

ПОИСК ТРАЕКТОРИИ ДВИЖЕНИЯ ЛЕТАТЕЛЬНОГО АППАРАТА ПО ПОЛЮ ВЫСОТ

С. Н. Васильев, Я. В. Малыгин, А. В. Мироненко

UNMANNED AIRBORNE VEHICLE TRAJECTORY SEARCH USING ELEVATION FIELD

S. N. Vasilyev, Y. V. Malygin, A. V. Mironenko

Рассматривается задача поиска траектории движения летательного аппарата, проходящей через области с минимальной ошибкой навигации. Навигация осуществляется по информации о поле высот в целом и фрагменту поля, снятому при движении аппарата. Описывается параллельный алгоритм поиска и полученные результаты.

Ключевые слова: навигация летательного аппарата, поиск траектории движения, параллельный алгоритм

The problem of unmanned airborne vehicle trajectory search is considered. The target trajectory must pass over the regions with small navigation error. The navigation is performed using the entire elevation field and its fragments measured by the vehicle in flight. Parallel algorithm of trajectory search and obtained results are described.

Keywords: navigation of an airborne vehicle, trajectory search, parallel algorithm

Введение

Задача навигации состоит в определении местоположения летательного аппарата (ЛА): координат центра масс t и ориентации a по информации о геофизическом поле в целом и по фрагменту поля, снятому ЛА при движении. В качестве геофизического поля рассматривается поле высот F , заданное на области $Q \subset \mathbb{R}^2$. Рельеф определяется графиком $\text{graph } F$.

Пусть ЛА находится в точке $v = (t, a) \in \mathbb{R}^6$. Фрагмент поля снимается ЛА посредством сканирующего луча $l \in \Delta$, где $\Delta = \{l\}$ - заданный конус сканирования. Через $\rho(t, a, l, F)$ обозначим наклонную дальность между точкой t и $\text{graph } F$ вдоль луча $a(l)$ (см. рис. 1). Набор наклонных дальностей при $l \in \Delta$ назовем фрагментом поля и обозначим его $\varphi_{t,a}(l, F)$. Для функций, заданных на Δ , определим норму, например

$$\|\varphi(l)\|_{\Delta} = \left(\int_{\Delta} |\rho(t, a, l, F)|^2 dl \right)^{1/2}.$$

Задача навигации сводится к поиску минимального уклонения между фрагментами

$$d(v, F) = \inf_{(T,A) \in V} \|\varphi_{t,a}(l, F) - \varphi_{T,A}(l, F)\|_{\Delta}$$

по множеству положений (T, A) из некоторой ограниченной области V . Эта задача решается на бортовом компьютере [1, 2].

Пусть $v^*(v, F)$ - один из аргументов, при котором достигается этот инфимум. Будем считать его вычисленным местоположением ЛА. Расстояние между истинным положением ЛА и вычисленным назовем ошибкой навигации в точке u . $D(v) = \rho(v, v^*(v))$.

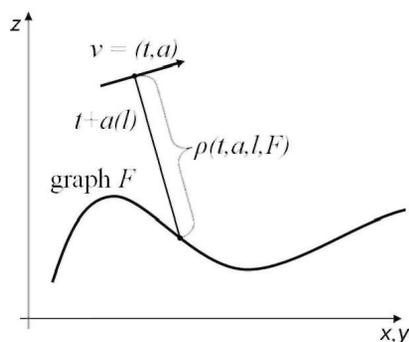


Рис. 1. Местоположение ЛА над рельефом

1. Построение алгоритма

Расчетная траектория движения (ЛА) состоит из чередующихся участков двух типов:

- 1) участок коррекции, на котором ЛА осуществляет сканирование поля высот и определение координат с последующей корректировкой курса.

- 2) участок «слепого» полета, на котором движение ЛА определяется заданной траекторией с учетом последней корректировки; накапливаемое на этом участке отклонение от курса определяется техническими параметрами ЛА и должно быть скомпенсировано в следующей зоне коррекции.

Оптимальная с точки зрения навигации траектория движения ЛА должна быть выбрана так, чтобы ее участки коррекции находились в районах с ошибкой навигации не больше некоторой наперед заданной величины D_{\max} . Нахождение траектории движения осуществляется до запуска ЛА и, в связи со значительным объемом вычислений, может выполняться на многопроцессорном вычислительном комплексе [2].

Для упрощения расчетов предполагаем, что высота полета ЛА постоянна, т.е. его положение определяется двумя пространственными координатами (x, y) и азимутом α : $v = (x, y, \alpha) \in \mathbb{R}^3$. Сама же траектория представляет собой ломаную, в ее вершинах производится коррекция, а отрезки являются участками «слепого» полета.

Для задачи поиска траектории используются следующие ограничения:

- допустимая величина ошибки навигации D_{\max} ;
- функция ошибки навигации для поля высот в точке в зонах коррекции $D(v) \leq D_{\max}$;
- начальная и конечная точка траектории v_0 и v_n ;
- расстояние между соседними зонами коррекции удовлетворяет условию: $r_{\min} \leq \rho((x_i, y_i), (x_{i+1}, y_{i+1})) \leq r_{\max}$;
- изменение угла азимута между соседними зонами коррекции $|\alpha_i - \alpha_{i+1}| \leq \alpha_{\min}$, (рис. 2).

В нашем алгоритме мы предполагаем, что поле высот F задано на сетке $N_1 \times N_2$, а допустимые значения азимута α дискретизированы. Для поиска кратчайшей расчетной траектории, которая удовлетворяет указанным выше условиям, была использована модификация алгоритма Дейкстры поиска кратчайшего пути на графе [3]. Вершинами этого графа являются все узлы $v(x_i, y_j, \alpha_k)$ исходной сетки или ее прореженной версии, а дуги есть только между вершинами, удовлетворяющими набору ограничений (см. рис. 2). Вес дуги равен расстоянию $\rho((x_i, y_i), (x_j, y_j))$. При переходе к следующему шагу алгоритма Дейкстры к текущей найденной величине длины пути добавляется небольшой штраф, что дает преимущество тем траекториям, которые состоят из как можно меньшего числа отрезков.

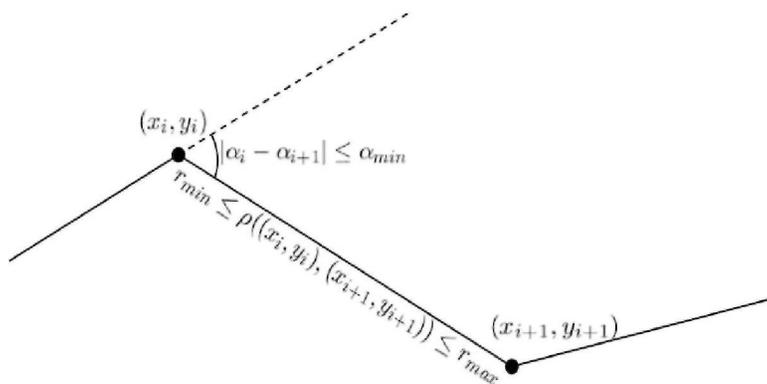


Рис. 2. Ограничения на траекторию

Для вычислительных экспериментов в качестве поля высот были использованы данные нескольких географических областей. В качестве конуса сканирования Δ взят симметричный набор из 5 лучей, лежащих в одной плоскости, с угловым расстоянием между соседними лучами в 5° . Данные полей высот были взяты в Интернете:

<http://srtm.csi.cgiar.org/SELECTION/inputCoord.asp>.

Результат работы алгоритма поиска оптимальной траектории представлен на рис. 3. Здесь поле высот задано на сетке размером 430×161 с шагом 30 м.

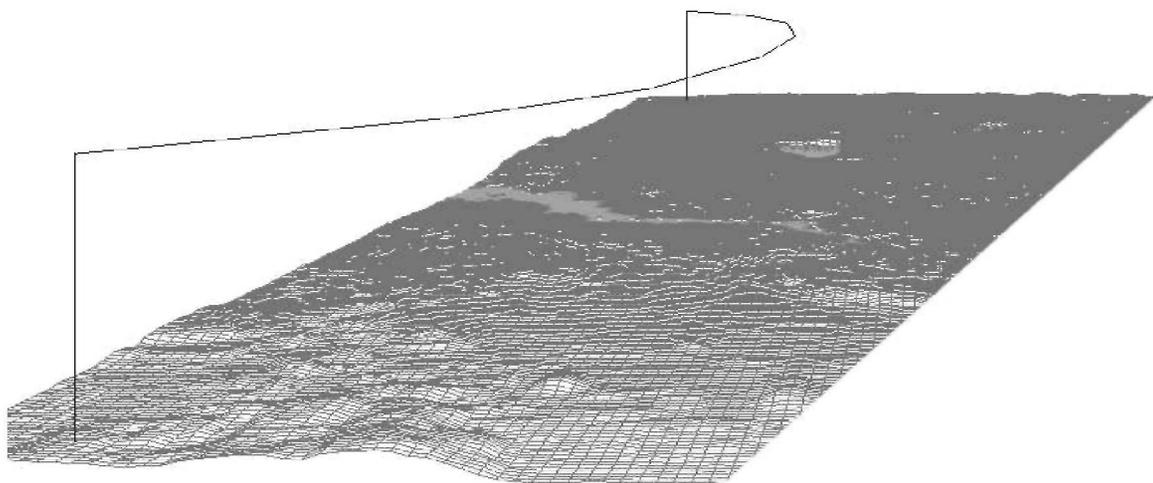


Рис. 3. Вычисленная траектория

На рис. 4 в плоскости XOY изображены найденные траектории при разных ограничениях на D_{\max} и α_{\max} .

2. Распараллеливание алгоритма

Ошибку навигации можно просчитать заранее для всего региона, подробнее см. в [2]. Как показывают эксперименты, в реальных задачах вычисление ошибок навигации требуются не во всех узлах региона, а лишь для 5 - 10% от их общего количества. Между

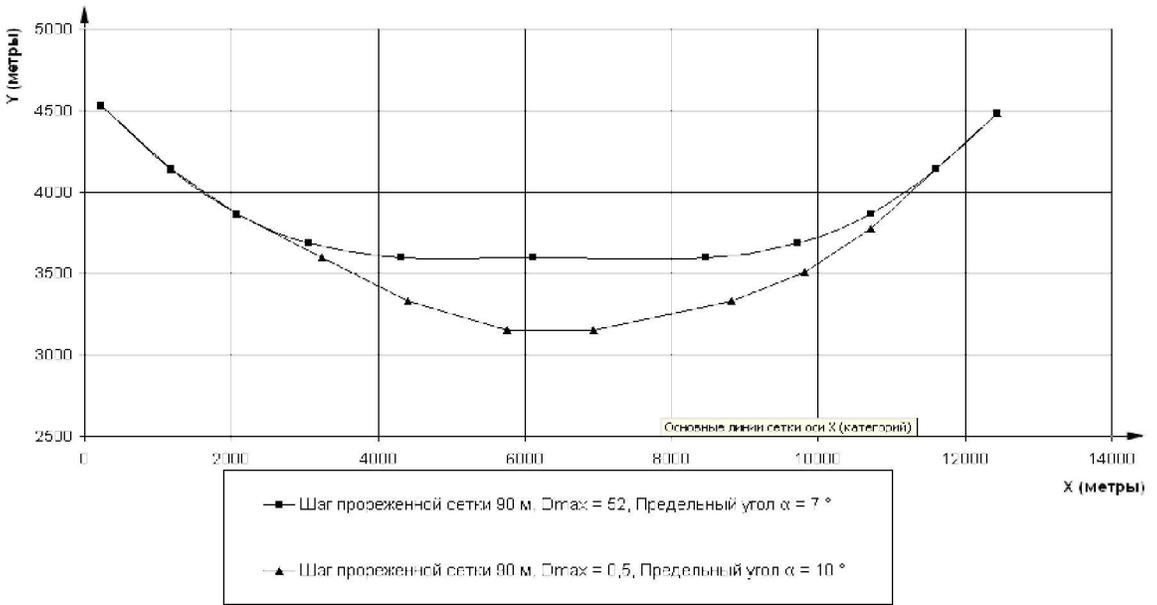


Рис. 4. Сравнительные схемы траекторий

тем, такие вычисления требуют больших вычислительных затрат. Поэтому была разработана модификация алгоритма, в которой ошибка навигации вычисляется непосредственно при проходе алгоритма Дейкстры, т.е. лишь для некоторых узлов сетки. Чтобы исключить повторные вычисления $D(v)$ в уже посчитанных узлах, значения этих вычислений необходимо сохранять в оперативной памяти. Узлы с уже посчитанными значениями могут быть достаточно сильно разбросаны по сетке, и для их хранения предложено использовать механизм квадродеревьев [4], который позволяет осуществлять поиск и добавление элементов за время, примерно пропорциональное глубине дерева.

Вся область Q (как правило, прямоугольная), на котором задано поле высот F , разбивается на 4 потока, с каждым из которых связан список вычисляемых значений ошибок навигации для v , пространственные координаты (x, y) которого принадлежат данному потоку. Вновь вычисляемое значение ошибки добавляется в соответствующий список. Если для какого-либо потока длина списка становится слишком большой, то этот поток делится на четыре части, и его список элементов также распределяется между этими частями. Такое заполнение и деление квадродерева продолжается до завершения процедуры поиска (рис. 5).

Параллельный алгоритм поиска траектории движения ЛА действует следующим образом: один из вычислительных узлов (ВУ) комплекса назначается ведущим. На нем выполняется алгоритм Дейкстры, а также хранится квадродерево вычисленных значений ошибок навигации. При выполнении очередного шага алгоритма Дейкстры ведущий ВУ определяет список вершин графа, для которых необходимо вычислить $D(v)$, и назначает свободным ведомым вычислительным узлам соответствующие задания. Ведомые ВУ передают результаты счета ведущему ВУ, который записывает их в квадродерево (рис. 6).

На рис. 7 изображена зависимость скорости решения задачи поиска от количества вычислительных узлов. При относительно малом количестве узлов зависимость близка к линейной. Однако скорость выполнения алгоритма Дейкстры на ведущем узле является примерно постоянной, что сказывается на скорости решения всей задачи при увеличении количества узлов.

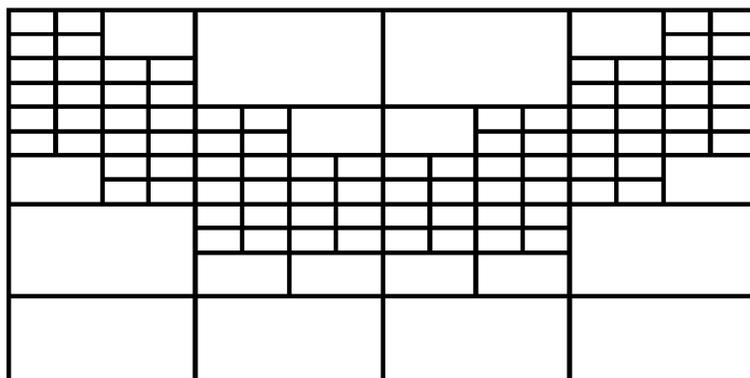


Рис. 5. Квадродерево вычисленных значений



Рис. 6. Схема параллельного алгоритма

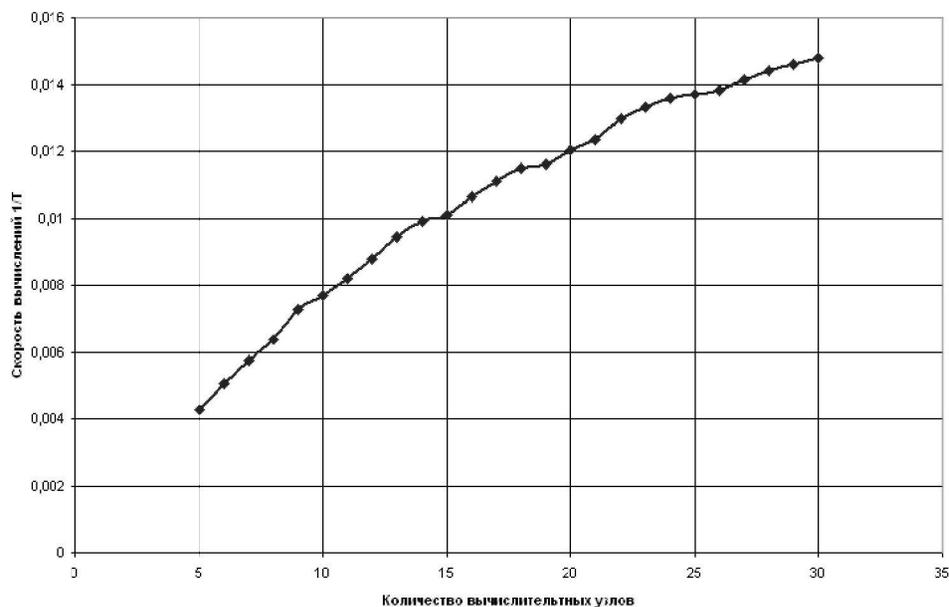


Рис. 7. Зависимость скорости вычисления от количества вычислительных узлов

3. Заключение

Планируется продолжить работу по поиску траектории движения ЛА с различными критериями оптимальности, проходящей через области с минимальной ошибкой навигации.

Статья рекомендована к печати программным комитетом международной научной конференции "Параллельные вычислительные технологии 2009" <http://agora.guru.ru/pavt>.

Работа поддержана программой 14 Президиума РАН, раздел 1 и программой 29 Президиума РАН, раздел 7.

Литература

1. Бердышев, В.И. Экстремальные задачи и модели навигации по геофизическим полям / В. И. Бердышев, В. Б. Костоусов // Научное издание. Екатеринбург: УрО РАН, 2007. - 270 с.
2. Бердышев, В.И. Задача навигации. Аппроксимация поля высот / / В. И. Бердышев, Я. В. Малыгин // Параллельные вычислительные технологии (ПАВТ' 2007): тр. Международ. науч. конф. (Челябинск, 29 января - 2 февраля 2007г.). - Челябинск: Изд-во ЮУрГУ, 2007. - Т. 1. - С. 28 - 30.
3. Ахо А. Построение и анализ вычислительных алгоритмов / А. Ахо, Дж. Хопкрофт, Дж. Ульман. - М.: Мир, 1979. - 536 с.
4. Препарата, Ф. Вычислительная геометрия: введение / Ф. Препарата, М. Шеймос. - М.: Мир, 1989. - 478 с.

Отдел теории приближения функций,
Институт Математики и Механики УрО РАН
malygin@imm.uran.ru

Поступила в редакцию 16 августа 2009 г.