**Отчёт**

**о выполнении практического задания на тему**

**«Разработка базы знаний на основе продукционных**

**правил и машины вывода для решения задачи выбора**

**метода командного наведения летательного аппарата»**

**по курсу «Методы моделирования интеллектуальных систем управления»**

Магистрант \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / С.А. Минеев

Группа КММО-02-23

2024 Москва

# 1. Цель и задачи работы

Цель работы: углубление и закрепление знаний по моделям представлениям и обработки знаний в интеллектуальных системах, приобретение навыков анализа предметной области и формализации экспертных знаний.

Необходимо реализовать экспертную базу знаний для следующих трех методов наведения:

• метод погони;

• метод маневра;

• метод перехвата.

Для достижения поставленной цели необходимо:

* изучить соответствующие источники информации;
* проанализировать эти методы и выделить существенные факторы, которые позволяют сделать выбор в пользу того или иного метода наведения;
* на основе факторов построить ситуационный вектор;
* построить продукционные правила для связи вводимых параметров и рекомендуемых методов наведения в зависимости от их значений;
* определить структуру машины вывода и алгоритм ее работы;
* определить правила разрешения конфликтных ситуаций;
* реализовать алгоритмы;
* произвести эксперимент;
* сделать выводы.

# 2. Краткая характеристика методов наведения, анализ условий их применимости

Отметим, выделяются следующие *методы наведения* [Guidance methods]:

* методы командного наведения [Command guidance methods];
* методы самонаведения [Homing guidance methods].

*Методы командного наведения* [*Command guidance methods*] – это методы наведения, когда наземная станция, корабль или самолёт по радиоканалу передаёт ракете зашифрованные команды управления движением ракеты, которые приводят к перехвату цели. Радиокоманды могут также включать команды на подрыв боевой части, так как ракете не обязательно физически поразить цель, достаточно, чтобы цель находилась в секторе поражения осколками боевой части.

*Методы самонаведения* [Homing guidance methods]– это метод наведения, при котором в наводчике (истребитель, ракета, торпеда) имеется cистема, с помощью которой этот наводчик направляется к цели с помощью *автономного механизма*, который приводится в действие некоторыми отличительными характеристиками цели.

Описанные далее методы наведения могут использоваться как методы командного наведения, так и как методы самонаведения.

! В данной работе рассматриваются только *методы командного наведения* и без совместного использования с методами самонаведения, поэтому *выделять из описанных ниже характеристик* методов наведения в ситуационный вектор *характеристики, связанные с последовательностью выполнения* командных методов наведения и самонаведения *не требуется*.

! Также в данной работе рассматривается ситуация, когда истребитель и цель находятся только в горизонтальной плоскости и поэтому не требуется из характеристик методов наведения выделять в качестве требований характеристики, связанные с разными высотами истребителя и цели.

## 2.1. Метод погони (~ Прямой метод)

### 2.1.1. Качественный смысл метода

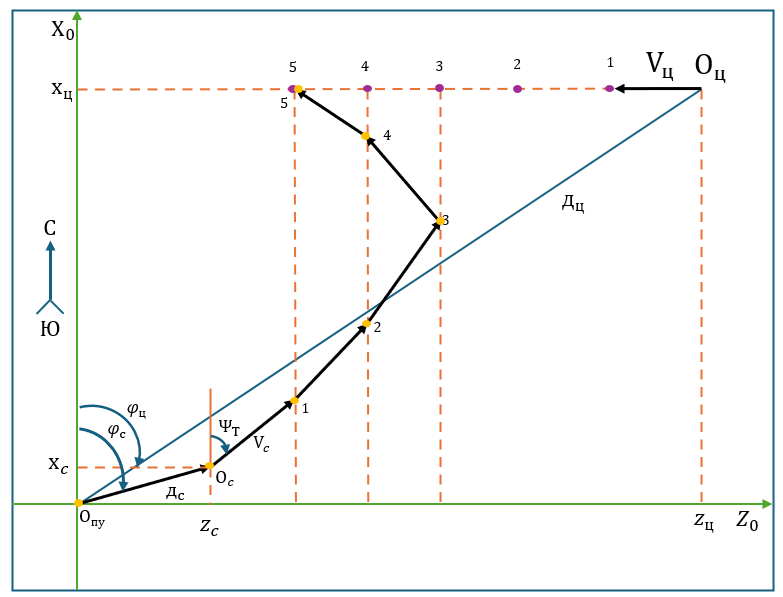


Рис. 1 – метод погони

Описание:

При прямом методе, называемом также методом погони, требуется все время совмещать продольную ось истребителя с направлением на цель (рис. 17.7). На этом рисунке точки и соответствуют положению истребителя и цели; и - скорости наводимого самолета и цели; , и , - текущие координаты цели и самолета. Из рис. 1 следует, что требуемый курс

где все координаты , и , формируются на основании измеренных в РЛС дальностей , и азимутов , цели и самолета по правилу:

# 2.1.2. Достоинства и недостатки метода

* **Достоинства**:
* инвариантность к дальности наведения (пригоден при различных дальностях) и высоте полета цели и самолёта;
* наведение в заднюю полусферу цели почти при любом исходном состоянии наводимого самолёта и цели;
* хорошее сопряжение с методами самонаведения самолётов и ракет   
  «В-В».
* **Недостатки**:
* криволинейная траектория наведения на подвижные цели;
* практическая невозможность использования для наведения на цель из её передней полусферы.

### 2.1.3. Ограничения на условия применения метода

На основе недостатков данного метода в качестве ограничений данного метода выступают:

* наведение на цель только из *задней полусферы*;
* *траектория является криволинейной* и, как следствие время, затрачиваемое на наведение ракеты на цель не минимально в общем случае.

## 2.2. Метод маневра

### 2.2.1. Качественный смысл метода

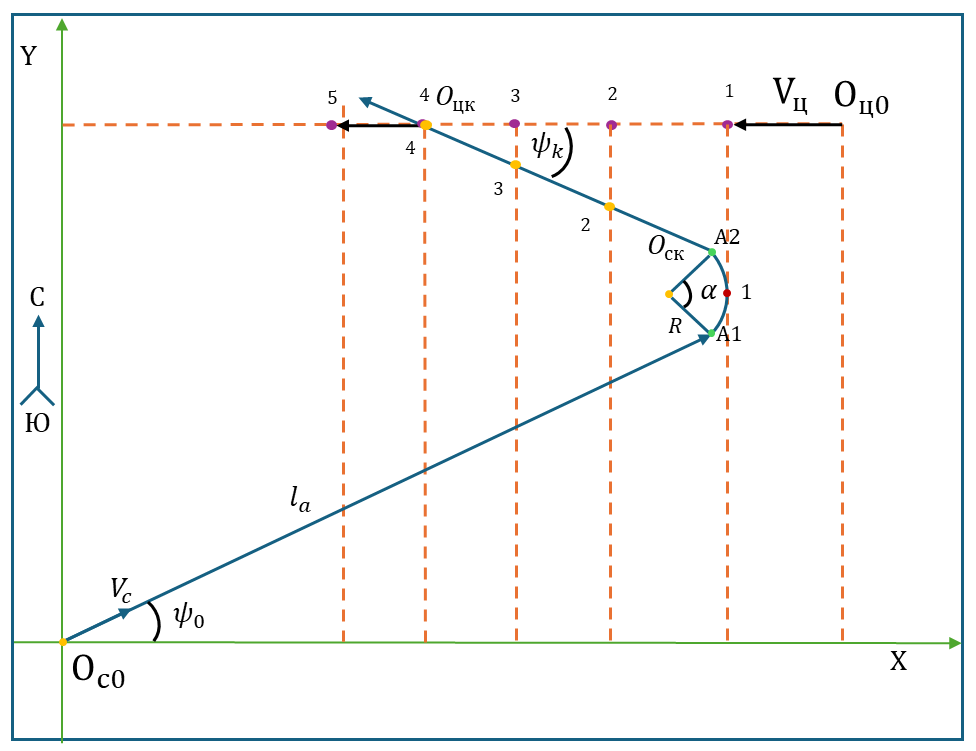


Рис. 2 – Метод маневра

Описание:

Метод маневра, называемый также методом прямой с разворотом, обеспечивает вывод истребителя в зону обнаружения цели бортовой РЛС, ОЭС или оптическим прицелом под заданным угломна заданном рас­ стоянии (рис. 2). На этом рисунке точки и соответствуют положению самолета и цели в начале дальнего наведения, в то время как и - в момент его окончания.

Траектория наведения при этом методе состоит из трех участков: отрезка прямой , дуги с радиусом , и отрезка прямой . Расчет этих участков выполняется по довольно сложным формулам, для реализации которых необходимо оценивать дальности до цели и истребителя, их азимуты и скорости. Кроме того, необходимо знать допустимые радиус разворота и диапазон углов , дальность рубежа захвата и суммарное время, затрачиваемое на вывод самолета в точку .

### 2.2.2. Достоинства и недостатки метода

* **Достоинства**:
* хорошее сопряжение со всеми методами самонаведения;
* возможность использования как радиолокационных, так и   
  оптико-электронных визирных систем (истребитель выводится на рубеж захвата в ЗПС, в которой ОЭС имеют наибольшую дальность захвата по факелу двигателя).
* **Недостатки**:
* большое время, затрачиваемое на выход самолёта на рубеж захвата;
* большой расход топлива;
* ограничения на ракурсы перехвата из ППС, обусловленные необходимостью вывода самолёта в ЗПС.

### 2.2.3. Ограничения на условия применения метода

На основе недостатков данного метода в качестве ограничений данного метода можно выделить:

* наличие ограничений на ракурсы перехвата из передней полусферы;

## 2.3. Метод перехвата

### 2.3.1. Качественный смысл метода

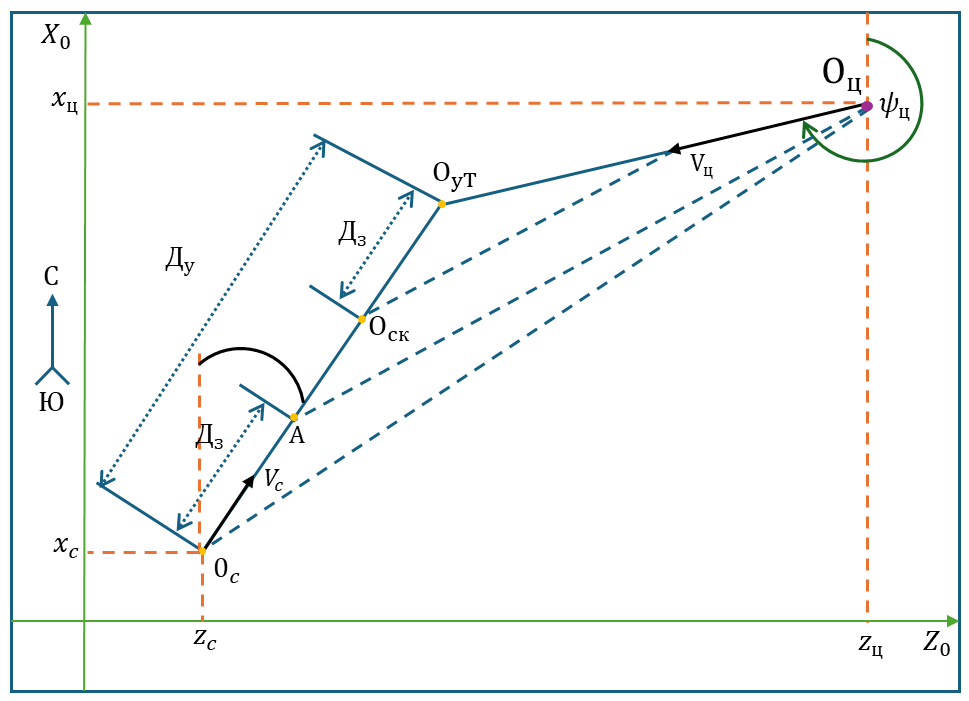


Рис. 3 – Метод перехвата

Описание:

**Метод перехвата** представляет собой разновидность метода параллельного сближения. Особенностью является то, что по методу параллельного сближения наводится не сам истребитель, а некоторая фиктивная точка **А**, расположенная по вектору скорости на расстоянии Д захвата цели бортовой визирной системой (рис. 17.9). Это означает, что в процессе дальнего наведения прямая перемещается параллельно самой себе. Такой прием обеспечивает нахождение истребителя в точке на рубеже захвата в момент, когда точка А «встретится» в упрежденной точке встречи с целью.

### 2.3.2. Достоинства и недостатки метода

* **Достоинства**:
* высокая экономичность наведения, обусловленная наведением в Упрежденную точку практически по прямолинейной траектории;
* обеспечение заданного рубежа перехвата при любом ракурсе наведения.
* **Недостатки**:
* невозможность сопряжения с прямыми методами самонаведения при перехвате цели в ППС (Передняя полусфера);
* отсутствие фиксированного ракурса атаки в момент окончания дальнего наведения (неудобно для дальнейшего применения визирных систем различной физической природы, например ОЭС);

### 2.3.3. Ограничения на условия применения метода

На основе недостатков данного метода в качестве ограничений данного метода можно выделить:

* если используется вместе с методами самонаведения, то только с методом перехвата;
* отсутствие ракурса, при котором метод позволяет истребителю выйти на заданный рубеж перехвата. (траектория в основном линейная)

# 3. Разработка базы знаний

## 3.1. Ситуационный вектор

Ситуационный вектор представляет собой описание параметров с помощью, которых задается условие критерия выбора метода наведения из экспертной базы знаний.

В ситуационный вектор будут входить некоторые *текущие параметры истребителя* и *параметры методов командного наведения.*

В качестве *текущих параметров истребителя* будут выступать, параметры, представленные в таблице № 1:

*Таблица № 1 – Текущие параметры истребителя*

|  |  |  |
| --- | --- | --- |
| Параметр | Значения | Описание |
| Pos | Передняя (0)/  задняя полусфера (1) | Текущее местоположение истребителя по отношению к цели. |
| Type | Радиолокационная (0)/  Оптическая (1) /  Радиолокационно-оптический (2) | Тип системы наведения ракет, предустановленный на истребителе. |

Пояснение:

В зависимости от *текущего местоположения истребителя* по *отношению к цели* и *наличия требования к скрытности* может быть определен *параметр скрытности*. Так, например, для *прямого метода* командного наведения и *метода маневра*, если истребитель находится в передней полусфере, а поразить цель должен из задней полусферы цели, то параметр скрытности выполняется, но если используется *метод перехвата и имеется наличие требования к скрытности, а* сам *истребитель находится в передней полусфере*, то, как следствие поразить цель он должен из передней полусферы (из-за свойств метода перехвата: Прямолинейная траектория), и в итоге требование к скрытности не будет выполнено для данного метода. Стоит выделить, что для каждого метода командного наведения в ситуационном векторе должно задаваться требование к скрытности (наличие либо отсутствие режима скрытности при использовании конкретного метода командного наведения) и если данное требование для конкретного метода командного наведения должно выполняться, то необходимо определить *параметр скрытности*, который будет определять, а выполняется ли *требование к скрытности* при *текущих параметрах ситуационного вектора* для выбранного метода командного наведения? Если выполняется, то происходит проверка остальных параметров продукционного правила, и в случае успеха выбирается данный метод в качестве рекомендуемого метода командного наведения, иначе данный метод командного наведения не рекомендуется и в окне программы должно быть определено соответствующее сообщение.

Рассмотрим параметр, связанный с типом системы наведения ракет.

Выделяют два типа, оптический и радиолокационный. В случае самонаведения ракет у них может быть только один тип наведения *–* либо оптический, либо радиолокационный. В случае с истребителем (как в нашей ситуации) тип системы наведения может включать как оба варианта, так и какой-то один из вариантов. В нашем случае *метод командного наведения* будет выбираться в зависимости от *типа системы наведения ракет заданного в качестве текущего параметра истребителя* и *параметров методов наведения, показывающих, при каких типах системы наведения пригодны эти методы наведения*.

При выборе методов наведения всегда стараются выбрать метод, тот который минимален по времени, и как следствие, по длине траектории, объёму израсходованного топлива и другим затрачиваемым ресурсам. Поэтому в качестве приоритета выполнения продукционных правил введем для каждого из методов командного наведения параметр длины траектории полета истребителя от начального местоположения до места поражения истребителем цели, который будет определять приоритет. Чем меньше длина, тем выше приоритет. Стоит отметить, что исходя из характеристик методов наведения самой короткой длиной траектории будет обладать метод перехвата т. к. его траектория прямолинейная в отличие от двух других методов. У двух оставшихся всегда траектория криволинейная. Будем также рассматривать ситуацию, когда все *длины траекторий различны*.

! В случае если оператор задает длину траектории для метода перехвата больше, чем у двух других методов должно выдаваться соответствующее сообщение о некорректности введенных данных.

Отметим, что т. к. мы используем разработку экспертной системы с помощью продукционных правил и с учетом условий задачи, то конфликтных ситуаций при правильном построении базы знаний можно избежать, что и предпринимается в данном практическом задании. Поэтому в этой работе не рассматриваются ситуации, когда могут возникнуть конфликты, а проектирование и реализация должны быть осуществлены так, чтобы проблем не возникало.

На основе выделенных качеств (достоинств и недостатков), ограничений трех методов наведения и предварительного выделения приоритета получены следующие параметры ситуационного вектора, которые представлены таблицей № 2 на следующей странице:

Таблица № 2 – Параметры методов наведения ситуационного вектора

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Значения** | **Описание** |
| Len | [0.0: …] (*Неотрицательное действительное число*) | Предварительно рассчитанная длина траектории для конкретного метода командного наведения. Согласно свойствам методов наведения она должна быть у метода перехвата меньше, чем у прямого метода наведения и метода маневра. У последних двух длины траекторий могут быть различными как в пользу метода маневра, так и в пользу прямого метода. Что и будет определять данный параметр. |
| S\_1 | Передняя (0) /Задняя (1) полусфера | Требование к исходному местонахождению ракеты (истребителя) по отношению к цели |
| S\_2 | Радиолокационная (0)/  Оптическая (1) | Тип системы наведения ракет, установленный на истребителе |
| S\_3 | В переднюю полусферу цели (0) / В заднюю полусферу цели (1) | Необходимость наведения в переднюю/заднюю полусферу цели |
| S\_4 | 0/1 | Наличие требование к скрытности |
| S\_5 | 0/1 | Реализуемость траектории наведения |
| S\_6 | 0/1 | Требования к отсутствию перегрузок |

Стоит отметить, что эти параметры должны быть реализованы для каждого метода, т. е. ситуационный вектор будет описываться (6 + 1) ∙ 3 + 2 параметрами ситуационного вектора и в следующем пункте приводится соответствующие выкладки.

## 3.2. Продукционные правила

В данном пункте задаются определенные критерии выбора метода наведения, по которым алгоритм ЭС (экспертной системы) сможет предоставить наиболее подходящий метод командного наведения.

! Условие критерия должно строиться на основе параметров, заложенных в ситуационном векторе, и должно включать как обязательные параметры, так и необязательные.

Ниже приводится предварительная таблица № 3, которая позволяет определить какой из методов *может быть выбран* при тех или иных значениях ситуационного вектора:

Таблица № 3 – Критерии и соответствующие значения в продукционных правилах для методов наведения

|  |  |  |  |
| --- | --- | --- | --- |
| **Описание** | **Метод погони (Direct guidance method [dgm])** | **Метод маневра (Maneuver guidance method [mgm])** | **Метод перехвата (Interception guidance method [igm])** |
| Длина траектории для конкретного метода командного наведения () (Автоматически сопоставляется с критерием минимального времени) [Определяет приоритет выполнения продукционных правил (см. п. 3.1)] | ! Длина меньше, чем у м. маневра и не выполнены критерии м. перехвата  [0.0: …] (*Неотрицательное действительное число*) | ! Длина меньше, чем у м. погони и не выполнены критерии м. перехвата  [0.0: …] (*Неотрицательное действительное число*) | ! Длина всегда меньше, чем у двух других.  [0.0: …] (*Неотрицательное действительное число*) |
| Требование к *исходному* местонахождению ракеты (истребителя) по отношению к цели ( | Передняя полусфера (1) | Передняя полусфера (1) | Передняя (1) / Задняя (0) полусфера |
| Тип системы наведения ракет, установленный на истребителе ( | Радиолокационная (1) | Радиолокационная (1) / Оптическая (0) | Радиолокационная (1) / Оптическая (0) |
| Необходимость наведение переднюю/заднюю полусферу цели ( | В заднюю полусферу цели (0) | В заднюю полусферу цели (0) | В передняя полусферу цели (1) [Только для (Может наводится в переднюю полусферу, только если нет требования к скрытности, иначе возникнет конфликт. И наводится в переднюю полусферу может только из передней полусферы. Как и для случая с задней полусферой) /  В задняя полусферу цели (0) [При любых ], но только если исходное местоположение истребителя определяется как местоположение в задней полусфере цели |
| Требование к скрытности ( [Выбирается оператором] | 0 / 1 | 0 / 1 | 0 [Для любых ] /  1 [Только если ] |
| Реализуемость траектории наведении ( | 0 / 1 | 0 (может изменить траекторию в наиболее предпочтительный момент времени) | 0 (Отсутствие фиксированного ракурса атаки) |
| Требования к отсутствию перегрузок ( | 0 | 0 | 1 |

В дальнейшем будем пользоваться следующим обозначением для составления продукционных правил:

− ситуационный вектор, - компонента ситуационного вектора,   
 − текущее местоположение истребителя по отношению к цели.

− номер параметра ситуационного вектора, - метод, для которого данный параметр устанавливается (см. таб. 2).

**–** это 1 для переменной, метода,

**–** это 0 для переменной, метода.

Продукционные правила:

* Предварительная версия:
* *|*

*|*

Т. к. имеются переменные, от которых результат не зависит, то по итогу получим следующие итоговые продукционные правила:

* Окончательная версия:
* *|*

*|*

* *,*

Т. к. в полученных правилах используется изначальное количество переменных, то это означает, что лишних выводов на предварительном этапе не было сформулировано и можно переходить к следующему шагу.

# 4. Разработка машины вывода

## 4.1. Входные данные

В качестве входных данных выступает условие критерия выбора метода наведения (Запрос). Условие критерия должно строиться на основе параметров, заложенных в ситуационном векторе, и должно включать значения для всех переменных ситуационного вектора.

## 4.2. Выходные данные

В качестве выходных данных выступает информация о наиболее подходящем методе при текущих заданных условиях.

## 4.3. Структура машины вывода

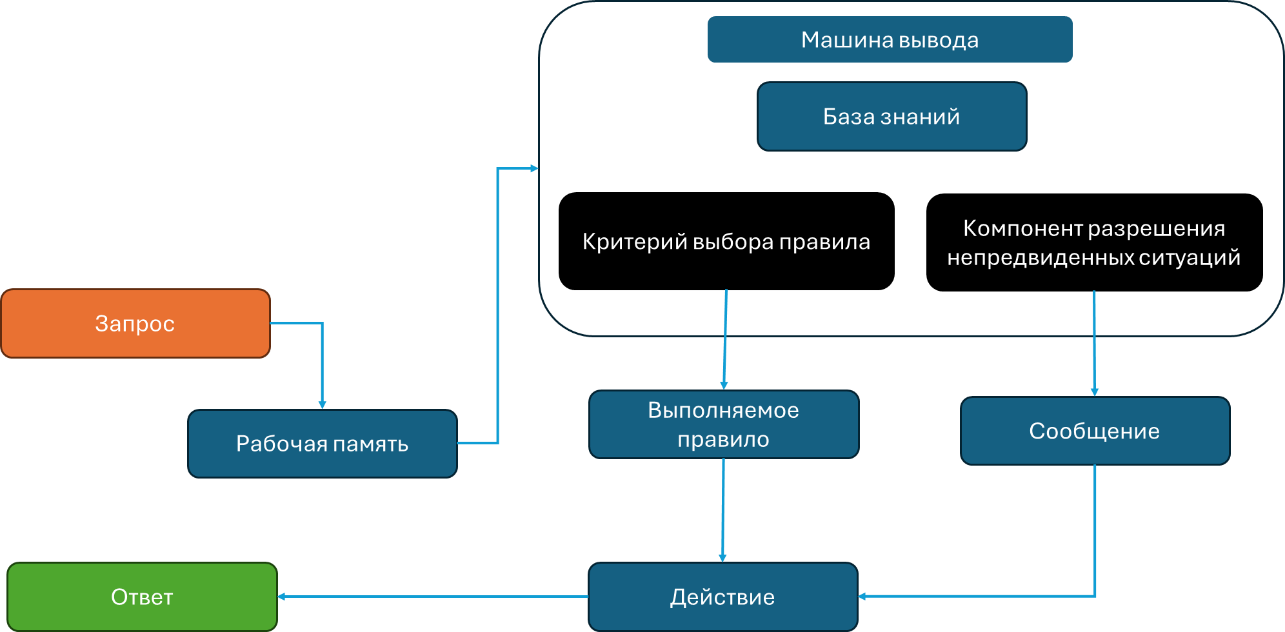


Рис. 4 - Структура машины вывода

Описание:

Оператор в виде *запроса* формирует в *рабочей памяти* последовательность параметров выбора метода наведения. После ввода всех необходимых параметров он отправляет запрос в *машину вывода,* которая осуществляет сопоставление введенных в запросе *параметров* с *критериями вывода* и ищет соответствующее *продукционное правило* в *Базе знаний.* В случае наличия соответствующих *совпадений из запроса* с одним из *продукционных правил* происходит *его выполнение (действие) и на выходе (в ответе)* оператор получает *рекомендуемый метод командного наведения.* Если в базе знаний не нашлось продукционного правила для конкретного запроса, то машина вывода формирует и выводит соответствующее сообщение либо и некорректности введенных данных, либо отсутствия продукционных правил для этого запроса.

## 4.4. Алгоритмы работы машины вывода

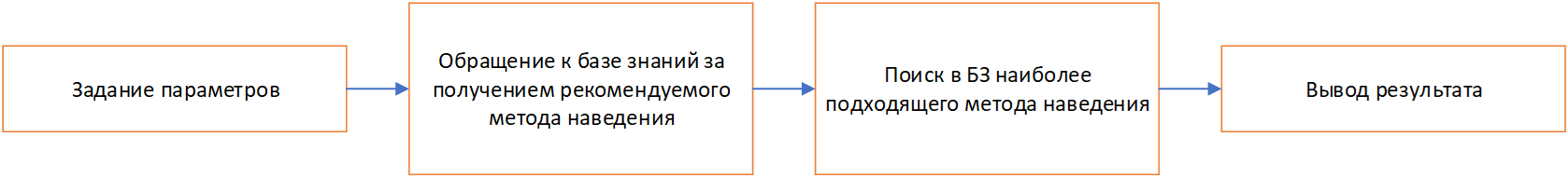


Рис. 5 – Алгоритм работы машины вывода

## 4.5. Исходный код машины вывода

Ниже приводится непосредственно код машины вывода, которая реализована через метод **:**

void **selectionChangedDecView**(*const* QItemSelection &**selected**, *const* QItemSelection &**deselected**){

*//* *Проверк* *выполнения* *условий* *длям.* *перехвата:*

*//* *lst* *[0];* *-* *Запрос* *№* *1;*

*//* *lst* *[1]* *-* *текущее* *местоположение* *истребителя;*

*//* *lst* *[2]* *-* *Тип* *системы* *наведения,* *установленный* *на* *истребителе;*

*//* *lst* *[3:7:9]* *-* *параметры* *"Метод* *погони";*

*//* *+* *lst[3]* *-* *Длина* *м.* *погони;*

*//* *+* *lst[4]* *-* *Требование* *к* *начальному* *положению* *истребителя;*

*//* *+* *lst[5]* *-* *Тип* *системы* *наведения;*

*//* *+* *lst[6]* *-* *Необходимость* *наведения* *;*

*//* *+* *lst[7]* *-* *Скрытность;*

*//* *+* *lst[8]* *-* *Реализуемость* *траектории;*

*//* *+* *lst[9]* *-* *Требования* *к* *перегрузкам;*

*//* *lst* *[10:7:16]* *-* *параметры* *"Метод* *маневра";*

*//* *+* *lst[10]* *-* *Длина* *м.* *пманевра;*

*//* *+* *lst[11]* *-* *Требование* *к* *начальному* *положению* *истребителя;*

*//* *+* *lst[12]* *-* *Тип* *системы* *наведения;*

*//* *+* *lst[13]* *-* *Необходимость* *наведения* *;*

*//* *+* *lst[14]* *-* *Скрытность;*

*//* *+* *lst[15]* *-* *Реализуемость* *траектории;*

*//* *+* *lst[16]* *-* *Требования* *к* *перегрузкам;*

*//* *lst* *[17:7:23]* *-* *параметры* *"Метод* *перехвата";*

*//* *+* *lst[17]* *-* *Длина* *м.* *перехвата;*

*//* *+* *lst[18]* *-* *Требование* *к* *начальному* *положению* *истребителя;*

*//* *+* *lst[19]* *-* *Тип* *системы* *наведения;*

*//* *+* *lst[20]* *-* *Необходимость* *наведения* *;*

*//* *+* *lst[21]* *-* *Скрытность;*

*//* *+* *lst[22]* *-* *Реализуемость* *траектории;*

*//* *+* *lst[23]* *-* *Требования* *к* *перегрузкам;*

QList<QVariant> **lst**;

foreach(auto **ix**, selected.indexes()) *//* *Получение* *списка* *выделенной* *записи*

lst << model\_workMemory->*data*(model\_workMemory->*index*(ix.row(), ix.column())); *//* *//func2(ix.row(),* *ix.column());*

qDebug() << lst;

*//* *lst\_len[0]* *-* *len* *м.* *погони.*

*//* *lst\_len[1]* *-* *len* *м.* *маневра.*

*//* *lst\_len[2]* *-* *len* *м.* *перехвата.*

QList<int> **lst\_len**;

*if*(lst.isEmpty() || lst.count() < model\_workMemory->*rowCount*()) *return*;

*for*(int **i** = 0; i < 3; i++)

lst\_len << lst.at(3 + 7\*i).toDouble();

qDebug() << lst\_len;

double **cond** = lst\_len.last();

std::sort(lst\_len.begin(), lst\_len.end());

*//* *Если* *длина* *м.* *перехвата* *меньше* *длины* *других* *методов* *наведения,то* *выход.*

*if*(std::abs(cond - lst\_len.first()) > 0.00001) {qDebug() << ""; *return*;}

*else* *if*(lst\_len[0] == lst\_len[1] || lst\_len[0] == lst\_len[2] || lst\_len[1]== lst\_len[2]) {

ui->line\_rec->setText(" - ");

ui->line\_result->setText("Установите корректную длину траектории для методов наведения!"); *return*;

}

*auto* **ff** = [&](int **row**, int **pos**)-> bool { *// Проверка критериев выбора*

*if*((model\_rule->*data*( *//* *Tекущее* *местонахождение* *истребителя* *по* *отношению* *к* *цели.*

model\_rule->*index*(row, 2)).toString().compare(lst[pos].toString()) != 0) &&

(model\_rule->*data*( *//* *Tекущее* *местонахождение* *истребителя* *по* *отношению* *к* *цели.*

model\_rule->*index*(row, 2)).toString().compare(lst[1].toString()) != 0)

) *return* *false*;

*if*((model\_rule->*data*( *//* *Тип* *системы* *наведения* *ракет* *установленный* *на* *истребителе.*

model\_rule->*index*(row, 3)).toString().compare(lst[pos + 1].toString()) != 0) &&

(model\_rule->*data*(model\_rule->*index*(row, 3)).toString().compare("Радиолокационно-Оптический") != 0)) *return* *false*;

*if*(model\_rule->*data*( *//* *Наведение* *в* *переднюю* *или* *заднюю* *полусферу.*

model\_rule->*index*(row, 4)).toString().compare(lst[pos + 2].toString()) != 0) *return* *false*;

*if*( model\_rule->*data*( *//* *Скрытность.*

model\_rule->*index*(row, 5)).toString().compare(lst[pos + 3].toString()) != 0) *return* *false*;

*if*( model\_rule->*data*( *//* *Реализуемость.*

model\_rule->*index*(row, 6)).toString().compare(lst[pos + 4].toString()) != 0) *return* *false*;

*if*( model\_rule->*data*( *//* *Перегрузки.*

model\_rule->*index*(row, 7)).toString().compare(lst[pos + 5].toString()) != 0) *return* *false*;

ui->line\_result->setText( *//* *Вывод рекомендуемого метода наведения*

model\_rule->*data*(

model\_rule->*index*(row, model\_rule->*columnCount*() - 1))

.toString());

ui->line\_rec->setText( *//* *Вывод рекомендатора (Продукционного правила из которого вывелся рекомендуемый метод.*

model\_rule->*data*(

model\_rule->*index*(row, 0))

.toString());

*return* *true*;

};

bool **status** = *false*;

qDebug() << "rowC: " << model\_rule->*rowCount*();

*for*(int **i** = 0; i < model\_rule->*rowCount*(); i++) *//* *Получаем* *выводы* *(методы* *наведения)*

*if*( model\_rule->*data*(

model\_rule->*index*(i, model\_rule->*columnCount*() - 1))

.toString()

.compare("Метод перехвата") == 0){

*if*(ff(i, 18)) *return*;

}*else* *if*(

(model\_rule->*data*( *//* *м.* *погони.*

model\_rule->*index*(i, model\_rule->*columnCount*() - 1))

.toString()

.compare("Метод погони") == 0) &&

(lst\_len[0] - lst\_len[1]) < 0.00001){ *//* *(lst\_len[0]* *(Длина* *м.* *погони)* *-* *lst\_len[1]* *(Длина* *м.* *маневра))* *<* *0.00001;*

*if*(ff(i, 4)) *return*; *//* *Цикл* *по* *продукционным* *правилам* *прекращается!*

} *else* *if*(

(model\_rule->*data*( *//* *м.* *маневра.*

model\_rule->*index*(i, model\_rule->*columnCount*() - 1))

.toString()

.compare("Метод маневра") == 0) &&

(lst\_len[1] - lst\_len[0]) < 0.00001) {

*if*(ff(i, 11)) *return*; *//* *Цикл* *по* *продукционным* *правилам* *прекращается!*

*else* status = *true*; *//* *пойдем* *на* *второй* *круг* *по* *м.* *погони*

}*else* *if*( *//* *Если* *мы* *тут* *оказались,* *значит* *приоритет* *для* *м.* *погони* *выполнился,* *но* *не* *выполнились* *требования.* *Переходим* *к* *м.* *маневра* *с* *нимжим* *приоритетом.*

(model\_rule->*data*( *//* *м.* *маневра.*

model\_rule->*index*(i, model\_rule->*columnCount*() - 1))

.toString()

.compare("Метод маневра") == 0) &&

(lst\_len[0] - lst\_len[1]) < 0.00001){

*if*(ff(i, 11)) *return*; *//* *Цикл* *по* *продукционным* *правилам* *прекращается!*

*else* {ui->line\_result->setText("Нет рекомендуемого метода наведения!");

ui->line\_rec->setText(" - ");}

}

*if*(status) *//* *Выполнился* *приоритет* *длины* *м.* *маневра,* *но* *раз* *мы* *тут* *оказались,* *значит* *не* *выполнились* *требования* *данного* *метода.* *Переходим* *к* *м.* *погони* *с* *нимжим* *приоритетом.*

*for*(int **i** = 1; i < model\_rule->*rowCount*(); i++)

*if*((model\_rule->*data*( *//* *м.* *погони.*

model\_rule->*index*(i, model\_rule->*columnCount*() - 1))

.toString()

.compare("Метод погони") == 0) &&(lst\_len[1] - lst\_len[0]) < 0.00001){

*//* *Идем* *на* *второй* *заход* *по* *м.* *погони.*

*if*(ff(i, 4)) *return*; *//* *Цикл* *по* *продукционным* *правилам* *прекращается!*

*else* {ui->line\_result->setText("Нет рекомендуемого метода наведения!");

ui->line\_rec->setText(" - ");}

}

}

# 5. Описание эксперимента

## 5.1. Исходные данные эксперимента

Рассмотрим следующие четыре случая (запроса) работы программы:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название\_запроса | Запрос № 1 | Запрос № 2 | Запрос № 3 | Запрос № 4 |
| Местоположение\_истребителя\_по\_отношению\_к\_цели | Передняя полусфера | Передняя полусфера | Передняя полусфера | Передняя полусфера |
| Тип\_системы\_наведения\_ракет | Радиолокационный | Радиолокационный | Радиолокационный | Оптический |
| Длина\_траектории\_метода\_погони | 100 | 100 | 90 | 0 |
| Метод\_погони\_Начальное\_местоположение\_истребителя\_по\_отношению\_к\_цели | Передняя полусфера | Передняя полусфера | Передняя полусфера | Передняя полусфера |
| Метод\_погони\_Тип\_системы\_наведения\_ракет | Радиолокационный | Радиолокационный | Радиолокационный | Радиолокационный |
| Метод\_погони\_Необходимость\_наведения | В переднюю полусферу | В переднюю полусферу | В заднюю полусферу | В переднюю полусферу |
| Метод\_погони\_Скрытность | Не скрытен | Не скрытен | Не скрытен | Не скрытен |
| Метод\_погони\_Реализуемость\_траектории | Не реализуема | Не реализуема | Не реализуема | Не реализуема |
| Метод\_погони\_Требование\_к\_перегрузкам | Отсутствуют перегрузки | Отсутствуют перегрузки | Присутствуют перегрузки | Отсутствуют перегрузки |
| Длина\_траектории\_метода\_маневра | 90 | 90 | 100 | 0 |
| Метод\_маневра\_Начальное\_местоположение\_истребителя\_по\_отношению\_к\_цели | Передняя полусфера | Передняя полусфера | Передняя полусфера | Передняя полусфера |
| Метод\_маневра\_Тип\_системы\_наведения\_ракет | Радиолокационный | Радиолокационный | Радиолокационный | Радиолокационный |
| Метод\_маневра\_Необходимость\_наведения | В переднюю полусферу | В заднюю полусферу | В заднюю полусферу | В переднюю полусферу |
| Метод\_маневра\_Скрытность | Не скрытен | Не скрытен | Не скрытен | Не скрытен |
| Метод\_маневра\_Реализуемость\_траектории | Не реализуема | Не реализуема | Не реализуема | Не реализуема |
| Метод\_маневра\_Требование\_к\_перегрузкам | Отсутствуют перегрузки | Присутствуют перегрузки | Отсутствуют перегрузки | Отсутствуют перегрузки |
| Длина\_траектории\_метода\_перехвата | 80 | 80 | 100 | 0 |
| Метод\_перехвата\_Начальное\_местоположение\_истребителя\_по\_отношению\_к\_цели | Передняя полусфера | Передняя полусфера | Передняя полусфера | Передняя полусфера |
| Метод\_перехвата\_Тип\_системы\_наведения\_ракет | Радиолокационный | Радиолокационный | Радиолокационный | Радиолокационный |
| Метод\_перехвата\_Необходимость\_наведения | В переднюю полусферу | В заднюю полусферу | В заднюю полусферу | В переднюю полусферу |
| Метод\_перехвата\_Скрытность | Не скрытен | Не скрытен | Не скрытен | Не скрытен |
| Метод\_перехвата\_Реализуемость\_траектории | Не реализуема | Не реализуема | Не реализуема | Не реализуема |
| Метод\_перехвата\_Требование\_к\_перегрузкам | Отсутствуют перегрузки | Отсутствуют перегрузки | Отсутствуют перегрузки | Отсутствуют перегрузки |

Таблица № 4 – Данные экспериментов

Пояснение к таблице № 4:

• В случае с запросом № 1 сначала определяется проверяется корректность длин траекторий для всех методов. Наибольшая длина траектории у прямолинейного метода перехвата, она ровна 100 и является корректной т. к. представляет собой максимальную для трех методов наведения. Далее т. к. длина траектория м. Перехвата корректна происходит проверка оставшихся требований для м. Перехвата. В данном случае, как видно по результатам параметры совпадают с продукционным правилом № 3 (Рекомендатор).

• В случае с запросом № 2 т. к. начальное (текущее местонахождение истребителя определяется в передней полусфере цели, а для м. перехвата требуется запросом преследовать(атаковать) цель из задней полусферы цели, то м. перехвата отпадает. В этом случае, определяется приоритет по длине траекторий двух оставшихся методов командного наведения, а именно м. маневра и м. перехвата. Т. к. e метода маневра длина траектории меньше, чем у м. перехвата () и т. к. требования запроса совпадают с продукционным правилом № 5, то рекомендуется м. маневра.

• В случае с запросом № 3 ситуация аналогичная предыдущему пункту, за исключением того, что минимальная длина траектории теперь у метода погони и т. к.

Выполняются требования из запроса с продукционным правилом № 9 для данного метода, то в рекомендациях на выходе его и получаем.

• В случае с запросом № 4 рассматривается ситуация, когда введены некорректные длины траекторий методов наведения. К длине траектории предъявляются следующие требования. Длина траектории м. погони должна быть меньше, чем и двух других. Длины траекторий не могут быть равными. В нашем случае они равны 0 и поэтому формируется соответствующее сообщение.

## 5.2 Демонстрация разработанных базы знаний и графического интерфейса программы

### 5.2.1 Демонстрация разработанной базы знаний

Разработка базы знаний велась с использованием SQLiteStudio v3.4.4 .

|  |  |
| --- | --- |
| Рис. 6 – Иконка базы знаний формата \*.db | Рис. 7 – Таблицы Базы знаний |

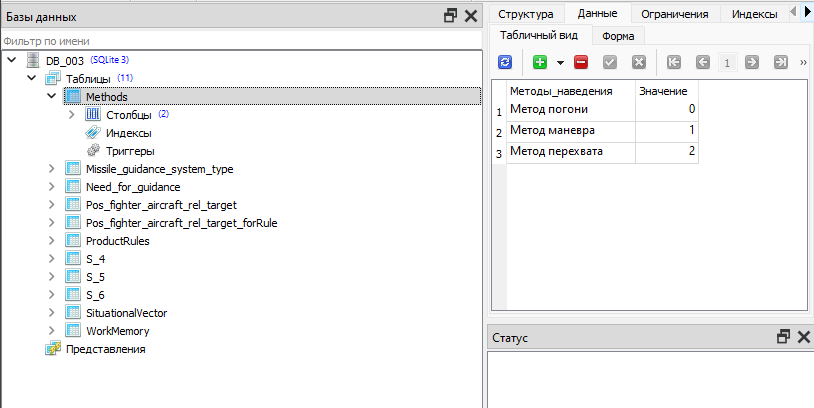


Рис. 8 – Содержимое таблицы предопределенных методов командного наведения

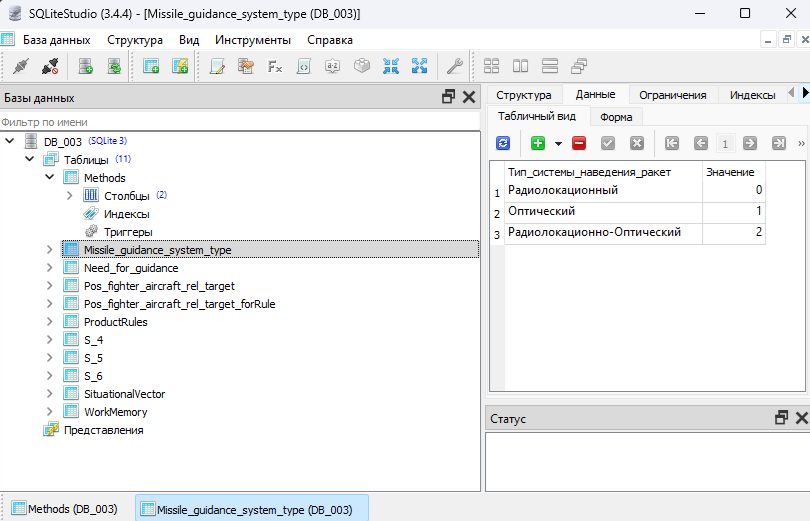


Рис. 9 – Содержимое таблицы типов систем установленных на истребителе

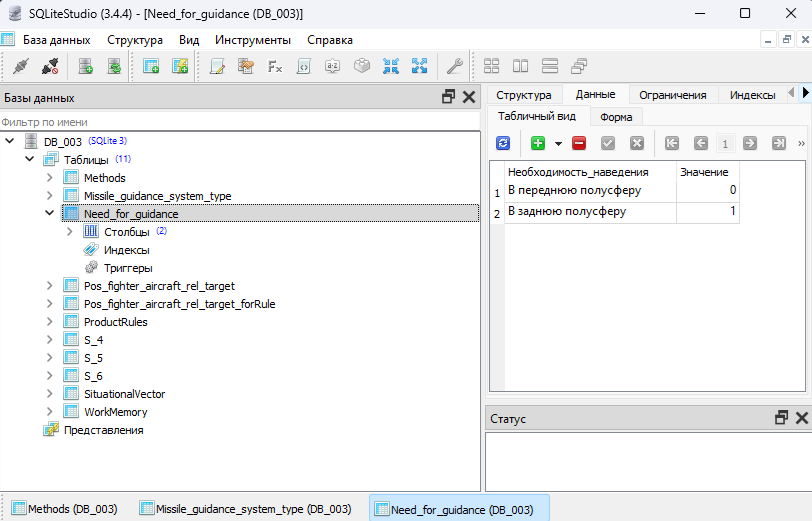


Рис. 10 – Содержимое таблицы Need\_for\_guidance

(Задает место наведение на цель по отношению к цели)

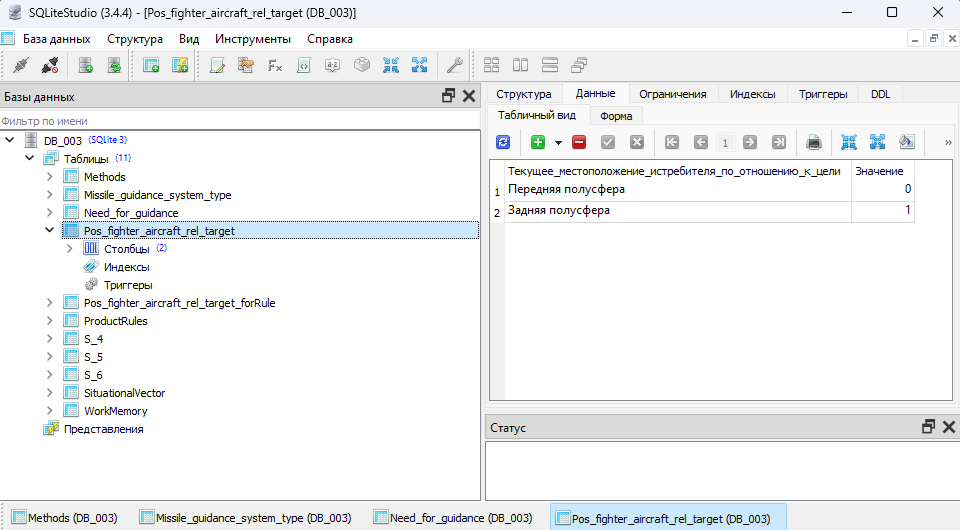


Рис. 11 – Таблица текущего местоположения истребителя по отношению к цели

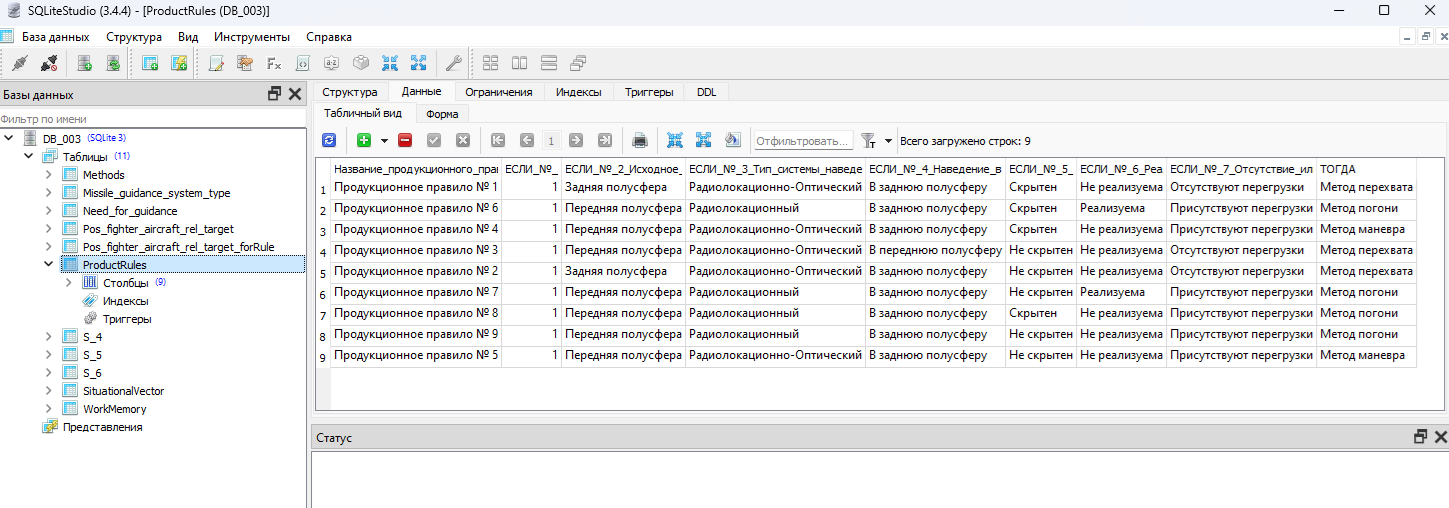


Рис. 12 – Таблица продукционных правил

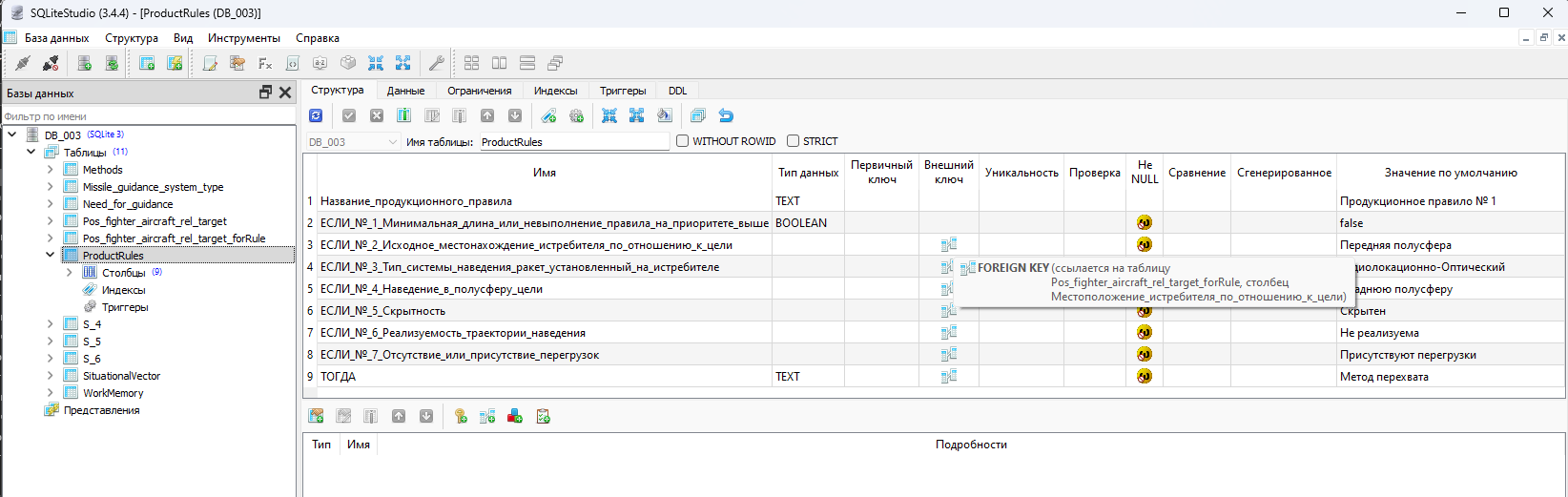


Рис. 13 – Демонстрация установки в окне ‘Структура’ таблицы внешнего ключа с другой таблицей в одной БЗ, которая позволяет организовать связь (отношение)

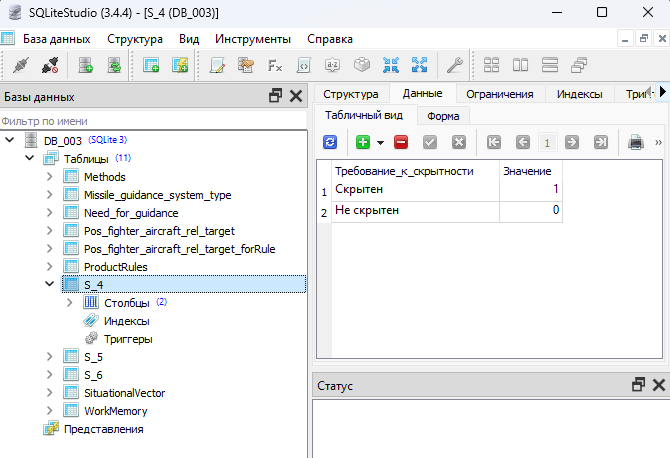


Рис. 14 – Содержимое таблицы S\_4 (Варианты требования к скрытности)

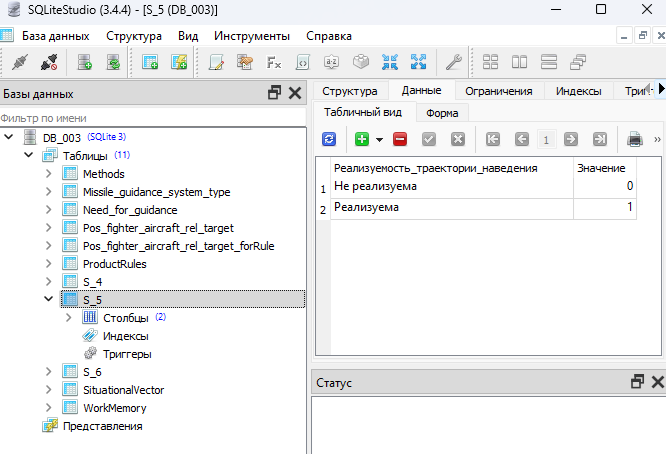


Рис. 15 - Содержимое таблицы S\_5 (Варианты требования к реализуемости)

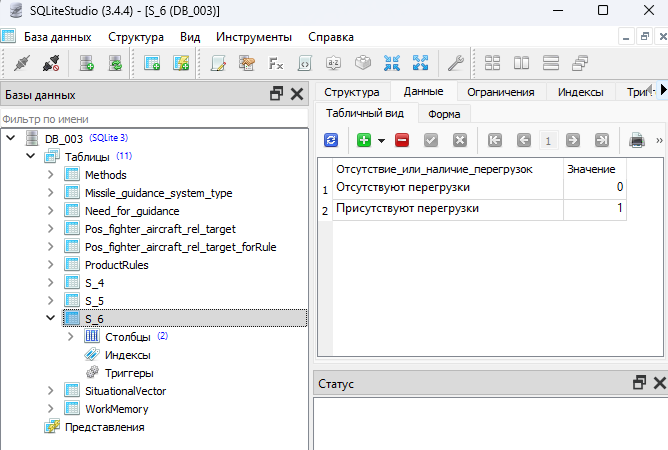


Рис. 16 - Содержимое таблицы S\_5 (Варианты требований к перегрузкам)

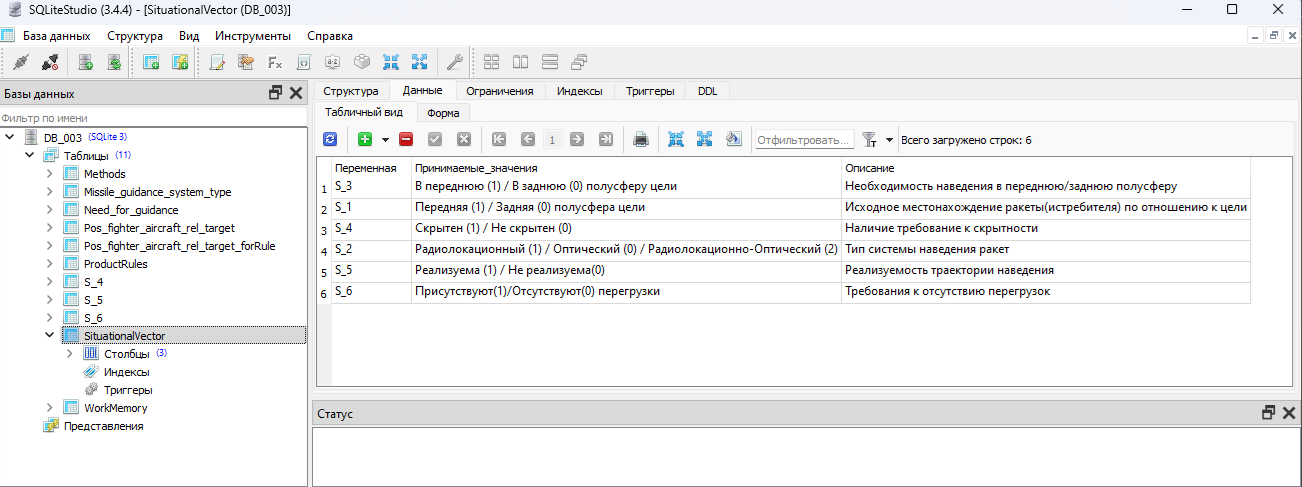


Рис. 17 – Таблица ситуационного вектора

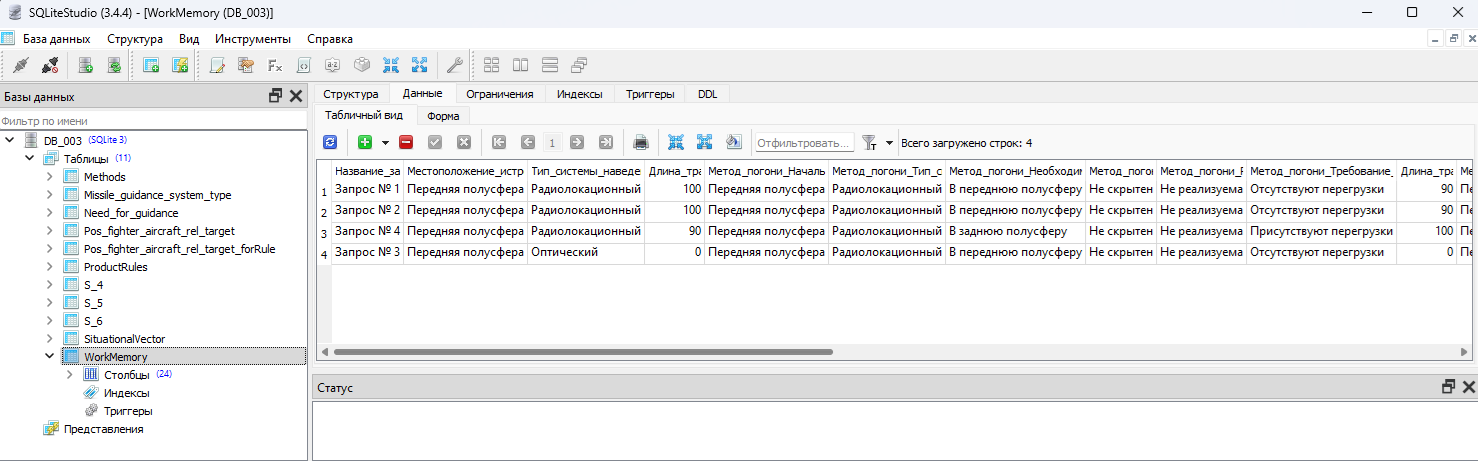


Рис. 18 – Таблица рабочей памяти

Программная реализация включает в себя использования в Qt Creator (Qt 6) двух составляющих, позволяющих упростить работу с базой знаний:

QSqlDatabase – компонент, отвечающий за основное взаимодействие с БЗ;

QSqlRelationalTableModel – компонент, позволяющий реализовать работу с БЗ с использованием отношений;

QSqlRelation – сами отношения, позволяют связать основную таблицу с зависимыми таблицами по внешним ключам таблиц.

Программа позволяет создавать запросы и продукционные правила, при этом используются предустановленные значения по умолчанию, которые при необходимости могут быть изменены оператором.

### 5.2.2 Демонстрация разработанной программы

Реализация графического интерфейса велась с использованием Qt Creator (Qt 6) и позволяет через графические представления и пиктограммы взаимодействовать с базой знаний. Пре использовании данного ПО необходимо сначала загрузить БЗ

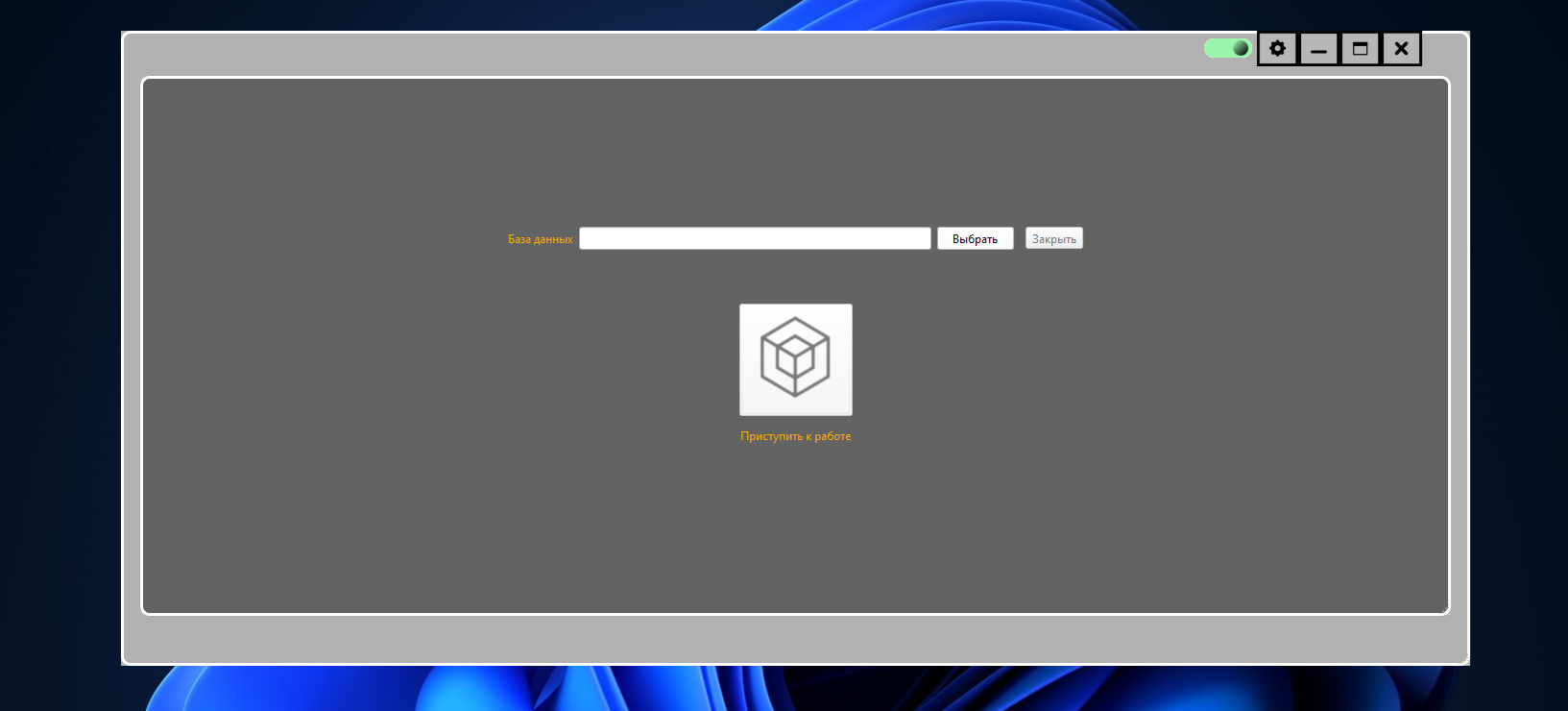


Рис. 19 – Демонстрация основного окна программы

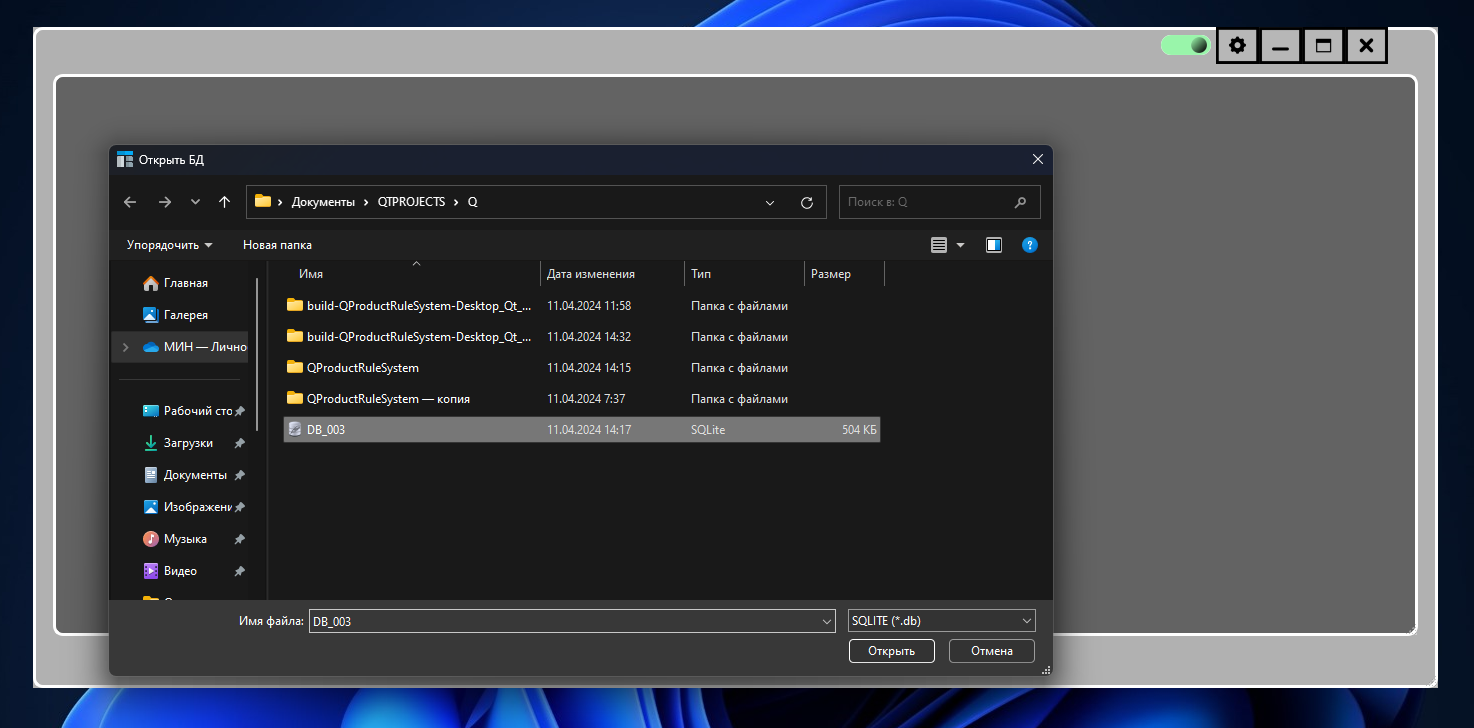


Рис. 20 – Демонстрация выбора базы знаний

При успешной загрузке БЗ формата \*.db(sqlite) загорится зеленый индикатор и оператор сможет приступить к работе, нажав на пиктограмму в центре.

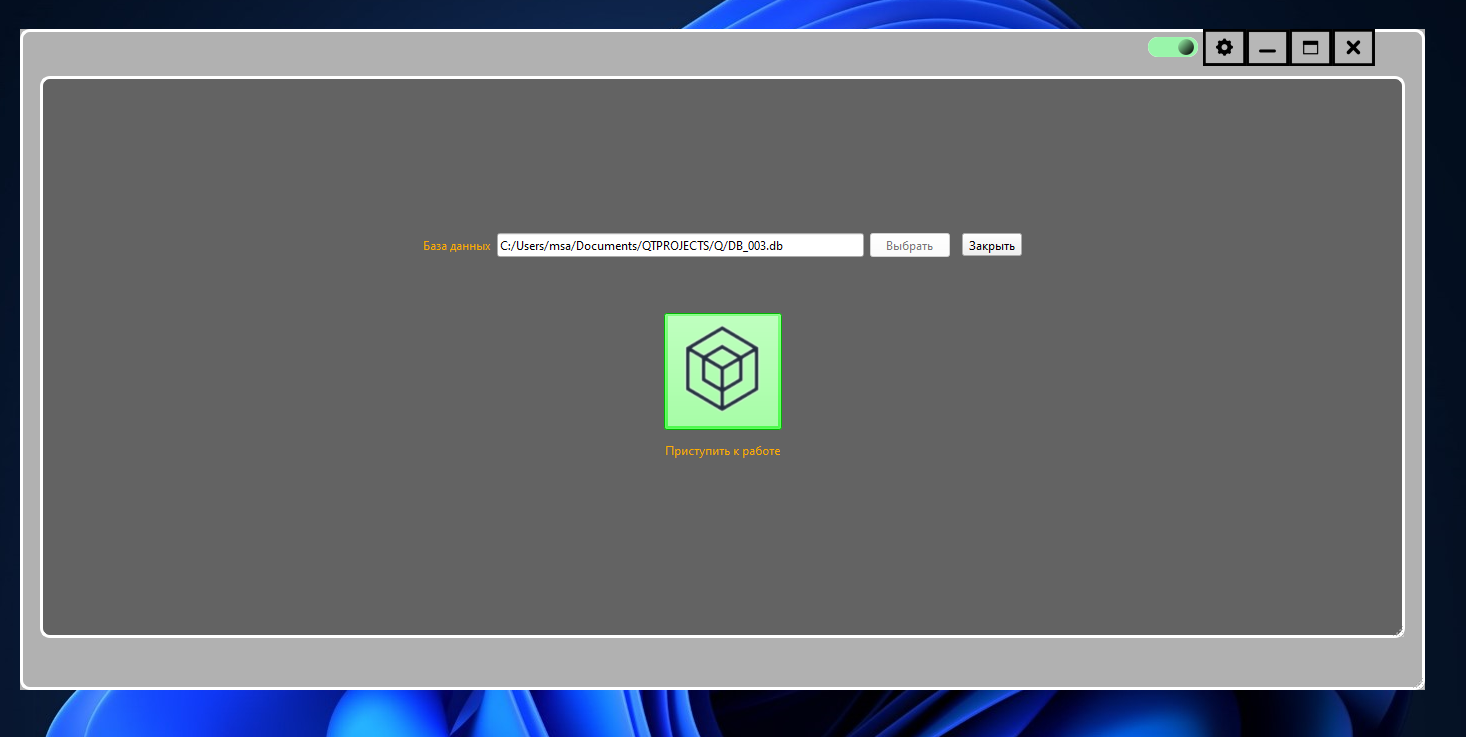


Рис. 21 – Демонстрация успешного подключения БЗ

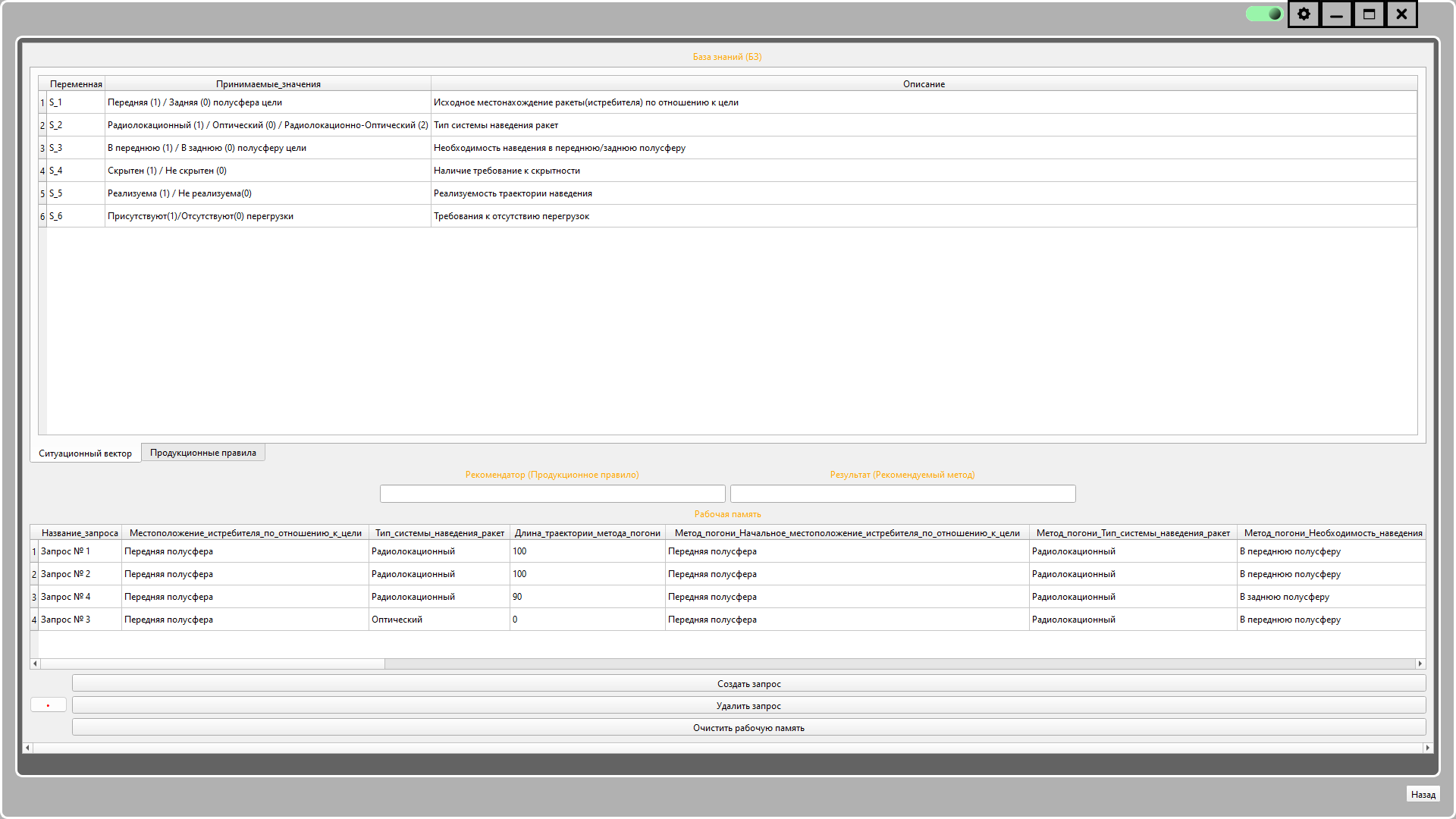
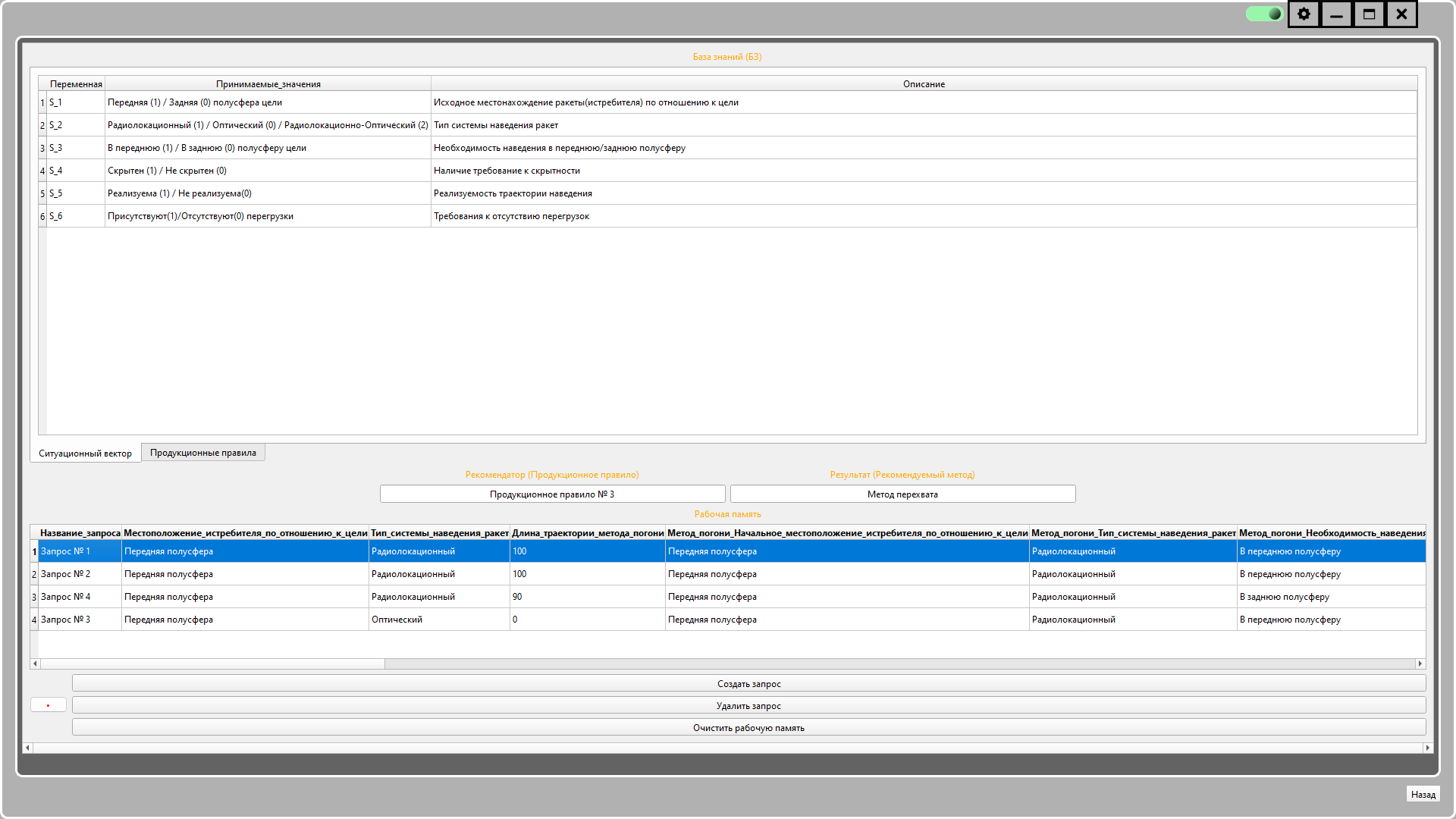


Рис. 22 – Демонстрация пользовательского интерфейса по работе с БЗ



Выберите строку запроса, чтобы получить результат

Результат

Рис. 23 – Демонстрация способа получения результата для конкретного запроса

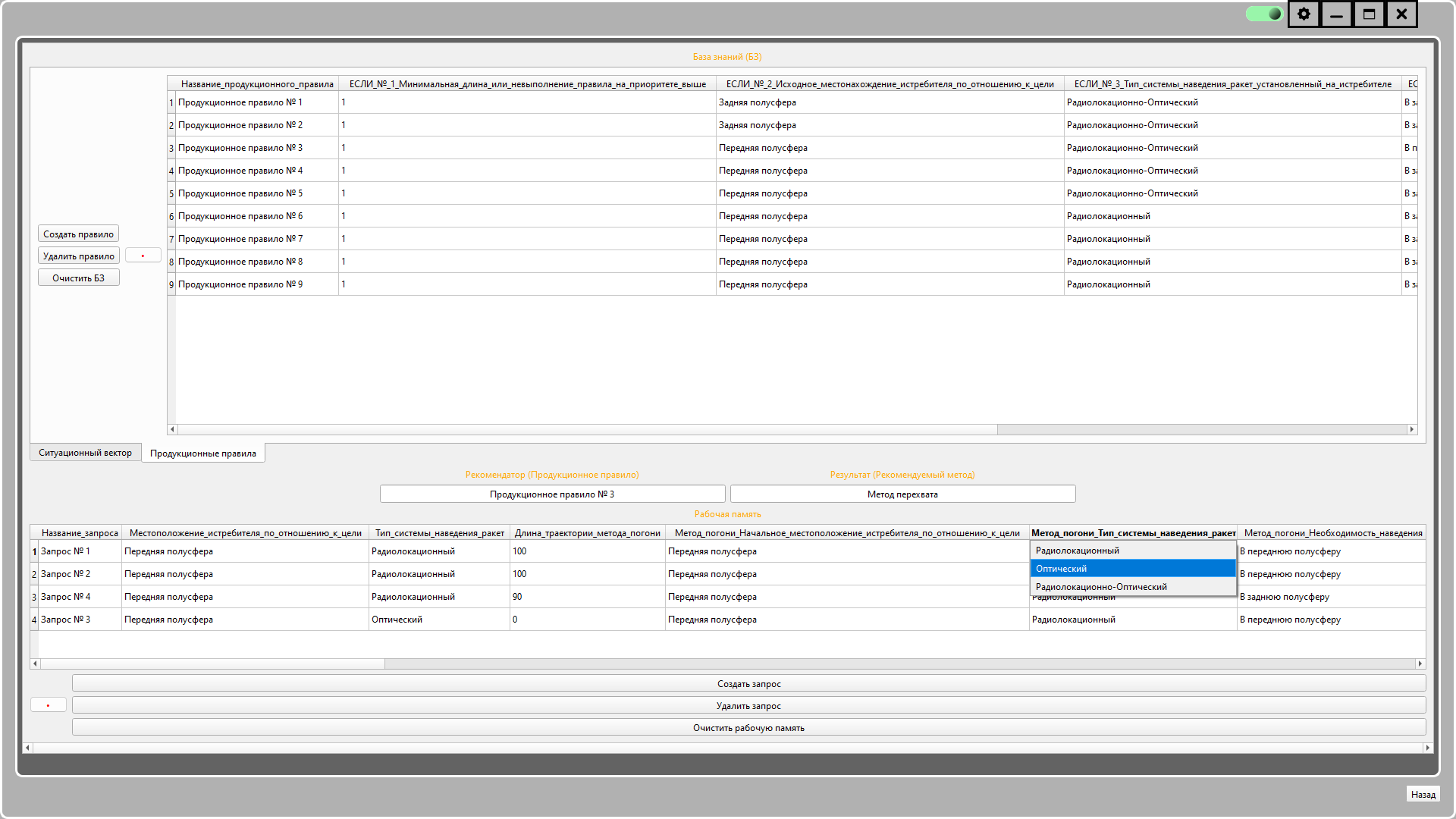


Рис. 24 – Демонстрация основного взаимодействия с запросами

Заметим также, что если оператор создает или удаляет строки в таблице продукционных правил или в таблице рабочего пространства, то он должен нажать на пиктограмму с красной точкой радом с соответствующей таблицей, чтобы изменения были сохранены в базе знаний.

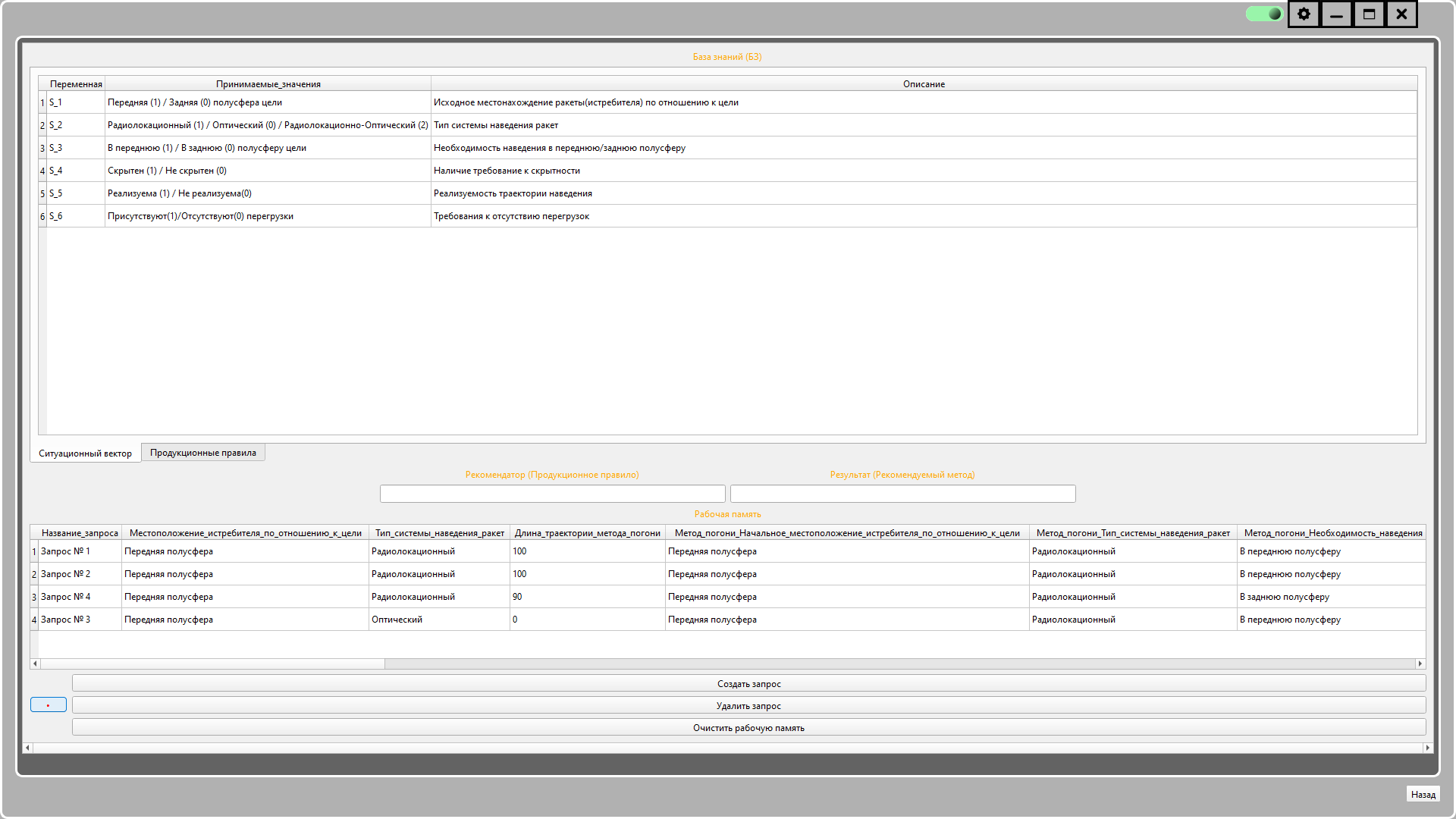


Рис. 25 – Демонстрация выделенной кнопки, внизу экрана (слева) для сохранения изменений, связанных с таблицей рабочей памяти

Отметим, что БЗ реализована таким образом, чтобы пользователь(оператор) мог в основном выбирать варианты значений из списка, предлагаемых для конкретного параметра. Исключение составляет длина траектории метода наведения. Её оператор должен задать вручную в соответствующее поле запроса. Аналогичная ситуация и с таблицей продукционных правил. Таблица ситуационного вектора предполагается неизменяемой.

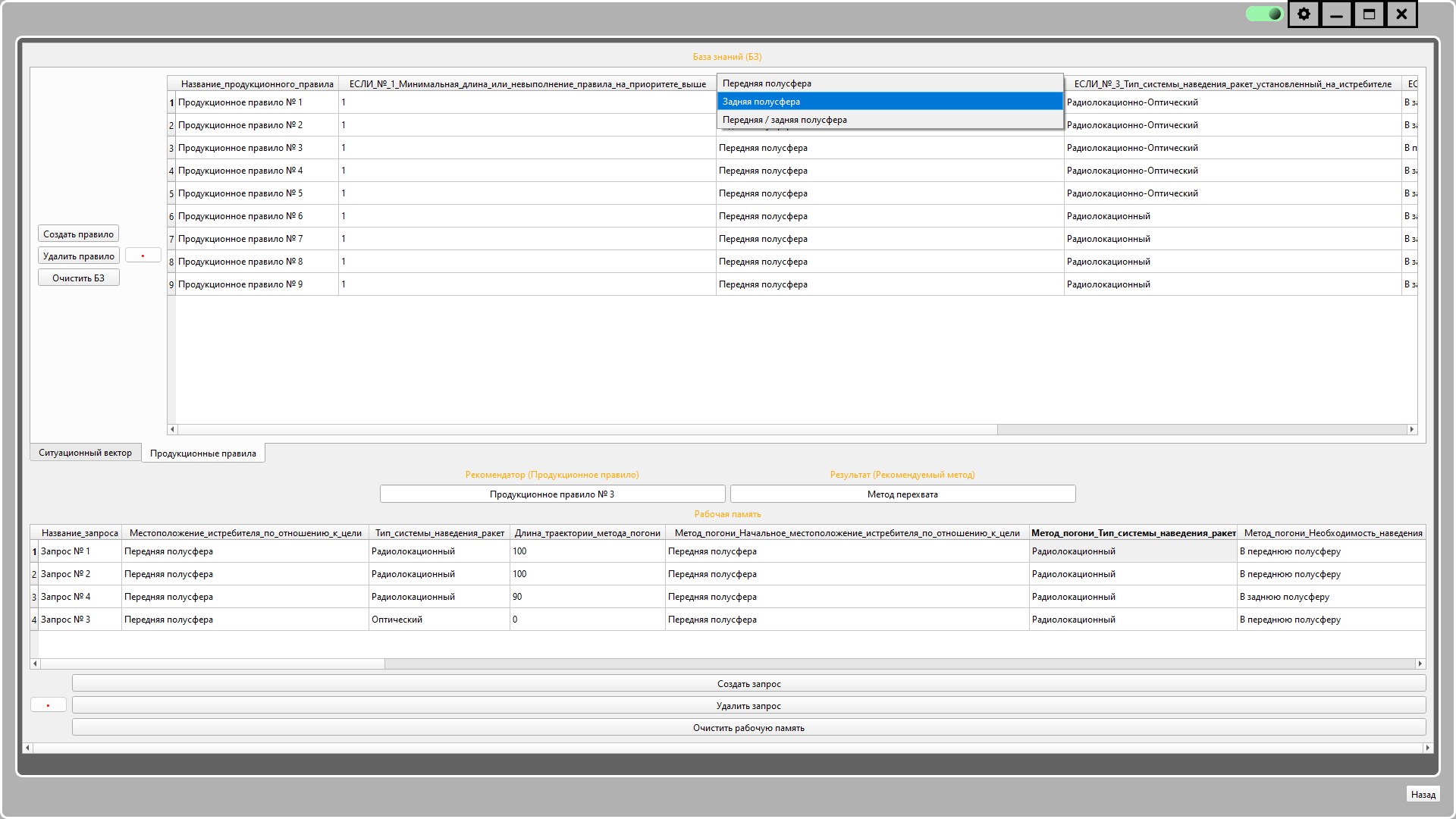


Рис. 26 – Демонстрация взаимодействия с таблицей продукционных правил

После того, как оператор поработал с программой необходимо отключить БЗ от программы. Для этого внизу окна имеется кнопка `назад` позволяющая выйти в основное окно программы и отключить безопасно БЗ. Если оператор просто завершит работу с программой не закрыв БЗ, это не должно привести к потере данных или порче БЗ или самого ПО, но рекомендуется закрывать соединение перед закрытием программы.

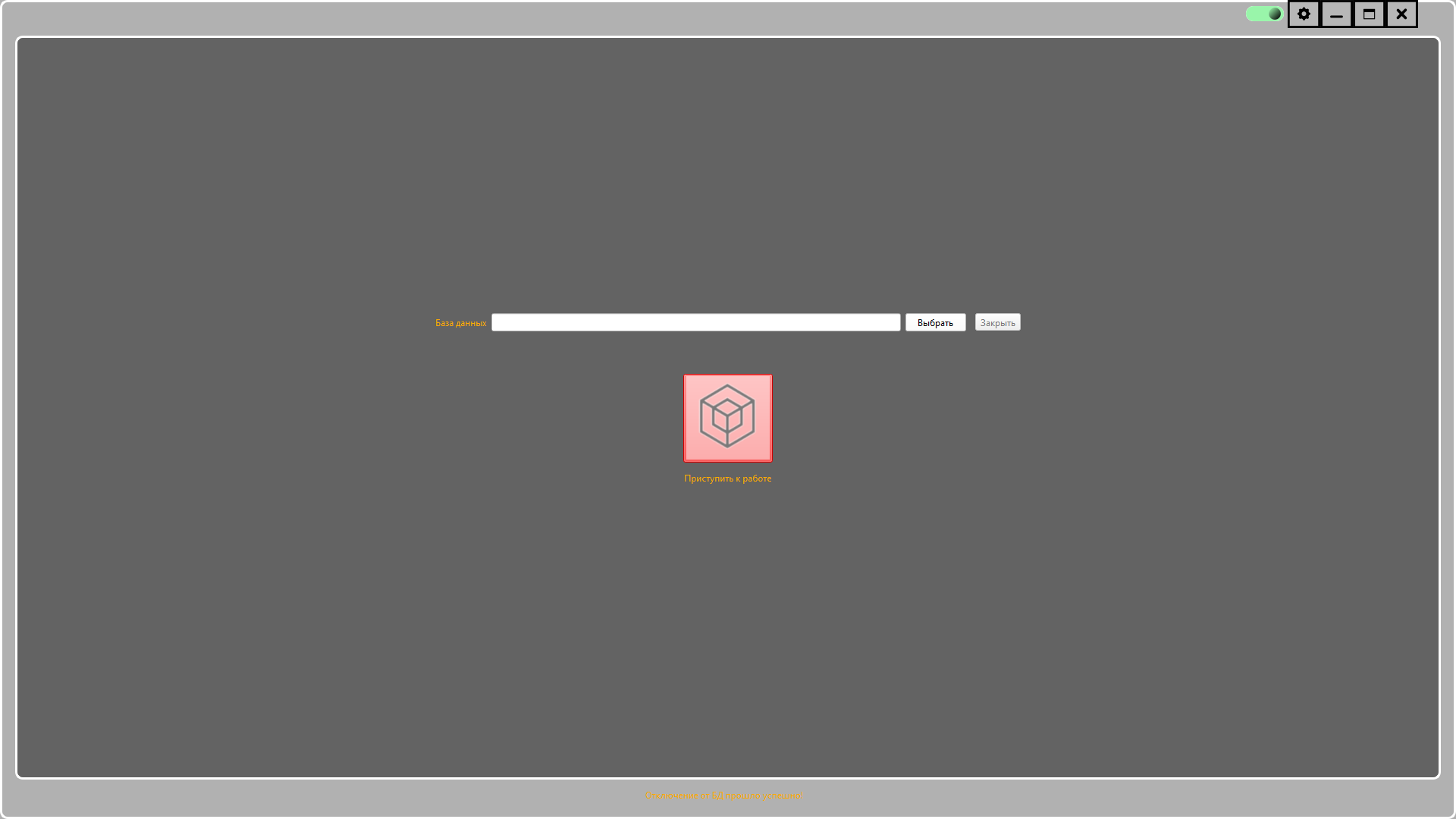


Рис. 27 – Демонстрация успешного отключения БЗ

## 5.3. Результат вывода

Таблица № 5 - Результаты

|  |  |  |
| --- | --- | --- |
| **Запрос** | **Рекомендатор** | **Рекомендуемое**  **(**Рекомендуемый метод**)** |
| Запрос № 1 | Продукционное правило № 3 | Метод перехвата |
| Запрос № 2 | Продукционное правило № 5 | Метод маневра |
| Запрос № 3 | Продукционное правило № 9 | Метод погони |
| Запрос № 4 |  | Установите корректную длину траектории для методов наведения! |