Графический метод приближенного описания динамически изменяющихся множеств в задачах поиска

Сергей Ежков

Московский Государственный Университет им. М.В.Ломоносова

Москва, Россия

Аннотация

Предложен простой и в то же время эффективный метод представления произвольных динамически изменяющихся множеств, возникающих в процессе решения поисковых задач.

Идея метода заключается в разбиении поисковой области *G* семейством кривых *C* и использовании для приближенного описания любого подмножества *A* поисковой области оболочки, порожденной множеством сегментов кривых из *C*, лежащих внутри *A*. Такой подход позволяет визуализировать информацию о форме и положении запретного множества при поиске как в двумерных, так и в трехмерных областях. При этом в последнем случае решается не трехмерная, а одна или несколько более простых двумерных задач, что дает выигрыш в производительности на порядок.

В качестве примеров, иллюстрирующих работу метода в трехмерном случае, рассмотрены задачи поиска в цилиндре и торе.

Ключевые слова: *задача поиска, динамически изменяющееся множество, приближенное представление переменных множеств.*

1. ВВЕДЕНИЕ

В настоящее время трудно найти область человеческой деятельности, в которой в той или иной степени не использовалась бы вычислительная техника. При проведении сложных вычислений, моделировании, построении прогнозов, хранении и обработке информации и т.п. незаменимым помощником человека становится компьютер.

Стало обычным использование компьютера в научных исследованиях. Причем, если раньше здесь оказывались востребованными лишь его вычислительные возможности, то сейчас все чаще находит применение и другой инструмент – компьютерная графика.

Средства современной машинной графики позволяют создавать на экране компьютера объемные реалистические изображения, причем не только статичные, но и изменяющиеся. Это дает возможность разрабатывать качественно новые алгоритмы решения задач, связанных с исследованием динамически изменяющихся объектов. К такому классу относятся и задачи поиска [1].

2. ПОСТАНОВКА ЗАДАЧИ ПОИСКА

В рассматриваемой ниже задаче поиска принимают участие два объекта – *ищущий* и *уклоняющийся*. Каждый объект может перемещаться с ограниченной скоростью в пределах заданного множества *G (поисковой области)*, которое представляет собой подмножество евклидова пространства.

Каждый из участвующих в задаче поиска объектов имеет свои цели и возможности. Так, уклоняющийся объект, как явствует уже из самого названия, стремится избежать встречи с ищущим и остаться необнаруженным. Цель ищущего объекта состоит в обнаружении уклоняющегося. При этом уклоняющийся считается обнаруженным, если в некоторый момент времени расстояние между ним и ищущим объектом оказывается не больше некоторого заданного числа $l>0$ *(радиуса обнаружения).* В связи с этим удобно считать, что ищущий объект как бы несет на себе *круг контроля* радиуса l (в трехмерном случае будем называть круг контроля *шаром контроля*).

Относительно скоростей и характера движения участников поискового процесса делаются следующие предположения:

- ищущий объект перемещается с постоянной скалярной скоростью $\alpha>0$ и в любой момент времени может произвольно менять направление своего движения;
- уклоняющийся объект перемещается со скалярной скоростью β, причем выполняется соотношение $\beta<\alpha$. Как и ищущий, уклоняющийся объект способен произвольно изменять направление своего движения в любой момент времени.

На ход поискового процесса существенное влияние оказывает уровень информированности его участников. В общем случае она неодинакова. Далее будем считать, что поисковая область *G*, скорости α и β и местоположение ищущего объекта в каждый момент времени, а также условия обнаружения известны всем участникам. Ищущий не имеет информации о местоположении уклоняющегося в каждый момент времени. Таким образом, возникает понятие *информационной дискриминации* объектов игры, характеризующее различие в уровне их осведомленности о ситуации на момент выбора собственной стратегии.

2.1 Множества, возникающие в процессе решения задачи

Уклоняющийся объект, избегая ищущего, волен выбирать, вообще говоря, любую траекторию движения. Однако, в силу ограниченности скорости он не способен оказаться мгновенно там, откуда только что ушел ищущий. Таким образом, в каждый момент времени возникает *остаточное множество* [2,3], в которое уклоняющийся объект на этот момент времени не успевает попасть.

Объединение круга контроля и остаточного множества назовем *запретным множеством*, поскольку в нем гарантировано отсутствие уклоняющегося.

Остальная часть поисковой области называется *допустимым множеством* – это множество, в котором может находится необнаруженный уклоняющийся объект.

Естественно, что положение в поисковой области и форма запретного и допустимого множеств существенно зависят от времени.

Ищущий не осведомлен о поведении уклоняющегося. Поэтому цель его состоит в том, чтобы запретное множество, нарастая, в конечном итоге совпало со всем поисковым множеством. Это гарантирует успешное (с точки зрения ищущего) завершение поиска.

3. ПРИМЕНЕНИЕ КОМПЬЮТЕРНОЙ ГРАФИКИ К ИССЛЕДОВАНИЮ ПОИСКОВЫХ ЗАДАЧ

При решении задач поиска важнейшую роль играет вопрос выбора траектории ищущего объекта. При этом в первую очередь траектория строится на качественном уровне, т.е. выделяется класс кривых, в котором ищется решение поставленной задачи. После нахождения такого класса можно приступать к определению конкретных числовых параметров искомой траектории [3,4].

В некоторых ситуациях – когда поисковое множество устроено достаточно просто – выбор траектории очевиден; в более сложных случаях увидеть искомую траекторию с помощью лишь воображения практически невозможно, так как принципиальную важность приобретает детальное изучение того, как эволюционируют запретное и допустимое множества с течением времени.

Компьютерная графика становится здесь необходимым инструментом, дополняющим и расширяющим возможности нашего воображения и обычных изобразительных средств за счет введения дополнительного измерения – времени, и позволяющим тем самым изучать процесс поиска в его динамике.

Ниже мы остановимся на описании предлагаемого метода получения динамически изменяющегося изображения всех множеств, вовлечённых в решение задачи поиска, что дает возможность оценить пригодность выбираемого класса траекторий, увидеть и исследовать его особенности.

3.1 Возможные подходы к решению

Графическое решение задачи поиска сводится к нахождению эффективного метода представления сложных (фактически произвольных) меняющихся во времени множеств и алгоритмов работы с ними. Все подобные методы могут быть разделены на две группы:

- *Аналитические методы,* основанные на аналитическом описании границ искомых множеств [5,6]. Применительно к задаче поиска суть аналитического метода заключается в том, что делается попытка описать рассматриваемое множество с помощью набора формул (равенств или неравенств), определяющих принадлежность точки в пространстве к внутренности или границе искомой области. Аналитический метод – единственный, обеспечивающий абсолютную точность при описании рассматриваемых множеств. К самым существенным его недостаткам относятся, во-первых, трудность практической реализации в большинстве случаев и, во-вторых, малопригодность с точки зрения компьютерной графики. В задачах поиска последнее вызвано чрезвычайной сложностью множеств, с которыми приходится иметь дело в процессе решения. Например, граничные эффекты (изменение формы следящей области под влиянием близости границы поискового множества) практически невозможно учесть аналитически.
- *Методы приближенного описания множеств.* Недостатки аналитических методов приводят к необходимости искать другие пути решения задачи. Например, можно попытаться описать интересующие нас множества не точно, а приближенно. Естественно, в отличие от аналитического, приближенное описание не гарантирует абсолютной точности получаемого решения; тем не менее приближенные методы

позволяют достичь хороших результатов при значительно меньших затратах (например, вычислительных).

Основная проблема, неизбежно возникающая при решении задачи визуализации поискового процесса, заключается в выборе метода, позволяющего удобно и эффективно описывать динамически изменяющиеся множества. Далее будем говорить в основном о запретном множестве – с одной стороны как о множестве, представляющем наибольший интерес при исследовании поисковых задач, с другой – как о множестве, на примере которого можно увидеть все особенности данного класса областей: невозможность эффективного применения аналитических методов для их описания и способность динамически изменять свою форму.

3.1.1 Требования к методу приближенного описания множеств переменной структуры для задачи поиска

Прежде чем переходить к обсуждению предлагаемого метода описания сложных множеств переменной структуры, необходимо сформулировать набор критериев, которым он должен удовлетворять. С учетом сказанного выше можно выделить следующие требования.

Высокая эффективность. Именно эффективность оправдывает замену аналитических методов методом приближенного описания.

Ориентация на компьютерографическую реализацию. Метод должен быть хорошо формализован, что позволило бы построить его программную реализацию.

Метод должен строить решение задачи с любой, наперед заданной *степенью точности*.

4. МЕТОД ПРИБЛИЖЕННОГО ПРЕДСТАВЛЕНИЯ ПЕРЕМЕННЫХ МНОЖЕСТВ ДЛЯ ЗАДАЧИ ПОИСКА

4.1 Идея метода

Основной идеей предлагаемого метода [7] является использование в качестве приближенного описания любого множества $A \subset G$ оболочки, порожденной множеством сегментов кривых из C, лежащих внутри A, где C – семейство кривых, равномерно «заполняющих» поисковую область. В общем случае построенная оболочка может быть невыпуклой, иметь внутренние полости или не быть связной (рис.1).

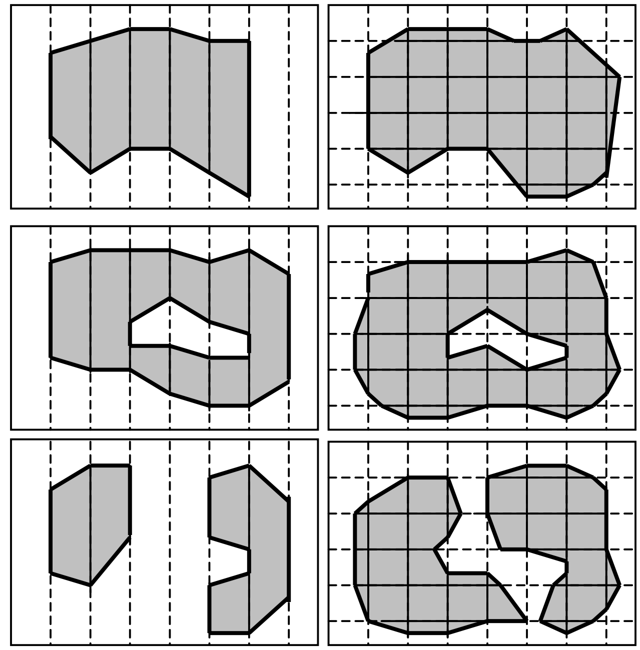

Рис. 1. Построение оболочки, порожденной множеством сегментов кривых в двумерном случае. Вертикальные и горизонтальные прямые образуют множество C. Сплошные тонкие линии – сегменты кривых, лежащие внутри описываемого множества.

Что касается рассматриваемой задачи поиска, то при ее решении в каждый момент времени нас будет интересовать описание запретного множества Ω_t. Для его построения мы будем пользоваться трехэтапной процедурой. Предположим, что нам известно описание множества Ω_t в момент времени t (в начальный момент времени это круг контроля, приближенное описание которого мы построим при помощи семейства кривых C). Тогда в момент времени $t+\Delta t$ последовательно проделаем следующие действия.

1. Выделим из множества Ω_t подобласть, в которой в момент времени $t+\Delta t$ уклоняющийся не может оказаться в силу ограниченности скорости перемещения. Для этого уменьшим сегменты кривых из C, которые лежали внутри Ω_t, на $\beta\Delta t$ с обоих концов («выедание» уклоняющимся объектом). В результате получим множество $C'_{t+\Delta t}$ сегментов семейства кривых C, лежащих внутри искомой части запретного множества. $\beta\Delta t$ – максимальное расстояние, на которое уклоняющийся (скорость которого равна β) может углубиться внутрь множества Ω_t за время Δt.

2. Зная положение ищущего в момент времени t, его траекторию и скорость, найдем точку, описывающую положение ищущего в момент времени $t+\Delta t$, и построим множество $C''_{t+\Delta t}$ сегментов кривых из C, лежащих внутри круга контроля с центром в новой точке. Таким образом происходит как бы «нарастание» запретного множества, вызванное смещением ищущего за время Δt на расстояние $\alpha\Delta t$.

3. Объединив множества $C'_{t+\Delta t}$ и $C''_{t+\Delta t}$, получим множество $C_{t+\Delta t}$ сегментов кривых из C, лежащих внутри $\Omega_{t+\Delta t}$. Оболочка, порожденная множеством

$C_{t+\Delta t}$, представляет собой приближенное описание множества $\Omega_{t+\Delta t}$ в момент времени $t+\Delta t$.

Отметим, что все приведенные выше действия не представляют сложности с точки зрения программной реализации и позволяют учесть многие особенности задачи поиска, связанные с характеристиками участников поискового процесса. Свобода выбора семейства кривых C, позволяет варьировать степень точности и строить приближенное решение, сколь угодно близкое к аналитическому. Таким образом, предлагаемый метод удовлетворяет всем основным требованиям пункта 3.1.1.

Ниже приволится формальное описание метода и рассматриваются примеры его использования. Для удобства назовем первый этап построения описания множества «выеданием», второй – «нарастанием», а весь метод в целом – методом «выедания» и «нарастания».

4.2 Двумерный случай

Двумерный случай, как наиболее простой, наилучшми образом подходит для иллюстрации работы метода «выедания» и «нарастания».

4.2.1 Пример. Поиск в квадрате

Рассмотрим простой пример – поиск в квадрате. Введем в квадрате прямоугольную сетку и рассмотрим несколько последовательных шагов поисковой игры. Курсивом выделены действия, проводимые в соответствии с методом «выедания» и «нарастания».

Начало.

t=0. Ищущий находится в начальной точке P_0 траектории (рис.2).

В качестве приближения к запретному множеству используется оболочка, порожденная отрезками, которые лежат на пересечении линий сетки с кругом контроля.

«Выедание».

За время Δt между первым и вторым шагами уклоняющийся может пройти расстояние $\beta\Delta t$ (рис.3).

Так как при t=0 известно, что его нет в начальном круге контроля с центром в P_0, то к моменту времени Δt можно гарантировать его отсутствие лишь в круге с центром в P_0 и радиусом l-$\beta\Delta t$, где l, напомним, – это радиус обнаружения.

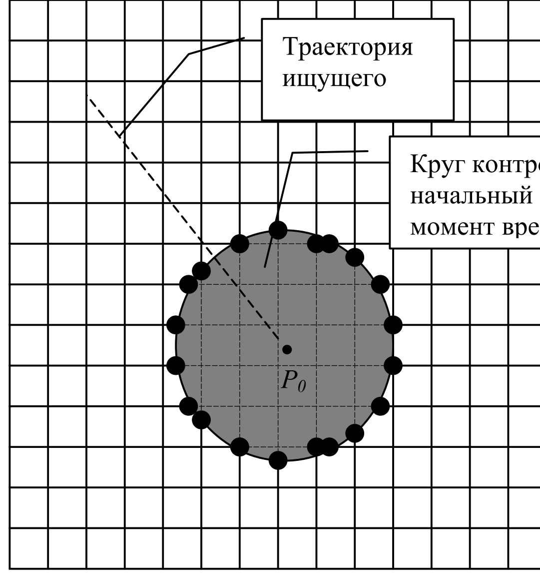

Рис. 2. Поиск в квадрате. Начало. Пунктиром выделены отрезки, лежащие внутри круга контроля в начальный момент времени.

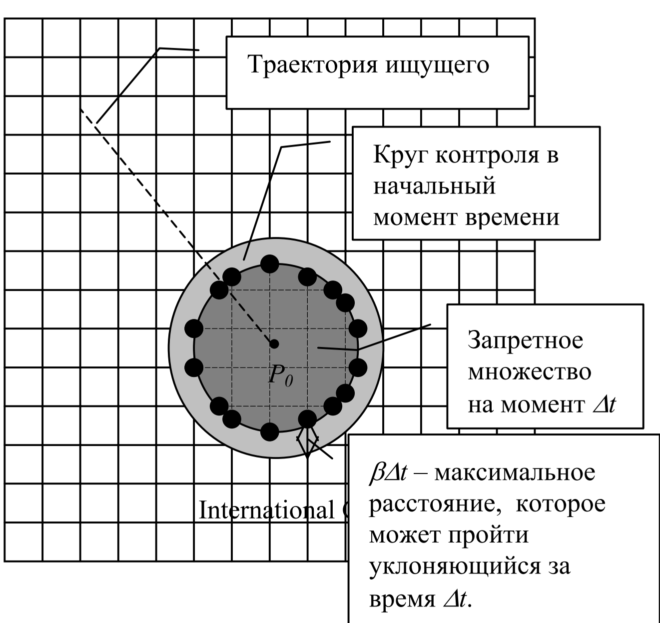

Рис. 3. Поиск в квадрате. «Выедание». Пунктиром выделены отрезки из множества $C'_{\Delta t}$.

«Выедание» осуществляется следующим образом – все отрезки, которые использовались для постоения начальной оболочки, укорачиваются на $\beta\Delta t$ с обоих концов. В результате получается множество отрезков $C'_{\Delta t}$, входящих в часть запретного множества в момент времени Δt.

«Нарастание».

За время Δt между первым и вторым шагами ищущий переместится в новую точку своей траектории – точку P_1 (рис.4).

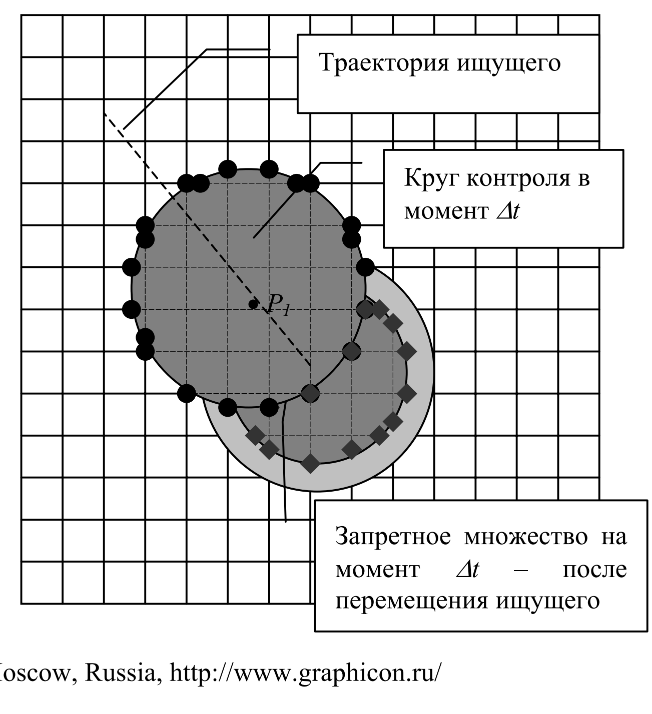

Рис. 4. Поиск в квадрате. «Нарастание». Пунктиром выделены отрезки из множества $C_{\Delta t}$. Отрезки с

круглыми концами образуют множество $C''_{\Delta t}$, отрезки с квадратными концами – множество $C'_{\Delta t}$.

Находится пересечение всех линий сетки с кругом контроля в новом положении. Множество $C''_{\Delta t}$ отрезков, по которым происходит пересечение, также принимает участие в построении оболочки, приближающей запретное множество в момент времени Δt.

Приближенным описанием запретного множества будет являться оболочка, порожденная множеством $C_{\Delta t}$ отрезков, полученным в результате объединения $C'_{\Delta t}$ и $C''_{\Delta t}$.

Далее в каждый момент времени t к запретному множеству применяются последовательно процедуры «выедания» и «нарастания».

4.3 Трехмерный случай

4.3.1 Формальное описание метода

Построим подробное формальное описание метода в трехмерном случае (все сказанное ниже без труда переносится на двумерный случай).

Введем в пространстве систему координат UVW и рассмотрим поисковое множество G.

Фиксируя последовательно пары $(u_i,v_j)\in U\times V,\ i=1,\ldots,N,\ j=1,\ldots,M$, построим семейство кривых

$$c^w = \{\, c^w_{ij}(w) = G(u_i,\ v_j,\ w),\ (u_i,\ v_j,\ w) \in G,\quad i = 1,\ldots,N,\ j = 1,\ldots,M\}.$$

Аналогично, фиксируя пары $(u_i,w_k)\in U\times W$, $(v_j,w_k)\in V\times W$, где $i=1,\ldots,N$, $j=1,\ldots,M$, $k=1,\ldots,L$, строим семейства кривых

$$c^v = \{\, c^v_{ik}(v) = G(u_i,\ v,\ w_k),\ (u_i,\ v,\ w_k) \in G,\quad i = 1,\ldots,N,\ k = 1,\ldots,L\},$$

$$c^u = \{\, c^u_{jk}(u) = G(u,\ v_j,\ w_k),\ (u,\ v_j,\ w_k) \in G,\quad j = 1,\ldots,M,\ k = 1,\ldots,L\}.$$

Объединение семейств кривых c^u , c^v и c^w обозначим через C.

В качестве приближенного описания произвольного подмножества $A\subset G$ будем использовать оболочку, порожденную множеством сегментов из C, лежащих внутри A. Если параметризация множества G выбрана удачно (что достаточно легко сделать, зная конкретный вид множества), то мы можем добиться сколь угодно высокой точности при приближенном описании любого подмножества $A\subset G$ (за счет увеличения числа кривых, то есть за счет увеличения параметров N, M и L).

В случае поисковой задачи роль изменяющегося во времени объекта, форму которого необходимо описать, играет запретное множество. Оказывается, что важнейшая задача изучения динамики его изменений решается очень просто – кривые, пронизывающие поисковое множество играют роль индикатора, показывая текущее состояние поискового процесса. Напомним, что запретное множество Ω_t в каждый момент времени t представляет собой объединение шара контроля и остаточного множества.

Зная радиус обнаружения и положение шара контроля в каждый момент времени, мы без труда можем построить приближенное описание круга контроля с помощью семейства кривых C. Но как описать динамически изменяющееся остаточное множество?

В момент времени $t=0$ остаточное множество пусто, и область Ω_0 состоит только из шара контроля, так что приближенное описание области Ω_0 мы можем построить без труда.

Предположим, что приближенное описание области Ω_t в момент времени t известно. Для построения $\Omega_{t+\Delta t}$ предлагается воспользоваться следующей процедурой. Определим в момент времени $t+\Delta t$ положение ищущего объекта (мы можем это сделать, поскольку знаем положение ищущего в момент времени t, траекторию его движения и скорость α). Пусть это точка $P_{t+\Delta t}$. Построим сферу $S_{t+\Delta t}$ с центром в точке $P_{t+\Delta t}$ и радиусом, равным радиусу l шара обнаружения. Рассмотрим произвольную кривую c из множества C. Пусть $c_s(t)$ – множество сегментов кривой c, входящих в область Ω_t. Применим ко всем кривым из множества C последовательно процедуры «выедания» и «нарастания».

Процедура «выедания». За время Δt уклоняющийся объект проходит расстояние $\beta\Delta t$, где β - скорость уклоняющегося объекта. Ищущий не имеет информации о положении и намерениях уклоняющегося до тех пор, пока последний не будет обнаружен. Поэтому при построении остаточного множества в момент времени $t+\Delta t$ надо исходить из худшей для ищущего ситуации – когда уклоняющий в момент времени t находится на границе следящей области и в момент времени $t+\Delta t$ не попадает в шар контроля, а проникает вглубь остаточного множества на расстояние $\beta\Delta t$. Чтобы реализовать такую ситуацию, для каждого сегмента $[t_1,\ t_2] \in c_s(t)$ каждой кривой $c\in C$ построим две сферы $s_1(t+\Delta t)$ и $s_2(t+\Delta t)$ с центрами в концах сегмента и радиусом $\beta\Delta t$ (назовем такие сферы сферами уклоняющегося) и найдем их пересечение со всеми кривыми множества C. Проделаем подобную процедуру со всеми сегментами всех кривых из множества C. Далее для каждой кривой $c\in C$ все сегменты, попавшие в сферы уклоняющегося (точнее, те части сегментов, которые принадлежат множеству $c_s(t)$), выкинем из $c_s(t)$. Получим промежуточное множество $c'_s(t+\Delta t)$ сегментов кривой c.

Процедура «нарастания». За время Δt ищущий проходит расстояние $\alpha\Delta t$. При этом изменяется положение шара контроля, и вся следящая область «продвигается» вдоль траектории ищущего. Чтобы учесть этот факт, найдем пересечение каждой кривой $c \in C$ со сферой $S_{t+\Delta t}$. Если кривая c не пересекает сферу $S_{t+\Delta t}$, то положим $c_s(t+\Delta t) = c'_s(t+\Delta t)$. Иначе найдем на кривой c сегмент $[t_1, t_2]$, лежащий внутри сферы $S_{t+\Delta t}$, и объединим $[t_1, t_2]$ с уже существующим множеством $c'_s(t+\Delta t)$. В результате получим множество $c_s(t+\Delta t)$ сегментов кривой c, входящих в область $\Omega_{t+\Delta t}$ в момент времени $t+\Delta t$. Пусть $C_s(t+\Delta t)$ – объединение всех множеств $c_s(t+\Delta t)$ по всем кривым c из C.

Построив оболочку, порожденную множеством $C_s(t+\Delta t)$, получим приближенное описание области $\Omega_{t+\Delta t}$ в момент времени $t+\Delta t$.

Остановимся более подробно на вопросе выбора параметризации поискового множества. Естественно, что общего алгоритма, позволяющего по заданному множеству строить оптимальную для данного метода параметризацию, не существует. Однако, руководствуясь общими геометрическими соображениями, можно предложить следующий подход: параметризация должна выбираться таким образом, чтобы соответствующие ей множества кривых «достаточно равномерно» пронизывали поисковую область. При этом, при увеличении чисел N, M и L эти кривые должны стремиться заполнить собой все поисковое множество.

Выбор параметризации, в общем, зависит от вида данной поисковой области. Как правило используются стандартные способы параметризации в зависимости от выбранной системы координат:

1. в прямоугольной декартовой системе координат XYZ:
   $x = x, y = y, z = z$;
2. в цилиндрической системе координат:
   $x = r\cos\varphi, y = r\sin\varphi, z = h$;
3. в сферической системе координат:
   $x = r\sin\theta\cos\varphi, y = r\sin\theta\sin\varphi, z = r\cos\theta$

и т.д.

Часто, учитывая специфику поискового множества и исследуемого класса траекторий, можно обойтись всего двумя или даже одним множеством кривых c (имеются ввиду множества c^u , c^v и c^w). Например, в случае цилиндра достаточно использовать только семейство кривых c^w=c^h при стандартной параметризации в цилиндрических координатах (угол, радиус, высота), то есть только вертикальные прямые. Аналогичная ситуация имеет место и в задаче с тором.

4.3.2 Пример 1. Поиск в цилиндре

Рассмотрим задачу визуализации поиска в цилиндре при траектории движения ищущего объекта, показанной на рисунке 5 (жирная линия).

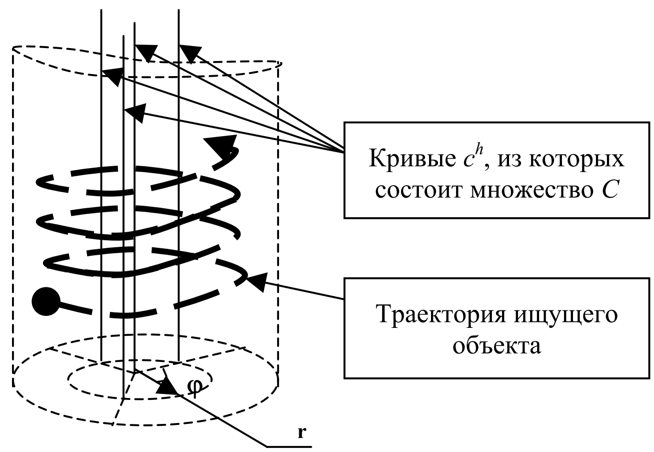

Рис. 5. Поиск в цилиндре.

Выберем стандартную параметризацию поисковой области:

$x = r\cos\varphi, y = r\sin\varphi, z = h,$

где $r \in [0,R]$, R – радиус цилиндра, $\varphi \in [0,2\pi]$ – угол между радиусом и осью абсцисс, $h \in [0,\infty]$ – высота цилиндра.

В этом случае можно обойтись лишь одним набором кривых

$c^w = c^h = \{c^h_{ij}(h) = G(r_i, \varphi_j, h), r_i = i \cdot R/N, \varphi_j = j \cdot 2\pi/M, i = 1,\ldots,N, j = 1,\ldots,M\}$ –

отрезками вертикальных прямых $x=const$, $y=const$, проходящих через точки пересечения концентрических окружностей с центром на оси цилиндра и радиусов, проведенных из этой точки. «Плотность» множества C влияет на точность моделирования и определяется по конкретным числовым параметрам задачи, например, необходимо учитывать скорость β движения уклоняющегося объекта (точнее, максимальное расстояние $\beta\Delta t$, проходимое уклоняющимся за один временной шаг, которое ограничивает сверху расстояние между кривыми из C).

Моделирование проводится в соответствии с приведенным выше алгоритмом. Можно отметить некоторые особенности задачи, проистекающие из принятых нами ограничений на поисковую область и вид траектории. Например, значительно упрощается построение оболочки, натягиваемой на множество получаемых в процессе моделирования сегментов, так как каждая кривая из множества C имеет (по истечении определенного промежутка времени, за который процесс «устанавливается») и притом единственный параметрический сегмент, соответствующий пересечению данной конкретной кривой и следящей области. Из этого следует, что для построения указанной оболочки достаточно построить всего лишь его верхнюю и нижнюю «крышки», что дает возможность применить значительно более простой и эффективный алгоритм.

4.3.3 Пример 2. Поиск в торе

Рассмотрим еще один интересный пример поисковой задачи – поиск в торе (рис.6). Поиск осуществляется по спиралевидной траектории, похожей на траекторию, использовавшуюся для поиска в цилиндре.

Множество кривых состоит из набора окружностей, построенного следующим образом:

- все окружности лежат в плоскостях параллельных *Oxy,*
- центры окружностей лежат на оси *Oz,*
- окружности целиком принадлежат тору,
- в плоскости *Oxz* вводится прямоугольная сетка и выбираются окружности, удовлетворяющие предыдущим условиям и проходящие через узлы этой сетки.

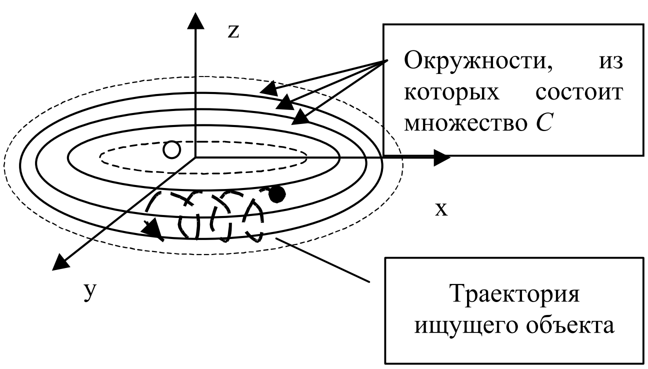

Рис. 6. Поиск в торе.

4.3.4 Комментарий

В более общем случае, когда вид траектории и/или поискового множества заранее неизвестен (например, при интерактивном решении поисковых задач), необходимо применение общего алгоритма построения оболочки множества сегментов кривых.

5. РЕЗУЛЬТАТЫ

Описанный в работе метод приближенного представления переменных множеств предназначен для качественного решения поисковой задачи, т.е. построения траектории движения ищущего объекта при заданных параметрах. Он позволяет оценить пригодность выбранной траектории путем визуализации динамики изменения формы и положения запретного множества. При этом метод не накладывает никаких ограничений на вид поисковой области. Другие же существующие на сегодняшний день методы решения задач поиска применимы лишь к ограниченному классу поисковых множеств – плоскостям и поверхностям.

Метод ориентирован на эффективную программную реализацию и учитывает все особенности поисковой задачи, которые сложно описать аналитически, например, изменение формы следящей области вблизи границы поискового множества [7].

По сравнению с распространенным в машинной графике методом представления объемов в виде объединения вокселов (элементов объема), данный метод обеспечивает значительный выигрыш в производительности, так как позволяет понизить размерность задачи на 1 (например, вместо трехмерной решается одна или несколько двумерных задач). Это позволяет строить переменные множества в режиме псевдореального времени.

В настоящее время существуют программные реализации метода для поиска в квадрате, в цилиндре и в торе.

6. ЗАКЛЮЧЕНИЕ

Подведем итог. Использование компьютерной графики значительно облегчает решение поисковых задач. Однако прежде, чем использовать этот удобный и красивый инструмент, необходимо научиться описывать множества, участвующие в задаче поиска, так, чтобы была возможность эффективно применять существующие методы визуализации. Аналитические методы, несмотря на их абсолютную точность, имеют ряд существенных недостатков, что приводит к необходимости искать другие пути решения этой проблемы. В работе был предложен простой и в то же время эффективный метод представления произвольных динамически изменяющихся множеств. Это особенно важно в случае трёхмерного поиска, где практически не существует способа аналитического описания переменных множеств, вовлеченных в процесс решения задачи.

7. ЛИТЕРАТУРА

[1] Петросян Л.А., Гарнаев А.Ю. Игры поиска. – СПб.: Изд-во С.-Петербургского ун-та, 1992.

[2] Чхартишвили А.Г., Шикин Е.В. Динамический поиск объектов. Геометрический взгляд на проблему // Фундаментальная и прикладная математика. 1995. Т. 1, №4. С. 827-862.

[3] Чхартишвили А.Г., Шикин Е.В. Метод следящих областей в задачах поиска // Математический сборник. 1993. Т.184, №10. С. 107-134.

[4] Ким Д.П. Методы поиска и преследования подвижных объектов. – М.: Наука, 1989.

[5] Айзекс Р. Дифференциальные игры. – М.: Мир, 1967.

[6] Gal J. Search Games. – New York, 1980.

[7] Ежков С.В. Графический метод представления переменных множеств для решения поисковых задач //Вестн. Моск. ун-та. Сер. 15, Вычислительная математика и кибернетика. 1998, №4. С.33-35.

Автор:

Ежков Сергей Владимирович, аспирант факультета вычислительной математики и кибернетики Московского Государственного Университета им. М.В. Ломоносова

Адрес: Москва, 119899, Воробьевы горы, МГУ, 2-й учебный корпус, факультет ВМиК, кафедра автоматизации научных исследований
E-mail: SergeyY@hotmail.com

Graphical method for dynamic sets approximate representing in search problems

Sergey Yezhkov

Abstract

Search problems are very interesting from both practical and theoretical points of view but yet very difficult to solve analytically – theoretical solutions exist only for very limited number of (mostly 2D) areas.

In this paper we propose an easy to use yet flexible and effective way to solve multidimensional search problems using real-time information visualization algorithms. This method was successfully applied to solving search problems in torus, cylinder and other 3D-dimensional bodies.

Keywords: *search problems, dynamic set, approximate representing of dynamic sets.*