

Лабораторная работа №5: Исследование моделей планирования действий в системах искусственного интеллекта

Цель работы

Изучение принципов описания и решения задачи планирования на основе продукционной модели с использованием механизма поиска в пространстве состояний.

Основные теоретические положения

Задача построения башни из блоков (кубиков) заключается в последовательном выборе из неупорядоченной кучи блоков и постановки их друг на друга. План решения задачи представляет собой чередование двух фаз: выбора блока из кучи и установки его в башню. При этом больший по размерам блок не может ставиться на меньший по размерам и, следовательно, на каждом шаге решения задачи необходимо выбирать из кучи самый большой блок.

Каждый блок характеризуется *размером, цветом и положением*, поэтому для его описания можно использовать следующий шаблон:

```
(deftemplate block
  (slot size (type INTEGER))
  (slot place (type SYMBOL))
  (slot color (type SYMBOL))
)
```

Каждый блок может находиться либо в куче (*heap*), либо в руке (*hand*), когда он выбран из кучи, либо в башне (*tower*).

В результате решения задачи необходимо последовательно вывести информацию о действиях, составляющих план её решения и взаимное положение блоков в построенной по данному плану башне с указанием их цвета. С этой целью для блоков, находящихся в башне, необходимо определить отношение, указывающее, какой блок (*upper*) на каком (*lower*) стоит:

```
(deftemplate on
  (slot upper (type SYMBOL))
  (slot lower (type SYMBOL))
  (slot place (type SYMBOL)(default heap))
)
```

Если блок ставится в основание башни, слоту *place* присваивается значение *tower*, а слоты *upper* и *lower* остаются неопределенными. Если блок ставится на другой блок, слотам *upper* и *lower* должны присваиваться значения цветов соответствующих блоков.

Поскольку план решения задачи представляет собой чередование фаз поиска в компоненте пространства состояний («куче») самого большого блока (`find`) и его установки в башню (`build`), необходимо иметь факт, определяющий текущую задачу. Его можно определить с помощью следующего шаблона:

```
(deftemplate goal
  (slot current-task (type SYMBOL))
)
```

Для решения задачи необходимы правила для следующих действий:

- установки начального значения текущей задачи;
- выбора из кучи самого большого блока;
- установки первого блока в основание башни;
- установки в башню всех последующих блоков;
- определения окончания процесса, когда в куче нет больше блоков.

Правило установки начального значения текущей задачи должно активироваться исходным фактом (`initial-fact`) и устанавливаться в качестве текущей задачи `find`.

Правило выбора из кучи самого большого блока должно активироваться следующими условиями:

- текущая задача - `find`;
- наличие в куче блока, для которого не существует большего по размерам.

Действия правила:

- взять найденный блок в руку (изменить его местоположение на `hand`);
- изменить текущую задачу на `build`.

Правило установки блока в основание башни (первого блока) имеет следующие условия активации:

- текущая задача - `build`;
- имеется блок в руке;
- отсутствуют блоки в башне - нет блока, у которого значение слота `place` равно `tower`.

Действия правила:

- изменить местоположение блока в руке на `tower`;
- изменить текущую задачу на `find`.

Правило установки в башню последующих блоков имеет условия активации:

- текущая задача - `build`;
- имеется блок в руке (для него надо запомнить цвет);
- имеется блок в башне, на котором не стоит другой блок (для него также надо запомнить цвет).

Действия правила:

- модифицировать местоположение блока в руке на `tower`;

- установить факт, что новый блок находится на блоке, который был до этого верхним;
- изменить текущую задачу на find.

Правило определения окончания процесса должно активироваться, когда в куче больше нет блоков и удалять из рабочей памяти факт текущей задачи.

Постановка задачи

Разработать и отладить на языке CLIPS программу решения задачи планирования постройки башни из блоков с пошаговым отображением плана построения башни.

Порядок выполнения работы

1. Разработать программу решения задачи планирования на языке CLIPS, реализующую описанные выше правила, в соответствии с вариантом задания, указанным в таблице. Исходное состояние должно описываться конструкцией `deffacts`, содержащей четыре факта, соответствующих блокам в куче.
2. Вывод результатов на экран должен обеспечивать пошаговое отображение плана построения башни.

Варианты заданий

Для каждого блока в таблице через слеш указаны параметры: цвет/размер.

№ варианта	Блок-1	Блок-2	Блок-3	Блок-4
1	синий/10	зелёный/8	красный/18	жёлтый/15
2	красный/20	коричневый/5	жёлтый/12	зелёный/8
3	жёлтый/15	красный/10	белый/25	коричневый/5
4	зелёный/10	жёлтый/14	чёрный/18	красный/11
5	коричневый/5	белый/25	синий/14	жёлтый/12
6	красный/10	чёрный/18	жёлтый/15	белый/25
7	жёлтый/12	синий/10	зелёный/8	чёрный/18
8	белый/16	красный/5	коричневый/15	синий/10
9	чёрный/18	жёлтый/15	красный/20	белый/10

Содержание отчёта

- Цель работы.
- Краткое изложение основных теоретических понятий.
- Постановка задачи с кратким описанием порядка выполнения работы.
- Пошаговый план решения задачи.
- Результаты работы программы с краткими выводами.

- Общий вывод по проделанной работе.
- Код программы.

From:

<https://se.moevm.info/> - **МОЭВМ Вики** [se.moevm.info]

Permanent link:

https://se.moevm.info/doku.php/courses:knowledge_representation_and_artificial_intelligence_systems:lab5

Last update:

