**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 Н.В. Коминцев

Группа КММО-01-23

**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.

Задачи:

* Изучить тему
* Проанализировать методы наведения и выделить ключевые факторы, влияющие на выбор метода
* Создать ситуационный вектор на основе выявленных факторов
* Разработать продукционные правила
* Определить структуру и алгоритм работы машины вывода
* Разработать машину ввода на основе продукционных правил
* Провести тестирование
* Сделать выводы

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

**2.1. Прямой метод**

2.1.1. Качественный смысл метода

Требуется всё время совмещать продольную ось истребителя с направлением на цель.



Рисунок 1 – пример использования прямого метода наведения

2.1.2. Достоинства и недостатки метода

Достоинства:

* Инвариантность к дальности наведения 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

Метод, как правило, применим:

− 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.

− при наведении на маневрирующие воздушные цели.

**2.2. Метод манёвра**

2.2.1. Качественный смысл метода

Метод обеспечивает вывод истребителя в зону обнаружения цели бортовой РЛС, ОЭС или оптическим прицелом под заданным углом Ψk к на заданном расстоянии.

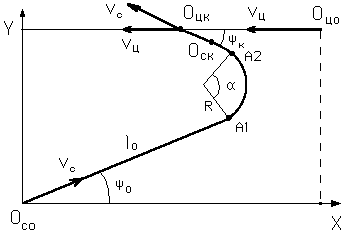


Рисунок 2 - пример использования метода манёвра

2.2.2. Достоинства и недостатки метода

Достоинства:

* Хорошее сопряжение со всеми методами самонаведения.
* Возможность использования как радиолокационных, так 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

* Наличие ограничений на ракурсы перехвата из передней полусферы;

**2.3. Метод перехвата**

2.3.1. Качественный смысл метода

Метод перехвата представляет собой разновидность метода параллельного сближения (рисунок 3). Особенностью является то, что по методу параллельного сближения наводится не сам истребитель, а некоторая фиктивная точка А, расположенная по направлению вектора скорости на расстоянии от самолёта – дальности захвата цели бортовой визирной системой.



Рисунок 3 – пример использования метода перехвата

2.3.2. Достоинства и недостатки метода

Достоинтва:

* 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.
* 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

• 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

|  |  |  |
| --- | --- | --- |
| Координаты | Смысл | Область значений |
| S1 | Текущее положение истребителя относительно цели | {Передняя (1); Задняя (0) полусфера} |
| S2 | Необходимость наведения в полусферу цели | {Передняя (1); Задняя (0) полусфера} |
| S3 | Тип установленный системы наведения на истребителе | {Радиолокационная (1); Оптическая (0)} |
| S4 | Требование к скрытности | {1;0} |
| S5 | Реализуемость траектории | {1;0} |
| S6 | Наличие перегрузки | {1;0} |

Так как возможны ситуации в котоорых реализуемы несколько подходящих методов, то принято решение добавить дополнительный параметр который не будет входить в ситуационный вектор (S\_доп) — длина траектории. Метод перехвата будет иметь наивысший приоритет [3], метод погони — [2], а метод манёвра — [1].

|  |  |  |
| --- | --- | --- |
| Sдоп | Длина траектории | {3;2;1} |

Если ни один из подходящих методов не реализуем, то выбор метода наведения невозможен.

3.2. Продукционные правила

Переменные, содержащие результат вывода:

* Решение – выбранный метод;

1.

**ЕСЛИ** Текущее положение истребителя относительно цели = 1

**И** Необходимость наведения в полусферу цели = 1

**И** (Тип установленный системы наведения на истребителе = 1

**ИЛИ** Тип установленный системы наведения на истребителе = 0)

**И** Требование к скрытности = 0

**И** Реализуемость траектории = 0

**И** Наличие перегрузки = 1

**ТО** Решение = Метод перехвата

2.

**ЕСЛИ** Текущее положение истребителя относительно цели = 0

**И** Необходимость наведения в полусферу цели = 0

**И** (Тип установленный системы наведения на истребителе = 1

**ИЛИ** Тип установленный системы наведения на истребителе = 0)

**И (**Требование к скрытности = 1 **ИЛИ** Требование к скрытности = 0)

**И** Реализуемость траектории = 0

**И** Наличие перегрузки = 1

**ТО** Решение = Метод перехвата

3.

**ЕСЛИ** Текущее положение истребителя относительно цели = 0

**И** Необходимость наведения в полусферу цели = 1

**ТО** Решение = Метод наведения невозможен.

4.

**ЕСЛИ** Текущее положение истребителя относительно цели = 1

**И** Необходимость наведения в полусферу цели = 1

**И** Требование к скрытности = 1

**ТО** Решение = Метод наведения невозможен

5.

**ЕСЛИ** Текущее положение истребителя относительно цели = 1

**И** Необходимость наведения в полусферу цели = 0

**И** Тип установленный системы наведения на истребителе = 1

**И** (Требование к скрытности = 1 **ИЛИ** Требование к скрытности = 0)

**И** (Реализуемость траектории = 1 **ИЛИ** Реализуемость траектории = 0)

**И** Наличие перегрузки = 0

**ТО** Решение = Метод погони

6.

**ЕСЛИ** Текущее положение истребителя относительно цели = 1

**И** Необходимость наведения в полусферу цели = 0

**И** Тип установленный системы наведения на истребителе = 0

**И** (Требование к скрытности = 1 **ИЛИ** Требование к скрытности = 0)

**И** Реализуемость траектории = 0

**И** Наличие перегрузки = 0

**ТО** Решение = Метод манёвра

7.

**ЕСЛИ** Текущее положение истребителя относительно цели = 1

**И** Необходимость наведения в полусферу цели = 0

**И** Тип установленный системы наведения на истребителе = 1

**И** (Требование к скрытности = 1 **ИЛИ** Требование к скрытности = 0)

**И** Реализуемость траектории = 0

**И** Наличие перегрузки = 0

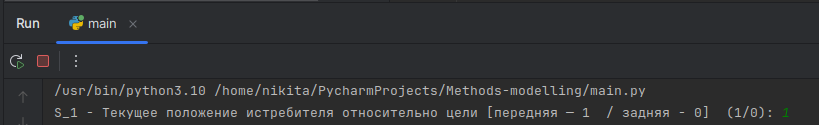
**ТО** Решение = Метод погони\*

\* на основании предпочтительности метода с наименьшей длиной пути.

**4. Разработка машины вывода**

4.1. Входные данные

Входные данные для программы представляют собой ситуационный вектор, который запрашивается у пользователя для каждого измерения, определенного в таблице правил. Каждое измерение имеет определенные варианты значений, которые пользователь должен выбирать. Программа принимает значения вектора ситуации от пользователя в формате, соответствующему указанным в таблице правил вариантам значений для каждого измерения.

Рисунок 4 - Входные данные

4.2. Выходные данные

Выходные данные программы представляют собой введённый ранее ситуационный вектор и решение, принятое на основе входных данных и правил, загруженных из файла.



Рисунок 5 - Выходные данные

4.3. Структура машины вывода

Структура программы:

1. Загрузка таблицы правил:

* Функция **load\_rules\_table(file\_path)** отвечает за загрузку таблицы правил из файла Excel.
* В случае успешной загрузки возвращает таблицу правил, иначе возвращает None.

2. Интерпретация ситуации:

* Функция i**nterpret\_situation(situation\_vector, rules\_df)** принимает вектор ситуации и таблицу правил.
* Проходит по каждой строке таблицы и сравнивает условия с введенным вектором ситуации.
* Если находится соответствие, возвращает рекомендованное решение.

3. Главная функция:

* Функция **main()** является точкой входа в программу.
* Загружает таблицу правил и запрашивает у пользователя ситуационный .
* Для каждой переменной, описанной в таблице правил, запрашивает значение и формирует вектор ситуации.
* После получения вектора вызывает функцию interpret\_situation() и выводит результат.

4.4. Алгоритм работы машины вывода

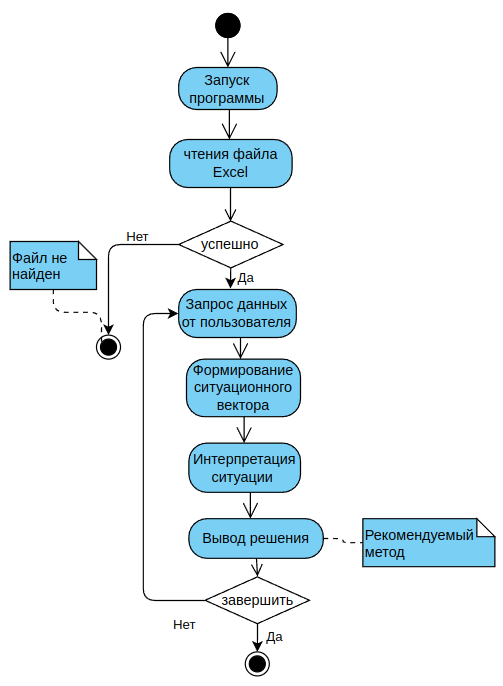


Рисунок 6 - Алгоритм программы

4.5. Исходный код

https://github.com/Nikita-Komintsev/Methods-modelling

**5. Описание эксперимента**

1. Исходные данные эксперимента.

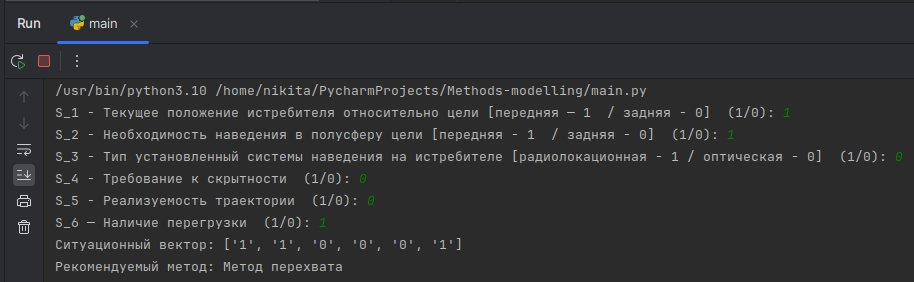
Excel таблица:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Метод перехвата | | Метод погони | Метод манёвра |
| S\_1 - Текущее положение истребителя относительно цели [передняя — 1 / задняя - 0] (1/0) | 1 | 0 | 1 | 1 |
| S\_2 - Необходимость наведения в полусферу цели [передняя - 1 / задняя - 0] (1/0) | 1 | 0 | 0 | 0 |
| S\_3 - Тип установленный системы наведения на истребителе [радиолокационная - 1 / оптическая - 0] (1/0) | 1 / 0 | 1 / 0 | 1 | 1 / 0 |
| S\_4 - Требование к скрытности (1/0) | 0 | 1 / 0 | 1 / 0 | 1 / 0 |
| S\_5 - Реализуемость траектории (1/0) | 0 | 0 | 1 / 0 | 0 |
| S\_6 — Наличие перегрузки (1/0) | 1 | 1 | 0 | 0 |
| S\_доп - Длина траектории (3/2/1) | 3 | 3 | 2 | 1 |

Эксперимент №1

Ситуационный вектор:['1', '1', '0', '0', '0', '1']

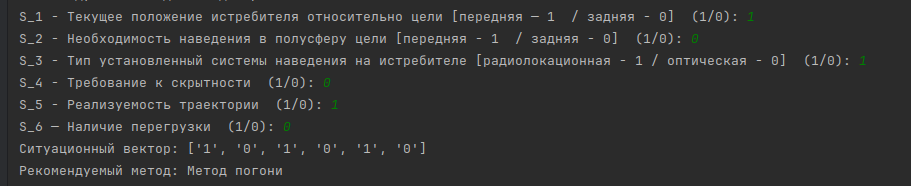
Результат вывода:

Рисунок 7 - Результат эксперимента №1

Эксперимент №2

Ситуационный вектор: ['1', '0', '1', '0', '1', '0']

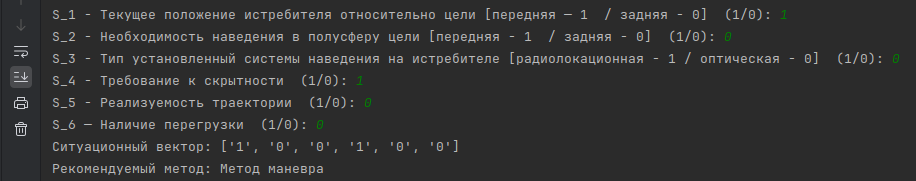
Результат вывода:

Рисунок 8 - Результат эксперимента №2

Эксперимент №3

Ситуационный вектор: ['1', '0', '0', '1', '0', '0']

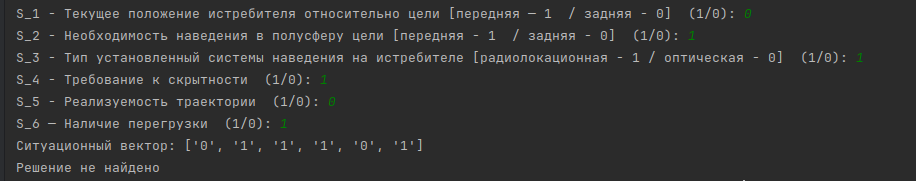
Результат вывода:

Рисунок 9 - Результат эксперимента №3

Эксперимент №4

Ситуационный вектор: ['0', '1', '1', '1', '0', '1']

Результат вывода:

Рисунок 10 - Результат эксперимента №4