

Алгоритм принятия решений в интерактивной модели адаптивной системы

А. М. Евтухов

Введение

В данной работе рассматриваются модели систем, состоящих из большого количества объектов, которых мы будем называть роботами из-за схожих с реальными роботами функций. Интерактивная модель позволяет исследователю вводить корректировки непосредственно во время проведения эксперимента. Способность системы к адаптации означает, что отдельные компоненты системы и их объединения будут стремиться к достижению поставленных перед ними задач на основе своего предыдущего опыта [1]. Предлагаемая модель системы может применяться, как правило, для имитации разумной деятельности большой группы людей с целью нахождения наиболее эффективных вариантов их действий. Задачей, поставленной в данном исследовании, является нахождение алгоритма, способного управлять целенаправленными перемещениями и действиями роботов в условиях меняющейся во времени и пространстве внешней среды.

Описание модели

В модели могут одновременно сосуществовать несколько сторон S , каждая из которых преследует свои собственные интересы. Обозначим как $\tilde{S} = \{S\}$ множество всех независимых сторон в модели. В свою очередь, оно делится на два непересекающихся множества R и \bar{R} , такие как $\tilde{S} = R \cup \bar{R}$, $R \cap \bar{R} = \emptyset$, где R – множество таких сторон S , что для них существует цель, которая должна быть достигнута этой стороной. Назовём множество R *целенаправленным множеством*. Для множества \bar{R} таких целей не существует, и назовём \bar{R} *множеством природных сил*. Рассматриваемые в данной работе алгоритмы есть только у сторон из множества R (по одному у каждой из сторон), причём они функционируют независимо друг от друга. Каждой стороне S принадлежит также множество роботов $S.\tilde{r} = \{r\}$, которые должны действовать согласованно. Множество $\{r\}$ назовём *группой роботов*. Введём *множество действий* $A = \{A\}$. Каждый робот r обладает атрибутом $r.\tilde{a} = \{a\}$, содержащим то множество действий $\{a\} \subset \tilde{A}$, которое он может совершить здесь и в данное время. Вид этого множества зависит от текущего состояния робота $r.\tilde{s}$, представляющего собой вектор числовых значений. Результатом применения действия является изменение состояния \tilde{s} у субъекта и объекта действия, а также, в некоторых случаях, и координат \tilde{x} .

Все роботы в модели обладают пространственными координатами $r.\tilde{x}$, при этом сами роботы находятся внутри какого-то внешнего окружения. Непосредственное окружение робота оказывает влияние на его состояние, а, следовательно, и на множество $r.\tilde{a}$.

Всё пространство, в котором будет происходить моделирование, мы разбиваем на отдельные небольшие элементы – клетки, кубики, либо аналогичные им объекты, множество которых обозначим $\tilde{c} = \{c\}$. Основным требованием к каждой клетке c является то, что состояние робота внутри неё должно меняться настолько мало, что этими изменениями можно было бы пренебречь. Это означает, что состояние $c.\tilde{s}$ клетки c полностью определяет те параметры робота, которые зависят от окружающей среды. Сами размеры клетки должны быть достаточно малы для того, чтобы пренебречь в большинстве случаев локальными координатами робота внутри неё. Локальные координаты робота внутри каждой клетки могут как учитываться, так и игнорироваться. Поскольку длительность вычислений в модели прямо

пропорциональна количеству клеток в ней, то при моделировании придётся учесть обратную зависимость между точностью и скоростью расчётов при уменьшении размера клеток.

Использование клеток позволяет описывать протяжённые объекты при подаче команд. Так, например, приказ о перемещении робота в комнату описывается как приказ о перемещении в любую из клеток с номерами #4, #5, #6, #84, #85, #86 – клеток, которые комната занимает внутри модели.

Аналогично роботам к каждой клетке можно применить множество действий $s.\tilde{a} \subset \tilde{A}$, состав которого зависит от $s.\bar{s}$. Действия к клеткам применяют, как правило, роботы. Характерным примером такого действия является перемещение робота в заданную клетку: такое действие меняет атрибуты как самого робота (меняются его координаты), так и клеток, через которые он движется (множество расположенных в них роботов). Другим примером действия является «закрыть дверь» – в этом случае атрибуты робота, который эту дверь закрыл, могут и не измениться, а изменяются только атрибуты клетки, которая имитирует дверной проём. Примером действий, которые не связаны с роботами, являются природные воздействия: например, таяние и разламывание льда – в этом случае соответствующие клетки меняют своё состояние со «льда» на «разлом» или «водную поверхность».

Каждое действие, которое может быть применено к роботу либо клетке, требует выполнения каких-либо условий. Способ достижения этих условий и должна найти система принятия решений модели, индивидуальная версия которой есть у каждой группы роботов, действующих совместно.

Описание алгоритма

Работа системы является циклической, при этом в каждом цикле она последовательно проходит следующие этапы:

1. Оценка ситуации.
2. Построение графа шаблонов.
3. Распределение ресурсов между шаблонами.
4. Выполнение текущей стадии всех активных шаблонов.
5. Перерасчёт эвристических параметров модели.

Рассмотрим эти этапы более подробно.

Оценка ситуации. На данном этапе компьютер должен перебрать множества $\{r\}$ и $\{c\}$ и вычислить эвристическую оценку $e = Ef(w, a)$ применения каждого возможного действия $a \in r.\tilde{a} \cup c.\tilde{a}$ к этим объектам [2]. Эвристическая оценка e в нашем случае – это число, и чем оно больше, тем цель приоритетнее. Целью g мы будем называть пару объект-действие: $g = (w, a)$, где $w \in \{r\} \cup \{c\}$. Некоторые цели могут противоречить друг другу, но это не мешает попытаться достичь их одновременно, отказавшись от одной из них лишь в самый последний момент. Для ускорения вычислений часть объектов w может отбрасываться до анализа всех возможных действий.

На этапе оценки ситуации *стратегией-кандидатом* называется совокупность $F' = (w, a, e)$ для каждого из существующих объектов и возможных действий. После расчёта оценки приоритета (эвристической оценки) стратегии-кандидата производится перебор всех стратегий из стратегического списка $\tilde{F} = \{F\}$, и если новая стратегия достаточно приоритетна по сравнению с находящимися в списке, то она либо добавляется туда, либо заменяет существующую стратегию. Результатом этапа оценки ситуации является формирование либо корректировка стратегического списка.

Все стратегии, принятые к исполнению, объединяются в единый *стратегический список* \tilde{F} . Каждая стратегия F в этом списке отличается от стратегии-кандидата наличием дополнительного атрибута – плана P достижения поставленной цели: $F = (F', P)$. План представляет собой условный граф *шаблонов* p – это те элементарные действия, которые могут совершаться в модели. Примерами шаблонов являются: «взять указанный предмет», «переместиться по прямой в указанную точку».

Построение графа шаблонов. Как правило, для каждой стратегии-кандидата соответствующий граф шаблонов можно построить по заранее известному алгоритму

$F.P = Plan(F.g)$, и переменными величинами при этом являются вероятности выбора того или иного поддерева в точках ветвления графа. Расчётом наиболее оптимальных значений этих вероятностей непосредственно в ходе моделирования достигается эффект адаптируемости роботов к окружающей среде.

Каждый шаблон требует для своего выполнения наличия каких-то ограниченных в количестве ресурсов. Причём эти ресурсы не обязательно должны быть реальными объектами в модели. Например, предположим, что военный корабль имеет только две орудийные башни. Тогда для выполнения шаблона выстрела по мишени задействуется ресурс «одна башня». Таким образом, в этом случае не может выполняться одновременно более двух шаблонов стрельбы, поскольку в совокупности они задействуют все ресурсы орудийных башен. Здесь нужно отметить, что один и тот же робот может одновременно реализовывать несколько стратегий, если они не исключают друг друга. Точно так же каждая стратегия может использовать одновременно нескольких роботов, которые для неё представляются всего лишь ресурсами.

Распределение ресурсов между шаблонами производится с учётом текущих приоритетов стратегий. При этом стратегии с большим приоритетом могут забирать ресурсы у стратегий с меньшим приоритетом. Однако можно стремиться и к такому распределению ресурсов, при котором одновременно выполняется максимальное количество стратегий.

На каждой итерации, как правило, успевают выполниться лишь часть текущего шаблона каждой стратегии. Поэтому этот этап называется *этапом выполнения текущей стадии всех активных шаблонов*. После него производится удаление из стратегий тех шаблонов, у которых достигнуто заключительное состояние и удаление тех стратегий, цель которых либо достигнута, либо провалена.

Участие человека в рассматриваемой модели заключается во введении в стратегический список *пользовательских стратегий-кандидатов* (такие стратегии не могут быть удалены из списка, пока существует хоть какая-то возможность достижения их конечной цели), а также наложение запрета на помещение в него стратегий заданной разновидности.

Перерасчёт всех эвристических параметров модели производится на заключительном этапе работы цикла модели. Здесь производится оценка того, насколько:

- один участок графа шаблонов оказался эффективнее своей альтернативы при схожих условиях функционирования модели;
- реализованная стратегия позволила приблизить конечную цель;
- реализация стратегии сказалась на доступном количестве ресурсов;
- оказались прерваны другие стратегии.

Заключение

Как следует из вышеизложенного, разработанный алгоритм очень чувствителен к количеству клеток, роботов и действий, которых к ним можно применить. Таким образом, использовать его следует лишь для сравнительно небольших систем. Однако эти ограничения компенсируются достаточно неплохой степенью приближённости к реальным условиям, которая может быть достигнута в ходе моделирования.

Abstract. The decision-making system in an interactive model of a complex adaptable system is offered in this article. The given system allows to simulate a reasonable activity in certain limits.

Литература

1. Ю. М. Козлов, *Адаптация и обучение в робототехнике*, Москва, Наука, Гл. ред. физ.-мат. лит., 1990.
2. Дж. Ф. Люггер, *Искусственный интеллект: стратегии и методы решения сложных проблем*, 4-е издание, Пер. с англ., Москва, Издательский дом "Вильямс", 2003.