

РОССИЙСКАЯ АКАДЕМИЯ НАУК  
УРАЛЬСКОЕ ОТДЕЛЕНИЕ  
ИНСТИТУТ МАТЕМАТИКИ И МЕХАНИКИ

АЛГОРИТМЫ  
И  
ПРОГРАММНЫЕ  
СРЕДСТВА  
ПАРАЛЛЕЛЬНЫХ  
ВЫЧИСЛЕНИЙ

Сборник научных трудов

Выпуск 2

ЕКАТЕРИНБУРГ

1998

УДК 519.688+517.9

**Алгоритмы и программные средства параллельных вычислений:** [Сб. науч. тр.]. Екатеринбург: УрО РАН, 1998. Вып. 2. ISBN 5-7691-0859-2.

Сборник посвящен вопросам организации параллельных вычислений и параллельным алгоритмам, рассмотренным на семинаре ИММ УрО РАН по параллельным вычислениям (1995–1997 г.).

Материалы могут быть рекомендованы специалистам, разрабатывающим параллельные алгоритмы, а также разработчикам программного обеспечения для параллельных машин.

Ответственные редакторы доктор физ.-мат. наук Л.Д.Попов,  
С.В.Шарф

Издание подготовлено в пакете CugTUG-emTex с использованием кириллических шрифтов семейства LH

ISBN 5-7691-0859-2

A  $\frac{98(98)}{8П6(03) - 1998}$  БО

©УрО РАН, 1998 г.

## СТАТИСТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ЗАДАЧ УПРАВЛЕНИЯ КОСМИЧЕСКИМИ АППАРАТАМИ\*

С. С. Кумков

### Введение

В широком круге задач прикладного и исследовательского плана возникает необходимость применения прямых численных методов оценки качественных и количественных показателей поведения управляемых систем.

В настоящее время в Институте математики и механики УрО РАН проводятся исследования задач управления космическими аппаратами, предназначенными для наблюдения за земной поверхностью, изучения околоземного космического мусора и опасных космических объектов (астероидов, комет и др.) [1, 2]. При разработке новых методов управления используются результаты теории дифференциальных игр и минимаксной фильтрации [3 – 5]. Некоторые постановки и методы решения рассматриваемых задач управления космическими аппаратами описаны в [6 – 8].

Математические модели объектов и их систем управления включают нелинейные системы дифференциальных уравнений, случайные внешние воздействия, информационные шумы и помехи. Вследствие этого аналитическое описание процессов управления и конечных результатов исключено. Для исследования таких моделей применяется численное моделирование.

---

\*Работа выполнена при поддержке Российского фонда фундаментальных исследований по гранту №97-01-00672.

Наличие случайных факторов приводит к тому, что выходные результаты являются случайными величинами с неизвестными заранее статистическими характеристиками. Статистическое моделирование оказывается единственным приемлемым методом для получения этих характеристик.

Вследствие полноты и сложности моделей однократный расчет процесса управления аппаратом уже требует ощутимых затрат машинного времени. В случае применения статистического моделирования длительность счета возрастает пропорционально. Использование компьютеров с большим быстродействием не дает существенного улучшения. Кардинальное решение проблемы возможно при ином подходе к организации вычислительного процесса, например с использованием параллельных вычислений.

В статье описывается ориентированный на машину МВС-100 параллельный алгоритм статистического моделирования.

## **1. Задачи управления космическими аппаратами**

Поясним содержательно упомянутые модели управляемого движения. На рис. 1 приведен возможный внешний вид космического аппарата. Источниками информации для алгоритмов управления являются оптические или радиолокационные измерители (символ а) пеленгов, дальности, скорости сближения и т.д.

Отмечены основные органы управления:  
двигатели коррекции движения центра масс (символ b);  
двигатели угловой ориентации и стабилизации (символ с).

Задачами управления космическими аппаратами могут быть: наблюдение за земной поверхностью, наблюдение и идентификация обломков околоземного мусора, исследовательские полеты к астероидам и кометам в режимах пролетного наблюдения или подлета с зависанием на заданных расстояниях и т.д.

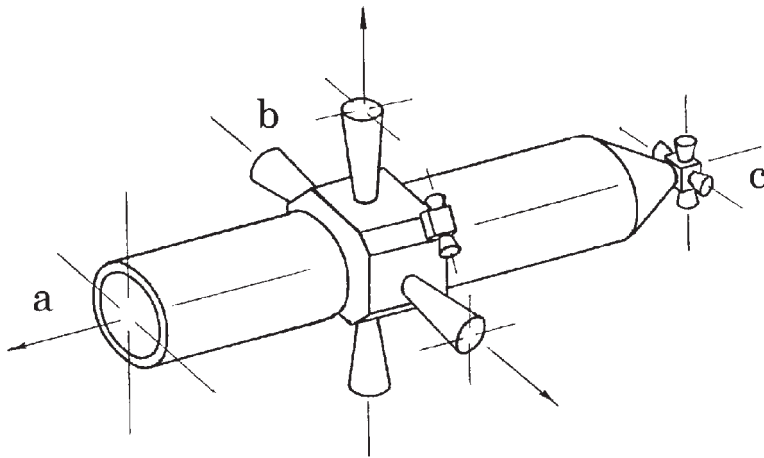


Рис. 1. Космический аппарат для наблюдения за земной поверхностью

Для выполнения указанных задач необходима разработка соответствующих алгоритмов и методов управления, обеспечивающих требуемые качественные и количественные показатели. Как отмечалось выше, исследование и отработка данных алгоритмов возможны лишь с использованием численного моделирования.

## 2. Применение параллельных вычислений

### 2.1. Пример постановки задачи

Рассматривается задача плоского преследования (рис. 2): космический аппарат (КА), обозначенный  $P$ , наводится на опасный космический объект (ОКО), обозначенный  $E$ . Рассмотрим разностную систему координат, связанную с КА. В этой системе координат скорость ОКО имеет отрицательную проекцию на продольную ось  $Ox$ . В процессе сближения скла-

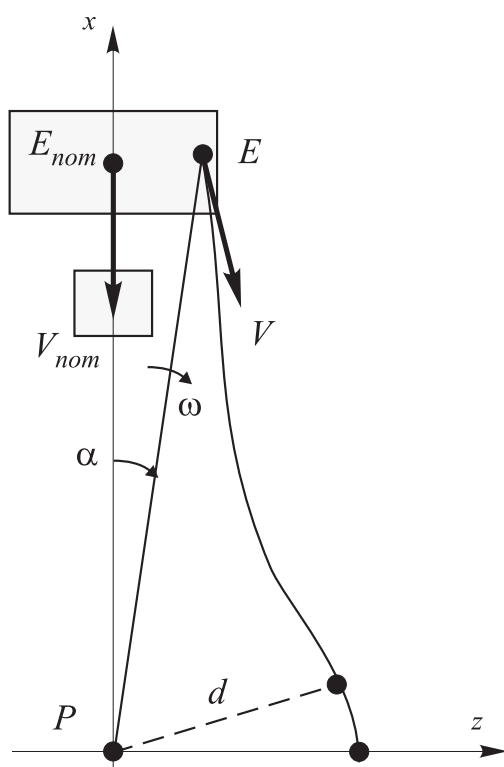


Рис. 2. Задача плоского преследования. Начальные неопределенности

дывается некоторая траектория (схематично показана кривой линией). Цель КА – минимизировать промах (на рисунке – расстояние  $d$ ), цель ОКО противоположная. В процессе преследования объекты могут воздействовать на свое поведение: КА, например, имеет возможность импульсно управлять своей боковой скоростью, что в разностной системе приводит к изменению боковой скорости ОКО. В реальной ситуации возможности такого управления стеснены: за один импульс скорость изменяется на фиксированную величину и, кроме того, запас топлива ограничен, поэтому количество таких импульсных воздействий тоже не превосходит известного числа.

Для выработки управления КА может привлекать некоторую информацию об ОКО, получаемую от различных измерительных приборов. Это может быть дальность до ОКО (длина  $PE$ ), угол линии визирования  $\alpha$ , угловая скорость ОКО  $\omega$  в разностной системе координат. Получаемые замеры могут “зашумляться”, т. е. отличаться от истинного значения на некоторую случайную величину. Это связано с погрешностью измерений реальных приборов. Считается, что некоторые параметры этих ошибок (максимальное значение, закон распределения и т. д.) нам известны.

Точное начальное положение ОКО нам не известно. Известно лишь номинальное (расчетное) начальное положение  $E_{\text{ном}}$  и зона возможного разброса (на рис. 2 обозначена прямоугольником около точки  $E_{\text{ном}}$ ). Аналогично о начальной скорости ОКО нам известно лишь номинальное ее значение и зона разброса ( $V_{\text{ном}}$  и прямоугольник вокруг  $V_{\text{ном}}$ ).

В процессе статистического моделирования проводится счет некоторого количества единичных сближений. Результатом статистического моделирования являются гистограммы распределений промаха и расхода управления. Под расходом управления понимается количество управляющих импульсов КА, истраченных в данной реализации на изменение своей боковой скорости.

Гистограмма распределения промаха строится следующим образом. Задаемся разумно большим значением промаха  $d_{\text{max}}$ . Далее исходный отрезок  $[0; d_{\text{max}}]$  разбивается на достаточно большое количество маленьких отрезков. Затем находится, сколько раз в каждый из них попадало значение промаха, получаемое в процессе моделирования единичных сближений. На основе полученного массива вычисляется гистограмма распределения промаха. Аналогично строится гистограмма распределения расхода управления (с учетом того, что расход управления есть величина ограниченная и изменяющаяся не непрерывно, а дискретно).

Целью статистического моделирования является выяснение свойств алгоритма, связи получаемых распределений по промаху и расходу управления с параметрами системы (номинальным начальным положением, начальными неопределенностями, ограничениями на управления и т. д.).

## 2.2. Организация последовательной программы

Рассмотрим существующий последовательный алгоритм организации статистических испытаний. Его блок-схема приведена на рис. 3.

Блоки 3,4 заключают в себе алгоритмы, реализующие модель единичного сближения. В блоке 3 содержится инициализация начальных положений, рабочих переменных и другого числового материала, необходимого для просчета единичного сближения. Счет осуществляется в блоке 4; сюда включаются процедуры интегрирования движения КА и ОКО, моделирования замеров, выработки управления объектов.

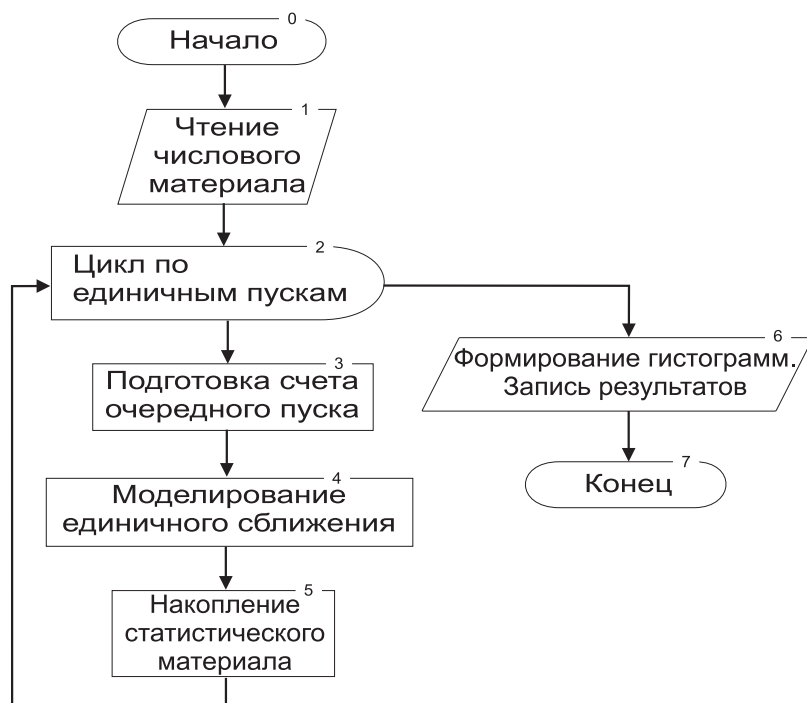


Рис. 3. Блок-схема последовательного алгоритма



Часть, ответственная за расчет единичного пуска, помещена в цикл по отдельным сближениям (блок 2). Числовой материал статистического моделирования накапливается в блоке 5 – после расчета каждого единичного сближения. После проведения всех пусков накопленный материал обрабатывается (блок 6): формируются гистограммы распределения. По ним вычисляются вероятностные характеристики распределений промаха и расхода управления: среднее значение и дисперсия, которые выдаются пользователю. Возможен просмотр полученных гистограмм на экране компьютера.

Последовательный вариант алгоритма был реализован для IBM-совместимых компьютеров с процессорами типа i80x86 на языке Microsoft QuickBasic 4.5.

### **2.3. Организация параллельной программы**

Для МВС-100 имеются компиляторы языков Си и Фортран. Поэтому для переноса программ на вычислитель требуется их предварительный перевод на один из этих языков. Отметим, что при переносе программы, написанной на языке Си или Фортран, с персонального компьютера на вычислитель для счета в однопроцессорном режиме дополнительных изменений не требуется.

Перевод осуществлялся на язык Си, наиболее мощный и удобный как в вычислительном плане, так и по имеющимся средствам написания и отладки программ.

В настоящее время реализована наиболее простая и очевидная модель распараллеливания. В ней каждый из рабочих процессоров выполняет статистическое моделирование по некоторому числу единичных пусков. Общее число пусков равно заданному представительному числу испытаний. Полученные после проведения счета частичные массивы счетчиков со всех процессоров суммируются, в результате образуется итоговый массив, как если бы он был получен на одном процессоре.

При параллельном счете процессоры делятся на две категории: ведущий и ведомый. Ведущим является процессор с номером 0. Алгоритм сбора информации устроен так, что после окончания работы всех ведомых процессоров итоговый массив

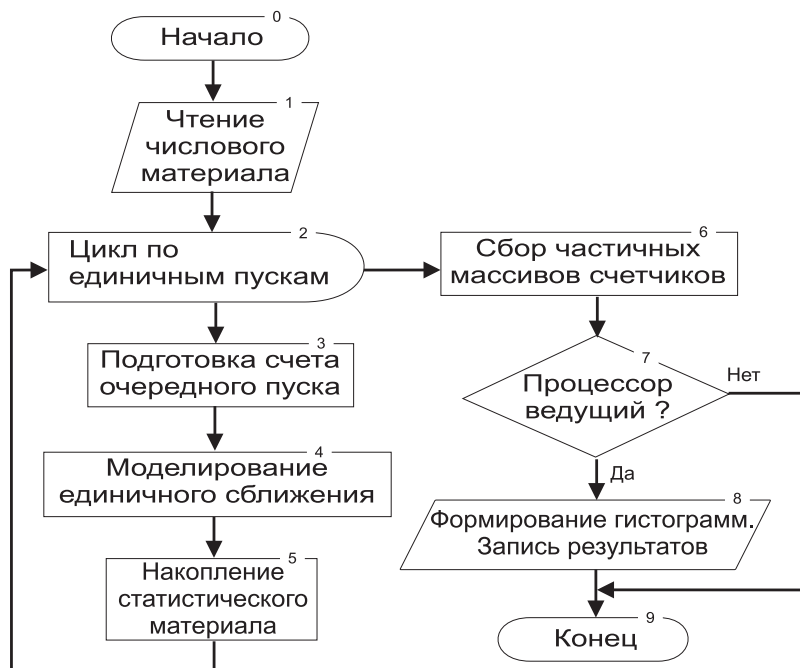


Рис. 4. Блок-схема параллельного алгоритма

счетчиков оказывается на ведущем процессоре (см. п. 2.4). Поэтому, кроме непосредственно счета, в его функции входит формирование гистограмм распределения и запись их на жесткий диск головной машины.

Ведомые процессоры производят счет статистических испытаний, после чего в процессе суммирования выдают накопленные ими массивы счетчиков ведущему процессору.

Блок-схема параллельного алгоритма приведена на рис. 4. Отличие параллельного алгоритма от последовательного заключается в добавлении блока суммирования частичных массивов счетчиков (блок 6 на рис. 4). Сам расчетный алгоритм (блоки 2 – 5) сохраняется без изменения. Это позволяет применять предлагаемую схему распараллеливания и для других содержательных задач.

Выполняемый модуль, реализующий параллельный алгоритм, один как для ведущего, так и для ведомых процессоров. Унификация выполняемого файла связана с тем, что алгоритм ведущего процессора по сравнению с ведомым имеет лишь одну дополнительную процедуру – процедуру формирования гистограмм распределения и записи их на диск (блок 8 на рис. 4). Все прочее: расчетный алгоритм (блоки 0 – 5) и алгоритм суммирования (блок 6) – те же, что и для ведомого процессора. Объем дополнительной процедуры по отношению к общему объему программы невелик (примерно 100 строк из 3000). Выполняемый код этой процедуры загружается и на ведомые процессоры. Но для того, чтобы она там не выполнялась, введена специальная логика (блок 7).

Еще раз отметим, что при использовании описанной схемы распараллеливания для организации параллельного алгоритма нужно добавить в последовательный алгоритм лишь блок суммирования частичных массивов счетчиков и немного подправить блок чтения числового материала (чтобы на каждом процессоре получить “частичное” число пусков, которые будут на нем просчитаны).

## 2.4. Обмен информацией и ее выходная обработка

Результатом работы каждого процессора являются массивы, содержащие счетчики гистограмм законов распределения промаха и истраченных импульсов. Для построения выходных законов распределения требуется два массива счетчиков “суммарных” гистограмм. При суммировании массивов используется метод двоичного дерева. Рассмотрим процесс объединения информации на примере системы из 11 процессоров (рис. 5).

На первом шаге каждый процессор с нечетным номером пересылает насчитанные им массивы процессору с четным номером (на единицу меньшим), который производит поэлементное сложение посланного ему массива со своим. После этого процессоры с нечетными номерами завершают работу. Если при

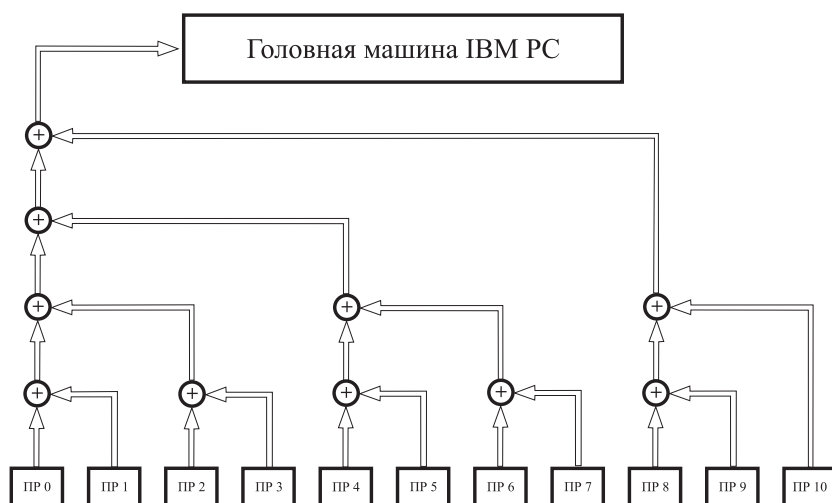


Рис. 5. Схема суммирования массивов счетчиков гистограмм

этом у какого-то процессора с четным номером нет соответствующей пары (как, например, на рис. 5 у процессора с номером 10), то такой процессор на этом шаге суммирования информацию не пересылает. На втором шаге суммирования происходит аналогичная процедура: процессоры “разбиваются” на пары. Второй процессор в паре передает свои массивы, полученные в результате сложения на предыдущем шаге, первому процессору, который и производит сложение. После передачи данных второй процессор в паре прекращает работу. Процессор, которому не досталось пары, информацию не пересылает.

Процедура повторяется, пока в работе не останется лишь ведущий процессор с номером 0, который производит запись на диск полученных “суммарных” данных.

Поскольку часть сложений делается одновременно, общее число шагов суммирования уменьшается с  $N - 1$  до  $\log_2 N$ , где  $N$  – число процессоров. В нашем примере – с 10 шагов до 4. Вследствие такой организации достигается определенная экономия временных затрат.

### 3. Результаты исследований

Для проведения временных оценок был взят случай, когда в обычной программе на персональной ЭВМ (i80486DX2/66) расчет одной реализации сближения аппарата с астероидом требует 35 с машинного времени. Суммарное машинное время (количество испытаний — 200) составило около 2 ч. При переходе на язык Си на той же ЭВМ наблюдается ускорение расчетов в 6.7 раза. Это достигается за счет лучшей внутренней организации выполняемого кода при использовании компилятора Borland C++ 3.1. Временные затраты сократились до 18 мин. После переноса программы с персональной ЭВМ на вычислитель в однопроцессорном режиме время моделирования сократилось в 4.5 раза за счет RISC-архитектуры процессоров MBC-100 и составило 4 мин.

При проведении временных оценок был проведен счет на 20 процессорах по 10 испытаний на каждом. Время счета составило 18 сек. Таким образом, ускорение при переходе на вычислитель от однопроцессорного счета к многопроцессорному составило 13 раз, против ожидаемых 20. Расхождение реального и ожидаемого ускорений обусловлено тем, что при счете 10 пусков на каждом процессоре доля “накладных” расходов (загрузка выполняемого модуля, обмена процессоров, выдача результатов) в общем времени весьма ощутима. Увеличивая число испытаний, мы должны ожидать приближение реального ускорения к ожидаемому. При счете 2000 пусков на 20 процессорах (по 100 на каждом) время счета составило 126 с, а в однопроцессорном режиме — 2386 с. Таким образом, в этом случае ускорение составило 18.3 раза.

На рис. 6 представлены типичные результаты статистического моделирования процессов управления КА в задаче наведения на опасный космический объект (астероид или комету). Изображены интегральные законы (гистограммы) распределения вероятностей промаха и затрат импульсного управления. Они построены для закона управления, описанного в [6 – 8], и двух вариантов формирования помех: RM — случайная помеха с равномерным законом распределения и GM — игровая поме-

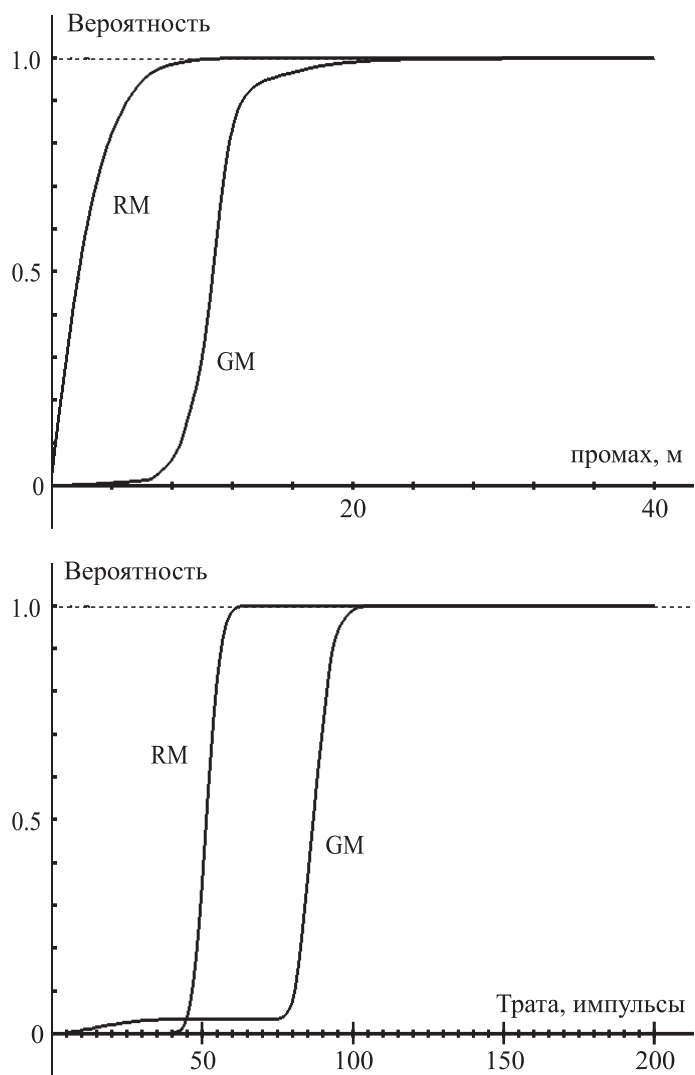


Рис. 6. Оценка эффективности управления КА в условиях неопределенности информации

ха. Подобные гистограммы распределения рассчитываются для различных вариантов параметров задачи (величина исходных неопределенностей, тип помехи и т.д.) и используются для тестирования алгоритмов управления.

Проведенные исследования показывают, что распараллеливание вычислений существенно уменьшает время, необходимое для статистического моделирования.

Приведенная схема распараллеливания при своей реализации не затрагивает основного расчетного алгоритма. Поэтому она легко реализуема. Кроме того, она применима и к другим содержательным задачам, в которых в результате статистического моделирования требуется найти законы распределения и связанные с ними вероятностные характеристики.

#### СПИСОК ЛИТЕРАТУРЫ

1. *Bendis J., Hoffmann J.P., Liebscher R., Rollenhagen F.* Detection of Space Debris by the Use of Space Based Optical Sensors // Proceedings of the First European Conference on Space Debris. Darmstadt, Germany, 5 – 7 Apr. 1993. p. 91 – 97.
2. *Рудин В.Н., Ильин В.В.* Проблемы сближения и перехвата опасных космических объектов // Тез. докл. междунар. конф. “Проблемы защиты Земли от столкновения с опасными космическими объектами”. Снежинск, 1994. Часть 1. С. 78 – 79.
3. *Красовский Н.Н., Субботин А.И.* Позиционные дифференциальные игры. М.: Наука, 1974. 456 с.
4. *Куржанский А.Б.* Управление и наблюдение в условиях неопределенности. М.: Наука, 1977. 392 с.
5. *Черноусько Ф.Л., Меликян А.А.* Игровые задачи управления и поиска. М.: Наука, 1978. 270 с.
6. *Кумков С.И., Пацко В.С.* Задача преследования с неполной информацией. [Препринт.] Екатеринбург: ИММ УрО РАН, 1993. 64 с.

7. *Кумков С.И., Пацко В.С.* Импульсные коррекции в задаче преследования с неполной информацией // Изв. РАН. Техн. кибернетика. 1994. N 3. С. 115 – 126.
8. *Кумков С.И., Пацко В.С.* Информационные множества в задаче преследования. // Век радио: Перспективные пути развития антенных систем космической связи, теории управления и распознавания образов. Екатеринбург: УрО РАН, 1996. С. 172 – 190.