

ПРИЛОЖЕНИЕ А
РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ (MATLAB)

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

**Библиотека открытого доступа DataReconcile для
согласования неточных индустриальных данных за счет
учета взаимосвязей между ними для нужд
промышленных предприятий Индустрии 4.0 и целей
Национальной технологической инициативы на языке
MATlab**

A.1 НАЗНАЧЕНИЕ БИБЛИОТЕКИ

Библиотека предназначена для решения задач **согласования неточных данных** (Data Reconcile) в условиях, когда входные данные содержат ошибки, неточности или противоречия. Она позволяет корректировать данные таким образом, чтобы они удовлетворяли заданным ограничениям, представленным в виде систем линейных и нелинейных уравнений и неравенств.

Библиотека помогает устранить противоречия в данных, минимизируя отклонения от исходных значений при соблюдении всех заданных ограничений. Это особенно полезно в задачах, где данные поступают из разных источников и могут содержать ошибки измерений или несоответствия.

Применение в различных областях:

Промышленность: согласование данных в системах управления технологическими процессами.

Экономика и финансы: обработка данных в моделях прогнозирования и анализа.

Наука и инженерия: решение задач оптимизации и калибровки моделей.

Энергетика: балансировка данных в энергосистемах.

Гибкость и масштабируемость:

Библиотека поддерживает работу с большими объемами данных и может быть интегрирована в различные программные системы.

Она предоставляет инструменты для настройки и адаптации под конкретные задачи пользователя.

Примеры задач, где библиотека может быть полезна:

Балансировка данных в химических процессах (например, согласование материальных и энергетических балансов).

Корректировка данных в системах мониторинга (например, устранение противоречий в показаниях датчиков).

Оптимизация данных в моделях прогнозирования (например, согласование исторических данных с прогнозными значениями).

Библиотека является мощным инструментом для анализа, корректировки и оптимизации данных в условиях наличия ограничений, что делает её незаменимой в задачах, требующих высокой точности и согласованности данных.

А.2 ОСНОВНЫЕ ВОЗМОЖНОСТИ

Поддержка стационарных и динамических задач:

Стационарные задачи: работа с данными, которые не изменяются во времени (например, балансовые уравнения в статических системах).

Динамические задачи: обработка данных, которые изменяются во времени (например, временные ряды или процессы, описываемые дифференциальными уравнениями).

Работа с ограничениями:

Библиотека поддерживает ограничения в виде:

- **Линейных уравнений и неравенств** (например, $Ax = b$ или $Ax \leq b$).
- **Нелинейных уравнений и неравенств** (например, $F(x) = 0$ или $g(x) \leq 0$).

Библиотека разрешает использование дифференциальных и интегро-дифференциальных уравнений в качестве ограничений, но в форме разностных уравнений по выбранной пользователем вычислительной схеме.

Это позволяет учитывать физические, экономические или другие законы, которым должны соответствовать данные.

Минимизация отклонений:

Библиотека минимизирует отклонения скорректированных данных от исходных, используя различные метрики.

А.3 СИСТЕМНЫЕ ТРЕБОВАНИЯ

Минимальные требования позволяют запускать библиотеку и выполнять базовые задачи на компьютерах с ограниченными ресурсами. Однако производительность может быть низкой при работе с большими объемами данных или сложными задачами.

- **Операционная система:**
 - Windows 10 (64-битная) или новее.
 - macOS 10.15 (Catalina) или новее.
 - Linux: Ubuntu 20.04 LTS или совместимые дистрибутивы.
- **Процессор (CPU):**
 - Минимум: 2-ядерный процессор с тактовой частотой 2.0 ГГц (например, Intel Core i3 или аналогичный AMD).
- **Оперативная память (RAM):**
 - Минимум: 4 ГБ (для небольших задач).
 - Рекомендуется: 8 ГБ для более комфортной работы.

- **Место на диске:**
 - Минимум: 5 ГБ свободного места для установки MATLAB и библиотеки.
 - Дополнительное место для хранения данных и результатов.
- **Графика:**
 - Интегрированная видеокарта с поддержкой OpenGL 3.3.
- **Программное обеспечение:**
 - MATLAB R2019 или новее.
 - Optimization Toolbox (обязательно для работы библиотеки).
- **Дополнительные требования:**
 - Подключение к интернету для активации MATLAB и загрузки обновлений.
 - Мышь или другое указывающее устройство.

Рекомендуемые системные требования

Рекомендуемые требования обеспечивают высокую производительность при работе с большими объемами данных, сложными моделями и динамическими задачами.

- **Операционная система:**
 - Windows 10 (64-битная) или новее.
 - macOS 11 или новее.
 - Linux: Ubuntu 22.04 LTS или совместимые дистрибутивы.
- **Процессор (CPU):**
 - Рекомендуется: 4-ядерный процессор с тактовой частотой 3.0 ГГц или выше (например, Intel Core i7/i9, AMD Ryzen 5/7/9).
- **Оперативная память (RAM):**
 - Рекомендуется: 16 ГБ или больше (для работы с большими наборами данных и сложными задачами).
- **Место на диске:**
 - Рекомендуется: SSD с 20 ГБ свободного места для установки MATLAB и библиотеки.
 - Дополнительное место для хранения данных и результатов (в зависимости от объема данных).
- **Графика:**
 - Дискретная видеокарта с 2 ГБ видеопамяти и поддержкой OpenGL 4.0 (например, NVIDIA GeForce GTX 1050 или выше).
- **Программное обеспечение:**
 - MATLAB R2020 или новее.

- Optimization Toolbox (обязательно).
- Parallel Computing Toolbox (рекомендуется для ускорения вычислений на многоядерных процессорах).
- **Дополнительные требования:**
 - Подключение к интернету для активации MATLAB и загрузки обновлений.
 - Мышь или другое указывающее устройство.

Установка и настройка

Далее для краткости в качестве имени функции, которой пользователь желает воспользоваться, а для этого, соответственно, добавить в рабочую среду матлаб используется условное название `*LibFunctionName*`.

Для подключения и использования внешней функции `LibFunctionName`, находящейся в файле `LibFunctionName.m` из библиотеки `DataReconcile`, выполните следующие шаги:

Подготовка файлов библиотеки

1. Убедитесь, что у вас есть файл `LibFunctionName.m` и другие необходимые файлы библиотеки `DataReconcile`.
2. Скопируйте папку `DataReconcile` с файлами библиотеки в удобное место на вашем компьютере (например, в папку `C:\MATLAB\Libraries\DataReconcile`).

Добавление пути к библиотеке в MATLAB

Чтобы MATLAB мог найти функцию `LibFunctionName`, необходимо добавить путь к папке `DataReconcile` в список путей MATLAB.

1. Откройте MATLAB.

В командной строке MATLAB выполните следующую команду (matlab):

```
addpath('C:\MATLAB\Libraries\DataReconcile');
```

Здесь `C:\MATLAB\Libraries\DataReconcile` — это путь к папке, где находится файл `LibFunctionName.m`.

2. Чтобы путь сохранялся при следующих запусках MATLAB, добавьте его в список постоянных путей (matlab):

```
savepath;
```

Проверка подключения функции

1. Убедитесь, что функция `LibFunctionName` доступна. Для этого выполните команду (matlab):

```
which LibFunctionName
```

MATLAB должен вернуть путь к файлу `LibFunctionName.m`, например:

C:\MATLAB\Libraries\DataReconcile\LibFunctionName.m

2. Проверьте, что функция работает корректно. Вызовите её с тестовыми входными данными (matlab):

```
result = LibFunctionName(input1, input2, ...);
```

Здесь input1, input2 — это входные аргументы, которые требуются для функции LibFunctionName.

Дополнительные рекомендации

- Если библиотека DataReconcile содержит несколько функций, добавьте путь к папке библиотеки, как описано выше. Это позволит использовать все функции библиотеки.
- Если вы работаете в команде, убедитесь, что все участники проекта добавили путь к библиотеке в свои среды MATLAB.
- Если вы используете Git или другую систему контроля версий, добавьте папку DataReconcile в репозиторий, чтобы упростить совместную работу.

Устранение возможных проблем

- **Ошибка "Function not found":** Убедитесь, что путь к папке DataReconcile добавлен правильно. Проверьте команду `which LibFunctionName`.
- **Ошибка "Not enough input arguments":** Проверьте документацию функции LibFunctionName и убедитесь, что вы передаете все необходимые входные аргументы.
- **Ошибка "File not found":** Убедитесь, что файл LibFunctionName.m находится в указанной папке и его имя написано правильно (с учетом регистра).

A.4 ПОШАГОВАЯ ИНСТРУКЦИЯ ПО РАЗВЕРТЫВАНИЮ БИБЛИОТЕКИ DATARECONCILE (MATLAB)

1. Если ранее этого не было сделано, установите программное обеспечение MATLAB версии не старше 2019a. Скачать установочный файл можно с официального сайта по ссылке <https://www.mathworks.com/products/matlab.html>

Если программный пакет MATLAB не доступен, возможно использовать данную библиотеку

с одним из совместимых с MATLAB на уровне языка программирования свободно распространяемых продуктов.

Ниже приведены пошаговые инструкции по установке двух предпочтительных вариантов: GNU Octave, Scilab.

1. (альтернатива 1) Пошаговая инструкция по установке GNU Octave

1.А Установка GNU Octave в Windows

- о Загрузите установщик, перейдя по ссылке <https://www.gnu.org/software/octave/download>.

- о В Windows нажмите на ссылку для последней стабильной версии (например, octave-8.4.0-w64-installer.exe для 64-разрядной версии или octave-8.4.0-w32-installer.exe для 32-разрядной версии).

- о Запустите установщик
- о Откройте загруженный файл .exe.
- о Следуйте указаниям мастера установки (выберите параметры по умолчанию, если вам не нужна настройка).

- о Выберите Добавить Octave в PATH (рекомендуется для использования в командной строке).

- о Запустите Octave

- о После установки вы можете открыть Octave через:
Меню «Пуск» → GNU Octave → Octave GUI
Или запустите octave в командной строке (если добавлен в PATH).

1.Б Установка GNU Octave на Linux (путем передачи команд через оболочку sh)

- о Для Debian/Ubuntu (APT) (командная оболочка sh):
`sudo apt update`
`sudo apt install octave`

- о Для Fedora (DNF) (командная оболочка sh):
`sudo dnf install octave`

- о Для Arch Linux (Pacman) (командная оболочка sh):
`sudo pacman -S octave`

- о Для OpenSUSE (Zypper) (командная оболочка sh):
`sudo zypper install octave`

- о Запуск Octave

Откройте терминал и выполните (командная оболочка sh):

octave

Или используйте графический интерфейс (командная оболочка sh):

octave --gui

1.B Установка GNU Octave на macOS

Способ 1: использование Homebrew (рекомендуется)

- о Установите Homebrew (если не установлен) (командная оболочка sh)

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Следуйте инструкциям на экране.

- о Установите Octave (командная оболочка sh)

```
brew install octave
```

- о Запустите Octave, откройте терминал и выполните (командная оболочка sh):

octave

Или используйте графический интерфейс, выполнив (командная оболочка sh):

```
brew install --cask octave
```

Способ 2: Использование официального файла DMG

- о Загрузите файл .dmg с <https://www.gnu.org/software/octave/download>.
- о Откройте .dmg и перетащите Octave в Applications.
- о Запустите Octave из Launchpad или Applications.

1. (альтернатива 2) Пошаговая инструкция по установке Scilab

1.A Установка Scilab в Windows

- о Перейдите на <https://www.scilab.org/download>.
- о Выберите установщик Windows (например, scilab-6.1.1_x64.exe для 64-разрядной версии).
- о Запустите установщик
- о Дважды щелкните загруженный .exe-файл.

- о Следуйте указаниям мастера установки (используйте настройки по умолчанию, если не требуется настройка).

- о Отметьте «Добавить Scilab в PATH» (необязательно, для использования в командной строке).

- о Запустите Scilab

Откройте из меню «Пуск» → Scilab-X.Y.Z → Scilab.

Или передайте команду `scilab` в командной строке (если добавлен в PATH) (командная оболочка `sh`):

1.Б Установка Scilab на Linux

- о Debian/Ubuntu (APT) (командная оболочка `sh`)

`sudo apt update`

`sudo apt install scilab`

- о Fedora (DNF) (командная оболочка `sh`)

`sudo dnf install scilab`

- о Arch Linux (AUR) (командная оболочка `sh`)

`yay -S scilab-bin` # Требуется помощник AUR (`yay/paru`)

- о OpenSUSE (Zypper) (командная оболочка `sh`)

`sudo zypper install scilab`

1.В Установка Scilab на macOS (с помощью файла DMG)

- о Чтобы загрузить Scilab, перейдите по ссылке <https://www.scilab.org/download>.

- о Выберите файл macOS .dmg (например, `scilab-6.1.1.dmg`).

- о Установить Scilab

Откройте файл .dmg.

Перетащите Scilab в папку Applications.

- о Запустите Scilab

- о Откройте Applications → Scilab-X.Y.Z → Scilab.

Если заблокировано, щелкните правой кнопкой мыши → Open (чтобы обойти предупреждения безопасности macOS).

2. Скачайте (А) файлы библиотеки DataReconcile, либо клонируйте (Б) на вычислительную машину, где планируется сборка проекта, использующего функционал библиотеки.

2.А Как скачать содержимое репозитория DataReconcile:

– Перейдите на главную страницу репозитория по ссылке <https://github.com/scientific-soft/DataReconcile>.

– Кликните на кнопку «<>», расположенную над списком содержащихся в корневой директории файлов и папок библиотеки (рисунок A.1):

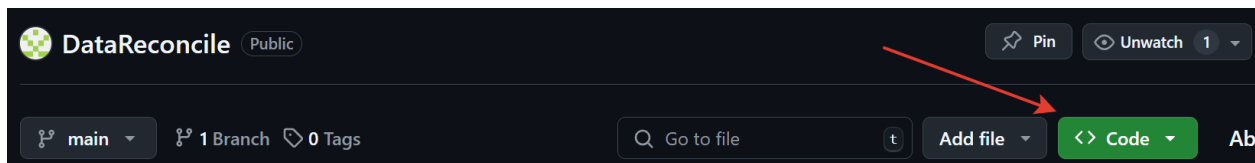


Рисунок A.1 – К разворачиванию библиотеки DataReconcile

– Во всплывающем меню выберите вкладку «Local» (1), после чего кликните на строку «Download ZIP» (2) (рисунок A.2).

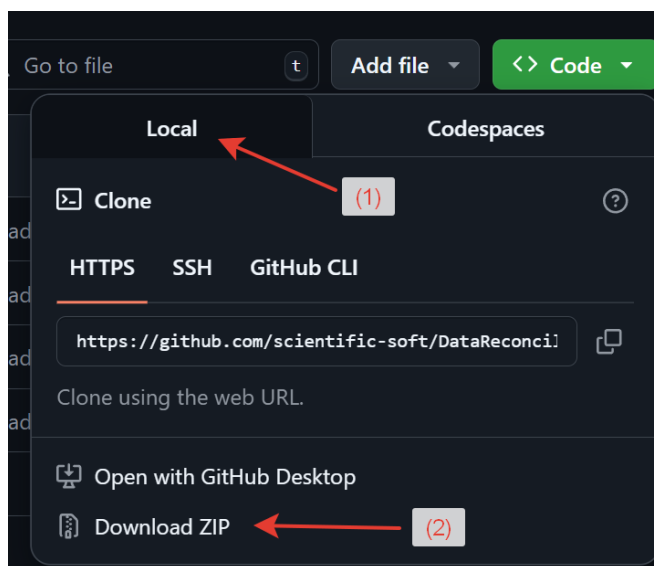


Рисунок A.2 – К разворачиванию библиотеки DataReconcile

– После выполнения данных операций содержимое библиотеки начнет скачиваться в директорию загрузки на локальном компьютере в виде файла с расширением «.zip».

– После того, как скачивание завершено, скаченный архивный файл «DataReconcile-main.zip» необходимо разархивировать программой архиватором, поддерживающей формат «.zip».

– Разархивация с использованием свободно распространяемого архиватора 7-Zip (рисунок A.3).

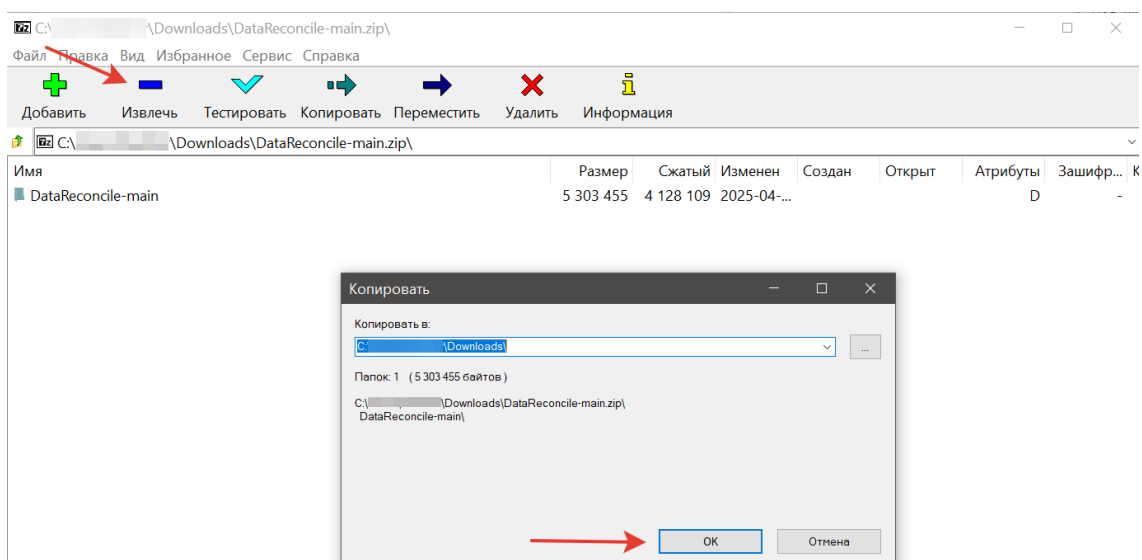


Рисунок А.3 – К разворачиванию библиотеки DataReconcile

– Результат скачивания файлов библиотеки – папка с именем DataReconcile-main, содержимое которой приведено на скриншоте ниже (рисунок А.4).

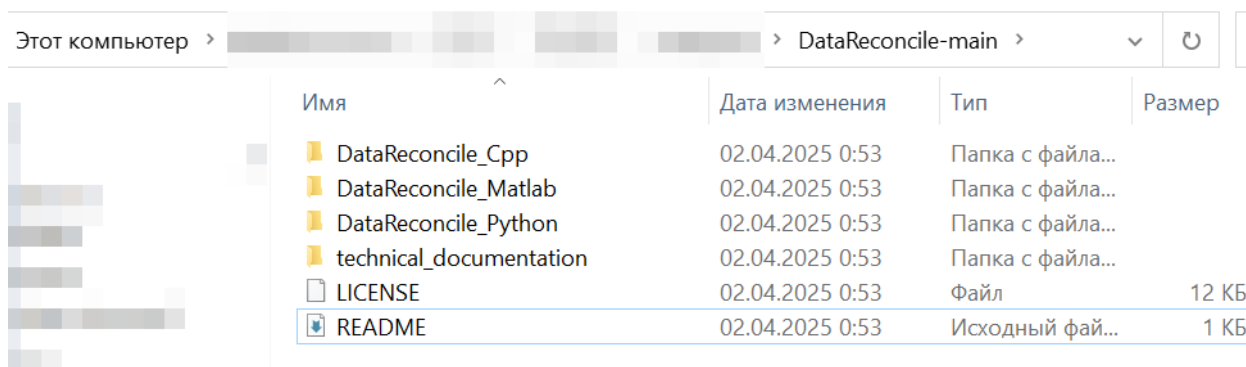


Рисунок А.4 – К разворачиванию библиотеки DataReconcile

2.Б Как клонировать содержимое репозитория DataReconcile.

– Перейдите на главную страницу репозитория по ссылке <https://github.com/scientific-soft/DataReconcile>.

– Кликните на кнопку «<>», расположенную над списком содержащихся в корневой директории файлов и папок библиотеки (рисунок А.5).

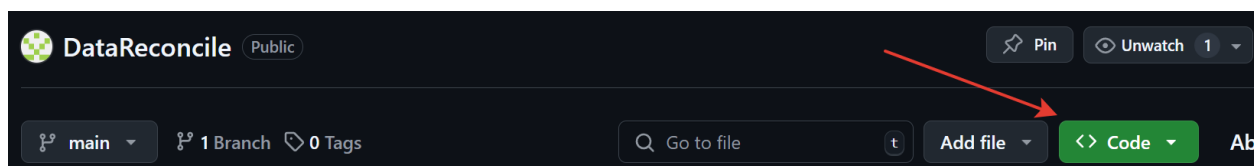


Рисунок А.5 – К разворачиванию библиотеки DataReconcile

– Скопируйте URL репозитория в буфер обмена:

1) Чтобы клонировать репозиторий с помощью HTTPS, нажмите кнопку копирования в разделе «HTTPS».

2) Чтобы клонировать репозиторий с помощью SSH ключа, включая сертификат, выданный центром сертификации SSH вашей организации, нажмите кнопку копирования в разделе SSH.

3) Чтобы клонировать репозиторий с помощью GitHub CLI, нажмите кнопку копирования в разделе GitHub CLI (рисунок А.6).

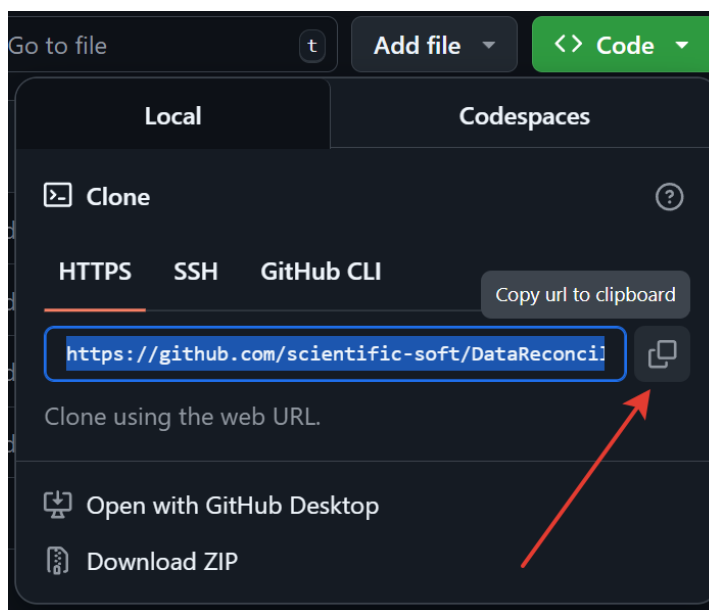


Рисунок А.6 – К разворачиванию библиотеки DataReconcile

- Откройте консоль Git Bash.
- Измените текущий рабочий каталог на место, куда вы хотите клонировать содержимое DataReconcile.
- Введите `git clone`, а затем вставьте URL-адрес, скопированный ранее в формате
`git clone https://github.com/scientific-soft/DataReconcile.git`
- Нажмите Enter, чтобы создать локальный клон библиотеки.

3. Запустите программный пакет MATLAB.

4. Смените рабочую директорию среды MATLAB на ту, в которой находится ваш проект, в котором планируется использовать функционал библиотеки.

5. Либо (А) скопируйте из папки DataReconcile/DataReconcile_Matlab, полученной в пункте 2, файлы с требуемыми функцией/ями, либо воспользуйтесь одним из методов добавления внешних функций в проект:

Метод 1: Временное добавление в путь MATLAB (код matlab)

```
% Добавить один каталог в путь
addpath('/full/path/to/external/functions');
% Добавить каталог и все подпапки
addpath(genpath('/full/path/to/external/functions'));
% Проверить, что он был добавлен
path
```

Метод 2: Постоянное добавление в путь MATLAB (код matlab)

```
% Открыть диалоговое окно Set Path
pathtool
% Или программно сохранить путь
addpath('/full/path/to/external/functions');
savepath; % Сохранение пути для будущих сеансов
```

Метод 3: Использовать абсолютные пути при вызове функций (код matlab)

```
% Вызвать функцию, используя ее полный путь
result = fullfile('/path/to/functions', 'function_name')(arguments);
```

6. Все доступные функции по названию совпадают с названием файла расширением «.m». Если возникает ошибка вызова функции из библиотеки, проверьте, действительно в директории по указанному пути присутствует файл с названием функции.

A.5 ОСНОВНЫЕ ФУНКЦИИ И API

A.5.1 Описание ключевых модулей

Решаемая в рамках программной библиотеки задача относится к задачам повышения точности результатов совместных измерений, выполняемых на сложном объекте измерений, для которого предоставлена или получена в ходе эмпирических исследований математическая модель (возможно, не вполне точная). Снижение неопределенности получаемых результатов достигается за счет учета функциональных зависимостей между измеряемыми величинами, формализованных в рамках упомянутой математической модели. В качестве последней могут выступать как различные уравнения (алгебраические – линейные и нелинейные, дифференциальные и интегро-дифференциальные), а также наборы ограничений, вытекающие из условий решаемой измерительной задачи и условий функционирования и эксплуатации наблюдаемого объекта измерений (системы неравенств

– односторонних и двусторонних). Решение задачи производится также при разном объеме известной информации, ключевое место среди которой занимают сведения о распределении погрешностей выполняемых измерений – как известно из математической статистики, при разных законах распределения случайных погрешностей разные числовые оценки для одного и того же параметра распределений будут являться эффективными (то есть будут обеспечивать наименьший статистический разброс от выборки к выборке). Данное обстоятельство указывает на обязательную необходимость привлечения сведений о распределении погрешностей (в том объеме, в котором она доступна). Решение задачи повышения точности (за счет согласования индустриальных данных) под перечисленными ограничениями представляет собой комплекс задач, решения которых имеет как общие черты, так и разнится в зависимости от типа ограничений. Эффективные вычислительные решения для разных значимых для измерительной практики ситуаций представляет собой основу настоящей библиотеки. На первом этапе основным типом рассматриваемых ограничений выступают математические модели, составленные из алгебраических уравнений (как линейных, так и нелинейных) при условии, что закон распределения случайных погрешностей отличается от нормального (возможно, несильно). Данный тип задачи должен включать в себя традиционные решения (в предположении гауссовости распределения погрешностей), так и новые, до сего момента не представленные в научной литературе по вопросу методы и подходы. Другим важным моментом является необходимость учета режима выполняемых измерений – статического (когда изменениями во времени значений измеряемых величин можно пренебречь без значимого снижения точности) или динамического (когда изменения сигналов измерительной информации во времени приводят к возникновению значимых составляющих погрешностей, отсутствие учета которых приводит к существенному снижению достоверности конечных результатов). Данная постановка задачи крайне важна для измерительной практики и предложена впервые в контексте задачи согласования индустриальных данных.

Рассматривая задачу согласования индустриальных данных целостно, следует отметить, что задача так или иначе сводится к условной оптимизационной задаче некоторой размерности и с тем или иным набором условий. Данное обстоятельство позволяет построить эффективную модель решения задачи согласования данных, выбрав для того или иного набора ограничений наиболее подходящий алгоритм оптимизации. Программная библиотека, реализующая разные методы согласования неточных данных друг с другом, также содержит в себе ряд автоматизированных инструментов.

На рисунке А.7 ниже представлена визуализация структуры библиотеки, а также входные и выходные данные.

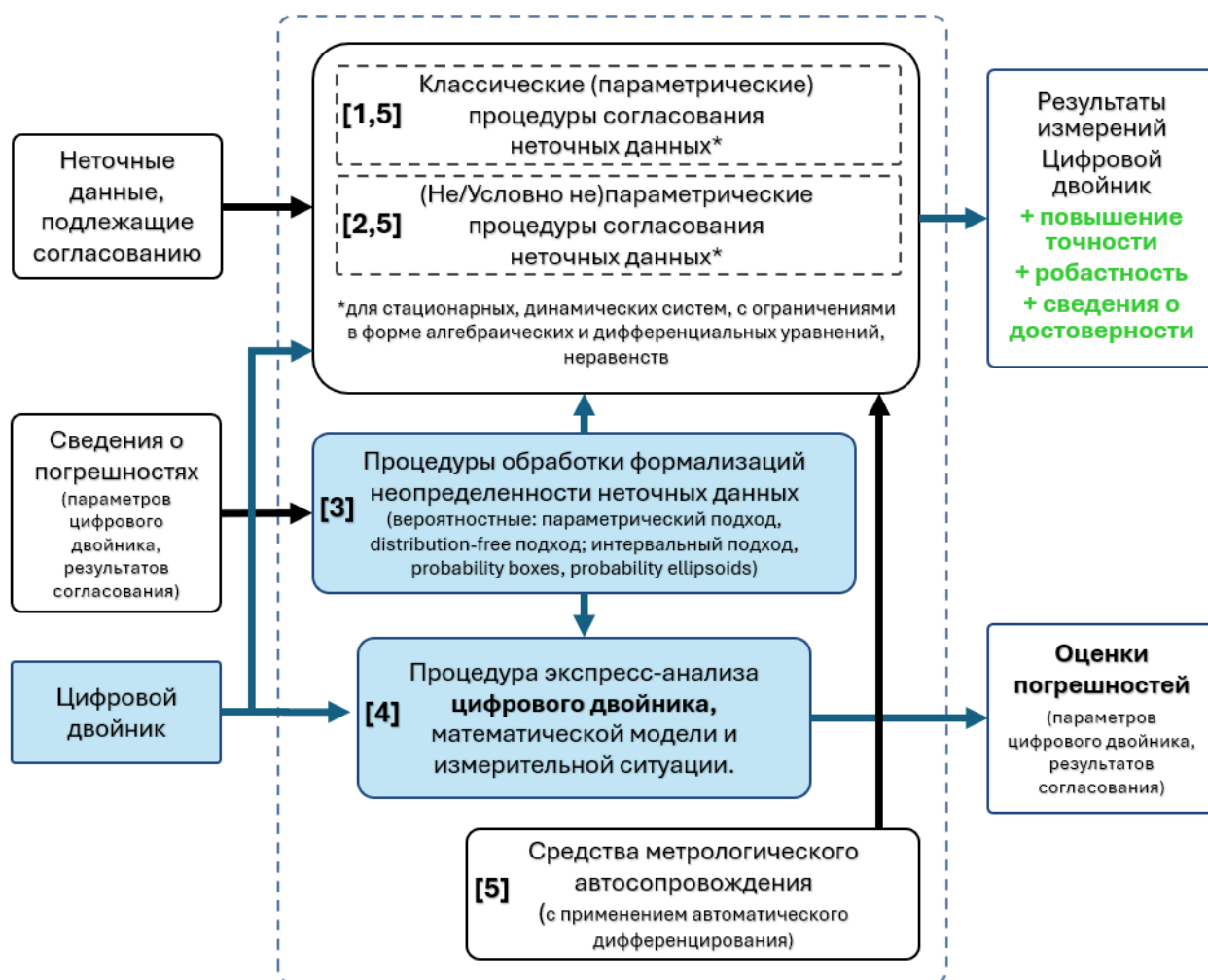


Рисунок А.7 – Структура библиотеки DataReconcile

В соответствии с приведенной на рисунке нумерацией, библиотека содержит:

- 1) программные средства, реализующие традиционные методы согласования промышленных данных, которые содержатся и применяются в зарубежных коммерческих продуктах,
- 2) программные средства, реализующие непараметрические и условно непараметрические методы согласования промышленных данных, позволяющие учитывать в рамках согласования существенные или малые отклонения действительного закона распределения погрешностей от нормального, применимые как в статическом режиме измерения, так и в динамическом,
- 3) программные представления и преобразования современных формализмов описания погрешности результатов измерений, таковых, чтобы результат

преобразования/представления мог быть эффективно использован в качестве мера неопределенности неточных индустриальных данных при их согласовании по известной модели.

4) программные предварительной экспресс-оценки потенциального уточнения, достигаемого при математической обработке результатов выполняемых измерений за счет выполнения их согласования на основе известной априорной информации,

5) программные средства для обеспечения метрологического автосопровождения производимых вычислений [2-4], поскольку без него нет возможности оценить точность конечных результатов согласования, а также – при необходимости – произвести согласование моментов остановки выполняемых итерационных процедур с точностью исходных данных (то есть результатов выполненных измерений).

А.5.2 Доступные функции, их параметры и возвращаемые значения

Презентуемая библиотека оформлена в виде независимых m-файлов, для работы с которым требуется программный пакет MATLAB, включающий расширение Optimization Toolbox. Ниже приведено описание процедур для не/полу/параметрического согласования данных и выполнения метрологического анализа потенциальных результатов согласования.

Для каждого параметра `msrd_data` функции каждой функции справедливо следующее: в случае согласования результатов однократного совместного измерения нескольких взаимосвязанных величин, согласуемые результаты нужно передавать в формате вектор-столбца; если согласованию подлежит массив многократных совместных измерений, согласуемые значения нужно передавать в виде матрицы `msrd_data`, где каждый столбец – набор однократного совместного измерения всех согласуемых величин (таким образом в j -ой строке передаваемой матрицы `msrd_data` располагаются все результаты измерений j -ой измеряемой физической величины).

Процедура оценки степени потенциального уточнения совместных измерений за счет согласования

estAccuracyIncreaseByDR

Вызов	<code>[accuracy_increase_ratio, variances_of_DR_result] = estAccuracyIncreaseByDR(@dep_func, params_, vals_for_reconc, err_matrix);</code>
-------	--

Описание	Функция выполняет приближенную оценку потенциального уточнения совместных измерений, достигаемого за счет учета известных функциональных взаимосвязей между измеряемыми величинами, на
----------	--

основе локальной линеаризации модели и метода декомпозиции алгоритма условной оптимизации.

Аргументы	<p><code>dep_func</code> – функция, возвращающую значения уравнений, формализующих взаимосвязь между измеряемыми величинами.</p> <p><code>params_</code> – одномерный массив значений параметров уравнений, описывающих зависимости между измеряемыми величинами, известных точно.</p> <p><code>vals_for_reconc</code> – одномерный массив, содержащий результаты совместных измерений, подлежащих уточнению;</p> <p><code>err_matrix</code> – двумерный массив, содержащий ковариационную матрицу согласуемых результатов измерений и параметров модели, используемой для согласования.</p>
Возвращаемое значение	<p><code>accuracy_increase_ratio</code> – одномерный массив оценок степени повышения точности, которое может быть достигнуто за счет процедуры согласования;</p> <p><code>variances_of_DR_result</code> – одномерный массив оценок дисперсий результатов согласования.</p>

Функции согласования, учитывающего зависимости между взаимосвязанными величинами, выраженные в виде систем уравнений

Процедура согласования совместных измерений в предположении, что случайные погрешности распределены нормально

DRparamEq

Вызов	<pre>[reconciled] = DRparamEq(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, true_params)</pre>
Описание	<p>Функция выполняет параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей.</p>
Аргументы	<p><code>Func</code> – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами,</p> <p><code>msrd_data</code> – массив, содержащий согласуемые результаты измерений,</p> <p><code>error_params</code> – одномерный массив, содержащий меры погрешности независимо полученных результатов измерений, подлежащих согласованию,</p> <p><code>true_params</code> – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.</p>

Возвращаемое значение	reconciled – одномерный массив результатов параметрического согласования совместных измерений, выполненного при следующих допущениях: случайные погрешности результатов совместных измерений распределены нормально; согласуемые результаты измерений получены независимо, и, следовательно, корреляция между случайными погрешностями отсутствует
-----------------------	---

Процедура семи-непараметрического согласования с представлением неизвестного закона распределения погрешностей рядом Грама-Шарлье

DRsemiparamEq

Вызов	<pre>[reconciled] = DRsemiparamEq(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, alpha_params, true_params);</pre>
Описание	Файл содержит одноименную вызываемую функцию, которая выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грамма-Шарлье.
Аргументы	Func – функция, возвращающую значения левой части уравнений взаимосвязи между измеряемыми величинами; msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию; error_params – одномерный массив, содержащий меры погрешности результатов измерений; alpha_params – вектор-массив, содержащий оценки мер погрешностей результатов измерений, коэффициентов асимметрии, коэффициентов и эксцесса случайных отклонений результатов измерений; true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.
Возвращаемое значение	reconciled – одномерный массив результатов семи-непараметрического согласования совместных измерений.

Процедура непараметрического согласования с ядерной аппроксимацией распределения случайных погрешностей

DRnonparamEq

Вызов	<pre>[reconciled] = DRnonparamEq(@equalities_dep_func, @inequalities_dep_func, msrd_data, true_params, prior_params, bandwidth)</pre>
Описание	Функция выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается

методом ядерной аппроксимации с использованием ядра Гаусса.
Подлежат учету зависимости в виде равенств и неравенств.

Аргументы	<p>Func – функция, возвращающую значения левой части уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающих зависимости между измеряемыми величинами \mathbf{x} и параметрами \mathbf{a} модели;</p> <p>msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;</p> <p>true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи;</p> <p>prior_params – одномерный массив, содержащий априорно заданные или оцененные меры погрешности результатов измерений;</p> <p>bandwidth – одномерный массив, содержащий регуляризирующее условие на результат ядерной аппроксимации неизвестного распределения случайных погрешностей согласуемых измерений. Данные значения являются ширинами окон ядерной аппроксимации. В качестве аппроксимирующего ядра использовано ядро Гаусса.</p>
Возвращаемое значение	<p>reconciled – одномерный массив результатов непараметрического согласования результатов измерений взаимосвязанных величин. Предполагается, что измерения выполнялись независимо, и корреляция между случайными погрешностями отсутствует. В качестве процедуры идентификации неизвестного распределения случайных погрешностей согласуемых измерений применен метод ядерной аппроксимации ядром Гаусса.</p>

Процедура робастного согласования совместных измерений в предположении, что случайные погрешности распределены нормально

DRparamEqRobust

Вызов	<pre>[reconciled] = DRparamEqRobust(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, true_params)</pre>
Описание	<p>Функция выполняет робастное параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей.</p>
Аргументы	<p>Func – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами,</p> <p>msrd_data – массив, содержащий согласуемые результаты измерений,</p>

error_params – одномерный массив, содержащий меры погрешности независимо полученных результатов измерений, подлежащих согласованию,
true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.

Возвращаемое значение **reconciled** – одномерный массив результатов параметрического согласования совместных измерений, выполненного при следующих допущениях: случайные погрешности результатов совместных измерений распределены нормально; согласуемые результаты измерений получены независимо, и, следовательно, корреляция между случайными погрешностями отсутствует

Процедура семи-непараметрического согласования с представлением неизвестного закона распределения погрешностей рядом Грама-Шарлье

DRsemiparamEqRobust

Вызов `[reconciled] = DRsemiparamEqRobust(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, alpha_params, true_params);`

Описание Файл содержит одноименную вызываемую функцию, которая выполняет робастное полу-непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грама-Шарлье.

Аргументы **Func** – функция, возвращающую значения левой части уравнений взаимосвязи между измеряемыми величинами;
msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;
error_params – одномерный массив, содержащий меры погрешности результатов измерений;
alpha_params – вектор-массив, содержащий оценки мер погрешностей результатов измерений, коэффициентов асимметрии, коэффициентов и эксцесса случайных отклонений результатов измерений;
true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.

Возвращаемое значение **reconciled** – одномерный массив результатов семи-непараметрического согласования совместных измерений.

Процедура непараметрического робастного согласования с ядерной аппроксимацией распределения случайных погрешностей

DRnonparamEqRobust

Вызов	<pre>[reconciled] = DRnonparamEqRobust(@Func, msrd_data, true_params, error_params, bandwidths)</pre>
Описание	Функция выполняет робастное непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса.
Аргументы	<p>Func – функция, возвращающую значения левой части уравнений вида $f_M(\mathbf{x}, \mathbf{a}) = 0$, описывающих зависимости между измеряемыми величинами \mathbf{x} и параметрами \mathbf{a} модели;</p> <p>msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;</p> <p>true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи;</p> <p>prior_params – одномерный массив, содержащий априорно заданные или оцененные меры погрешности результатов измерений;</p> <p>bandwidth – одномерный массив, содержащий регуляризирующее условие на результат ядерной аппроксимации неизвестного распределения случайных погрешностей согласуемых измерений. Данные значения являются ширинами окон ядерной аппроксимации. В качестве аппроксимирующего ядра использовано ядро Гаусса.</p>
Возвращаемое значение	<p>Reconciled – одномерный массив результатов непараметрического согласования результатов измерений взаимосвязанных величин. Предполагается, что измерения выполнялись независимо, и корреляция между случайными погрешностями отсутствует. В качестве процедуры идентификации неизвестного распределения случайных погрешностей согласуемых измерений применен метод ядерной аппроксимации ядром Гаусса.</p>

Функции согласования, учитывающего зависимости между взаимосвязанными величинами, выраженные в виде систем уравнений и неравенств

Процедура согласования совместных измерений в предположении, что случайные погрешности распределены нормально

DRparamEqIneq

Вызов	<pre>[reconciled] = DRparamEqIneq(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, true_params)</pre>
Описание	Функция выполняет параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых

соответствует нормальному распределению вероятностей. Подлежат учету зависимости в виде равенств и неравенств.

Аргументы	<p><code>equalities_dep_func</code> – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами (размерность возвращаемого вектора соответствует размерности вектор-функции \mathbf{f}_M),</p> <p><code>inequalities_dep_func</code> – функция, возвращающую результат вычисления левых частей системы неравенств вида $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) \geq \mathbf{0}$, представленных в формате равенств $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$ (в соответствии с методом множителей Лагранжа), описывающую взаимосвязи (ограничения) между согласуемыми величинами,</p> <p><code>msrd_data</code> – массив, содержащий согласуемые результаты измерений,</p> <p><code>error_params</code> – одномерный массив, содержащий меры погрешности независимо полученных результатов измерений, подлежащих согласованию,</p> <p><code>true_params</code> – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.</p>
Возвращаемое значение	<p><code>reconciled</code> – одномерный массив результатов параметрического согласования совместных измерений, выполненного при следующих допущениях: случайные погрешности результатов совместных измерений распределены нормально; согласуемые результаты измерений получены независимо, и, следовательно, корреляция между случайными погрешностями отсутствует</p>

Процедура семи-непараметрического согласования с представлением неизвестного закона распределения погрешностей рядом Грама-Шарлье

DRsemiparamEqIneq

Вызов	<pre>[reconciled] = DRsemiparamEqIneq(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, alpha_params, true_params);</pre>
Описание	<p>Файл содержит одноименную вызываемую функцию, которая выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грамма-Шарлье. Подлежат учету зависимости в виде равенств и неравенств.</p>
Аргументы	<p><code>equalities_dep_func</code> – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами (размерность возвращаемого вектора соответствует размерности вектор-функции \mathbf{f}_M),</p> <p><code>inequalities_dep_func</code> – функция, возвращающую результат вычисления левых частей системы неравенств вида $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) \geq \mathbf{0}$, представленных в формате равенств $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$ (в соответствии с</p>

методом множителей Лагранжа), описывающую взаимосвязи (ограничения) между согласуемыми величинами,
msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;
error_params – одномерный массив, содержащий меры погрешности результатов измерений;
alpha_params – вектор-массив, содержащий оценки мер погрешностей результатов измерений, коэффициентов асимметрии, коэффициентов и эксцесса случайных отклонений результатов измерений;
true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.

Возвращаемое значение **reconciled** – одномерный массив результатов семи-непараметрического согласования совместных измерений.

Процедура непараметрического согласования с ядерной аппроксимацией распределения случайных погрешностей

DRnonparamEqIneq

Вызов `[reconciled] = DRnonparamEqIneq (@equalities_dep_func, @inequalities_dep_func, msrd_data, true_params, prior_params, bandwidth)`

Описание Функция выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса. Подлежат учету зависимости в виде равенств и неравенств.

Аргументы **equalities_dep_func** – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами (размерность возвращаемого вектора соответствует размерности вектор-функции \mathbf{f}_M);
inequalities_dep_func – функция, возвращающую результат вычисления левых частей системы неравенств вида $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) \geq \mathbf{0}$, представленных в формате равенств $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$ (в соответствии с методом множителей Лагранжа), описывающую взаимосвязи (ограничения) между согласуемыми величинами;
msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;
true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи;
prior_params – одномерный массив, содержащий априорно заданные или оцененные меры погрешности результатов измерений;
bandwidth – одномерный массив, содержащий регуляризирующее условие на результат ядерной аппроксимации неизвестного распределения случайных погрешностей согласуемых измерений.

Данные значения являются ширинами окон ядерной аппроксимации. В качестве аппроксимирующего ядра использовано ядро Гаусса.

Возвращаемое значение **Reconciled** – одномерный массив результатов непараметрического согласования результатов измерений взаимосвязанных величин. Предполагается, что измерения выполнялись независимо, и корреляция между случайными погрешностями отсутствует. В качестве процедуры идентификации неизвестного распределения случайных погрешностей согласуемых измерений применен метод ядерной аппроксимации ядром Гаусса.

Процедура робастного согласования совместных измерений в предположении, что случайные погрешности распределены нормально

DRparamEqIneqRobust

Вызов `[reconciled] = DRparamEqIneqRobust(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, true_params)`

Описание Функция выполняет робастное параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей. Подлежат учету зависимости в виде равенств и неравенств.

Аргументы **equalities_dep_func** – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами (размерность возвращаемого вектора соответствует размерности вектор-функции \mathbf{f}_M); **inequalities_dep_func** – функция, возвращающую результат вычисления левых частей системы неравенств вида $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) \geq \mathbf{0}$, представленных в формате равенств $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$ (в соответствии с методом множителей Лагранжа), описывающую взаимосвязи (ограничения) между согласуемыми величинами; **msrd_data** – массив, содержащий согласуемые результаты измерений, **error_params** – одномерный массив, содержащий меры погрешности независимо полученных результатов измерений, подлежащих согласованию, **true_params** – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.

Возвращаемое значение **Reconciled** – одномерный массив результатов параметрического согласования совместных измерений, выполненного при следующих допущениях: случайные погрешности результатов совместных измерений распределены нормально; согласуемые результаты

измерений получены независимо, и, следовательно, корреляция между случайными погрешностями отсутствует

Процедура семи-непараметрического согласования с представлением неизвестного закона распределения погрешностей рядом Грама-Шарлье

DRsemiparamEqIneqRobust

Вызов	<pre>[reconciled] = DRsemiparamEqIneqRobust(@equalities_dep_func, @inequalities_dep_func, msrd_data, error_params, alpha_params, true_params);</pre>
Описание	<p>Файл содержит одноименную вызываемую функцию, которая выполняет робастное полу-непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грама-Шарлье. Подлежат учету зависимости в виде равенств и неравенств.</p>
Аргументы	<p>equalities_dep_func – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами (размерность возвращаемого вектора соответствует размерности вектор-функции \mathbf{f}_M);</p> <p>inequalities_dep_func – функция, возвращающую результат вычисления левых частей системы неравенств вида $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) \geq \mathbf{0}$, представленных в формате равенств $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$ (в соответствии с методом множителей Лагранжа), описывающую взаимосвязи (ограничения) между согласуемыми величинами;</p> <p>msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;</p> <p>error_params – одномерный массив, содержащий меры погрешности результатов измерений;</p> <p>alpha_params – вектор-массив, содержащий оценки мер погрешностей результатов измерений, коэффициентов асимметрии, коэффициентов и эксцесса случайных отклонений результатов измерений;</p> <p>true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи.</p>
Возвращаемое значение	<p>Reconciled – одномерный массив результатов семи-непараметрического согласования совместных измерений.</p>

Процедура непараметрического робастного согласования с ядерной аппроксимацией распределения случайных погрешностей

DRnonparamEqIneqRobust

Вызов	<pre>[reconciled] = DRnonparamEqIneqRobust(@equalities_dep_func, @inequalities_dep_func, msrd_data, true_params, error_params, bandwidths)</pre>
Описание	Функция выполняет робастное непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса. Подлежат учету зависимости в виде равенств и неравенств.
Аргументы	<p>equalities_dep_func – функция, возвращающую результат вычисления левых частей системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, описывающую взаимосвязи между согласуемыми величинами (размерность возвращаемого вектора соответствует размерности вектор-функции \mathbf{f}_M);</p> <p>inequalities_dep_func – функция, возвращающую результат вычисления левых частей системы неравенств вида $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) \geq \mathbf{0}$, представленных в формате равенств $\mathbf{g}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$ (в соответствии с методом множителей Лагранжа), описывающую взаимосвязи (ограничения) между согласуемыми величинами;</p> <p>msrd_data – массив, значения результатов совместного измерения величин, подлежащих согласованию;</p> <p>true_params – одномерный массив, содержащий известные точно параметры уравнений взаимосвязи;</p> <p>prior_params – одномерный массив, содержащий априорно заданные или оцененные меры погрешности результатов измерений;</p> <p>bandwidth – одномерный массив, содержащий регуляризирующее условие на результат ядерной аппроксимации неизвестного распределения случайных погрешностей согласуемых измерений. Данные значения являются ширинами окон ядерной аппроксимации. В качестве аппроксимирующего ядра использовано ядро Гаусса.</p>
Возвращаемое значение	<p>Reconciled – одномерный массив результатов непараметрического согласования результатов измерений взаимосвязанных величин. Предполагается, что измерения выполнялись независимо, и корреляция между случайными погрешностями отсутствует. В качестве процедуры идентификации неизвестного распределения случайных погрешностей согласуемых измерений применен метод ядерной аппроксимации ядром Гаусса.</p>

Конструктор объектов типа PBox, служащих для отображения множества возможных значений согласуемых величин и/или их неопределенности в форме области возможных значений функции распределения

PBox

Вызов	[obj] = PBox(x, lowerCDF, upperCDF)
Описание	Функция представляет собой конструктор класса типа PBox, служащего для отображения возможных значений и/или неопределенности согласуемых величин в виде области возможных значений функции распределения.
Аргументы	x – набор значений, отражающих узлы сетки дискретизации значений аргумента кумулятивной функции распределения (cdf); lowerCDF – набор значений нижней границы области допустимых значений функции распределения; upperCDF – набор значений верхней границы области допустимых значений функции распределения.
Возвращаемое значение	obj – созданный объект типа PBox.

Конструктор объектов типа Hist, служащих для отображения множества возможных значений согласуемых величин и/или их неопределенности в форме интервальной гистограммы по Берлинту

Hist

Вызов	[obj] = Hist(x, lowerPDF, upperPDF)
Описание	Функция представляет собой конструктор класса типа Hist, служащего для отображения возможных значений и/или неопределенности согласуемых величин в виде гистограммы функции плотности распределения с интервальным заданием возможных значений высоты ее полос.
Аргументы	x – массив значений, последовательно содержащий границы полос гистограммы (pdf); lowerPDF – массив значений, отображающих нижние границы множества допустимых значений высоты полос гистограммы; upperPDF – массив значений, отображающих верхние границы множества допустимых значений высоты полос гистограммы.
Возвращаемое значение	obj – созданный объект типа Hist.

Конструктор объектов типа DempsterShafer, служащих для отображения множества возможных значений согласуемых величин и/или их неопределенности в форме структуры Демпстера-Шафера

DempsterShafer

Вызов	<code>[obj] = DempsterShafer(intervals, masses)</code>
Описание	Функция представляет собой конструктор класса типа DempsterShafer, служащего для отображения возможных значений и/или неопределенности согласуемых величин в виде структуры Демпстера-Шафера.
Аргументы	<code>intervals</code> – массив интервалов для фокальных элементов; <code>masses</code> – соответствующие им массы.
Возвращаемое значение	<code>obj</code> – созданный объект типа DempsterShafer.

Конструктор объектов типа Fuzzy, служащих для отображения множества возможных значений согласуемых величин и/или их неопределенности в форме нечеткой переменной по Заде

Fuzzy

Вызов	<code>[obj] = Fuzzy(universe, membership)</code>
Описание	Функция представляет собой конструктор класса типа Fuzzy, служащего для отображения возможных значений и/или неопределенности согласуемых величин в виде нечеткой переменной.
Аргументы	<code>universe</code> – значения носителя нечеткой переменной; <code>membership</code> – значения функции принадлежности (от 0 до 1).
Возвращаемое значение	<code>obj</code> – созданный объект типа Fuzzy.

Конструктор объектов типа FuzzyInterval, служащих для отображения множества возможных значений согласуемых величин и/или их неопределенности в форме нечеткого интервала

FuzzyInterval

Вызов	<code>[obj] = FuzzyInterval(alphaLevels, intervals)</code>
-------	--

Описание	Функция представляет собой конструктор класса типа FuzzyInterval, служащего для отображения возможных значений и/или неопределенности согласуемых величин в виде нечеткого интервала (включает в себя как предельный случай классический интервал, а исчисление нечетких интервалов – классическую интервальную арифметику по Муру).
Аргументы	<code>alphaLevels</code> – так называемые значения α -cut (уровни значений функции принадлежности); <code>intervals</code> – соответствующие вложенные интервалы (nested intervals).
Возвращаемое значение	<code>obj</code> – созданный объект типа FuzzyInterval.

Конструктор объектов типа Sample, служащих для отображения выборки значений результатов многократных измерений

Sample

Вызов	<code>[obj] = Sample(x)</code>
Описание	Функция представляет собой конструктор класса типа Sample, служащего для отображения выборки значений результатов многократных измерений одной и той же отображаемой величины.
Аргументы	<code>x</code> – массив значений, образующих выборку значений результатов многократных измерений отображаемой величины.
Возвращаемое значение	<code>obj</code> – созданный объект типа Sample.

Функция преобразования переменной типа PBox в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

PBox2Hist

Вызов	<code>[Histogram] = PBox2Hist(obj, numBins)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа PBox в переменную типа Hist с минимизацией потерь содержащейся в PBox информации о возможных значениях отображаемой величины.
Аргументы	<code>obj</code> – преобразуемый экземпляр класса PBox; <code>numBins</code> – количество полос в создаваемом экземпляре класса Hist
Возвращаемое значение	<code>obj</code> – созданный объект типа Hist.

Функция преобразования переменной типа PBox в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

PBox2DempsterShafer

Вызов	[ds] = PBox2DempsterShafer(obj, numFocal)
Описание	Функция представляет собой процедуру преобразования переменной типа PBox в переменную типа DempsterShafer с минимизацией потерь содержащейся в PBox информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса PBox; numFocal – количество фокальных элементов в создаваемом экземпляре класса DempsterShafer
Возвращаемое значение	ds – созданный объект типа DempsterShafer.

Функция преобразования переменной типа PBox в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

PBox2Fuzzy

Вызов	[fv] = PBox2Fuzzy(obj, numPoints)
Описание	Функция представляет собой процедуру преобразования переменной типа PBox в переменную типа Fuzzy с минимизацией потерь содержащейся в PBox информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса PBox; numPoints – количество значений универсального множества для создаваемой нечеткой переменной (экземпляре класса Fuzzy)
Возвращаемое значение	fv – созданный объект типа Fuzzy.

Функция преобразования переменной типа PBox в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

PBox2FuzzyInterval

Вызов	[fi] = PBox2FuzzyInterval(obj, numAlpha)
Описание	Функция представляет собой процедуру преобразования переменной типа PBox в переменную типа FuzzyInterval с минимизацией потерь содержащейся в PBox информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса PBox; numAlpha – количество значений уровней значений функции принадлежности для множества вложенных интервалов, образующих создаваемый экземпляр класса FuzzyInterval.
Возвращаемое значение	fv – созданный объект типа FuzzyInterval.

Функция преобразования переменной типа Hist в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Hist2PBox

Вызов	[pbox] = Hist2PBox(obj)
Описание	Функция представляет собой процедуру преобразования переменной типа Hist в переменную типа PBox с минимизацией потерь содержащейся в Hist информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса Hist.
Возвращаемое значение	pbox – созданный объект типа PBox.

Функция преобразования переменной типа Hist в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Hist2DempsterShafer

Вызов	[ds] = Hist2DempsterShafer(obj)
Описание	Функция представляет собой процедуру преобразования переменной типа Hist в переменную типа DempsterShafer с минимизацией потерь содержащейся в Hist информации о возможных значениях отображаемой величины.

Аргументы `obj` – преобразуемый экземпляр класса `Hist`.

Возвращаемое значение `ds` – созданный объект типа `DempsterShafer`.

Функция преобразования переменной типа `Hist` в переменную типа `Fuzzy` с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Hist2Fuzzy

Вызов `[fv] = Hist2Fuzzy(obj, numUniverse)`

Описание Функция представляет собой процедуру преобразования переменной типа `Hist` в переменную типа `Fuzzy` с минимизацией потерь содержащейся в `Hist` информации о возможных значениях отображаемой величины.

Аргументы `obj` – преобразуемый экземпляр класса `Hist`;
`numUniverse` – количество значений универсального множества для создаваемой нечеткой переменной (экземпляре класса `Fuzzy`)

Возвращаемое значение `fv` – созданный объект типа `Fuzzy`.

Функция преобразования переменной типа `Hist` в переменную типа `FuzzyInterval` с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Hist2FuzzyInterval

Вызов `[fi] = Hist2FuzzyInterval(obj, numAlpha)`

Описание Функция представляет собой процедуру преобразования переменной типа `Hist` в переменную типа `FuzzyInterval` с минимизацией потерь содержащейся в `Hist` информации о возможных значениях отображаемой величины.

Аргументы `obj` – преобразуемый экземпляр класса `Hist`;
`numAlpha` – количество значений уровней значений функции принадлежности для множества вложенных интервалов, образующих создаваемый экземпляр класса `FuzzyInterval`.

Возвращаемое значение `fv` – созданный объект типа `FuzzyInterval`.

Функция преобразования переменной типа DempsterShafer в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

DempsterShafer2PBox

Вызов	[pbox] = DempsterShafer2PBox(obj, numPoints)
Описание	Функция представляет собой процедуру преобразования переменной типа DempsterShafer в переменную типа PBox с минимизацией потерь содержащейся в DempsterShafer информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса DempsterShafer; numPoints – количество значений в сетке дискретизации границ области возможных значений функции распределения
Возвращаемое значение	pbox – созданный объект типа PBox.

Функция преобразования переменной типа DempsterShafer в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

DempsterShafer2Hist

Вызов	[Histogram] = DempsterShafer2Hist(obj, numBins)
Описание	Функция представляет собой процедуру преобразования переменной типа DempsterShafer в переменную типа Hist с минимизацией потерь содержащейся в DempsterShafer информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса DempsterShafer; numBins – количество полос в создаваемом экземпляре класса Hist (интервальнозначеной гистограмме)
Возвращаемое значение	Histogram – созданный объект типа Hist.

Функция преобразования переменной типа DempsterShafer в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

DempsterShafer2Fuzzy

Вызов	[fv] = DempsterShafer2Fuzzy(obj, numPoints)
Описание	Функция представляет собой процедуру преобразования переменной типа DempsterShafer в переменную типа Fuzzy с

минимизацией потерь содержащейся в DempsterShafer информации о возможных значениях отображаемой величины.

Аргументы	obj – преобразуемый экземпляр класса DempsterShafer; numPoints – количество значений универсального множества для создаваемой нечеткой переменной (экземпляре класса Fuzzy)
Возвращаемое значение	fv – созданный объект типа Fuzzy.

Функция преобразования переменной типа DempsterShafer в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

DempsterShafer2FuzzyInterval

Вызов	[fi] = DempsterShafer2FuzzyInterval(obj)
Описание	Функция представляет собой процедуру преобразования переменной типа DempsterShafer в переменную типа FuzzyInterval с минимизацией потерь содержащейся в DempsterShafer информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса DempsterShafer.
Возвращаемое значение	fv – созданный объект типа FuzzyInterval.

Функция преобразования переменной типа Fuzzy в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Fuzzy2PBox

Вызов	[pbox] = Fuzzy2PBox(obj , numPoints)
Описание	Функция представляет собой процедуру преобразования переменной типа Fuzzy в переменную типа PBox с минимизацией потерь содержащейся в Fuzzy информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса Fuzzy; numPoints – количество значений в сетке дискретизации границ области возможных значений функции распределения
Возвращаемое значение	pbox – созданный объект типа PBox.

Функция преобразования переменной типа Fuzzy в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Fuzzy2Hist

Вызов	[Histogram] = Fuzzy2Hist(obj, numBins)
Описание	Функция представляет собой процедуру преобразования переменной типа Fuzzy в переменную типа Hist с минимизацией потерь содержащейся в Fuzzy информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса Fuzzy; numBins – количество полос в создаваемом экземпляре класса Hist
Возвращаемое значение	Histogram – созданный объект типа Hist.

Функция преобразования переменной типа Fuzzy в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Fuzzy2DempsterShafer

Вызов	[ds] = Fuzzy2DempsterShafer(obj, numFocal)
Описание	Функция представляет собой процедуру преобразования переменной типа Fuzzy в переменную типа DempsterShafer с минимизацией потерь содержащейся в Fuzzy информации о возможных значениях отображаемой величины.
Аргументы	obj – преобразуемый экземпляр класса Fuzzy; numFocal – количество фокальных элементов в создаваемом экземпляре класса DempsterShafer
Возвращаемое значение	ds – созданный объект типа DempsterShafer.

Функция преобразования переменной типа Fuzzy в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

Fuzzy2FuzzyInterval

Вызов	[fi] = Fuzzy2FuzzyInterval(obj, numAlpha)
Описание	Функция представляет собой процедуру преобразования переменной типа Fuzzy в переменную типа FuzzyInterval с минимизацией потерь содержащейся в Fuzzy информации о возможных значениях отображаемой величины.

Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>Fuzzy</code> ; <code>numAlpha</code> – количество значений уровней значений функции принадлежности для множества вложенных интервалов, образующих создаваемый экземпляр класса <code>FuzzyInterval</code> .
-----------	--

Возвращаемое значение	<code>fv</code> – созданный объект типа <code>FuzzyInterval</code> .
-----------------------	--

Функция преобразования переменной типа `FuzzyInterval` в переменную типа `PBox` с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

FuzzyInterval2PBox

Вызов	<code>[pbox] = FuzzyInterval2PBox(obj, numPoints)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа <code>FuzzyInterval</code> в переменную типа <code>PBox</code> с минимизацией потерь содержащейся в <code>FuzzyInterval</code> информации о возможных значениях отображаемой величины.
Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>FuzzyInterval</code> ; <code>numPoints</code> – количество значений в сетке дискретизации границ области возможных значений функции распределения
Возвращаемое значение	<code>pbox</code> – созданный объект типа <code>PBox</code> .

Функция преобразования переменной типа `FuzzyInterval` в переменную типа `Hist` с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

FuzzyInterval2Hist

Вызов	<code>[Histogram] = FuzzyInterval2Hist(obj, numBins)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа <code>FuzzyInterval</code> в переменную типа <code>Hist</code> с минимизацией потерь содержащейся в <code>FuzzyInterval</code> информации о возможных значениях отображаемой величины.
Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>FuzzyInterval</code> ; <code>numBins</code> – количество полос в создаваемом экземпляре класса <code>Hist</code>
Возвращаемое значение	<code>Histogram</code> – созданный объект типа <code>Hist</code> .

Функция преобразования переменной типа FuzzyInterval в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

FuzzyInterval 2DempsterShafer
--

Вызов	<code>[ds] = FuzzyInterval2DempsterShafer(obj, method)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа FuzzyInterval в переменную типа DempsterShafer с минимизацией потерь содержащейся в FuzzyInterval информации о возможных значениях отображаемой величины.
Аргументы	<code>obj</code> – преобразуемый экземпляр класса FuzzyInterval; <code>method</code> – метод преобразования: 'fractional' или 'nested'.
Возвращаемое значение	<code>ds</code> – созданный объект типа DempsterShafer.

Функция преобразования переменной типа FuzzyInterval в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование)

FuzzyInterval2Fuzzy

Вызов	<code>[fv] = FuzzyInterval2Fuzzy(obj, numPoints)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа FuzzyInterval в переменную типа Fuzzy с минимизацией потерь содержащейся в FuzzyInterval информации о возможных значениях отображаемой величины.
Аргументы	<code>obj</code> – преобразуемый экземпляр класса FuzzyInterval; <code>numPoints</code> – количество значений универсального множества для создаваемой нечеткой переменной (экземпляре класса Fuzzy)
Возвращаемое значение	<code>fv</code> – созданный объект типа Fuzzy.

Функция преобразования переменной типа Sample в переменную типа PBox (построение области возможных значений функции распределения по выборке результатов многократных измерений согласуемой величины)

Sample2PBox

Вызов	<code>[pbox] = Sample2PBox(obj)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа Sample в переменную типа PBox.

Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>Sample</code> .
Возвращаемое значение	<code>pbox</code> – созданный объект типа <code>PBox</code> .

Функция преобразования переменной типа `Sample` в переменную типа `Hist` (построение интервальнозначной гистограммы по Берлинту по выборке результатов многократных измерений согласуемой величины)

Sample2Hist

Вызов	<code>[Histogram] = Sample2Hist(obj, numBins)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа <code>Sample</code> в переменную типа <code>Hist</code> .
Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>Sample</code> ; <code>numBins</code> – количество полос в создаваемом экземпляре класса <code>Hist</code>
Возвращаемое значение	<code>Histogram</code> – созданный объект типа <code>Hist</code> .

Функция преобразования переменной типа `Sample` в переменную типа `DempsterShafer` (построение структуры Демпстера-Шафера по выборке результатов многократных измерений согласуемой величины)

Sample2DempsterShafer

Вызов	<code>[ds] = Sample2DempsterShafer(obj, numFocal)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа <code>Sample</code> в переменную типа <code>DempsterShafer</code> .
Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>Sample</code> ; <code>numFocal</code> – количество фокальных элементов в создаваемом экземпляре класса <code>DempsterShafer</code> .
Возвращаемое значение	<code>ds</code> – созданный объект типа <code>DempsterShafer</code> .

Функция преобразования переменной типа `Sample` в переменную типа `Fuzzy` (построение нечеткой переменной по выборке результатов многократных измерений согласуемой величины)

Sample2Fuzzy

Вызов	<code>[fv] = Sample2Fuzzy(obj, numPoints)</code>
-------	--

Описание	Функция представляет собой процедуру преобразования переменной типа <code>Sample</code> в переменную типа <code>Fuzzy</code> .
Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>Sample</code> ; <code>numPoints</code> – количество значений универсального множества для создаваемой нечеткой переменной (экземпляре класса <code>Fuzzy</code>)
Возвращаемое значение	<code>fv</code> – созданный объект типа <code>Fuzzy</code> .

Функция преобразования переменной типа `Sample` в переменную типа `FuzzyInterval` (построение нечеткого интервала по выборке результатов многократных измерений согласуемой величины)

Sample2FuzzyInterval

Вызов	<code>[fi] = Sample2FuzzyInterval(obj, numAlpha)</code>
Описание	Функция представляет собой процедуру преобразования переменной типа <code>Sample</code> в переменную типа <code>FuzzyInterval</code> .
Аргументы	<code>obj</code> – преобразуемый экземпляр класса <code>Sample</code> ; <code>numAlpha</code> – количество значений уровней значений функции принадлежности для множества вложенных интервалов, образующих создаваемый экземпляр класса <code>FuzzyInterval</code> .
Возвращаемое значение	<code>fv</code> – созданный объект типа <code>FuzzyInterval</code> .

Функция графического отображения экземпляров классов, служащих для отображения множества возможных значений и/или неопределенности согласуемых величин

Plot

Вызов	<code>[] = plot(obj)</code>
Описание	Функция представляет собой процедуру графического отображения информации, содержащейся в объекте <code>obj</code> , который должен относиться к одному из типов представления неопределенности: <code>PBox</code> , <code>Hist</code> , <code>DempsterShafer</code> , <code>Fuzzy</code> , <code>FuzzyInterval</code> , <code>Sample</code> .
Аргументы	<code>obj</code> – отображаемый экземпляр одного из классов представления информации о неопределенности: <code>PBox</code> , <code>Hist</code> , <code>DempsterShafer</code> , <code>Fuzzy</code> , <code>FuzzyInterval</code> , <code>Sample</code> .
Возвращаемое значение	нет.

Функция построения интервала возможных значений (доверительного интервала) среднеквадратического отклонения того множества возможных значений или неопределенности согласуемой величины, что отражается экземплярами классов PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample

getStd

Вызов	<code>[stdCint] = getStd(obj, method)</code>
Описание	Функция оценки возможных значений среднеквадратического отклонения неточной величины, представленной в одном из форматов представления неопределенности (метод классов PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample).
Аргументы	<code>obj</code> – экземпляр одного из классов представления информации о неопределенности: PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample; <code>method</code> – метод оценки множества возможных значений среднеквадратического отклонения: ‘approximate’ – приближенная, но быстрая оценка; ‘accurate’ – точная, но более продолжительная по времени оценка.
Возвращаемое значение	<code>stdCint</code> – границы интервала возможных значений среднеквадратического отклонения (доверительного интервала), большее из значений которых используется для дальнейшего выполнения согласования индустриальных данных.

Процедура выполнения согласования неточных величин в рамках аналитической модели с решением задачи Каруша-Куна-Таккера с применением теоремы Ефремова-Козлова

AnalyticalModel

Вызов	<code>[y] = AnalyticalModel(x, A_y, b_y, c_y, r2_y)</code>
Описание	Процедура выполнения согласования неточных величин в рамках аналитической модели с решением задачи Каруша-Куна-Таккера с применением теоремы Ефремова-Козлова.
Аргументы	<code>x</code> – вектор значений согласуемых величин (результаты совместных измерений на объекте, чья математическая модель описывает ограничения на возможные значения согласуемых значений); <code>A_y</code> – матрица линейных ограничений, накладываемых на согласуемые значения, вида $A_y \cdot y = b_y$; <code>b_y</code> – вектор правых частей в линейных ограничениях, накладываемых на согласуемые значения, вида $A_y \cdot y = b_y$; <code>c_y</code> – вектор, соответствующий координатам центра эллипсоида в пространстве значений вектора <code>x</code> , накладывающего групповое условие типа неравенства на возможную область допустимых значений вектора <code>y</code> , вида $(y - c_y)^T \cdot (y - c_y) = r2_y$;

$r2_y$ – величина группового ограничения на возможную область допустимых значений вектора y , вида $(y - c_y)^T \cdot (y - c_y) = r2_y$.

Возвращаемое значение y – результаты согласования значений по аналитической модели процедуры Data Reconciliation, основанной на теореме Ефремова-Козлова.

Процедура выполнения оценки неопределенности результата согласования по аналитической модели AnalyticalModel при представлении неопределенности исходных данных в одном из перечисленных форматов: PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample

GetUncertainty

Вызов `[dy] = GetUncertainty(x, dx, A_y, b_y, c_y, r2_y)`

Описание Процедура выполнения оценки неопределенности результата согласования по аналитической модели AnalyticalModel при представлении неопределенности исходных данных в одном из перечисленных форматов: PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample.

Аргументы x – вектор значений согласуемых величин (результаты совместных измерений на объекте, чья математическая модель описывает ограничения на возможные значения согласуемых значений);
 dx – массив экземпляров одного из классов выражения неопределенности согласуемых величин x (на выбор пользователя – PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample);
 A_y – матрица линейных ограничений, накладываемых на согласуемые значения, вида $A_y \cdot y = b_y$;
 b_y – вектор правых частей в линейных ограничениях, накладываемых на согласуемые значения, вида $A_y \cdot y = b_y$;
 c_y – вектор, соответствующий координатам центра эллипсоида в пространстве значений вектора x , накладывающего групповое условие типа неравенства на возможную область допустимых значений вектора y , вида $(y - c_y)^T \cdot (y - c_y) = r2_y$;
 $r2_y$ – величина группового ограничения на возможную область допустимых значений вектора y , вида $(y - c_y)^T \cdot (y - c_y) = r2_y$.

Возвращаемое значение dy – массив оценок неопределенности результатов согласования значений по аналитической модели процедуры Data Reconciliation, основанной на теореме Ефремова-Козлова; результаты возвращаются в виде экземпляра того же класса, что и dx .

А.5.3 Примеры и особенности

Приведенная в листинге А.1 программа предназначена для оценки потенциального повышения точности результатов прямых совместных измерений взаимосвязанных величин за счет учета этих взаимосвязей. Программа выполняет метрологический анализ измерительных ситуаций, допускающих учет формализованных зависимостей между

измеряемыми величинами. Допускается две ситуации: когда погрешности параметров модели, описывающей эти взаимосвязи, известны; когда погрешностями параметров модели можно пренебречь.

Листинг А.1 – Пример функции, оценивающей повышение точности, достигаемое за счет согласования результатов измерений по известной модели (с использованием метода локальной линеаризации)

```

1 function [accuracy_increase_ratio, variances_of_DR_result] = ...
2   est_accuracy_increase_by_DR(dep_func, vals_for_reconc, no_error_params, cov_matrix)
3 num_equations = length(dep_func(vals_for_reconc, no_error_params));
4 num_vals = length(vals_for_reconc);
5 variances_of_DR_result = zeros([num_vals,1]);
6 accuracy_increase_ratio = zeros([num_vals,1]);
7 dF_dxj = zeros([num_equations,1]);
8 Jacobian = zeros([num_equations,num_vals - 1]);
9 for j = 1 : num_vals
10   nums_without_j_th = [1 : 1 : j - 1, j + 1 : 1 : num_vals];
11   tmp_mx = cov_matrix;
12   tmp_mx(j,:) = [];
13   tmp_mx(:,j) = [];
14   cov_matrix_without_j_th = tmp_mx;
15   for i = 1 : num_equations
16     dF_dxj(i) = get_imag_numeric_derivative(dep_func, vals_for_reconc, no_error_params, j, i);
17     for j_ = 1 : num_vals - 1
18       Jacobian(i, j_) = get_imag_numeric_derivative(dep_func, vals_for_reconc,
19 no_error_params, nums_without_j_th(j_), i);
20     end
21   end
22   var_of_delta_x_j = zeros([num_equations, 1]);
23   weights_for_WLS = zeros([1, num_equations]);
24   for j_ = 1 : num_equations
25     sum_df_j_dz = Jacobian(j_,:).^2*diag(cov_matrix_without_j_th);
26     var_of_delta_x_j(j_) = dF_dxj(j_)^(-2)*sum_df_j_dz;
27   end
28   for j_ = 1 : num_equations
29     tmp_w = var_of_delta_x_j(j_)./var_of_delta_x_j;
30     tmp_w(isnan(tmp_w)) = 1;
31     weights_for_WLS(j_) = 1/sum(tmp_w);
32   end
33   linear_est_of_indirect_measurement_variance = weights_for_WLS.^2*var_of_delta_x_j;
34   accuracy_increase_ratio(j) = sqrt(1 +
35 cov_matrix(j,j)/linear_est_of_indirect_measurement_variance);
36   variances_of_DR_result(j) = cov_matrix(j,j)/accuracy_increase_ratio(j)^2;
37 end
38 end
39 function df_dx = get_imag_numeric_derivative(func, params, no_error_params, deriv_param_num,
40 equation_num)
41   alpha = params.*0;
42   alpha(deriv_param_num) = params(deriv_param_num)*10^(-100);
43   df = func(params + alpha*1i, no_error_params) - func(params, no_error_params);
44   df_dx = imag(df(equation_num))/alpha(deriv_param_num);end

```

Программа представляет собой вызываемую функцию `est_accuracy_increase_by_DR` на языке MATLAB, которой передаются: указатель `dep_func` на математическую модель \mathbf{f}_M , описывающую взаимосвязи между измеряемыми величинами и формализованную в виде системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$, где $\mathbf{f}_M = (f_1, f_2, \dots)^T$ – вектор функций, описывающих взаимосвязи между согласуемыми величинами $\mathbf{x} = (x_1, x_2, \dots)^T$ и параметрами модели \mathbf{a} , заданными точно, если такие есть, $\mathbf{0}$ – вектор, заполненный нулями, той же длины, что и \mathbf{f}_M . Значения `no_error_params` представляют собой одномерный массив параметров \mathbf{a} , погрешностью которых можно пренебречь. Одномерный массив `vals_for_reconc` содержит результаты измерения, подлежащие согласованию, а элементы двумерного массива `cov_matrix` должны представлять собой коэффициенты ковариационной матрицы.

Основная идея – разложить функционал, описывающий зависимости между величинами \mathbf{x} , в ряд Тейлора в области результатов измерений `vals_for_reconc` с усечением его до линейных слагаемых. Значения частных производных по неточно известным параметрам \mathbf{x} вычисляются в строках 16–21, и функция `get_imag_numeric_derivative` вычисляет производную по выбранному параметру, которой передаются следующие параметры:

`func` – указатель на вектор функций, производную одной из которых нужно вычислить,

`params` – вектор значений параметров функций, известных с погрешностью,

`no_error_params` – параметры функции, известные точно,

`deriv_param_num` – номер параметра, по которому нужно взять производную,

`equation_num` – номер функции в векторе уравнений \mathbf{f} , производную которой нужно вычислить.

В строках 33–36 последовательно вычисляются:

линейное приближение к средневзвешенному значению дисперсии результата оценки искомой величины x_j по всем уравнениям взаимосвязей, входящим в систему $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$,

коэффициент уточнения результата прямого измерения величины x_j за счет их согласования между собой,

дисперсия результата согласования результата измерения величины x_j с результатами измерений прочих величин.

Функция `est_accuracy_increase_by_DR` возвращает два массива:

`accuracy_increase_ratio` – приближенные оценки коэффициентов уточнения прямых измерений за счет согласования,

`variances_of_DR_result` – приближенные оценки дисперсий результатов согласования.

Листинг А.2 – Пример функции, осуществляющей непараметрическую сверку результатов совместных измерений взаимосвязанных величин (с использованием метода реконструкции проекции плотности вероятности)

```

1 function reconciled_with_AlphaGramCh = DRsemiparamEq(Func, msrd_data, prior_params,
2   alpha_params)
3     l = zeros([length(Func(msrd_data)), 1]);
4     check_ = size(msrd_data);
5     if check_(1) > 1
6         msrd_data = msrd_data';
7     end
8     mu_and_l_start = [mu; l];
9     reconciled_with_AlphaGramCh = fsolve(@AlphaSystemToSolve, mu_and_l_start, [], Func, msrd_data,
10    prior_params, alpha_params);
11 end
12 function mustbezeros = AlphaSystemToSolve(mu_and_l, SquareDependFunc, msrd_data, prior_params,
13    alpha_params)
14     vars = prior_params;
```

```

15     xNum = size(msrd_data, 2);
16     n = size(msrd_data, 1);
17     mustbezeros = zeros([xNum+1, 1]);
18     for j = 1 : xNum
19         imagx = zeros([xNum,1]);
20         imagx(j) = 1i*mu_and_l(j)*10^(-100);
21         df_dx = imag(SquareDependFunc(mu_and_l(1:xNum)+imagx))/imag(imagx(j));
22         mustbezeros(j) = mu_and_l(j)/vars{j} - mean(msrd_data(:,j))/vars{j}...
23         - calculMeanAlpha(mu_and_l(j),msrd_data(:,j),alpha_params(j))...
24         + mu_and_l(xNum+1)*df_dx/n;
25     end
26     mustbezeros(xNum+1) = SquareDependFunc(mu_and_l(1:xNum));
27 end
28 function alpha_mean = calculMeanAlpha(mj,msrd_data,alpha_params)
29     sj = sqrt(alpha_params(1));
30     As = alpha_params(2);
31     Ex = alpha_params(3);
32     tji = (msrd_data - mj)./sj;
33     d_alpha_d_mu = (Ex - 3).*(1/2.*tji - 1/6.*tji.^3) + As.*(1/2 - 1/2.*tji.^2);
34     alpha_plus_one = (Ex - 3).*(1/24.*tji.^4 + 1/8 - 1/4.*tji.^2) + As.*(1/6.*tji.^3 - 1/2.*tji) + 1;
35     y = d_alpha_d_mu./alpha_plus_one;
36     alpha_mean = (1/sj)*mean(y);
37 end

```

Следующая функция (листинг А.2) предназначена для уточнения результатов совместных измерений посредством выполнения процедуры семи-непараметрического согласования. Реализованный алгоритм включает в себя процедуру аппроксимации неизвестной плотности распределения случайных погрешностей согласуемых измерений усеченным рядом полиномов Грама-Шарлье. Данный подход имеет смысл и подразумевает, что случайные отклонения результатов измерений имеют распределение близкое к нормальному. Данные отличия характеризуются коэффициентами асимметрии и эксцесса, оценки которых передаются функции в виде массива `alpha_params`.

Программа представляет собой вызываемую функцию `DRsemiparamEq` на языке MATLAB, которой передаются следующие параметры. Указатель `Func` на математическую модель \mathbf{f}_M , описывающую взаимосвязи между измеряемыми величинами и формализованную в виде системы уравнений вида $\mathbf{f}_M(\mathbf{x}, \mathbf{a}) = \mathbf{0}$. Значения результатов однократного совместного измерения `msrd_data`, подлежащих согласованию. Массив `prior_params` априорно заданных или оцененных характеристик неопределенности (variances) результатов измерений. Также функции передается двумерный массив параметров `alpha_params`, представляющий собой оценки среднеквадратического отклонения результатов измерений, коэффициентов асимметрии, коэффициентов и эксцесса случайных отклонений результатов измерений величин \mathbf{x} .

Согласование неточных данных с помощью `DRsemiparamEq` эквивалентно численному решению системы некоторой уравнений, полученной путем решения задачи условной оптимизации методом множителей Лагранжа. При этом оптимизируемый функционал задан усеченным полиномиальным рядом Грама-Шарлье. Численное решение системы уравнений выполняется штатным инструментом MATLAB Statistical Toolbox. Программное описание левой части решаемой системы приведено в строках 10–27 и 27–38. Функция `AlphaSystemToSolve`, вычисляющей левую часть системы уравнений, принимает следующие аргументы:

`mu_and_1` – одномерный массив содержащий values of measured quantities, Lagrange multipliers and model parameters при которых вычисляется левая часть решаемой системы,

`SquareDependFunc` – указатель на функцию вида $f_M(\mathbf{x}, \mathbf{a})$,

`msrd_data` – согласуемые результаты измерений величин \mathbf{x} и \mathbf{a} ,

`prior_params` – одномерный массив оценок дисперсий измерений \mathbf{x} , \mathbf{a} ,

`alpha_params` – двумерный массив, содержащий заданные пользователем оценки среднеквадратических отклонений результатов измерений

Функция возвращает одномерный массив `reconciled_with_AlphaGramCh`, содержащий результаты согласования значений `msrd_data` по модели f_M .

A.6 СПИСОК ФУНКЦИЙ И МЕТОДОВ

estAccuracyIncreaseByDR()	Функция выполняет приближенную оценку потенциального уточнения совместных измерений, достигаемого за счет учета известных функциональных взаимосвязей между измеряемыми величинами, на основе локальной линеаризации модели и метода декомпозиции алгоритма условной оптимизации.
DRparamEq()	Функция выполняет параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей. Подлежат учету зависимости в виде равенств и неравенств.
DRsemiparamEq()	Файл содержит одноименную вызываемую функцию, которая выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грамма-Шарлье. Подлежат учету зависимости в виде равенств и неравенств.
DRnonparamEq()	Функция выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса. Подлежат учету зависимости в виде равенств и неравенств.
DRparamEqRobust()	Функция выполняет робастное параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей.
DRsemiparamEqRobust()	Файл содержит одноименную вызываемую функцию, которая выполняет робастное полу-непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грамма-Шарлье.
DRnonparamEqRobust()	Функция выполняет робастное непараметрическое согласование совместно измеренных величин, закон

	распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса.
DRparamEqIneq()	Функция выполняет параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей. Подлежат учету зависимости в виде равенств и неравенств.
DRsemiparamEqIneq()	Файл содержит одноименную вызываемую функцию, которая выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грамма-Шарлье. Подлежат учету зависимости в виде равенств и неравенств.
DRnonparamEqIneq()	Функция выполняет непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса. Подлежат учету зависимости в виде равенств и неравенств.
DRparamEqIneqRobust()	Функция выполняет робастное параметрическое согласование совместно измеренных величин, закон распределения случайных погрешностей которых соответствует нормальному распределению вероятностей. Подлежат учету зависимости в виде равенств и неравенств.
DRsemiparamEqIneqRobust()	Файл содержит одноименную вызываемую функцию, которая выполняет робастное полу-непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается проекционным методом с применением в качестве модели усеченного ряда Грамма-Шарлье. Подлежат учету зависимости в виде равенств и неравенств.
DRnonparamEqIneqRobust()	Функция выполняет робастное непараметрическое согласование совместно измеренных величин, закон распределения которых оценивается методом ядерной аппроксимации с использованием ядра Гаусса. Подлежат учету зависимости в виде равенств и неравенств.
PBox()	Конструктор класса PBox для канонического представления неопределенности входных данных в форме области возможных значений функции распределения. Границы представляемой области дискретизированы и представлены кусочно-постоянными функциями.
Hist()	Конструктор класса Hist для представления неопределенности входных данных в форме интервальной гистограммы по Берлинту, высота полос которой представлена интервалом возможных значений.
DempsterShafer()	Конструктор класса DempsterShafer для представления неопределенности входных данных в форме структуры Демпстера-Шафера.
Fuzzy()	Конструктор класса Fuzzy для представления неопределенности входных данных в форме нечеткой переменной по Заде.

FuzzyInterval()	Конструктор класса FuzzyInterval для представления неопределенности входных данных в форме нечеткого интервала (допускающего в том числе представление классического интервала по арифметике Мура как частный вырожденный случай).
Sample()	Конструктор класса Sample для представления информации о входных данных в виде выборки значений результатов многократных измерений.
PBox2Hist()	Функция преобразования переменной типа PBox в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
PBox2DempsterShafer()	Функция преобразования переменной типа PBox в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
PBox2Fuzzy()	Функция преобразования переменной типа PBox в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
PBox2FuzzyInterval()	Функция преобразования переменной типа PBox в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Hist2PBox()	Функция преобразования переменной типа Hist в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Hist2DempsterShafer()	Функция преобразования переменной типа Hist в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Hist2Fuzzy()	Функция преобразования переменной типа Hist в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Hist2FuzzyInterval()	Функция преобразования переменной типа Hist в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
DempsterShafer2PBox()	Функция преобразования переменной типа DempsterShafer в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
DempsterShafer2Hist()	Функция преобразования переменной типа DempsterShafer в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
DempsterShafer2Fuzzy()	Функция преобразования переменной типа DempsterShafer в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
DempsterShafer2FuzzyInterval()	Функция преобразования переменной типа DempsterShafer в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Fuzzy2PBox()	Функция преобразования переменной типа Fuzzy в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).

Fuzzy2Hist()	Функция преобразования переменной типа Fuzzy в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Fuzzy2DempsterShafer()	Функция преобразования переменной типа Fuzzy в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Fuzzy2FuzzyInterval()	Функция преобразования переменной типа Fuzzy в переменную типа FuzzyInterval с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
FuzzyInterval2PBox()	Функция преобразования переменной типа FuzzyInterval в переменную типа PBox с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
FuzzyInterval2Hist()	Функция преобразования переменной типа FuzzyInterval в переменную типа Hist с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
FuzzyInterval2DempsterShafer()	Функция преобразования переменной типа FuzzyInterval в переменную типа DempsterShafer с минимизацией потерь информации о неопределенности.
FuzzyInterval2Fuzzy()	Функция преобразования переменной типа FuzzyInterval в переменную типа Fuzzy с минимизацией потерь информации о неопределенности (квазиэквивалентное преобразование).
Sample2PBox()	Функция построения переменной типа PBox по выборке Sample значений одной из согласуемых величин.
Sample2Hist()	Функция построения переменной типа Hist по выборке Sample значений одной из согласуемых величин.
Sample2DempsterShafer()	Функция построения переменной типа DempsterShafer по выборке Sample значений одной из согласуемых величин.
Sample2Fuzzy()	Функция построения переменной типа Fuzzy по выборке Sample значений одной из согласуемых величин.
Sample2FuzzyInterval()	Функция построения переменной типа FuzzyInterval по выборке Sample значений одной из согласуемых величин.
plot()	Функция графического отображения экземпляров классов, выражающих неопределенность согласуемых значений (метод классов PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample).
getStd()	Функция оценки возможных значений среднеквадратического отклонения неточной величины, представленной в одном из форматов представления неопределенности (метод классов PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample).
AnalyticalModel()	Процедура выполнения согласования неточных величин в рамках аналитической модели с решением задачи Каруша-Куна-Таккера с применением теоремы Ефремова-Козлова.
GetUncertainty()	Процедура выполнения оценки неопределенности результата согласования по аналитической модели AnalyticalModel при представлении неопределенности исходных данных в одном из перечисленных форматов: PBox, Hist, DempsterShafer, Fuzzy, FuzzyInterval, Sample.