире. В то же время необходимо максимально точно формулировать задачу со всеми ее ограничениями и особенностями. Почему это так важно было упомянуто в пункте 4. Принятие решений. Что необходимо знать?.

Второй и третий пункты содержат всю необходимую математику для получения требуемого результата. Однако подходы, которые в них используются, существенно различаются.

7.1. Получение оценки длительности одной задачи

Пусть $f_1, f_2, \dots, f_n$ – распределения, которые указали эксперты, оценивая случайную величину (время выполнения задачи).

Пусть $C_1, C_2, \dots, C_n$ – веса экспертов. Причем выполнено:

$$\{C_k\}_{k=1}^{n}: C_k \geq 0, \sum_{k=1}^{n} C_k = 1$$

Возможны 2 случая:

Более простой. Все эксперты считаются одинаково опытными и их веса равны. Либо менеджер проекта может сам определить веса экспертов в соответствии с их опытом. В любом случае эти веса известны априори.

Более сложный. Веса экспертов неизвестны и определяются согласно следующему правилу: более опытные эксперты дают оценки наиболее близкие к реальным срокам, поэтому их оценки друг между другом будут совпадать на достаточно большом интервале. Это значит для того, чтобы избежать расширения результирующего интервала из-за менее точных экспертных оценок следует брать близкие оценки с б**о**льшими коэффициентами. С математической точки зрения это значит следующее: подобрать веса экспертов так, чтобы результирующий интервал, удовлетворяющий уровню доверия, был наиболее коротким. Т.е. веса экспертов определяются как решение оптимизационной задачи найти минимум длинны результирующего интервала.

Основная идея агрегации экспертных оценок состоит в следующем: мы рассматриваем время выполнения задачи как случайную величину со смешанным распределением:

$$f = \sum_{k=1}^{n} C_k f_k \quad (1)$$

Нам необходимо решить следующую задачу: найти такие 2 числа a и b, что выполняется:

$$\begin{cases} P(a \leq \xi \leq b) \geq \gamma \\ b - a \to min \end{cases} \quad (2)$$

где γ – уровень доверия, P – функция распределения СВ с плотностью распределения (1). Перепишем (2) в более удобном виде:

$$\begin{cases} b - a \to min \\ \int_a^b f(t)dt \geq \gamma \end{cases}$$

Будем решать эту задачу численно. Поскольку $f(t)$ является плотностью некоторого распределения, то она обладает весьма важными свойствами, а именно:

$$\lim_{t \to -\infty} f(t) = 0$$

$$\lim_{t \to +\infty} f(t) = 0$$

Однако, исходя из специфики задачи и входных данных, мы можем сделать еще более сильные выводы. Используем то, что эксперты задают 4 числа (параметры распределения) и первое из них (назовем его временно А) означает, по сути, самый короткий срок выполнения задачи. Таким образом, плотность распределения $f(t) \approx 0$ при $t \leq A$. В случае треугольного, трапециевидного и равномерного распределений это утверждение превращается в равенство строго, а в случае с нормальным распределением значением $f(t)$ можно пренебречь и считать его так же практически равным нулю.

Это значит, что мы априори можем найти такое

$$\alpha = \min_{\mathrm{k}=1,\dots,\mathrm{n}} A_k \ ,$$

что $f(t) = 0, t \leq \alpha$. То есть мы знаем, что искомый интервал будет начинаться заведомо правее: $a \geq \alpha$.

Несложно также найти минимально возможное b (обозначим его как β), такое что выполняется:

$$\int_{\alpha}^{\beta} f(x)dx = \gamma$$

Основная идея поиска границ a и b заключается в следующем: мы будем одновременно двигать границы интеграла a и b начиная от α и β соответственно, так, чтобы значение интеграла оставалось равным γ. Причем левые конец мы будем двигать с постоянной скоростью, а скорость правого конца будет изменяться в зависимости от значений функции. Пусть, например $v(a) = c = const$, тогда $v(b) = \frac{f(a)}{f(b)} * v(a) = \frac{f(a)}{f(b)} * c$. Вывод этой формулы несложно понять, посмотрев на рис. 1 и учитывая, что изменение интеграла на левом конце должно быть равно изменению интеграла на правом конце, чтобы его значение было неизменным и оставалось равным γ.

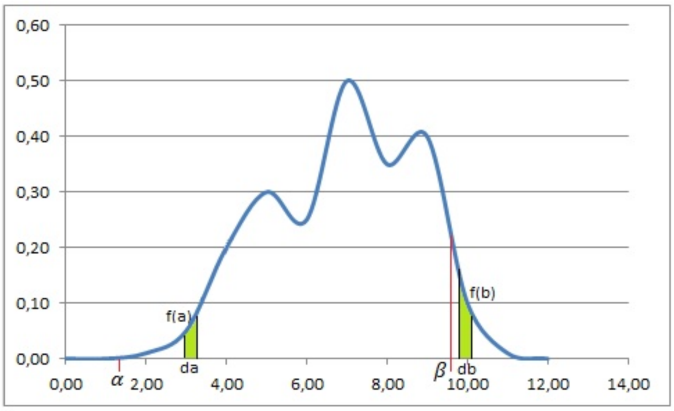


Рис. 1. Нахождение правой границы

Таким образом, мы можем выписать значения a и b в зависимости от времени:

$$a(t) = \alpha + \mathrm{c}t$$

$$b(t) = \beta + \int_0^t \mathrm{c}\frac{f(a(\tau))}{f(b(\tau))}d\tau$$

Заметим, что функция $b(t)$ задана неявно, чтобы получить ее значения в зависимости от t продифференцируем обе части равенства:

$$b'(t) = \mathrm{c}\frac{f(a(t))}{f(b(t))}$$

Получили простое линейное дифференциальное уравнение, которое несложно решить численно, например одним из методов Рунге-Кутты. Будем интегрировать это уравнение до тех пор, пока выполняется условие:

$$b(t) \le B$$

Как только мы нашли такое t_{max}, что выполняется $b(t_{max}) = B$, останавливаем вычислительный процесс и запоминаем t_{max}. Из-за специфики поставленных условий мы можем быть уверены, что всегда выполняется

$$\int_{a(t)}^{b(t)} f(x)dx = \gamma\,, \forall\, t \in [0, t_{\max}]$$

Это означает, что введя в рассмотрение функцию

$$d(t) = b(t) - a(t), \;\; t \in [0, t_{max}]$$

мы можем решать задачу **безусловной** минимизации $d(t) \to min$ численно на отрезке $t \in [0, t_{\max}]$. Решив ее, мы найдем t, а значит и оптимальные параметры a и b.

7.2. Получение интегрированной оценки длительности проекта

Длительность каждой работы выражена случайной величиной. Нам необходимо правильно их просуммировать и найти интервал дат, который также будет удовлетворять заранее заданному уровню доверия γ. Чтобы вычислить сумму введем несколько предположений:

- Все оцененные случайные величины являются независимыми
- Количество оцененных работ достаточно большое

Большинство проектов состоит из огромного числа небольших подзадач, поэтому предлагаемая техника для них применима. Для оценки суммы случайных величин мы будем пользоваться Центральной Предельной Теоремой (ЦПТ). Классическая формулировка нам не подходит, так как она требует одинакового распределения случайных величин, поэтому воспользуемся ЦПТ с наложенным условием Ляпунова (Натан, Горбачев, & Гуз, 2008):

$$a_k = MX_k, \qquad b_k^2 = DX_k, \qquad B_n^2 = \sum_{k=1}^{n} b_k^2 = D\left(\sum_{k=1}^{n} X_k\right)$$

$$\exists\, \delta, \delta > 0, \qquad \frac{\sum_{k=1}^{n} M|X_k - a_k|^{2+\delta}}{B_n^{2+\delta}} \xrightarrow[n\to\infty]{} 0,$$

Данное условие, по сути, означает, что случайные величины теперь могут иметь разные распределения, но внос дисперсии каждой случайной величины в общую суммарную дисперсию должен быть мал, чтобы отдельно взятая случайная величина не могла существенно влиять на сумму. Наш случай как раз подходит под данное условие, так как весь проект разбивается на достаточно маленькие подзадачи (более маленький объем работ проще и точнее оценивается экспертом), а оценка одной задачи скомбинированная из нормального, равномерного, треугольного и трапециевидного распределений имеет маленькую дисперсию по сравнению с суммой остальных дисперсий. Чтобы практически можно было пользоваться данной теоремой, достаточно оценивать несколько десятков задач, что так же не является проблемой для крупных проектов.

Итак, по теореме Ляпунова (Натан, Горбачев, & Гуз, 2008) мы имеем, что случайная величина:

$$Z_n = \frac{\sum_{i=1}^{n}(X_i - MX_i)}{B_n} \tag{3}$$

сходится по распределению к стандартной нормально распределенной случайной величине $N(0,1)$. Введя обозначения $M_0 = \sum_{i=1}^{n} MX_i$, $D_0 = \sum_{i=1}^{n} DX_i$ и $S_n = \sum_{i=1}^{n} X_i$, перепишем (3) в следующем виде:

$$Z_n = \frac{S_n - M_0}{\sqrt{D_0}} \tag{4}$$

S_n - это и есть интересующая нас величина – сумма длительностей задач.

Любую функцию распределения нормально распределенной случайной величины можно выразить через стандартное нормальное распределение по следующей формуле:

$$F(x, \mu, \sigma) = \Phi\left(\frac{x - \mu}{\sigma}\right) \tag{5}$$

где Φ – функция распределения стандартной нормальной случайной величины $\Phi(x) = N(0,1)$.

Сравнивая формулы (4) и (5) несложно видеть, что случайная величина S_n имеет нормальное распределение $N(M_0, \sqrt{D_0})$. Поскольку мы знаем плотности распределения всех случайных величин, то вычислить их математические ожидания и дисперсии не сложно:

$$MX_i = \int_{A_i}^{B_i} x f_i(x) dx$$

$$DX_i = MX_i^2 - (MX_i)^2$$

Учитывая что f – это смесь плотностей распределения (см. формулу (1)) получаем, что математическое ожидание смеси распределений является смесью математических ожиданий:

$$MX_i = \int_{A_i}^{B_i} x f_i(x) dx$$
$$= \int_{A_i}^{B_i} x \sum_{k=1}^{n_i} C_{ik} \, f_{ik}(x) dx = \sum_{k=1}^{n_i} C_{ik} \int_{A_i}^{B_i} x f_{ik}(x) dx = \sum_{k=1}^{n_i} C_{ik} \, M_{ik} X_i$$

Таким образом, нет необходимости вычислять математическое ожидание с помощью численного интегрирования. Дисперсию, тем не менее, по-другому посчитать не получится, поскольку компоненты смеси являются взаимозависимыми случайными величинами, а значит, мы не можем воспользоваться свойством линейности дисперсии (т.к. появляются слагаемые ковариации компонент смеси).

Просуммировав полученные величины, получим параметры распределения суммы S_n. Зная параметры распределения S_n и величину γ – уровень доверия, несложно вычислить предполагаемые интервалы дат окончания всего проекта.

Для этого можно воспользоваться подходом описанном в предыдущем пункте. Но поскольку мы имеем всего одну оценку и, кроме того, мы знаем точный вид распределения, то существует более простой и быстрый способ найти оценку.

По сути нам нужно найти наикратчайший отрезок, интеграл по которому от плотности распределения будет равен γ. Учитывая, что нормальное распределение симметрично относительно M_0 и, что значение плотности распределения монотонно убывает при удалении от M_0, то искомый интервал будет так же симметричен относительно M_0. Таким образом, мы можем искать лишь половину интервала, например в сторону большую, чем M_0.

Нам достаточно численно решить следующее уравнение:

$$F(x) - F(M_0) = \gamma/2$$

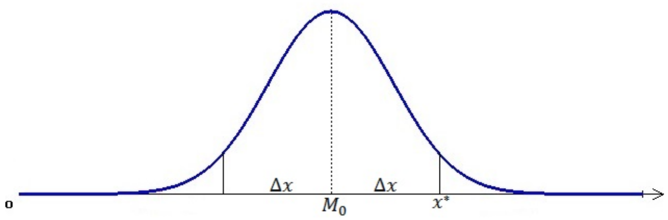


Рис. 2 Нахождение интервала дат из нормального распределения

Обозначив решение уравнения как x^* и введя обозначение $\Delta x = x^* - M_0$, получим интересующий нас интервал дат всего проекта: $[M_0 - \Delta x, M_0 + \Delta x]$.

Замечание: для того, чтобы выполнить расчеты, описанные в этом пункте, необходимо знать какие задачи реально влияют на длительность проекта. Такие задачи называются *критическими*, и они находятся при помощи Critical Path Method (Project Management Institute, Inc., 2008). Стоит отметить, что строго говоря, в нашем подходе оценки длительности задач являются случайными величинами и CPM использовать нельзя. Но поскольку зачастую набор критических задач сильно выделяется среди остальных задач, мы можем в качестве оценок длительности задачи брать математические ожидания полученных случайных величин, и выявить какие задачи являются критическими. Кроме того, для того, чтобы реализовалась ситуация, когда CPM дал бы неверные результаты необходимо выполнение следующих условий:

- Сумма МО задач, вошедших в CP близка (немного больше) к сумме МО задач, которые реально могут повлиять на проект, но не вошли в CP. Это необходимо для того, чтобы влияние было максимальным: если МО этой группы задач много меньше, то она не сможет повлиять на срок проекта, а если больше – то они войдут в CP.
- $\sqrt{D_0}$ задач, не вошедших в CP намного больше, чем аналогичная величина вошедших задач.

Шанс, что выполнятся оба эти условия, достаточно мал. Даже если такое произойдет, то поправка будет небольшой (вследствие того, что сами дисперсии достаточно маленькие).

Классическим описанным CPM можно пользоваться, если выделить группу задач не сложно (что верно для большинства проектов).

Преимущество CPM заключается в том, что он достаточно быстрый, но не гарантирует правильного результата.

Существует более точный, и вместе с тем более затратный способ посчитать длительность проекта, честно учитывая неопределенности в длительности задач. Этот способ – Monte-Carlo Analysis. Суть его в том, что зная распределения всех задач (они же являются СВ), мы совершаем большое количество итерации по подсчету суммарной длительности проекта методом CPM, беря в качестве длительности задач реализацию соответствующей СВ на текущей итерации. Таким образом мы получаем эмпирическое распределение длительности всего проекта. Построив гистограмму несложно убедиться, что это распределение очень похоже на нормальное распределение. Теперь имея доверительный интервал несложно из полученного распределения выделить интересующий нас интервал дат.

Несмотря на то, что существенных различий на проверяемых реальных проектах между методами CPM по МО и Monte-Carlo Analysis обнаружить не удалось, последний способ используется по умолчанию в программе, поскольку честно учитывает все возможные варианты. При проекте в 250 задач, длине критического пути в 21 задачу и выполнив 10000 итераций, время выполнения программы возрастает на 14 секунд. Учитывая, что данное вычисление проводится один раз, мы можем использовать Monte-Carlo Analysis даже несмотря на некоторые временные затраты.

8. Описание работы программной реализации

Как ранее уже было сказано достаточно много проектов ведутся в MS Project, а значит, нам будет удобно извлечь все необходимые данные по проекту (сделать экспорт в MS Excel). При экспортировании проекта в формат xls следует выбрать отображение свойств проекта “Task ‘Export Table’ map”. При этом в excel будут перенесены все необходимые свойства.

Название колонки	**Назначение**
ID	Идентификатор задачи
Task_Name	Имя задачи
Duration	Посчитанная длительность задачи
Duration1	
Duration2	
Duration3	
Duration4	
Distribution [3]	Вид распределения, которым оценивается задача
Predecessors	Зависимости задачи
Summary	Является ли задача суммирующей

Обозначения в экспортированном xls файле

[3] Поля Duration4 и Distribution не входят в стандартный экспортируемый файл, однако, могут быть добавлены как непосредственно в MS Project-е, так и в уже экспортированном файле

Как и было сказано на текущий момент программная реализация поддерживает 4 распределение (математическая модель не налагает ограничения на используемые распределения).

Чтобы показать каким из четырех распределений оценивается задача достаточно в колонке Distribution выставить одно из 4х значений:

- U – для равномерного распределения
- T – для треугольного распределения
- Tr – для трапециевидного распределения
- N – для нормального распределения

Параметры распределений считываются из файла следующим образом:

- U (Duration1, Duration2)
- T (Duration1, Duration2, Duration3)
- Tr (Duration1, Duration2, Duration3, Duration4)
- N[4] (Duration2, (Duration2 – Duration1) / 3)

Если распределение не было задано явно (либо отсутствует сама колонка Distribution), то полагается что использовалось треугольное распределение для оценки. В случае если Duration1 = Duration2 = Duration3 = 0 они автоматически генерируются из поля Duration по формулам (6) (см. далее).

Как мы видим, непосредственной завязки на MS Project нет, то есть можно вручную создать xls файл с описанной структурой или сгенерировать его из другой программы.

[4] Здесь нормальное распределение задается как обычно $N(\mu, \sigma)$, при этом полагается, что пользователь задал математическое ожидание в графе Duration2, а левую и правую границы нормального распределения соответственно в Duration1 и Duration3. «Границы» вычисляются согласно концепции 6σ, отсюда и формулы вычисления $\sigma = (Duration2 - Duration1)/\,3$.

Поле Summary необходимо для того чтобы фильтровать задачи, которые являются всего лишь агрегирующими сущностями и сами по себе не требуют времени для выполнения.

Поля Duration1, Duration2, Duration3 заполняются при выполнении PERT-анализа, либо вводятся экспертами вручную.

Поле Predecessors необходимо для описания зависимостей между задачами. Зная такие зависимости, можно определить какие из задач выполняются последовательно, какие параллельно и, что важнее всего, какие задачи влияют на длительность всего проекта.

В соответствие с пунктами 7.1 и 7.2 программа рассчитывает длительность каждой задачи с учетом оценок нескольких экспертов (при этом, если задана оценка только одного эксперта, программа корректно это обрабатывает), после этого вычисляет, какие задачи составляют критический путь и корректно вычисляет суммарную длительность проекта. Все оценки, как оценки каждой задачи, так и оценка длительности проекта в итоге преобразуются в интервалы дат с учетом входного параметра – уровня доверия.

Как было сказано ранее на вход программе подаются файлы xls, имеющие определенный формат и число – уровень доверия. При этом предполагается, что все файлы идентичны с точностью до оценок экспертов. То есть являются одним и тем же проектом (точнее его экспортом из MS Project в xls формат) и отличаются только значениями Duration(1, 2, 3). Каждый файл, таким образом, представляет оценки одного эксперта по задачам проекта.

В качестве результата работы программы генерируется 2 xls файла: один в точности повторяет проектный xls-файл с новыми вычисленными оценками (учтены мнения всех экспертов), а второй файл – имеет график результирующего распределения длительности всего проекта с обозначенными границами дат, которые соответствуют заданному уровню доверия.

Замечание 1: в качестве оценки длительности задачи по описанной математической модели является смешанное распределение или интервал дат, если учтен уровень доверия. В первый xls файл выставляются математические ожидания этих посчитанных случайных величин. Такой подход позволяет сделать импорт сгенерированного файла обратно в MS Project и работать в нем уже с новыми оценками[5].

Замечание 2: генерировать файл с полученным распределением для каждой задачи было бы расточительно, поэтому генерируется только файл с распределением длительности проекта. Как было рассказано в теории такое распределение является нормальным, если мы вычисляем сумму нескольких длительностей работ. Однако если в проекте всего одна работа, т.е. мы специально хотим оценить одну работу несколькими экспертами, то результирующее распределение будет не нормальным, а честно посчитанным смешанным. При этом так же будут указаны интервалы дат соответствующие заданному уровню доверия (примеры таких распределений будут представлены далее).

[5] При этом теряется весь смысл выработанной методики, суть которой в результирующих интервалах дат, а не в числе, как это было изначально. Однако это позволяет учесть мнения нескольких экспертов, тем самым улучшив оценку длительности задачи и продолжить работать в MS Project, если это необходимо.

9. Тестирование программной реализации на реальных проектах

Далее будут описаны процедуры исследования с результатов расчетов программы в двух случаях:

1. Одну задачу оценивают несколько экспертов (сгенерированный пример)
2. Много задач оценивает один эксперт (реальный проект)

Как уже было сказано программа поддерживает как каждый режим по отдельности, так и их комбинацию. Мы будем исследовать каждый режим по отдельности, в корректности работы комбинации этих режим сомневаться не приходится, поскольку как было сказано, ранее нигде не налагалось ограничений на вид распределения выступающего в роли оценки длительности задачи. А значит помимо четырех описанных распределений, полученное смешанное распределение также может выступать в роли оценки длительности задачи (можно представить, что мнения всех экспертов собрал один эксперт, который и выдал эту оценку).

9.1. Исследование результатов оценки длительности одной задачи несколькими экспертами

Рассмотрим некоторую задачу, длительность которой оценивают несколько экспертов (в рассматриваемом случае их было 5). Каждый эксперт оценил задачу, используя одно из описанных распределений (исключая нормальное распределение – оно достаточно специфичное). Ниже представлены оценки в виде таблицы, а также получившиеся результирующие графики распределений для разных уровней доверия γ.

Эксперт	1	2	3	4	5
Оценка	Tr (2,4,7,8)	Tr (3,5,6,8)	T (4, 6, 9)	U (4, 7)	Tr (3,3,7,8)

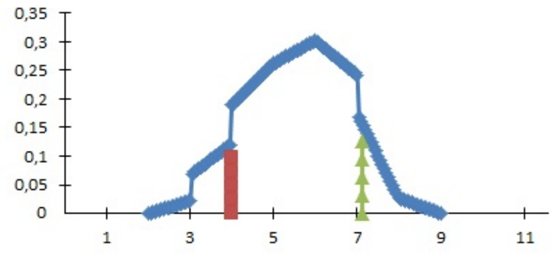


Рис 3. γ=0.8, A = 4, B = 7.1

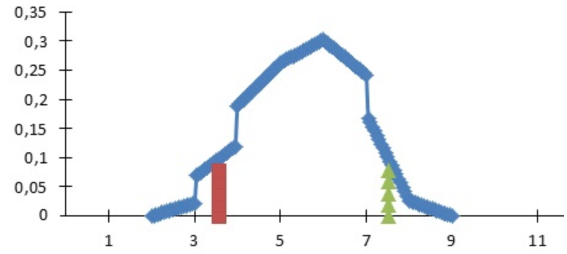


Рис 4. γ=0.9, A =3.6, B = 7.5

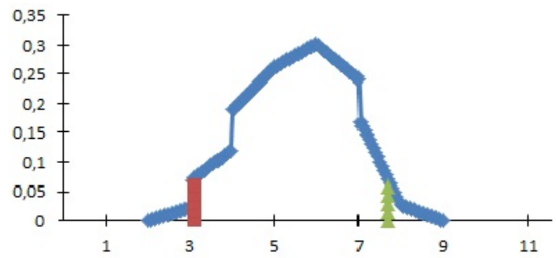


Рис 5. γ=0.95, A = 3.1, B = 7.7

Все экспертные оценки включают интервал [4, 7] и как несложно видеть этот же интервал входит в результирующий интервал при любых достаточно больших уровнях доверия, что подтверждает правильность составления итогового интервала. Кроме того наглядно видно как распределилось агрегированное мнение всех экспертов. Даже если не учитывать отметки, посчитанные с учетом уровня доверия, можно визуально выделить наиболее вероятный интервал [4, 7]. Данный факт является также достоинством предлагаемого метода.

Рассмотрим еще один особенный случай: оценки экспертов не пересекаются. Такой случай маловероятен, тем не менее, не исключен. Достаточно рассмотреть 2х экспертов, давших оценки: Tr(12,13,14,15) и T(16,17,18).

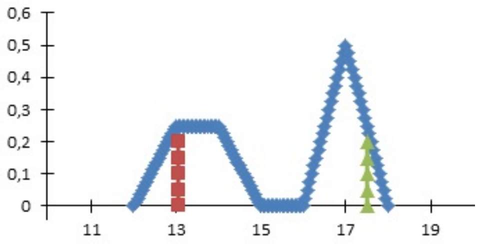


Рис 6. γ=0.8, A =13, B = 17.5

Как видно из графика, алгоритм корректно обрабатывает подобные разрывы, однако результирующий интервал получился достаточно широким и включающим интервал, который не вошел в оценки ни одного эксперта. Поскольку все эксперты считаются одинаково опытными, то они и ошибаются в равной степени, что и дало нам некоторую усредненную оценку этих экспертов.

9.2. Исследование реального проекта

Для исследования разработанного метода на практике был взят реальный проект, состоящий примерно из 250 задач (разделенных на 3 большие фазы). Этот проект был составлен в MS Project-е, в нем были описаны задачи, их взаимосвязи, а также оценки длительности. Одно число (количество дней) на каждую задачу. Данный проект был экспортирован в xls-формат и подан в качестве входного файла программе. Поскольку файл был один (что равносильно одному эксперту), а также, поскольку в качестве оценки было одно число на одну задачу, были сгенерированы дополнительные оценки (пессимистичная t_p и оптимистичная t_o). Первоначальная оценка эксперта была взята за ожидаемую оценку (t_e). В качестве распределения были взяты треугольные распределения. Для большей реалистичности t_p и t_o генерировались следующим образом:

$$
\begin{aligned}
t_o &= t_e * (0.9 + \varepsilon) \\
t_p &= t_e * (1.2 + \varepsilon)
\end{aligned} \tag{6}
$$

где ε – случайная величина, равномерно распределенная на интервале [-0.05, 0.05]. Оптимистичное время берется с коэффициентом примерно 0.9, а пессимистичное с коэффициентом 1.2, потому что человеку свойственно недооценивать сложности и возможные трудности в реализации. Поэтому математическое ожидание оценки будет немного смещено в сторону б*о*льших времен.

Поскольку оценки проводил один человек, то основная задача заключалась в подсчете итогового времени выполнения проекта, при заданных распределениях длительностей отдельных задач. Однако в процессе исследования обнаружилось, что в проекте существуют задачи которые имеют длительность сопоставимую (или просто равную) длительности всего проекта, например задачи «Управление рисками», «Управление изменениями». Длительность данных задач зависит от длительности проекта, поэтому пришлось вручную исключить их из оценки, чтобы они случайно не попали в критический путь (см. п. 9). Это может быть легко реализовано: достаточно в колонке Summary проставить значение «Yes» и данные задачи будут игнорироваться алгоритмом.

На данном проекте было выяснено, что 21 задача является критической и влияет непосредственно на длительность всего проекта. В зависимости от уровня доверия получили различные финальные интервалы длительностей проекта:

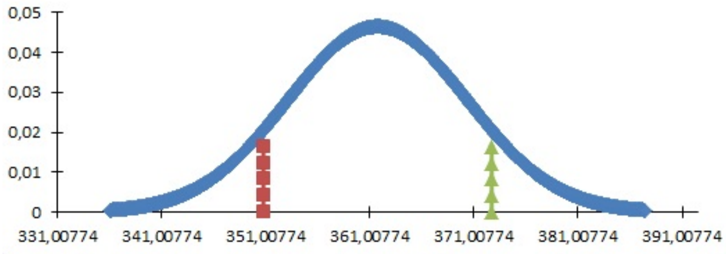


Рис 7. γ=0.8, A = 350.8, B = 372.7

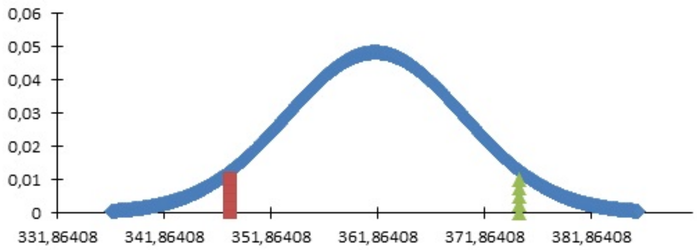


Рис 8. γ=0.9, A = 348.1, B = 375.1

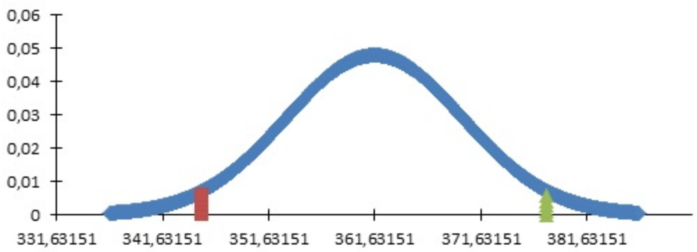


Рис 9. γ=0.95, A = 345.4, B = 377.8

Если взять начальные оценки задач, которые являлись критическими и сложить их, мы получим длительность проекта равную 350 дней. Данная оценка входит в интервал, но она меньше, чем мат. ожидание полученного распределения (361 день), из-за характера сгенерированных распределений каждой задачи.

Мы получили достаточно широкое нормальное распределение из-за того, что неопределенность каждой из 250 задач вносила свои поправки. Таким образом, мы получили, что сроки проекта варьируются в интервале около 15 дней от значения 361 день.

Данный результат свидетельствует о том, что проект на самом деле на стадии оценки не может быть оценен с точностью до дня и интервальная оценка представляется более естественной и реалистичной. Чем больше неопределенности в отдельно взятой задаче, тем больше неопределенность в суммарных сроках проекта (возрастает дисперсия).

10.Заключение

В данной работе были исследованы проблемы оценки сроков выполнения работ в проектах, а также оценки длительности самих проектов. Были установлены некоторые предложения, выполнение которых поможет улучшить оценки:

- Оценки желательно проводить группой экспертов
- Оценки длительности должны быть интервальными, а не точечными

Согласно этим предложениям были разработаны алгоритмы оценки одной задачи группой экспертов и алгоритм суммирования уже существующих оценок для получения оценки длительности проекта. Оцениваемая величина (длительность) принимается, как случайная величина, что существенно, так как всегда есть некоторая неопределенность относительно того, сколько времени в действительности будет выполняться задача.

В качестве результата мы получаем некоторое агрегированное распределение, из которого, задавшись некоторым уровнем доверия, можно получить наиболее вероятный интервал выполнения задачи.

Стоит отметить, что разработанный подход для учета экспертных мнений, которые оценивают некоторую величину можно применять не только для оценки длительности выполнения задачи или проекта. Так, например, данную методику можно адаптировать и для других областей Project Management-а, а именно для управления стоимостью, оценки бюджета проекта, оценки и управление рисками и другие области. Актуальным и эффективным будет продолжить текущее исследование по оценке длительности задач в новые области применения. Это объясняется тем, что многие оцениваемые величины будут зависеть от оцененных длительностей задач, а значит, корректнее будет с ними работать как с уже оцененными случайными величинами, а не как с интервалами или тем более как с точечными оценками. Кроме того, управление стоимостью, управление рисками и прочие перечисленные предметные области очень быстро развиваются и являются одними из ключевых процессов, как в проектном управлении, так и в других управленческих областях.

Удалено: пр.

11.Список литературы

Allais, M. F. (1953). Le Comportement de l'Homme Rationnel devant le Risque. Critique des Postulats et Axiomes de l'Ecole Americaine. *Econometrica* , 503—546.

Baker, D., Bridges, D., Hunter, R., Johnson, G., Krupa, J., Murphy, J., и др. (2001). *Guidebook to decision-making methods.*

Edwards, W. (1977). How to use multiattribute utility measurement for social decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics , 5* (7), 326 - 340.

Fulop, J. (б.д.). Introduction to Decision Making Methods.

Gass, S., & Rapcsak, T. (2004). Singular value decomposition in AHP. *European Journal of Operational Research , 154* (3), 573–584.

Hsu, H. M., & Chen, C. T. (1996). Aggregation of fuzzy opinions under group decision making. *Fuzzy Sets and Systems* (79), 279-285.

Lu, C., Lan, J., & Wang, Z. (2006). Aggregation of fuzzy opinions under group decision-making based on similarity and distance. *Journal of Systems Science and Complexity* (19), 63-71.

Meszaros, C., & Rapcsak, T. (1996). On sensitivity analysis for a class of decision systems. *Decision Support Systems , 16* (3), 231 - 240.

Project Management Institute, Inc. (2008). *Project Management Body Of Knowladge (PMBOK Guide).*

Saaty, T. (1980). *The Analytic Hierarchy Process.* McGraw Hill.

Saaty, T., & Vargas, L. (1984). Comparison of eigenvalue, logarithmic least squares and least squares methods in estimating ratios. *Mathematical Modelling , 5* (5), 309–324.

Savage, L. J. (1954). *The Foundations of Statistics.* New York.

Simon, H. A. (1986). Decision Making and Problem Solving. *Report of the Research Briefing Panel on Decision Making and Problem Solving.* Washington, DC.

Натан, А. А., Горбачев, О. Г., & Гуз, С. А. (2008). *Теория вероятностей.* Москва: МЗ Пресс.