

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

по направлению 09.03.04 Программная инженерия

тип программы академическая

профиль «Разработка программно-информационных систем»

Разработка системы анализа эффективности использования ресурсов в
региональном управлении МЧС

Студент

Группа ПИ-16

Посаженников И.А.

Руководитель

д-р техн. наук, доцент

Корнеев А.М.

Консультант по

программному обеспечению

канд. техн. наук, доцент

Ведищев В.В.

Нормоконтроль

Болдырихин О.В.

Заведующий кафедрой

канд. техн. наук

Алексеев В.А.

Липецк 2020 г.

ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Факультет ФАИ

Кафедра АСУ

Заведующий кафедрой Алексеев В.А.

« » _____ 2020 г.

ЗАДАНИЕ НА ВЫПОЛНЕНИЕ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

Студенту _____ *Посаженникову Ивану Алексеевичу* группы *ПИ-16*

Направление (специальность) *9.03.04 «Программная инженерия»*

1. Тема *Разработка системы анализа эффективности использования ресурсов в региональном управлении МЧС*

2. Цель и задачи работы *Спроектировать и разработать систему анализа эффективности использования ресурсов в рамках регионального управления МЧС.*

3. Характеристика предметной области *анализ оболочки данных (Data Envelopment Analysis)*

4. Содержание расчетно-пояснительной записки *введение, литературный и патентный обзор предметной области, постановка задачи, изучение и моделирование предметной области, программно-аппаратная реализация, заключение, список источников*

5. Перечень графического материала *скриншоты программы, схема алгоритма работы программы, концептуальная диаграмма данных.*

6. Срок сдачи ВКР руководителю *18.06.2020*

7. Консультанты по ВКР _____

8. Дата выдачи задания *10.02.2020*

9. Руководитель ВКР _____

доктор технических наук, доцент Кортеев А. М.

10. Задание принял к исполнению студент *Посаженников И. А.*

Заведующий кафедрой АСУ Алексеев В.А.

Аннотация

С. 48, Ил. 19, Табл. 2, Формул 3, Литература 28 назв.

Данный отчёт является пояснительной запиской к выпускной квалификационной работе бакалавра «Разработка системы анализа эффективности использования ресурсов в региональном управлении МЧС».

Целью работы является разработка системы анализа эффективности использования ресурсов для регионального управления МЧС.

Оглавление

Введение.....	7
1 Постановка задачи.....	9
1.1 Литературный обзор постановки задач	9
1.2 Патентный обзор постановки подобных задач. Анализ стандартных средств и существующих способов решения задачи.....	10
1.3 Объекты управления, информационные объекты и автоматизируемые процессы. Пользователи и внешние сущности	11
1.4 Цели разработки, функции системы	11
2 Изучение и моделирование предметной области.....	13
2.1 Основные понятия и процессы, их свойства и закономерности.	13
2.1.1 Коэффициент эффективности.....	14
2.1.2 Эффективная граница	15
2.1.3 Приближение к границе эффективности	16
2.1.4 Возврат к масштабу (Returns to scale)	17
2.1.5 Технические и смешанные неэффективности.....	19
2.1.6 Группы равных (Peers).....	20
2.2 Теоретическое изучение предметной области. Построение теоретических математических моделей	20
2.3 Экспериментальное изучение предметной области	22
3 Разработка информационной базы для решения задачи	24
3.1 Построение концептуальной модели данных	24
3.2 Описание источников информации, входных сигналов и документов.....	24
3.3 Описание выходной информации: сигналов, документов и видеокадров	25
4 Программно-аппаратная реализация решения задачи.....	27
4.1 Аппаратное обеспечение	27
4.2 Программное обеспечение	27
4.3 Разработанные программные средства.....	28
4.3.1 Описание использованных средств, подходов, методов, языков, библиотек	28

4.3.2	Описание программы.....	31
4.3.2.1	Общие сведения.....	31
4.3.2.2	Функциональное назначение.....	31
4.3.2.3	Описание логической структуры.....	32
4.3.2.4	Используемые технические средства.....	34
4.3.2.5	Вызов и загрузка.....	35
4.3.2.6	Входные данные	35
4.3.2.7	Выходные данные.....	36
4.3.3	Описание применения программы.....	36
4.3.3.1	Назначение программы.....	36
4.3.3.2	Условия применения	37
4.3.3.3	Описание задачи	37
4.3.3.4	Входные и выходные данные.....	38
4.3.4	Описание результатов работы программы.....	41
5	Результаты внедрения и использования системы. Достижение целей разработки	43
	Заключение.....	45
	Список источников.....	46

Введение

Вместе с непрерывным ростом плотности населения в городской местности, аварии, вызванные чрезвычайными происшествиями, становятся всё более и более серьёзными. Как правило, чрезвычайные происшествия, которые происходят в городских районах относятся к инцидентам, вызванных взаимодействием людей с окружающей средой или антропогенными системами, такие как пожары в городе, промышленные аварии, сбои связи, террористические акты и т.д. Стало необходимо чаще тушить пожары на интенсивно эксплуатируемых землях, в густонаселённых районах и на территориях заводов с опасным производством. Люди полагаются на то, что спасательные службы смогут принять соответствующие меры для предотвращения чрезвычайных ситуаций и спасения людей. Однако, например, тушение пожаров характеризуется предотвращением надвигающихся катастроф в то время, как оценка работоспособности современных пожарных частей основывается на устаревших данных. Несмотря на то, что результаты анализа покажут определённые заключения, с учётом будущих стратегий они не будут достаточными для регулирования противопожарных ситуаций. Спасатели же склонны к физическому истощению в связи с длительным рабочим днём и повышенной опасности работы, эти факторы приводят к снижению работоспособности. Поэтому с экономической точки зрения создание оптимизированной и корректной модели распределения человеческих ресурсов для тушения пожаров означает выявление критической точки между возможностями тушения пожаров, проведением аварийно-спасательных работ и оказанием первой помощи. С помощью подобной модели можно избежать излишних инвестиций в человеческие ресурсы и оборудование, а также с её помощью можно избежать неудачи в обеспечении безопасности людей из-за недостатка человеческих ресурсов и оборудования. В условиях постоянного увеличения числа чрезвычайных ситуаций, а также ограниченного финансирования очень сложно рассчитать относительную эффективность пожарных частей в регионе

в целом, с учётом общего объёма «производства». Поэтому необходимо выбрать правильный, справедливый и разумный подход к оценке будущей производительности каждого пожарного подразделения, поставить рациональные цели, а также провести корректировку ресурсов для неэффективных подразделений.

1 Постановка задачи

1.1 Литературный обзор постановки задач

В 1978 году Гэтри назвал три составляющих выполнения работы: эффективность, продуктивность, производительность [1]. Ричман и Фармер (1975) выделили две составляющие для достижения цели: эффективность и продуктивность [2]. Соотношение между затратами и выпуском определяется как эффективность [3], уровень производства ожидаемой цели в системе определяется как продуктивность [4, 5]. На самом деле эффективность и продуктивность по отдельности представляют различные требования к эксплуатации (рабочим характеристикам), и их достигают не одновременно. Однако продуктивная организация должна думать о том, и о другом, добиваясь максимальной производительности наиболее эффективным способом [2].

Наиболее часто используемыми оценками эффективности работы являются метод коэффициента (Ratio Approach), регрессионный анализ (Regression Analysis), анализ множественных критериев (Multiple Criteria Analysis), метод анализа иерархий (Analytic Hierarchy Process), система сбалансированных показателей (Balanced Scorecard), дельфийский метод (Delphi Hierarchy Process), совокупная факторная производительность (СФП) (Total Factor Productivity) и анализ среды функционирования (АСФ) (Data Envelopment Analysis) [6-11]. АСФ является наиболее подходящим методом оценки продуктивности некоммерческих организаций, так как при измерении эффективности используются множество критериев оценки. Производительность пожарного подразделения классифицируется как множество затрат и выпусков и их функциональная связь не может быть предопределена. Таким образом, в данной системе используется АСФ как основной метод оценки эффективности работы пожарных подразделений.

АСФ, по Чарнзу, Куперу и Роудзу (ЧКР) [12] – это метод оценивания эффективности общественных некоммерческих организаций, но позже стал часто применяться и для оценивания коммерческих организаций. АСФ, как и

СФП – это метод оценивания эффективности, основанный на соотношении затрат и выпуска. При анализе общей эффективности подразделения пытаются оценить эффективность каждой пожарной части с учётом следующих целей: предотвращение стихийных бедствий, проведение спасательных операций, защита людей и их собственности. АСФ применяется в сравнении эффективности при условии, что сравниваются одинаковые субъекты принятия решений (СПР). В данной системе взят за основу АСФ, так как данный метод не только объективно оценивает продуктивность работы пожарных подразделений, определяет, как правильно распределять продуктивность пожарных подразделений, но и предлагает новый виток в развитии всестороннего мышления для измерения продуктивности пожарных подразделений.

1.2 Патентный обзор постановки подобных задач. Анализ стандартных средств и существующих способов решения задачи

Задачи оценки эффективности функционирования различных объектов и систем является актуальной для любой отрасли. Для решения указанной задачи, на мой взгляд, целесообразно использовать метод Data Envelopment Analysis (DEA), который в России известен под названием Анализ среды функционирования (АСФ).

В этой сфере существует как коммерческое ПО [13, 14, 15, 16], так и свободное ПО [17, 18, 19, 20, 21], а есть и такое, которое имеет свободную и коммерческую лицензии [22, 23, 24, 25]. Программы, как правило, предназначены для работы в среде операционной системы (ОС) Microsoft Windows, но есть и такие, которые работают в среде ОС Linux [17, 19]. Среди рассматриваемого ПО есть Web-приложения [24, 20]. Некоторые программы реализуют много моделей метода DEA (АСФ) [22, 17, 13, 18, 23], но есть и такие, которые предлагают лишь несколько моделей [19, 25, 20, 21]. В качестве метода решения задачи линейного программирования, которая лежит в основе метода DEA (АСФ), как правило, используется симплекс-метод, но есть ПО, в

котором применяется метод внутренней точки [25]. Большая часть программ имеет графический интерфейс пользователя, однако, консольные приложения тоже существуют [19, 21]. В качестве языков программирования используются Visual Basic [22, 15], Java [17], Fortran [19, 21], JavaScript [24] и др.

В существующих программных продуктах не используются «большие» системы управления базами данных – Oracle, PostgreSQL, MySQL и др. Данные хранятся в таблицах Excel [22, 15], базе данных формата Microsoft Access (mdb) [23, 16], в текстовых файлах [19, 21].

Таким образом, задача создания удобной, простой и недорогой системы анализа эффективности всё ещё является актуальной.

1.3 Объекты управления, информационные объекты и автоматизируемые процессы. Пользователи и внешние сущности

Объектом управления данной информационной системы является процесс расчёта выбранной DEA модели.

Информационный объект – файл с данными, которые импортируются в систему и которые потом выбираются как входные для модели.

Автоматизируемые процессы в данном проекте – это решения задачи линейного программирования выбранной модели DEA.

Данная система однопользовательская, то есть пользователь – это человек, который производит расчёт эффективности использования ресурсов, выбранными объектами.

Внешней сущностью может выступать заказчик, который получает уже готовые значения расчёта анализа эффективности использования ресурсов, не работая при этом с данными и моделями.

1.4 Цели разработки, функции системы

Необходимо разработать систему анализа эффективности использования ресурсов для регионального управления МЧС.

Целью работы является автоматизация процесса расчёта эффективности использования ресурсов по выбранным подразделениям.

К ключевым задачам можно отнести:

а) в клиентском приложении должна быть осуществлена поддержка плоских файлов;

б) в приложении должны приводиться результаты анализа среды функционирования;

в) в приложении должно быть осуществлено сохранение текущего расчёта;

г) в приложении должен быть осуществлён экспорт полученных результатов анализа.

2 Изучение и моделирование предметной области

2.1 Основные понятия и процессы, их свойства и закономерности.

Классическое исследование эффективности и производительности изучает структуру и закономерности динамики рынка, обосновывает необходимость и возможности конкретного предприятия повысить эффективность производства, технологий и структуры, а также выводимой на рынок продукции или услуг к спросу и требованиям потребителя. Стратегическое исследование эффективности тесно связано с применением принципов, методов, инструментария и философии бенчмаркинга.

Бенчмаркинг используется для сравнения индексов показателей результативности и основан на сравнении этих показателей с оценками лучших и передовых технологий (Best-in-Class) на основе определения расстояния между исследуемым результатом и границей эффективности. Метод анализа оболочки данных (Data Envelopment Analysis, DEA) является широко известным и применяемым на практике аналитическим методом бенчмаркинга. Он представляет собой классическое применение методов линейного программирования к решению задач сравнения многопараметрических объектов по эффективности функционирования и их производительности.

При анализе оболочки данных основополагающим понятием является эффективность, которая определяется как частное от деления взвешенной суммы всех выходных параметров на взвешенную сумму всех входных факторов.

DEA определяет их количества задействованных объектов анализа эффективные единицы путём построения границы эффективности, а для всех остальных – меру их неэффективности, причём эффективность или неэффективность конкретного предприятия определяется принятием собственных решений.

Критерием для выявления эффективности при DEA является достижение оптимума по Парето, а в основе лежит идея относительной

эффективности: предприятие является на 100% эффективным [12, 26, 27], если:

а) ни один из выходных параметров не может быть повышен без повышения одного или более входных факторов либо понижения других выходных параметров;

б) ни один из входных факторов не может быть уменьшен без понижения одного или более выходных параметров либо повышения других факторов.

Такое определение применительно только к понятию относительной эффективности и может быть нестрогим, поскольку истинная эффективность в большинстве случаев неизвестна. 100% относительной эффективности может быть достигнуто только тогда, когда в сравнении с аналогами не существует оснований для неэффективности по отношению к одному или нескольким входным или выходным факторам. Как сопоставимые, при этом, обозначаются такие предприятия, которые при одинаковых входных и выходных факторах преследуют одинаковые цели.

2.1.1 Коэффициент эффективности

Эффективность – это отношение произведённых результатов к количеству используемых ресурсов. «Объём продаж на одного сотрудника», «ВВП на душу населения» или «среднее количество пассажиров на рейс» являются типичным примером коэффициента эффективности. DEA измеряет эффективность объекта (называемой лице принимающее решение или DMU – Decision Making Units), используя взвешенный коэффициент, как показано на формуле 1 ниже:

$$\text{Эффективность} = \frac{u_1 y_{10} + \dots + u_s y_{s0}}{v_1 x_{10} + \dots + v_m x_{m0}}, \quad (1)$$

где x и y – входные и выходные значения, u_s – взвешенная мощность выходных значений и v_m – взвешенная мощность входных значений.

Соотношение выше учитывает все результаты и входы. Этот тип меры называется фактором общей производительности. Все модели DEA реализуют схожие типы коэффициентов производительности, хотя и со своими

специфическими характеристиками. Веса, назначенные каждому входу и каждому выходу, являются переменными, используемые в процессе оптимизации DEA.

2.1.2 Эффективная граница

Эффективная граница представляет лучшую наблюдаемую производительность в наборе данных. Например, на таблице 1 показано количество сотрудников и показателей продаж для некоторых продуктовых магазинов. Коэффициент эффективности (продажи/сотрудник) можно использовать для измерения эффективности различных магазинов.

Таблица 1 – Пример сотрудники/продажи

Магазин	A	B	C	D	E	F	G	H
Сотрудник	2	1.5	3	4	5	5	6	8
Продажи	1	1	2	3	5	2	3	5
Продажи/сотрудник	0.5	0.667	0.667	0.75	1	0.4	0.5	0.625

При построении этих данных с сотрудниками по оси x (горизонтальная ось) и продажи по оси y (вертикальная ось) будет получен график, изображённый на рисунке 1.

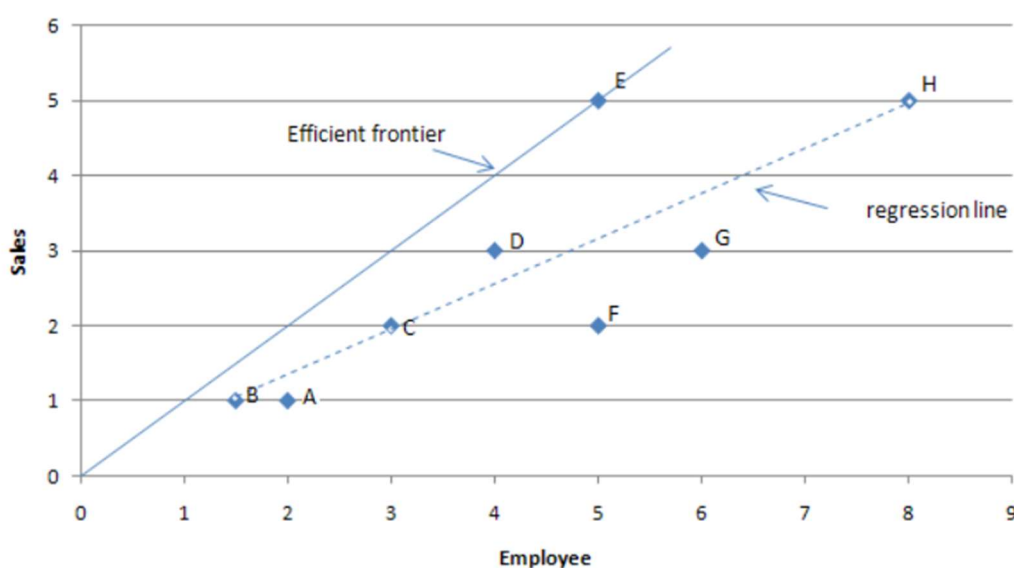


Рисунок 1 – Графическое вычисление коэффициента эффективности

Таблица и график позволяют найти лучших исполнителей. В этом случае магазин Е показывает лучшую производительность с эффективностью 1. Линия, которая проходит от начала координат до магазина Е, является границей эффективности. Граница эффективности иллюстрирует наилучшую наблюдаемую производительность, и в математике говорится, что она «охватывает» данные; отсюда и название анализ охвата данных (Data Envelopment Analysis). Регион, охваченный эффективной границей, называется «Набор производственных возможностей». Это область возможного производства (на основе наблюдаемой наилучшей производительности).

В DEA производительность DMU всегда рассчитывается путём сравнения этой области с границей эффективности, непосредственно определённой из данных.

2.1.3 Приближение к границе эффективности

Все DMU, которые не находятся на границе эффективности, являются неэффективными. Неэффективное подразделение должно, следовательно, достичь границы эффективности, чтобы стать эффективным. Есть три варианта:

- уменьшить входы, сохраняя при этом выходы постоянными (это подход, ориентированный на вход);
- увеличить выходы, сохраняя при этом входы постоянными (это подход, ориентированный на выход);
- увеличение выходов и уменьшение входов одновременно (это может быть сделано с неориентированными версиями моделей, такими как модель SBM).

По сути, чем дальше от границы находится объект, тем хуже его производительность.

Это можно оценить на рисунке 2, смотря на DMU С на предыдущем рисунке.

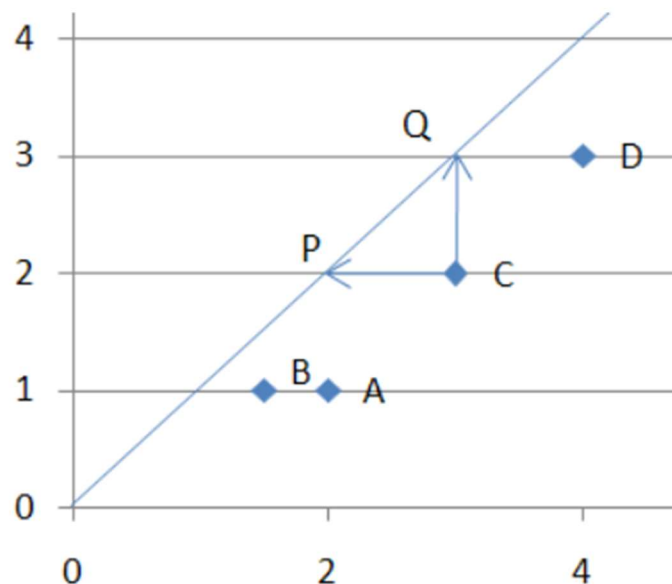


Рисунок 2 – Пример достижения границы эффективности

На этом рисунке можно увидеть как DMU C может либо достичь границы эффективности, уменьшив свои входы (сохраняя свои уровни выхода) и достигнув P, либо увеличив свою производительность (сохраняя постоянными свои входные значения) и достигнув Q.

2.1.4 Отдача масштаба (Returns to scale)

Важно отметить, что пограничные изображения на графиках выше предполагаются растянутыми до бесконечности, то есть, что уровни производительности DMU E (единственно эффективного магазина) возможны независимо от количества сотрудников в магазине. Это называется постоянной отдачей масштабу (Constant RTS). Хотя постоянное допущение RTS иногда верно для локального диапазона производства, его иногда нужно ослаблять. Это возможно, например, при возврате переменной в масштабе, как показано на рисунке 3 ниже.

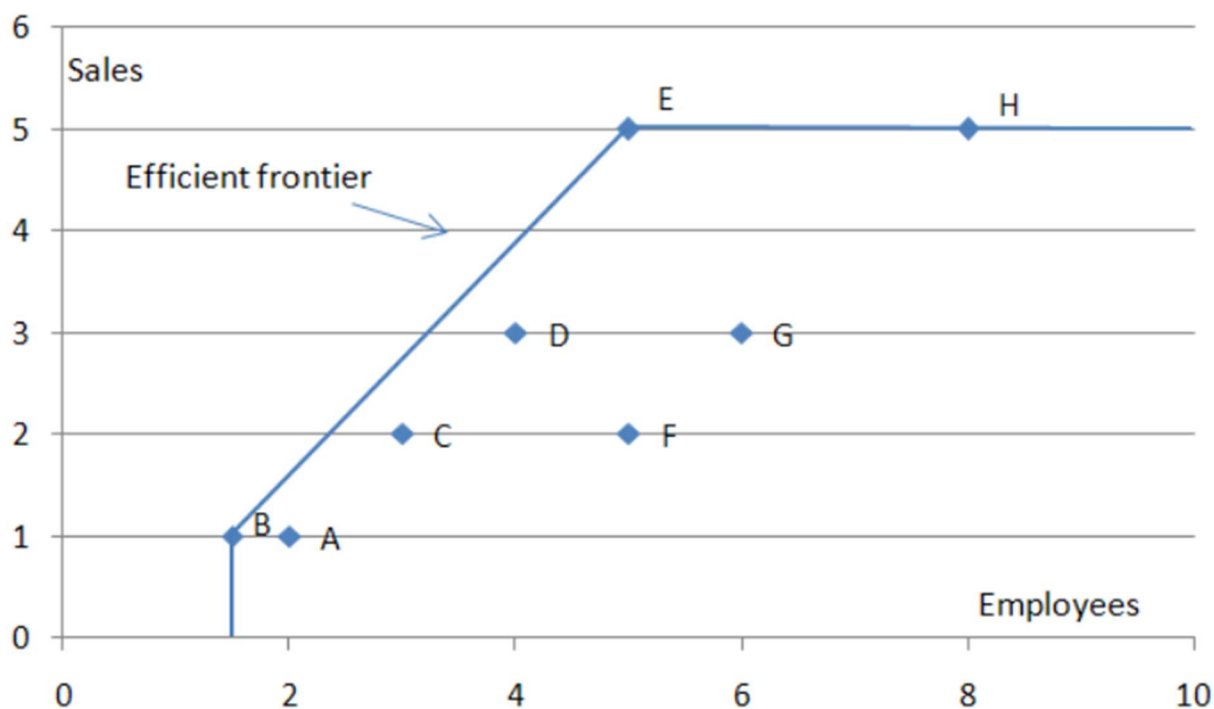


Рисунок 3 – Пример отдачи масштаба

Важно отметить, что при переменной отдаче масштаба DMU В становится эффективным. А хотя DMU Н находится на границе эффективности, он не эффективен. Это вызвано тем, что DMU Е производит аналогичное количество продаж (то есть 5), но использует для этого на 3 единицы меньше сотрудников. Для того, чтобы Н был эффективен, ему нужно будет сократить свои продажи на 3 сотрудника, чтобы достичь координат Е. DMU Н также может стать эффективным за счёт увеличения продаж. Однако нет никакого способа точно узнать, возможно ли это, поскольку такие уровни производства не наблюдались (то есть выше текущей границы эффективности).

Существуют и другие типы возвратов к шкале, такие как Increasing Returns to Scale (иногда также называемое Non Decreasing Returns to Scale [28]), которые предполагают, что невозможно уменьшить масштаб DMU, но что RTS может стремиться к бесконечности. Другим типом RTS, который является противоположным допущению IRTS, является Decreasing Returns To Scale (иногда называемое non-increasing RTS). Наконец, существует общая модель

RTS, которая позволяет контролировать, насколько масштаб DMU может быть уменьшен или увеличен.

2.1.5 Технические и смешанные неэффективности

Существует два типа неэффективности:

- Техническая неэффективность;
- Смешанная неэффективность.

Следующий рисунок 4 иллюстрирует производительность различных складов автомобилей на основе 2 выходов: использование и продажи, и один вход: количество автомобилей. Можно отметить использование каждого склада, нанеся на график значения загрузки/автомобили и соотношения продажи/автомобили.

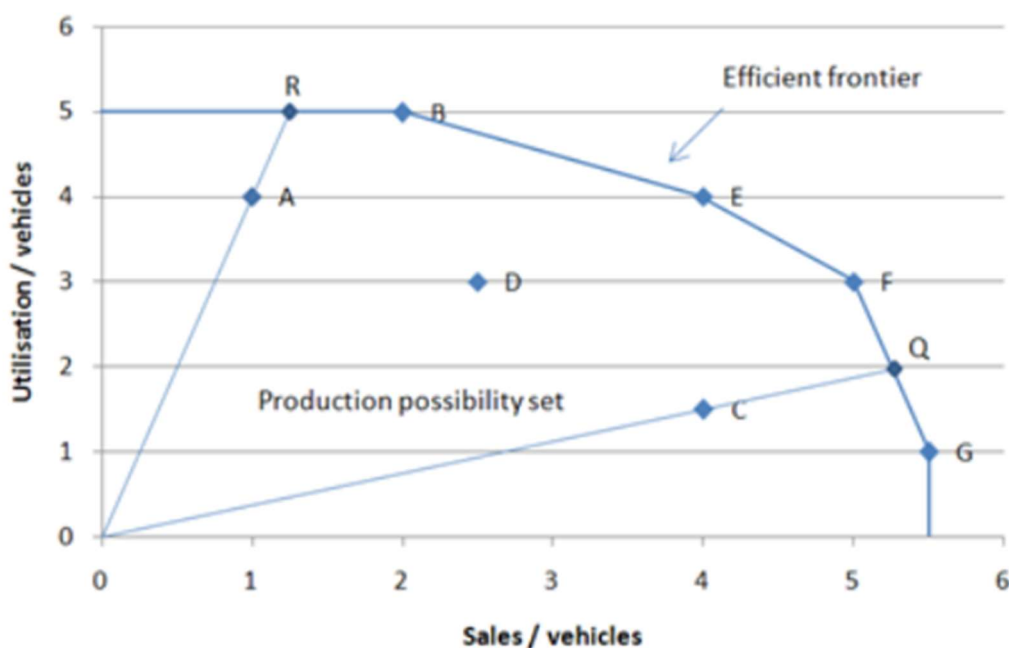


Рисунок 4 – Пример технической/смешанной неэффективности

На этом графике DMU C неэффективен, так как он не находится на границе эффективности. Для того чтобы DMU C стал эффективным, он должен достичь границы, проецируя на Q. Это радиальная проекция соответствует технической неэффективности DMU C. DMU A также неэффективен, так как не находится на эффективной границе. Однако при

проецировании на эффективную границу R (техническая неэффективность) DMU A всё ещё не эффективен, поскольку существует ещё один DMU (DMU B), который демонстрирует пропорционально более высокий объём продаж на транспортные средства. Для того, чтобы DMU A стал эффективным, ему сначала необходимо проецировать на границу эффективности, а затем увеличивать свои продажи на единицу продукции, пока не достигнет B (смешанная неэффективность). Степень улучшения между Q и B измеряется неотрицательным провалом A в показателях продаж.

Таким образом, техническая неэффективность может быть устранена без изменения пропорций между входами и выходами, в то время как смешанная неэффективность может быть устранена только путём изменения пропорции между входами и выходами. Модели DEA используют разные подходы к оценке технической к смешанной неэффективности. DMU эффективны, когда они не проявляют технической неэффективности и не смешивают неэффективность.

2.1.6 Группы равных (Peers)

На приведённом выше рисунке 4, проекция DMU C на эффективную границу, находится между DMU F и DMU G. DMU F и DMU G называются группой равных (или эталонный набор) DMU C. Группа равных данного DMU состоит из списка эффективных DMU, производительность которых была использована для расчёта эффективности данного DMU.

2.2 Теоретическое изучение предметной области. Построение теоретических математических моделей

Как объяснялось ранее, DEA использует коэффициент общей производительности факторов для измерения эффективности использования ресурсов (то есть уникальное соотношение со всеми входными и выходными значениями). DEA присваивает виртуальный вес каждому из этих входных и выходных данных.

Производительность объектов рассчитывается с использованием процесса линейной оптимизации, который пытается максимизировать коэффициент каждого объекта путём нахождения наилучшего набора веса для этого конкретного объекта.

Процесс оптимизации противоречит существующим данным, так что каждый объект сравнивается с лучшими наблюдаемыми показателями.

Базовыми моделями DEA-анализа являются прямая и двойственная CCR- (Charnes, Cooper, Rhodes – Чарнеса, Купера и Родоса) и ВСС-модели (Бранкера, Чарнеса и Купера), которые представлены в таблице 2.

Таблица 2 – Базовые DEA-модели.

Название	Функциональное представление DEA-модели	Характеристика
Двойственная input-ориентированная CCR-input модель	$\sum_{j=1}^s \mu_j y_{j0} \rightarrow \max$ <p>при $\sum_{i=1}^r t_i x_i = 1; \mu_j, t_i \geq 0$</p>	Максимальный эффект при минимизации взвешенных входов по отношению к любому нормированному выходу
Двойственная input-ориентированная ВСС-модель	$\sum_{j=1}^s \mu_j y_{j0} + u_0 \rightarrow \max$ <p>при $\sum_{i=1}^r t_i x_i = 1; \mu_j, t_i \geq 0$</p>	Максимальный эффект при минимизации взвешенных входов к любому выходу эффекту масштаба и изменения входных параметров x относительно выходных y .
Двойственная суммарная модель VarMulti	$\sum_{j=1}^s \mu_j \log(y_{j0}) - \sum_{i=1}^r t_i \log(x_{i0}) \rightarrow \max$ <p>при $\sum_{i=1}^r t_i x_i = 1; \mu_j, t_i \geq 0$</p>	Интерпретируется для производственных функций

Примечание: использованы обозначения x_i – входные показатели, y_i – выходные показатели, t_i и μ_i – весовые коэффициенты входных и выходных

показателей соответственно, i – число единиц, которые сравниваются; r – число входных факторов; s – число выходных параметров.

Прямая модель определяет изменение входных показателей по отношению только к одному выходному показателю. Двойственная модель определяет отношение изменения входных показателей по отношению к любому набору выходных показателей.

2.3 Экспериментальное изучение предметной области

Метод DEA (АСФ) основан на построении так называемой границы эффективности в многомерном пространстве входных и выходных переменных, описывающих объекты, эффективность которых требуется определить. Граница эффективности представляет собой гиперповерхность, огибающую (охватывающую) точки, соответствующие оцениваемым объектам. Степень эффективности конкретного объекта зависит от расстояния между ним и границей эффективности: чем дальше объект находится от границы, тем его эффективность ниже. Объекты, находящиеся на границе эффективности, считаются эффективными.

Представим формализованное описание метода на примере одной из его моделей. Пусть требуется определить показатель эффективности каждого из n объектов. Для описания каждого объекта $o_j, j = \overline{1, n}$, служит пара векторов (x_j, y_j) . При этом вектор $x_j = (x_{j1}, \dots, x_{jr}, \dots, x_{jm})^T$ содержит входные показатели (входы) для объекта o_j , а вектор $y_j = (y_{j1}, \dots, y_{jr}, \dots, y_{js})^T$ содержит выходные показатели (выходы) для объекта o_j . Тогда матрица $X = (x_j)$, имеющая размерность $m \times n$, содержит вектор-столбцы с входными данными для всех n объектов, а матрица $Y = (y_j)$, имеющая размерность $s \times n$, содержит вектор-столбцы с выходными данными для всех n объектов. В основе метода DEA (АСФ) лежит метод линейного программирования, поэтому модель формируется в таком виде [15, с. 43]:

$$\begin{aligned}
& \min_{\theta, \lambda} (\theta), \\
& \theta x_j - X\lambda \geq 0, \\
& -y_j + Y\lambda \geq 0, \\
& \lambda \geq 0
\end{aligned} \tag{2}$$

Скаляр θ и является мерой эффективности j -го объекта, $\theta \in (0; 1]$. Объекты, имеющие значения показателя $\theta = 1$, считаются эффективными и находятся, как принято говорить, на границе эффективности. Аналогичная задача решается для каждого объекта, т.е. n раз.

Таким образом, граница эффективности формируется путём многократного решения задачи линейного программирования.

3 Разработка информационной базы для решения задачи

3.1 Построение модели данных

Среда реализации постоянного хранения данных: текстовый файл, в котором содержатся данные.

Модель данных DEA модели представлена на рисунке 5.

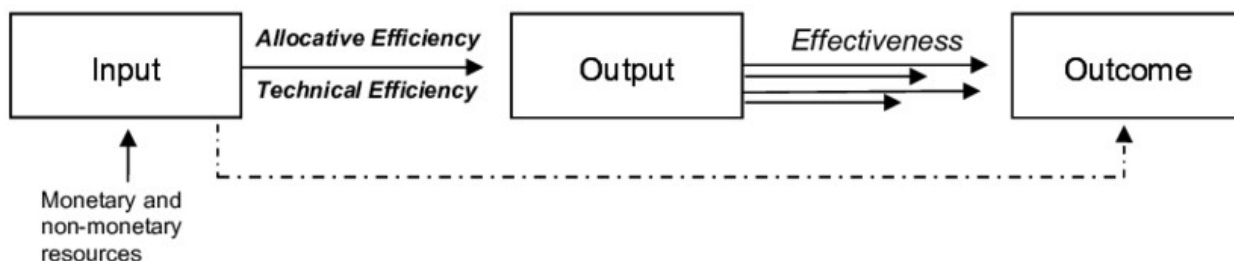


Рисунок 5 –Модель данных

3.2 Описание источников информации, входных сигналов и документов

Входной информацией являются входные/выходные данные для DEA модели, полученные из импортируемого файла в формате CSV UTF-8 (*.csv).

Включает в себя столбец названий объектов анализа и входные данные для анализа (Рисунок 6).

DMU Names	Число дежурного персонала	Финансовые издержки	Количество пожарных машин	Плата за обслуживание трансп
Sinying fire ...	18.0	268.575	6.0	175.795
Yanshuei fire...	7.0	188.476	3.0	36.459
Liouying fire...	7.0	115.367	3.0	12.062
Baihe fire br...	8.0	210.896	4.0	35.368
Guanling fire...	5.0	82.326	2.0	3.35
Houbi fire br...	7.0	125.684	3.0	38.511
Dongshan fir...	7.0	97.813	3.0	41.086
Dongshan fi...	7.0	57.188	2.0	2.25

Рисунок 6 – Пример данных для анализа

3.3 Описание выходной информации: сигналов, документов и видеок кадров

Выходной информацией является результаты анализа среды функционирования (Data envelopment analysis), а именно:

- Оценки эффективности объектов анализа (Objectives).

Значения относительной эффективности согласно текущей модели. Общая формула расчёта эффективности, она отличается в зависимости от выбранной модели анализа, выглядит так:

$$\text{Техническая эффективность} = \frac{\sum \text{взвешенные выходные параметры}}{\sum \text{взвешенные входные параметры}}, \quad (3)$$

- Проекции (Projections).

Проецирование неэффективных объектов анализа, на границу эффективности.

- Группы равных (Peer Group).

Группа равных неэффективного объекта включает в себя эффективные объекты, с ближайшими значениями входных/выходных значений в сравнении со сравниваемым объектом.

- Оставшиеся резервы (Slacks).

Показывают возможное дальнейшим увеличением объёма производства (output) или сокращения расходуемых ресурсов (input).

- Факторы взвешивания (Lambdas) и веса (Weights).

Необходимы при проведении суммирования, интегрирования или усреднения с целью придания некоторым элементам большего веса в результирующем значении по сравнению с другими элементами.

Результаты анализа можно сохранить в виде файла в формате DEA Problem file (*.deap) и после открыть в разработанной системе или экспортировать в файл в формате Excel 2007 (*.xls) или Excel 2010 (*.xlsx).

Результаты анализа представлены в виде таблиц по каждому объекту анализа (DMU).

4 Программно-аппаратная реализация решения задачи

4.1 Аппаратное обеспечение

При разработке системы использовались следующие аппаратные средства:

- процессор Intel® Core™ i3-6100 CPU @ 3.70 GHz, 3.70 GHz;
- объём оперативной памяти 16 GB;
- 64-разрядная система;
- видеоадаптер NVIDIA GeForce GTX 1050 Ti;
- диски 120 GB SSD + 932 GB HDD;
- минимальный набор периферийного оборудования (клавиатура, мышь, монитор).

Система состоит из одного клиентского приложения и внешней библиотеке lp_solve.

Минимальные системные требования:

- Операционная система Windows 7 x64 (64-bit) SP1 или выше, или Ubuntu Linux x64 (64-bit) версии 19.04 или выше;
- Процессор с тактовой частотой 1,6 ГГц;
- ОЗУ объёмом 1 ГБ;
- Видеокарта;
- 350 МБ доступного пространства на жёстком диске;
- минимальный набор периферийного оборудования (клавиатура, мышь, монитор).

Для работы программы в системе должна быть установлена библиотека lp_solve 5.5.

4.2 Программное обеспечение

Для корректной работы системы анализа эффективности использования ресурсов требуются следующие программные средства:

- Java Runtime Environment 11;
- установленная система Windows 7 x64 (64-bit), с пакетом обновления 1 (SP1) или выше, или Ubuntu Linux x64 (64-bit) версии 19.04 или выше;
- библиотека `lp_solve 5.5` установленная в системе (под каждую операционную систему своя версия этой библиотеки).
- Excel 2007 или новее, или любой другой редактор электронных таблиц.

Импортирование данных в систему выполняется с помощью файла в формате csv, который можно редактировать в редакторе электронных таблиц.

4.3 Разработанные программные средства

4.3.1 Описание использованных средств, подходов, методов, языков, библиотек

Система разработана на языке Java. В качестве инструментальных средств программирования использовались:

- среда разработки IntelliJ IDEA – интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains.

В первую очередь от IDE требовалось следующее:

- Поддержка синтаксиса. Подсветка ключевых слов, автоподстановка, подсветка ошибок.
- Навигация. Переход к объявлению, поиск использований, поиск по текстовой строке, файлу или названию и т.д.
- Анализ. Иерархия классов и вызовов, а также свойства и действия класса.
- Рефакторинг. Переименование классов, свойств и действий.
- Визуализация форм. Отображению разработчику текущего дизайна определенной формы.
- Отладчик. Возможность ставить breakpoint'ы (в том числе и с условиями), отлаживать императивную логику, смотреть watches.

- среда разработки Eclipse – свободная интегрированная среда разработки модульных кроссплатформенных приложений. Развивается и поддерживается Eclipse Foundation.

Eclipse написана на Java, потому является платформо-независимым продуктом, за исключением библиотеки SWT, которая разрабатывается для всех распространённых платформ (см. ниже).

- Standard Widget Toolkit, или SWT – библиотека с открытым исходным кодом для разработки графических интерфейсов пользователя на языке Java.

SWT не является самостоятельной графической библиотекой, а представляет собой кросс-платформенную оболочку для графических библиотек конкретных платформ, например, под Linux SWT использует библиотеку GTK+. SWT написана на стандартной Java и получает доступ к OS-специфичным библиотекам через Java Native Interface, который рассматривается в качестве сильного средства несмотря на то, что это не является чистой Java.

SWT – альтернатива AWT и Swing (Sun Microsystems) для разработчиков, желающих получить привычный внешний вид программы в данной операционной системе. Использование SWT делает Java-приложение более эффективным, но снижает независимость от операционной системы и оборудования, требует ручного освобождения ресурсов и в некоторой степени нарушает Sun-концепцию платформы Java.

- библиотека lp_solve – это свободно распространяемое решение для решения задач линейного программирования, основанная на пересмотренном симплекс-методе и метода Branch-and-bound для целых чисел.

lp_solve написан на ANSI C и может быть скомпилирована на многих различных платформах, таких как Linux и Windows. По сути, lp_solve – это библиотека, набор подпрограмм, которые можно легко вызывать из языков программирования, таких как C, C++, C# и VB. К сожалению, не существует

простого способа использовать нативные библиотеки C, такие как `lp_solve`, в программах на Java. Однако существует так называемый «Java wrapper» (Java-упаковщик), которая предназначена для устранения этого недостатка. Он состоит из двух основных частей:

- библиотека классов Java, используемая клиентскими программами Java. Она предоставляет доступ ко всем подпрограммам `lp_solve` через `LpSolve` класс.

- Собственная библиотека, написанная на C++, также называемая библиотекой-заглушкой, которая использует API JNI (Native Interface) для преобразования вызовов методов Java в вызовы соответствующих подпрограмм библиотеки `lp_solve`. Клиентские программы Java не взаимодействуют напрямую с библиотекой-заглушкой. Эта библиотека должна быть скомпилирована для каждой целевой платформы. Предварительно скомпилированные двоичные файлы включены для операционных систем Windows и Linux. Существует также скрипт сборки для компиляции библиотеки-заглушки в Mac OS X.

Текущая версия оболочки была написана для работы с `lp_solve 5.5.0.9` и была протестирована под Windows XP и Linux. Оболочка требует Java Runtime Environment 1.3 или более поздней версии.

- библиотека OSDEA – это свободно распространяемая библиотека, написанная на языке Java и основанная на алгоритмах анализа среды функционирования (DEA) В ней используется библиотека `lp_solve` и реализована имплементация «Java wrapper».

Ключевые возможности включают в себя:

- неограниченное количество единиц принятия решений (DMU);
- неограниченное количество переменных;
- доступно до 40 моделей DEA;

- можно использовать недискреционные и неконтролируемые переменные (то есть переменные, которые субъекты не могут полностью контролировать);
- полный доступ к решению.

4.3.2 Описание программы

4.3.2.1 Общие сведения

Наименование программы: «Система анализа эффективности использования ресурсов».

Программное обеспечение необходимое для функционирования программа – операционная система Linux/Windows 7 и старше; Java Runtime Environment 11; установленная в системе библиотека lp_solve версии 5.5.

Система написана на языке программирования Java с использованием нативной библиотеки lp_solve.

4.3.2.2 Функциональное назначение

Программное решение предназначено для автоматизации процессов расчётов эффективности использования ресурсов подразделений МЧС.

Созданная программа позволяет решить около 40 разных DEA моделей и предлагает детализированное решение для каждой из них. Модели, которые данная система предлагает решить:

- CCR (Charnes Cooper and Rhodes model). Модель доступна в Input и Output ориентированных версиях;
- BCC (Banker, Charnes & Cooper model). Модель доступна в Input и Output ориентированных версиях;
- SBM (Slack Based Model – Model developed by Kaoru Tone). Модель доступна в Input и Output ориентированных версиях, а также с переменной, увеличивающейся, уменьшающейся и обобщённой отдачей масштаба;
- IRS (Increasing Return to Scale – technical efficiency). Модель доступна в Input и Output ориентированных версиях;

- DRS (Decreasing Return to Scale – technical efficiency). Модель доступна в Input и Output ориентированных версиях;
- GRS (Generalized Return to Scale – technical efficiency). Модель доступна в Input и Output ориентированных версиях;
- NC (Non-Controllable Model – позволяет установить некоторые переменные неконтролируемыми, это переменные, в которых нет оставшихся резервов). Доступна в IRS, DRS, GRS and input/output ориентированных версиях;
- ND (Non-Discretionary Model – позволяет установить некоторые переменные недискреционными, это переменные для которых могут быть оставшиеся резервы, но они не влияют на оценку). Доступна в IRS, DRS, GRS and input / output ориентированных версиях.

4.3.2.3 Описание логической структуры

Алгоритм работы приложения представлен в виде диаграммы деятельности на рисунке 7.

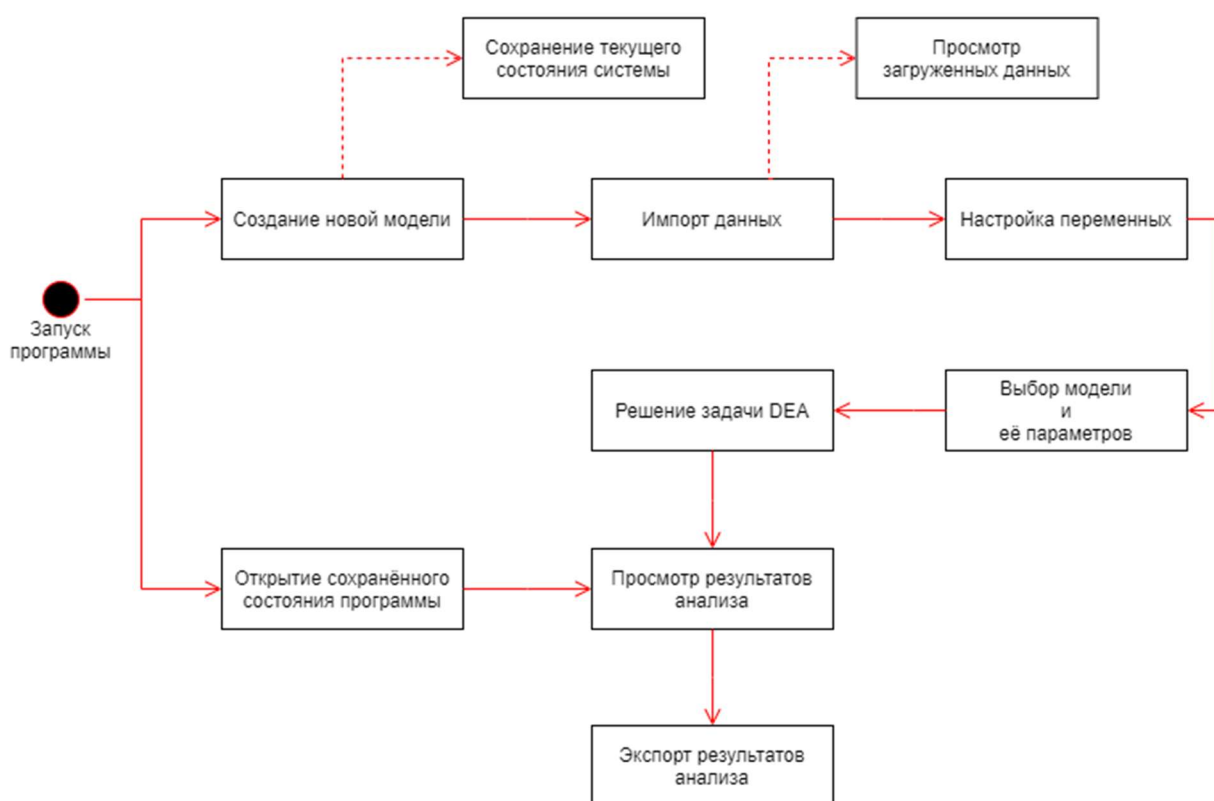


Рисунок 7 – Диаграмма деятельности системы

Используемые методы:

- решение выбранной DEA модели;
- импорт данных в программу;
- экспортирование данных;
- сохранение данных.

Структура программы представлена на рисунке 8.

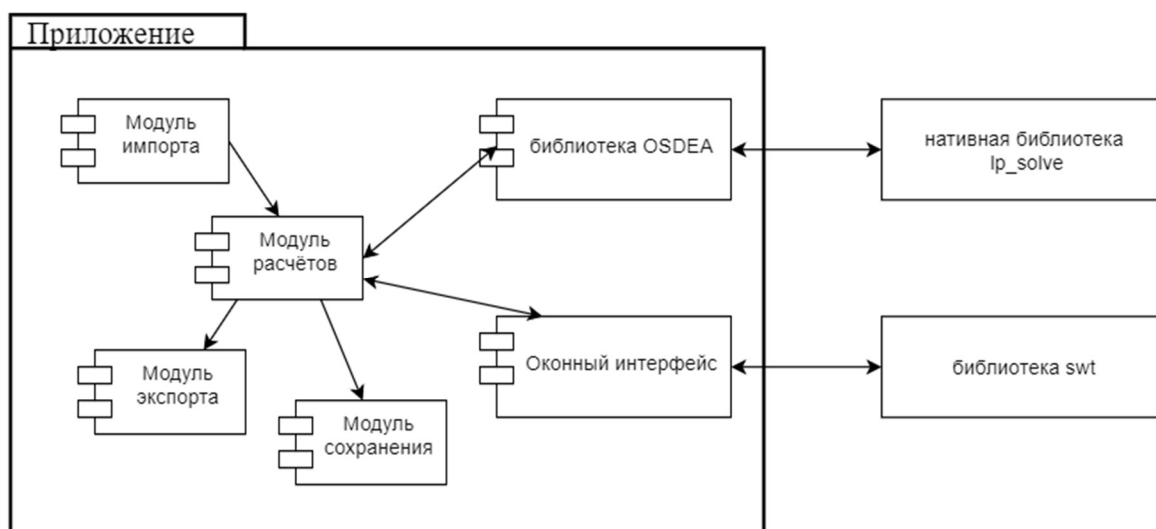


Рисунок 8 – Структура программы

Описание функций структурных модулей программы:

- Оконный интерфейс приложения отрисовывается на экране, используя кросс-платформенную оболочку swt для графических библиотек конкретных платформ (например, под Linux SWT использует библиотеку GTK+);
- Модуль импорта отвечает за импорт данных для анализа в систему;
- После импорта данные попадают в модуль расчётов, в котором производится решение модели DEA;

– При расчёте модели DEA модуль расчётов обращается к библиотеке OSDEA, которая в свою очередь является Java-wrapper'ом для нативной внешней библиотеки lp_solve;

– После результаты анализа среды функционирования могут быть экспортированы, используя модуль экспорта, либо сохранены, используя модуль сохранения.

4.3.2.4 Используемые технические средства

При разработке системы использовались следующие аппаратные средства:

- процессор Intel® Core™ i3-6100 CPU @ 3.70 GHz, 3.70 GHz;
- объём оперативной памяти 16 GB;
- 64-разрядная система;
- видеоадаптер NVIDIA GeForce GTX 1050 Ti;
- диски 120 GB SSD + 932 GB HDD;
- минимальный набор периферийного оборудования (клавиатура, мышь, монитор).

Система состоит из одного клиентского приложения и внешней библиотеке lp_solve.

Минимальные системные требования:

- Операционная система Windows 7 x64 (64-bit) SP1 или выше, или Ubuntu Linux x64 (64-bit) версии 19.04 или выше;
- Процессор с тактовой частотой 1,6 ГГц;
- ОЗУ объёмом 1 ГБ;
- Видеокарта;
- 350 МБ доступного пространства на жёстком диске;
- минимальный набор периферийного оборудования (клавиатура, мышь, монитор).

Для работы программы в системе должна быть установлена библиотека lp_solve 5.5.

4.3.2.5 Вызов и загрузка

До запуска приложения в системе должна быть установлена библиотека lp_solve версии 5.5, соответствующая используемой операционной системе. В папке с библиотекой прилагается инструкция по установке.

Также в системе должна быть установлена JRE 11 (можно в составе JDK 11).

В соответствии с системой Windows/Linux/Mac и её разрядностью x64\х86 должна использоваться соответствующая библиотека swt. Нужно из папки SWT выбрать соответствующую системе пользователя swt.jar и загрузить его в папку OSDEA-v0.2_lib (по умолчанию в папке уже лежит swt.jar для Windows x64).

Само приложение не требует установки и доступно к работе сразу после скачивания программы и установки необходимых зависимостей (lp_solve 5.5, JRE 11 и swt.jar).

Чтобы запустить программу пользователь должен запустить ярлык с названием в папке программы «start».

4.3.2.6 Входные данные

Для начала работы с приложением необходимо сперва выбрать данные для анализа. Данные должны быть в файле в формате CSV UTF-8 (*.csv). Первый столбец в файле должен содержать объекты анализа (DMU). Первая строка данных в файле должна содержать названия объектов анализа и переменных и также не используется в расчётах. Остальные поля будут считаться как переменные, используемые в модели расчётов.

После выбора и подготовки данных для анализа они могут быть импортированы в систему.

Также входными данными являются выборы пользователем в системе модели для анализа, параметров моделей и типы переменных (input/output и т.п.).

Также пользователь может открыть, созданные в данной системе модели в формате dear и продолжить работу с ними.

4.3.2.7 Выходные данные

Выходными данными системы анализа использования ресурсов являются:

- таблица с оценками эффективности и статусом эффективности по каждому DMU;
- таблица проекций на каждое значение;
- таблица значения параметров Lambdas для каждой переменной;
- таблица групп равных для каждого DMU;
- таблица оставшихся резервов для каждой переменной;
- таблица весовых мощностей для каждой переменной.

Все эти таблицы отображаются в окне программы и их можно экспортировать в виде xls или xlsx файла.

Также на любом этапе работы программы, можно сохранить её текущее состояние в формате dear и потом открыть с помощью этой же программы.

4.3.3 Описание применения программы

4.3.3.1 Назначение программы

Программное решение предназначено для решения задачи анализа эффективности использования ресурсов регионального управления МЧС.

Разработанное приложение позволяет экспортировать результаты анализа в удобном и популярном формате электронных таблиц.

Хотя приложение предполагается для решения вопросов связанных с использованием ресурсов в региональном управлении МЧС, оно способно

решать любые типы задач анализа эффективности использования ресурсов, используя методы DEA.

4.3.3.2 Условия применения

Минимальные системные требования:

- Операционная система Windows 7 x64 (64-bit) SP1 или выше, или Ubuntu Linux x64 (64-bit) версии 19.04 или выше;
- Процессор с тактовой частотой 1,6 ГГц;
- ОЗУ объёмом 1 ГБ;
- Видеокарта;
- 350 МБ доступного пространства на жёстком диске;
- минимальный набор периферийного оборудования (клавиатура, мышь, монитор).

Для работы программы в системе должна быть установлена библиотека lp_solve 5.5 и Java Runtime Environment 11.

Для просмотра и редактирования импортируемых данных необходим любой текстовый редактор.

Для просмотра экспортируемых результатов анализа необходим редактор электронных таблиц (Excel, LibreOffice или Google Таблицы).

4.3.3.3 Описание задачи

Основные задачи, решаемые информационной системой – автоматизация решения анализа среды функционирования (АСФ), а также предоставление необходимые данные для объективной оценки продуктивности использования ресурсов пожарными подразделениями.

Модели DEA, которые позволяет решить данная система:

- CCR (Charnes Cooper and Rhodes model). Модель доступна в Input и Output ориентированных версиях;

- BCC (Banker, Charnes & Cooper model). Модель доступна в Input и Output ориентированных версиях;
- SBM (Slack Based Model – Model developed by Kaoru Tone). Модель доступна в Input и Output ориентированных версиях, а также с переменной, увеличивающейся, уменьшающейся и обобщённой отдачей по масштабу;
- IRS (Increasing Return to Scale – technical efficiency). Модель доступна в Input и Output ориентированных версиях;
- DRS (Decreasing Return to Scale – technical efficiency). Модель доступна в Input и Output ориентированных версиях;
- GRS (Generalized Return to Scale – technical efficiency). Модель доступна в Input и Output ориентированных версиях;
- NC (Non-Controllable Model – позволяет установить некоторые переменные неконтролируемыми, это переменные, в которых нет оставшихся резервов). Доступна в IRS, DRS, GRS and input/output ориентированных версиях;
- ND (Non-Discretionary Model – позволяет установить некоторые переменные недискреционными, это переменные, для которых могут быть оставшиеся резервы, но они не влияют на оценку). Доступна в IRS, DRS, GRS and input / output ориентированных версиях.

4.3.3.4 Входные и выходные данные

Входными данными являются:

Список объектов анализа (DMU)

Соответствующие им входные и выходные значения (выбрать тип к какому относятся данные пользователь может в программе) (Рисунок 9).

Reset

VARIABLES

INPUTS

Число дежурного персонала
Финансовые издержки
Количество пожарных машин
Плата за обслуживание транс

NON-DISCRETIONARY INPUTS

NON-CONTROLLABLE INPUTS

OUTPUTS

Количество случаев пожаров
Количество других спасательных мероприятий

NON-DISCRETIONARY OUTPUTS

NON-CONTROLLABLE OUTPUTS

Рисунок 9 – Выбор типа параметров

Выходными данными являются результаты анализа, представленные в виде таблиц по каждому объекту анализа. Полученные решения включают в себя:

- Оценки подразделения (Рисунок 10);

DMU Names	Objective Value	Efficient
Sinying fire b...	1	Yes
Yanshuei fire ...	1	Yes
Liouying fire ...	1	Yes
Baihe fire bra...	0.871	
Guanling fire...	1	Yes
Houbi fire br...	0.943	

Рисунок 10 - таблица с оценками эффективности и статусом эффективности по каждому DMU

- Значения проекций подразделения на границу эффективности (Рисунок 11);

DMU Names	Число дежурного персонала	Финансовые издержки	Количество пожарных машин	Пл
Sinying fire branch	18.0	268.575	6.0	17
Yanshuei fire branch	7.0	188.476	3.0	36
Liouying fire branch	7.0	115.367	3.0	12
Baihe fire branch	6.969	171.348	3.14	30
Guanling fire branch	5.0	82.326	2.0	3.3

Рисунок 11 – Таблица проекций на каждый параметра

– Значения параметра λ для каждой входного/выходного параметра (Рисунок 12);

DMU Names	Sinying fire branch	Yanshuei fire branch	Liouying fire branch	Guanling fire branch
Sinying fire branch	1.0	0.0	0.0	0.0
Yanshuei fire branch	0.0	1.0	0.0	0.0
Liouying fire branch	0.0	0.0	1.0	0.0
Baihe fire branch	0.0	0.0	0.0	1.0
Guanling fire branch	0.0	0.0	0.0	0.0

Рисунок 12 – Значения параметра λ для каждой входного/выходного параметра

– Группы равных (Peer Group) (Рисунок 13);

DMU Names	Peer Group
Sinying fire branch	Sinying fire branch, Yanhang fire branch.
Yanshuei fire branch	Yanshuei fire branch.
Liouying fire branch	Liouying fire branch.
Baihe fire branch	Yanshuei fire branch, Guanling fire branch, Sigang fire branch.
Guanling fire branch	Guanling fire branch.
Houbi fire branch	Yanshuei fire branch, Liouying fire branch, Guanling fire branch, Dor

Рисунок 13 – Группы равных

– Оставшиеся резервы (Slacks) для каждого входного/выходного параметра (Рисунок 14);

DMU Names	Число дежурного персонала	Финансовые издержки	Количество пожарных машин
Sinying fire branch	0.0	0.0	0.0
Yanshuei fire branch	0.0	0.0	0.0
Liouying fire branch	0.0	0.0	0.0
Baihe fire branch	0.0	12.37	0.345
Guanling fire branch	0.0	0.0	0.0
Houbi fire branch	0.0	0.0	0.0

Рисунок 14 – Оставшиеся резервы

– Весы для каждого входного/выходного параметра (Рисунок 15).

DMU Names	Число дежурного персонала	Финансовые издержки	Количество пожарных машин
Sinying fire branch	0.046	0.172	0.0
Yanshuei fire branch	0.0	0.334	0.08
Liouying fire branch	0.1	0.656	0.092
Baihe fire branch	0.176	0.615	0.114
Guanling fire branch	1.157	3.047	0.063

Рисунок 15 – Значения весов для каждого входного/выходного параметра

4.3.4 Описание результатов работы программы

В ходе работы было разработано приложение анализа эффективности использования ресурсов с удобным интерфейсом и широким набором моделей для анализа. Использование нативной библиотеки на языке С обеспечивает эффективную скорость решения моделей, а выбор основным языком программирования Java позволяет программе быть кросс-платформенной.

В приложении реализованы следующие функции:

– импорт данных (Рисунок 16);

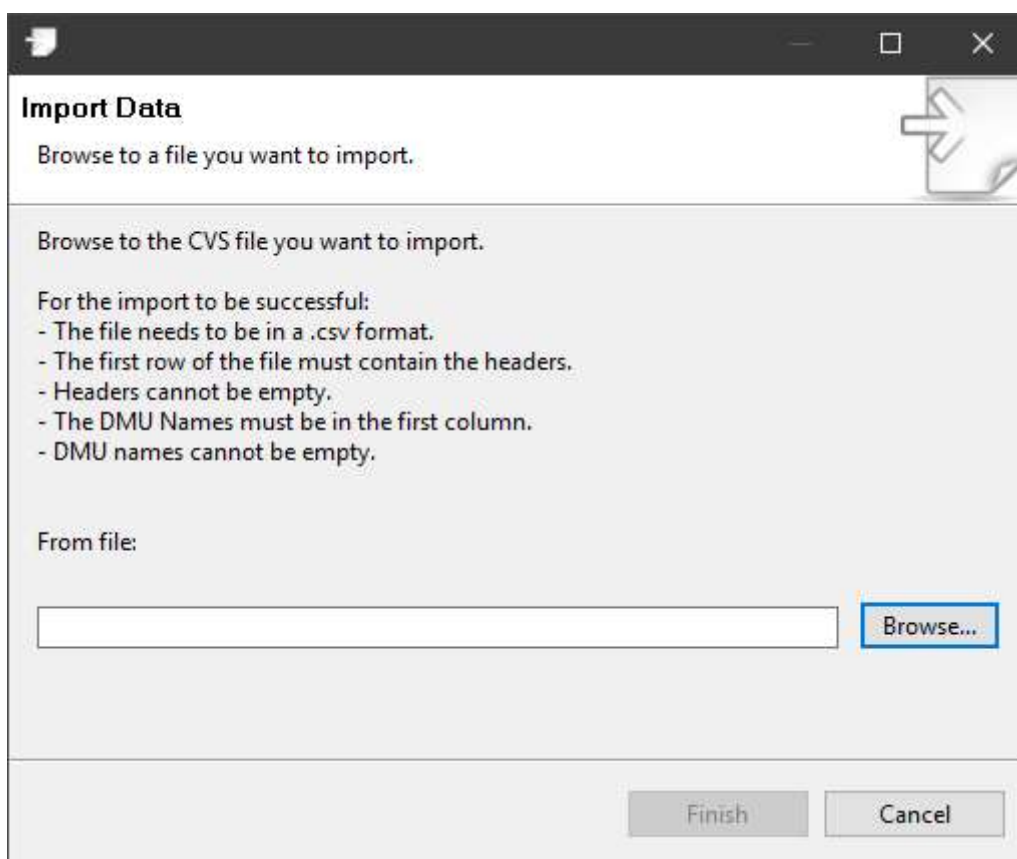


Рисунок 16 – Окно импорта данных

- сохранение состояния программы;
- экспорт данных в удобном и популярном формате электронных таблиц (Рисунок 17);

1	Model Name	New DEA Problem
2	Model Type	CCR_I
3	Model Orientation	INPUT_ORIENTED
4	Model Efficiency Type	TECH
5	Model RTS	CONSTANT
6	Model Description	The Charnes Cooper and Rhodes Model called CCR. This model was first introduced in 1978 and assumes CONSTANT RTS.
7		
8		
9		

Model details | Raw Data | Variables | Objectives | Projections | Lambdas | Peer Group | Slacks | Weights | (+)

Рисунок 17 – Пример экспорта результатов анализа

- широкий выбор моделей и их характеристик анализа оболочки данных (Data Envelopment Analysis) (Рисунок 18).

Please chose a Model Type:

CCR_I
CCR_O
BCC_I
BCC_O
GRS_I
GRS_O
IRS_I
IRS_O
DRS_I
DRS_O
SBM
SBM_V
SBM_GRS
SBM_I
SBM_I_V
SBM_I_GRS
SBM_O
SBM_O_V
SBM_O_GRS
NC_I
NC_I_V
NC_O
NC_O_V

ONS

ALL EFFICIENCIES

ALL RTS

Value: 0.80

RTS Upper Bound Value: 1.20

Рисунок 18 – Окно выбора модели и её характеристик

5 Результаты внедрения и использования системы. Достижение целей разработки

Разработанное приложение предоставляет простой и интуитивно понятный интерфейс для решения широко набора моделей DEA, а также обеспечивает наглядное представление для имеющихся данных (Рисунок 19). Данная система позволяет сохранять текущее состояние модели, а также импортировать результаты полученного анализа в файл электронных таблиц.

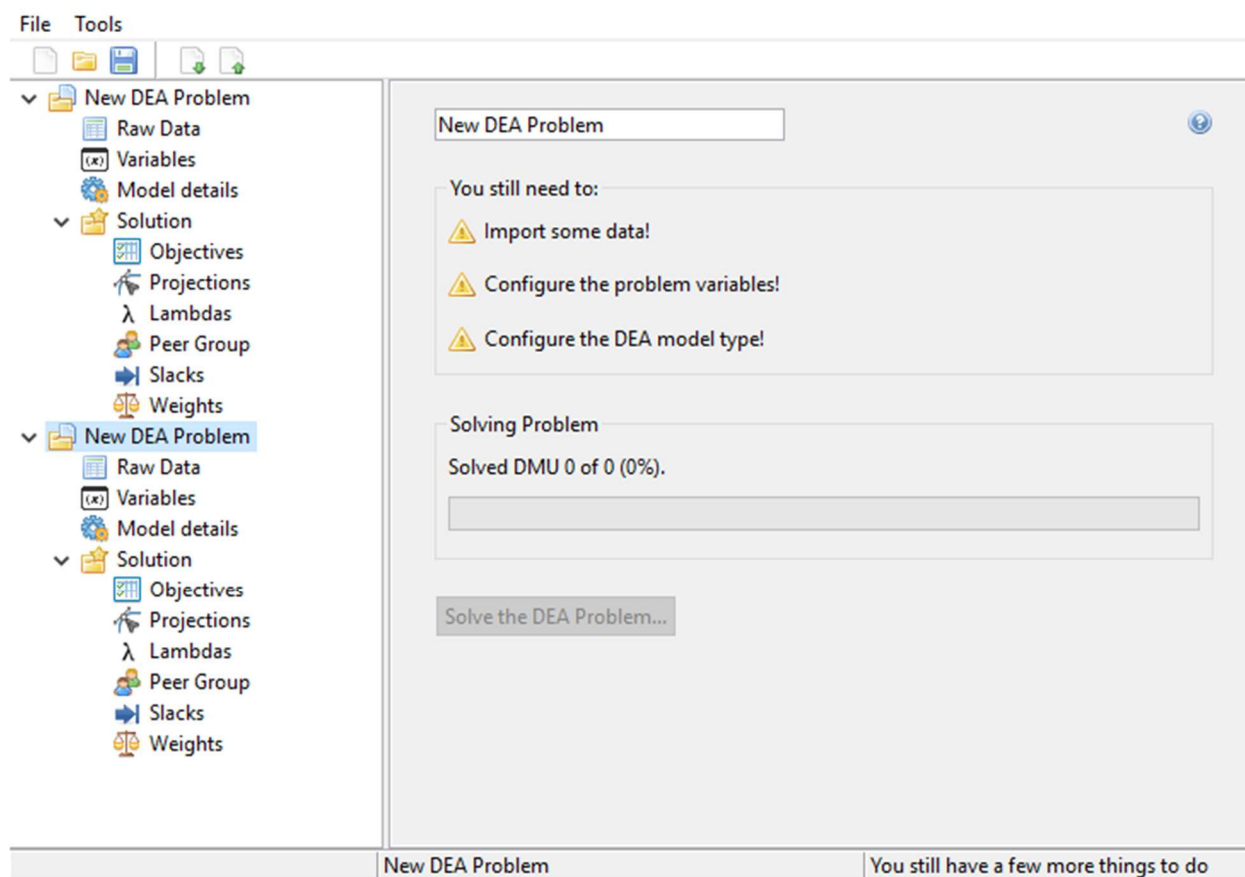


Рисунок 19 – Начальное окно приложения

В системе реализованы следующие функции:

- импорт данных для анализа в формате csv;
- сохранение состояния программы в формате deap;
- возможность решения большого набора моделей DEA;
- вывод результатов анализа в окне приложения в таблицах;
- экспорт результатов анализа в формате электронных таблиц.

Применение DEA (анализа среды функционирования) в рамках МЧС позволит лицам, принимающим решения:

- обеспечивать диагностику функционирования подразделений, т. е. давать панорамную картину деятельности подразделений в их взаимодействии с другими подразделениями;
- определять и различать эффективно и неэффективно работающие подразделения, находить количественную меру эффективности или же их неэффективности;
- указывать эффективные цели для каждого подразделения, т.е. эталонную группу эффективных подразделений, наиболее близких по своим показателям к исследуемому;
- находить наилучшие пути достижения эффективных целей;
- при дальнейшем анализе результатов эффективности работы подразделений с помощью эконометрических или статистических методов можно оценивать качество менеджмента, определять наиболее перспективные направления работы, их оптимальную величину и так далее.

Заключение

Результатом выпускной квалификационной работы стала программная реализация информационной системы анализа эффективности использования ресурсов.

Было проведено тестирование реализованной системы, обработаны различного рода исключения, которые могут возникать при её работе.

Данная система сможет помочь в распределении ресурсов пожарным подразделениям, которые находятся в городах с большим населением, а также городам, у которых существуют какие-либо региональные особенности. Данная система поможет лицам, принимающим решения, как правильно и эффективно распределять ресурсы по пожарным подразделениям. С помощью неё не только повысится эффективность пожарных подразделений, но и так же можно будет распределять ограниченные человеческие ресурсы. Таким образом, люди будут в безопасности. Стратегия АСФ объясняет, как рационально пользоваться и распределять ресурсы лицам, принимающим решения.

В итоге, данное программное решение предлагает решение для широкого набора моделей распределения ресурсов, и дополнительно создаёт новую эру в сфере достижения более высокой эффективности.

Также при выполнении выпускной квалификационной работы были закреплены знания и навыки, полученные в университете по различным дисциплинам, а также освоена предметная область, которая была необходима при рассмотрении поставленной задачи.

Список источников

- 1 Hatry H.P. (1978) 'The status of productivity measurement in the public sector', *Public Administration Review*, January–February, Vol. 38, No. 1, p.28.
- 2 Richman B.M. and Farmer, R.M. (1975) *Management and Organizations*, New York: Random House Press, p.364.
- 3 Farrell M.J. (1957) 'The measurement of productivity efficiency', *Journal of the Royal Statistical Society, Series A, Part 3*, Vol. 120, pp.253–281.
- 4 Szilagyi A.D., Jr. (1984) *Management and Performance*, 2nd ed., New Jersey: Scott, Foreman and Company.
- 5 Robbins S.P. (1996) *Organizational Behavior, Concept, Controversies, and Application*, 7th ed., New Jersey: Prentice-Hall, Inc.
- 6 Clarke R.L. (1992) 'Evaluating USAF vehicle maintenance productivity over time: an application of data envelopment analysis', *Decision Science*, Vol. 23, No. 2, pp.376–384.
- 7 Studit E.F. (1995) 'Productivity measurement in industrial operations', *European Journal of Operational Research*, Vol. 85, pp.435–453.
- 8 Griliches Z. and Regev H. (1995) 'Firm productivity in Israeli industry', *Journal of Econometrics*, Vol. 65, pp.175–203.
- 9 Feng C.M. and Wang R.T. (2000) 'Performance evaluation for airlines including the consideration of financial ratios', *Journal of Air Transport Management*, Vol. 6, pp.133–142.
- 10 Kaplan R.S. and Norton D.P. (2001) *The Strategy-Focused Organization: How Balanced Scorecard Companies Thrive in the New Business Environment*, 1st ed., Boston, MA: Harvard Business School Press.
- 11 Agrell P. and West B.M. (2001) 'A caveat on the measurement of productive efficiency', *International Journal of Production Economics*, Vol. 69, pp.1–14.

12 Charnes A. Measuring the Efficiency of Decision Making Units / A. Charnes, W. W. Cooper, E. Rhodes // European Journal of Operational Research. 1978. V. 2.

13 PIM-DEA Soft (Performance Improvement Management Software). Available at: <http://deazone.com/en/software>

14 Frontier Analyst. Available at: <http://www.banxia.com/frontier>

15 DEA-SolverPro. Available at: <http://www.saitech-inc.com/Products/Prod-DSP.asp>

16 KonSi – Data Envelopment Analysis for Benchmarking. Available at: <https://konsi-data-envelopment-analysis-75-units.soft112.com/>

17 Open Source DEA. Available at: <https://opensourcedea.org/>

18 DEAS (Data Envelopment Analysis using Stata). Available at: <http://sourceforge.net/projects/deas>

19 OpenDEA. Available at: <http://sourceforge.net/projects/opendea>

20 DEA Solver Online. Available at: <http://www.dea.fernuni-hagen.de>

21 DEAP. Available at: <http://www.uq.edu.au/economics/cepa/deap.php>

22 DEA Frontier. Available at: <http://www.deafrontier.net/deasoftware.html>

23 MaxDEA. Available at: <http://www.maxdea.cn>

24 DEAOS (DEA Online Software). Available at: <https://www.deaos.com>

25 EMS (Efficiency Measurement System). Available at: <http://www.holger-scheel.de/ems>

26 Cooper W. W. Data Envelopment Analysis [Text]: A Comprehensive Text with Models, Applications, References, and DEA-Solver Software / W. W. Cooper, L. M. Seiford, K. Tone. Boston: Kluwer Academic Publishers, 2000

27 Кривоножко В. Е. Анализ эффективности финансовых институтов в экономике переходного периода / В. Е. Кривоножко [и др.] // Нелинейная динамика и управление: сборник трудов. М. : ИСА РАН, 2000.

28 Seiford L.M., Zhu J., An investigation of returns to scale in data envelopment analysis, *International Journal of Management Science*, Vol. 27, 1999, pp. 1-11