

С. РАХМАТИ, К. КИАНФАР, А. А. КАЛАТ

ОПРЕДЕЛЕНИЕ МЕСТОПОЛОЖЕНИЯ ПО ДАННЫМ ГРАВИТАЦИОННОГО ГРАДИЕНТОМЕТРА С ПРИМЕНЕНИЕМ НЕЙРО-НЕЧЕТКОГО МОДЕЛИРОВАНИЯ

В статье представлен новый метод определения местоположения по данным гравитационного градиентометра с помощью нейро-нечеткого моделирования, который также может использоваться для коррекции показаний инерциальной навигационной системы с применением фильтра Калмана. Поскольку изменения гравитационного градиента главным образом связаны с особенностями рельефа, данные о высоте используются для моделирования гравитационных градиентов. Чтобы продемонстрировать потенциальные возможности метода, исследуются два типа рельефа – изрезанный и гладкий, в обоих случаях метод обеспечивает высокую точность навигации. Также анализируется применимость метода при полете на разных высотах.

Ключевые слова: нейро-нечеткое моделирование, гравитационный градиент, изрезанный и гладкий рельеф, фильтр Калмана

1. Введение

Прецизионная навигация нашла широкое применение во многих военных и коммерческих приложениях. В настоящее время GPS служит эффективным дополнением инерциальной навигационной системы (ИНС), обеспечивая субметровую точность позиционирования. Однако позиционирование по GPS возможно только при использовании по крайней мере четырех спутников [1]; кроме того, сигналы GPS подвержены интерференции и преднамеренным помехам. Существуют различные методы для осуществления навигации в районах с ослабленным GPS-приемом, в частности применение ИНС, метод сравнения с рельефом местности (Terrain Referenced Navigation – TRN), отслеживание рельефа местности (Terrain Contour Matching – TERCOM), и навигация на основе изображений [2, 3].

Еще один метод навигации, который в последнее время приобретает все большую популярность, состоит в использовании гравитационных градиентометров (ГГ) – приборов, измеряющих пространственные производные силы тяжести. В настоящее время точность ГГ значительно возросла за счет улучшения качества акселерометров. Исследования, посвященные изучению возможности использования ГГ в навигации, основаны на алгоритмах

Рахмати Саид. Аспирант факультет электротехники Университета г. Бирджанд (Иран).

Кианфар Кавех. Старший научный сотрудник Университета имени имама Хусейна. Исследовательский центр Гадр (Тегеран, Иран).

Калат Али Акбарзаде. Доцент факультета электротехники Технологического университета г. Шахруд (Иран).

Научный редактор перевода д.т.н. Д.А.Кошаев.

сопоставления с картой [4, 5]. В этих алгоритмах обычно используются функции плотности распределения вероятности, однако, если наблюдаются значительные изменения гравитационного градиента, будет получено множество возможных значений местоположения объекта [4].

В статье предлагаются метод определения местоположения и коррекции данных ИНС за счет обработки данных ГГ с применением нейро-нечеткого моделирования. Рассматривается процесс расчета компонентов гравитационных градиентов. Данные о высоте местности используются для моделирования гравитационного градиента на окружающей местности, а градиенты, вызванные подземной гравитационной аномалией, считаются систематической погрешностью. Чтобы продемонстрировать потенциальные возможности метода, он апробируется для изрезанного и гладкого рельефа и на разных высотах. Указаны ограничения на применение предлагаемого метода.

2. Карты гравитационного градиента

Модель гравитационного градиента служит основой для достижения поставленной задачи. В соответствии с [6] гравитационное поле содержит два основных компонента, длинноволновый (~400 км) и коротковолновый (25–100 км). Длинноволновый компонент вызван аномалиями подземных пластов и может быть рассчитан с помощью глобальных моделей геопотенциала. Коротковолновый компонент возникает вследствие локальных эффектов. Для их расчета можно использовать различные методы, например метод конечных элементов или быстрое преобразование Фурье.

2.1. Гравитационный градиент

Гравитационный потенциал V в точке (x, y, z) вследствие распределения масс с функцией плотности $\rho(x', y', z')$ и объемной массой v' равен

$$V = G \iiint_{v'} \frac{\rho}{r} dv', \quad (1)$$

где $r = \sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2}$ – расстояние между точкой элемента (x', y', z') и точкой расчета (x, y, z) , G – гравитационная постоянная Ньютона, $dv' = dx' dy' dz'$ – элементарный объем, ρ – массовая плотность. Вектор силы гравитации F равен

$$F = \nabla V = \left[\frac{\partial V}{\partial x}, \frac{\partial V}{\partial y}, \frac{\partial V}{\partial z} \right]^T. \quad (2)$$

Гравитационный градиент является тензором гравитационного потенциала второго порядка

$$\|V_{ij}\| = \nabla F = \begin{bmatrix} \frac{\partial^2 V}{\partial x^2} & \frac{\partial^2 V}{\partial x \partial y} & \frac{\partial^2 V}{\partial x \partial z} \\ \frac{\partial^2 V}{\partial y \partial x} & \frac{\partial^2 V}{\partial y^2} & \frac{\partial^2 V}{\partial y \partial z} \\ \frac{\partial^2 V}{\partial z \partial x} & \frac{\partial^2 V}{\partial z \partial y} & \frac{\partial^2 V}{\partial z^2} \end{bmatrix} = \begin{bmatrix} V_{xx} & V_{xy} & V_{xz} \\ V_{yx} & V_{yy} & V_{yz} \\ V_{zx} & V_{zy} & V_{zz} \end{bmatrix}, i, j = 1, 2, 3. \quad (3)$$

Градиент симметричен. След тензора удовлетворяет уравнению Пуассона:

$\nabla^2 V = -4\pi G\rho$. При нулевой плотности (в свободном воздухе) это уравнение преобразуется в уравнение Лапласа. Таким образом, из девяти компонентов тензора только пять являются независимыми.

Сила тяжести определяется как комбинация гравитационной и центробежной сил.

Возмущающий потенциал T определяется как разница между реальным потенциалом силы тяжести W (который является суммой гравитационного потенциала V и центробежного потенциала) и нормальным потенциалом U . Эллипсоид вращения используется для аппроксимации поверхности Земли и считается эквипотенциальной поверхностью поля нормальной силы тяжести с потенциалом U . Поскольку для определенного эллипсоида (напр. WGS84) известны нормальный потенциал силы тяжести и его градиенты, требуется только рассчитать T_{ij} . Градиенты гравитационных возмущений определяются как

$$\|T_{ij}\| = \begin{bmatrix} \frac{\partial^2 T}{\partial x^2} & \frac{\partial^2 T}{\partial x \partial y} & \frac{\partial^2 T}{\partial x \partial z} \\ \frac{\partial^2 T}{\partial y \partial x} & \frac{\partial^2 T}{\partial y^2} & \frac{\partial^2 T}{\partial y \partial z} \\ \frac{\partial^2 T}{\partial z \partial x} & \frac{\partial^2 T}{\partial z \partial y} & \frac{\partial^2 T}{\partial z^2} \end{bmatrix} = \begin{bmatrix} T_{xx} & T_{xy} & T_{xz} \\ T_{yx} & T_{yy} & T_{yz} \\ T_{zx} & T_{zy} & T_{zz} \end{bmatrix}, \quad i, j = 1, 2, 3. \quad (4)$$

Данный тензор, как и гравитационный тензор, имеет пять независимых компонентов.

Обычно для моделирования возмущающего градиента в ограниченном районе используются гравитационная аномалия и данные о высоте места. Поскольку большинство изменений гравитационных градиентов связаны с местностью, градиенты обычно моделируются с помощью данных о высоте [7]. Однако, если при расчетах используется большой объем таких данных, реализация обычных алгоритмов на основе метода конечных элементов требует много времени. Поэтому для корректного расчета коротковолнового компонента гравитационного поля используется частотный метод Паркера [5, 7], согласно которому

$$T_{ij}(p_1, p_2) = 2\pi G\rho FFT^{-1} \left(\mu_{ij} g^{-2\pi z f_{p_1, p_2}} \sum_{n=1}^{\infty} \frac{1}{n!} (2\pi f_{p_1, p_2})^{n-2} FFT(h^n)_{p_1, p_2} \right), \quad (5)$$

где $p_1 = 0, \dots, M_1 - 1$; $p_2 = 0, \dots, M_2 - 1$; M_1 и M_2 – количество используемых значений координат в направлениях x и y , h – высота, $f_{p_1, p_2} = \sqrt{f_{1p_1}^2 + f_{2p_2}^2}$, f_1 и f_2 – пространственные частоты, соответствующие координатам x и y :

$$f_{1p_1} = \frac{p_1}{\Delta x' M_1}, \quad f_{2p_2} = \frac{p_2}{\Delta y' M_2}, \quad \text{для } p_1 = 0, \dots, \frac{M_1}{2} - 1, \quad p_2 = 0, \dots, \frac{M_2}{2} - 1,$$

$$f_{1p_1} = \frac{p_1 - M_1}{\Delta x' M_1}, \quad f_{2p_2} = \frac{p_2 - M_2}{\Delta y' M_2}, \quad \text{для } p_1 = \frac{M_1}{2}, \dots, M_1 - 1, \quad p_2 = \frac{M_2}{2}, \dots, M_2 - 1,$$

где $\Delta x'$ и $\Delta y'$ – дискретность значений координат.

Коэффициенты μ_{ij} , которые связывают гравитационный потенциал и его вторые производные в частотной области, определяются как [8]:

$$\begin{aligned}\mu_{11} &= -(2\pi)^2 f_{1p1}^2 & \mu_{12} &= -(2\pi)^2 f_{1p1} f_{2p2} & \mu_{13} &= -i(2\pi)^2 f_{1p1} f_{p1,p2} \\ \mu_{22} &= -(2\pi)^2 f_{2p2}^2 & \mu_{23} &= -i(2\pi)^2 f_{2p2} f_{p1,p2} & \mu_{33} &= (2\pi)^2 f_{p1,p2}^2\end{aligned}\quad (6)$$

2.2 Моделированные карты гравитационных градиентов

Смоделированная карта гравитационного градиента для компонента T_{zz} на высоте 3000 м над эллипсоидом для изрезанного и гладкого рельефа представлена на рис. 1.

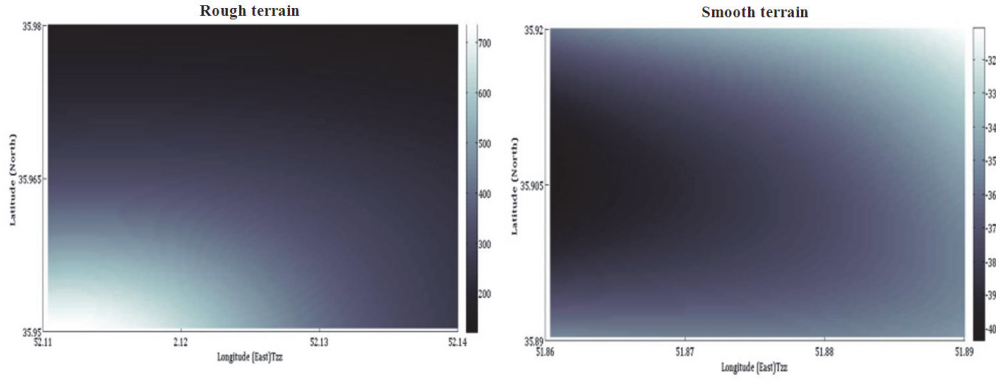


Рис. 1. Карта компонента T_{zz} гравитационного градиента для изрезанного (rough) и гладкого (smooth) рельефа на высоте 3000 м над эллипсоидом

3. Нейро-нечеткое моделирование

Определение местоположения предлагаемым методом производится путем подгонки модели гравитационного градиента к измеренным значениям градиента, которая выполняется в горизонтальной плоскости. При этом требуется смоделировать работу навигационной системы в соответствии с набором данных на ее входе и выходе [9, 10]. Поскольку изменения гравитационного градиента представляют собой очень сложный процесс, моделирование также будет непростым, поэтому требуется модель высокого порядка. В настоящее время существует много методов для моделирования систем. Аналитические методы стараются воспроизвести точную модель системы. Но для сложных систем они неприменимы. Иранский ученый, профессор Заде предложил теорию нечетких множеств для решения проблемы сложности и неопределенности системы [11], но в рамках этой теории не разработан конкретный метод для определения количества требуемых нечетких правил и соответствующих им функций принадлежности. Предлагаются также и другие методы, такие как моделирование на основе нейронных сетей [12, 13], они достигают высокой точности в части аппроксимирования функций и идентификации систем, но имеют некоторые недостатки: попадание в локальный минимум [14] и малая скорость сходимости [15]. В последнее время нейро-нечеткое моделирование становится все более популярным. Нечеткие и нейронные методики обычно реализуются в два этапа. Поэтому нейро-нечеткое моделирование использует преимущества как нечеткой логики, так и нейронных сетей.

В данной статье предлагается новый метод создания нейро-нечеткой сети для определения местоположения. В общем, процесс создания такой сети включает два этапа: идентификация структуры и идентификация параметров. На первом этапе производится определение нечетких правил или их выработка на основе имеющихся реализаций входных и выходных данных. Чтобы достичь высокой точности, на втором этапе параметры нечетких правил, настраиваются в процессе обучения. Разработан гибридный алгоритм обучения с уменьшенной зоной поиска и повышенной скоростью сходимости, который объединяет алгоритм оценивания по методу наименьших квадратов и модифицированный алгоритм прыжков лягушки (modified shuffled frog leaping algorithm, MSFLA). В следующем разделе дается обзор нейро-нечеткого алгоритма моделирования навигационной системы.

3.1 Обзор предлагаемого метода

Как уже отмечалось, цель предлагаемого метода заключается в том, чтобы по набору данных, содержащему n входов и m выходов, получить модель навигационной системы, т.е. определить функцию $f: R^n \rightarrow R^m$. Здесь пять независимых компонентов тензора гравитационного градиента считаются входами ($n = 5$), а широта и долгота – выходами ($m = 2$). Схема метода представлена на рис. 2.

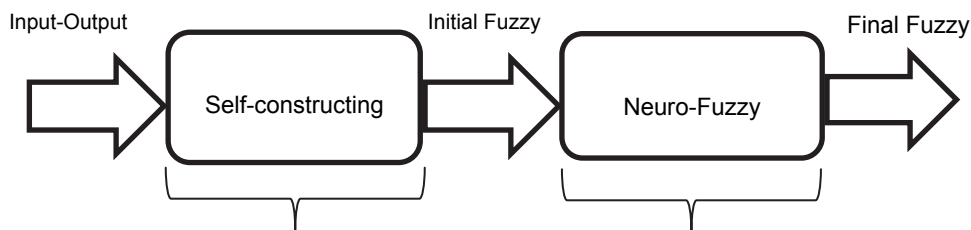


Рис. 2. Общая схема нейро-нечеткого алгоритма (надписи слева направо и сверху вниз):
Вход – выход, Базовые нечеткие правила, Финальные нечеткие правила, Самопостроение,
Нейро-нечеткие правила

В нейронно-нечеткой схеме Takagi-Sugeno-Kang реализуется нечеткая модель с j -м нечетким правилом вида (7), в которой w_j^M – вес j -го нечеткого правила, связанного с M -м выходом, $\mu_{ij}(x_i)$ – гауссовская функция принадлежности, для j -го нечеткого правила, связанного с x_i со средним c_{ij} и дисперсией σ_{ij} . Коэффициенты α_{kj}^M ($k = 0, \dots, n$ and $M = 1, \dots, m$) постоянны [16].

$$\text{Если } x_1 = \mu_{1j}(x_1) \text{ И } x_2 = \mu_{2j}(x_2) \text{ И } \dots \text{ И } x_n = \mu_{nj}(x_n), \text{ то } y^M = \quad (7)$$

$$w_j^M = \alpha_{0j}^M + \alpha_{1j}^M x_1 + \dots + \alpha_{nj}^M x_n, \quad (8)$$

$$\mu_{ij}(x_i) = \exp \left[- \left(\frac{x_i - c_{ij}}{\sigma_{ij}} \right)^2 \right]. \quad (9)$$

Выход сети равен взвешенной сумме правил:

$$Q^m = \frac{\sum_{j=1}^k \exp\left(\sum_{i=1}^n -\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right) w_j^M}{\sum_{j=1}^k \exp\left(\sum_{i=1}^n -\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right)}, M = 1, \dots, m. \quad (10)$$

В уравнениях (8) и (10) к параметрам, настраиваемым в системе и влияющим на качество работы и точность определения местоположения, относятся c_{ij} , σ_{ij} и w_j^M , где $k = 0, \dots, n$ и $M = 1, \dots, m$. Эти параметры оптимизируются с помощью гибридного алгоритма обучения сети (см 3.3).

3.2 Идентификация структуры

На этапе идентификации структуры нечеткие правила формируются для набора данных входа – выхода. Эти нечеткие правила образуют базовую модель неизвестной системы. К настоящему моменту было предложено множество методов идентификации структуры, каждый из которых имеет свои преимущества и недостатки. Что касается объема и размерности входных данных, то для идентификации структуры используется алгоритм вычитающей кластеризации [17], который при малых вычислительных затратах способен автоматически находить центр кластеров исходной базы данных. После классификации данных из каждого кластера создаются правила Takagi-Sugeno-Kang «если-то», и затем формируется база правил.

Алгоритм вычитающей кластеризации основан на кластеризации по плотности, используемой для генерирования правил. В соответствии с идеей вычитающей кластеризации каждая точка данных считается потенциальным центром кластера. Возможность каждой точки данных быть центром кластера n рассчитывается по плотности окружающих точек данных. Каждый кластер содержит определенное количество обучающих образов и характеризуется произведением n одномерных гауссовских плотностей.

Рассмотрим группу из L точек данных $(x_1, x_2, \dots, x_L)$, где x_i – вектор в n -мерном пространстве входов. Для всех обучающих данных рассчитывается плотность образца, чтобы решить, нужно ли создать новый кластер или заменить существующий. После обработки всех обучающих образцов мы получаем набор кластеров. Центр кластера определяет нечеткое правило, на этом этап идентификации структуры заканчивается. Ниже этот процесс описан более подробно.

Пусть k – количество существующих кластеров, плотность i -го кластера обозначается D_{ci}^t при появлении образа t . Изначально $k = 0$. Таким образом, вначале кластеров нет, и первые данные считаются центром первого кластера x_{c1} с плотностью $D_{c1}^1 = D_1^1 = 1$, r_a – радиус кластера, который можно определить экспериментально. Для образца t , т.е. x_t , $1 \leq t \leq N$ плотность центров существующих кластеров модифицируется следующим образом:

$$D_{ci}^t = D_{ci}^{t-1} + \exp\left[-\frac{\|x_{ci} - x_t\|^2}{(r_a/2)^2}\right] \quad i = 1, 2, \dots, k. \quad (11)$$

Плотность x_t , т.е. D_t^t , рассчитывается как

$$D_t^t = \sum_{j=1}^t \exp \left[\frac{\|x_t - x_j\|^2}{\left(\frac{r_a}{2}\right)^2} \right] \quad (12)$$

Если $D_t^t > \{D_{ci}^t, i = 1, 2, \dots, k\}$, новый кластер не будет сформирован и x_{cz} будет центром кластера, ближайшего к x_t с текущей плотностью D_{cz}^t . Значение x_t заменяет x_{cz} в качестве центра нового кластера, если $\frac{\delta_{min}}{r_a} + \frac{D_{cz}^t}{D_t^t} < 1$, где $\delta_{min} = \|x_{cz} - x_t\| = \min\{\|x_{ci} - x_t\|\}$. Тогда количество нечетких правил останется неизменным. В противном случае x_t становится центром нового кластера, и к количеству существующих кластеров, т.е. нечетких правил, добавляется еще один.

Предусматривается также удаление кластерного центра. Допустим, что существует s кластерных центров. Попробуем найти два кластерных центра x_{ca} и x_{cb} ($D_{ca}^t < D_{cb}^t$) с наименьшим расстоянием и зададим d_{min} следующим образом:

$$d_{min} = \|x_{ca} - x_{cb}\| = \min \left\{ \|x_{ci} - x_{cj}\| \quad \begin{matrix} i = 1, 2, \dots, s-1, \\ j = 1, 2, \dots, s \end{matrix} \right\} \quad (13)$$

Если $D_{max} = \max\{D_{ci}^t, i = 1, 2, \dots, s\}$ удовлетворяет следующей формуле:

$$\frac{d_{min}}{r_a} + \frac{D_{ca}^t}{D_{max}} < 1 \quad (14)$$

то x_{ca} удаляется из базы кластерных центров. В противном случае количество правил не меняется.

3.3 Идентификация параметров

После создания нейронной сети производится оценка ее параметров с помощью гибридного обучающего алгоритма, чтобы улучшить точность системы. В алгоритме для определения оптимальных параметров α_{kj}^M ($k = 0, \dots, n$ и $M = 1, \dots, m$) используется оценка по методу наименьших квадратов, а для оптимизации параметров σ_{ij} и c_{ij} – модифицированный алгоритм прыжков лягушки (MSFLA).

Модифицированный алгоритм прыжков лягушки (MSFLA)

Эволюционные алгоритмы типа MSFLA представляют собой методы случайного поиска, которые имитируют естественную биологическую эволюцию группы лягушек или их социальное поведение. К настоящему моменту алгоритм MSFLA прошел испытания на нескольких комбинаторных проблемах и подтвердил свою эффективность [18]. Алгоритм включает как детерминированные, так и случайные элементы стратегии для поиска оптимального решения. Детерминированная стратегия позволяет алгоритму MSFLA

воспользоваться предыдущими данными, чтобы производить эффективный поиск. Случайная стратегия обеспечивает гибкость и мощность схемы поиска [19]. В алгоритме MSFLA рассматривается группа лягушек, разбитая на подгруппы– мемеплексы, каждая из которых выполняет свой локальный поиск и несет идеи, на которые могут повлиять идеи других лягушек. Также идеи могут возникать в процессе эволюции. Локальный поиск и комбинирование продолжаются до достижения критериев сходимости.

Применение алгоритма MSFLA при оптимизации нелинейных параметров

В данном исследовании поисковые возможности алгоритма MSFLA используются для оптимизации параметров нейро-нечеткой сети (т.е. параметров σ_{ij} и c_{ij}) для определения местоположения в R^n ($n = 5$). Сначала формируется начальная популяция, состоящая из $F = MN$ лягушек, где M – количество мемеплексов, а N – количество лягушек в каждом мемеплексе. Каждая лягушка, т.е. вектор переменных, будет иметь $p = 2kn$ параметров, т.е. $U(i) = (U_i^1, U_i^2, \dots, U_i^{2kn})$ ($i = 1, \dots, F$), где k – количество полученных нечетких правил. На следующем этапе лягушки сортируются по пригодности в убывающем порядке f_i ($i = 1, \dots, F$), т.е. лучшие лягушки имеют наименьшую пригодность. Критерием пригодности считается средний корень из суммы квадратов ошибок положения. Затем множество делится на M мемеплексов $Y_1, Y_2, \dots, Y_M$ таким образом, что первая лягушка попадает в первый мемеплекс, вторая – во второй, и наконец M -я лягушка попадает в M -й мемеплекс. Затем процесс продолжается: $(M+1)$ -я лягушка попадает в первый мемеплекс и т.д. Данный процесс можно описать следующим выражением

$$Y_l = \begin{cases} U(j)^l, f(j)^l | U(j)^l = U(l + m(j-1)), f(j)^l \\ = f(l + m(j-1)), j = 1, \dots, N \end{cases}; \quad (15)$$

$$l = 1, \dots, M$$

После разделения лягушек на M мемеплексов, каждый мемеплекс эволюционирует T раз. В соответствии с данным алгоритмом меметические эволюции внутри каждого мемеплекса происходят следующим образом ([18]). Допустим, что P_w и P_b – положения худшей и лучшей лягушки в каждом мемеплексе соответственно. Положение лучшей лягушки в популяции обозначается P_g . В процессе эволюции P_w прыгает по направлению к P_b . Но из-за неидеального восприятия стратегии и неточного действия худшая лягушка не может прыгнуть точно к заданной позиции. С учетом этих неопределенностей и того факта, что худшая лягушка может перепрыгнуть через лучшую, модифицированное правило прыжков лягушки выглядит следующим образом ([19]):

$$D_i = rand() \times c \times (P_b - P_w) + W, \quad (16)$$

$$W = [r_1 w_{1max}, r_2 w_{2max}, \dots, r_p w_{pmax}], \quad (17)$$

$$P_w^{new} = \begin{cases} P_w + D & \text{if } \|D\| \leq D_{max} \\ P_w + \frac{D}{\sqrt{D}^T D} D_{max} & \text{if } \|D\| > D_{max} \end{cases}, \quad (18)$$

где $rand()$ – случайное число между 0 и 1; c – количество, которое принадлежит диапазону $[1 \ 2]$; $r_i (1 < i < p)$ – случайные числа между -1 и 1; D_{max} –

максимальное разрешенное изменение в положении худшей лягушки; $W_{i, \max}$ – максимальное разрешенное восприятие в i -м p -размерной области поиска [19]. На Рис. 3 показан закон улучшенных прыжков лягушки для двухмерной задачи. Если в процессе прыжков получается лучшее решение, оно закладывается в худшую лягушку.

В противном случае уравнения (16)-(18) повторяются относительно лучшей лягушки в популяции (т.е. P_g заменяет P_b). Если положение худшей лягушки не улучшается, случайным образом будет сгенерирована новая лягушка, которая ее заменит. После определенного количества T меметических эволюционных шагов в каждом мемеплексе вся популяция перемешивается для проведения глобального информационного обмена. Локальные эволюции и перемешивание продолжаются до достижения критериев сходимости.

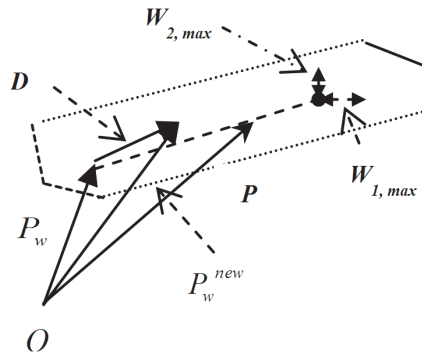


Рис. 3. Правило прыгающих лягушек

Алгоритм оценки по методу наименьших квадратов

Допустим $(\vec{p}_v, \vec{q}_v)$ – v -й обучающий образец данных, где $\vec{p}_v = [p_{1v}, \dots, p_{nv}]^T$ – входной вектор, состоящий из независимых компонент гравитационного градиента, $\vec{q}_v = [q_{1v}, q_{2v}]^T$ – выходной вектор – горизонтальные координаты. Очевидно, что мы хотим минимизировать $|\vec{q}_v - \vec{o}_v|$ для v -го обучающего образца, здесь $\vec{o}_v$ равен $[o_v^1, o_v^2]^T$. Для L образцов получим L уравнений вида (10), таким образом, мы хотим минимизировать $J(X) = \|Q - \Phi X\|$, где Q, Φ, X – матрицы $L \times 2$, $L \times ((n+1) \times k)$ и $((n+1) \times k) \times 2$, соответственно, которые описываются следующим образом:

$$Q = [\vec{q}_1, \vec{q}_2, \dots, \vec{q}_L]^T \quad (19)$$

$$\Phi = \begin{bmatrix} o_1^{(2)}(1) \cdot [1 \vec{p}_1^T] & \dots & o_k^{(2)}(1) \cdot [1 \vec{p}_1^T] \\ \vdots & \ddots & \vdots \\ o_1^{(2)}(L) \cdot [1 \vec{p}_L^T] & \dots & o_k^{(2)}(L) \cdot [1 \vec{p}_L^T] \end{bmatrix} \quad (20)$$

$$X = \begin{bmatrix} \alpha_{01}^1 & \dots & \alpha_{n1}^1 & \dots & \alpha_{0k}^1 & \dots & \alpha_{nk}^1 \\ \alpha_{01}^2 & \dots & \alpha_{n1}^2 & \dots & \alpha_{0k}^2 & \dots & \alpha_{nk}^2 \end{bmatrix}^T \quad (21)$$

В (20), $o_j^{(2)}(r) \{j = 1, \dots, k; r = 1, \dots, L\}$ – нормализованный выход j -го нечеткого правила, описываемый как

$$o_j^{(2)} = \frac{\exp\left(\sum_{i=1}^n -\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right)}{\sum_{j=1}^k \exp\left(\sum_{i=1}^n -\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}\right)} \quad j = 1, \dots, k. \quad (22)$$

Применяя метод наименьших квадратов к полному множеству обучающих данных, можно по следующему уравнению рассчитать веса оптимизированных выходных уровней

$$X = [\Phi^T \Phi]^{-1} \Phi^T Q \quad (23)$$

4. Экспериментальные результаты

4.1 Нейро-нечеткая система позиционирования

Точность разрабатываемой системы позиционирования была проанализирована в ходе двух численных экспериментов. На первом этапе рассматривалась работа системы на гладком и изрезанном рельефе до и после оптимизации параметров системы при полном отсутствии шума в измерениях ГГ. Топографические карты гладких и изрезанных рельефов показаны на рис. 4.

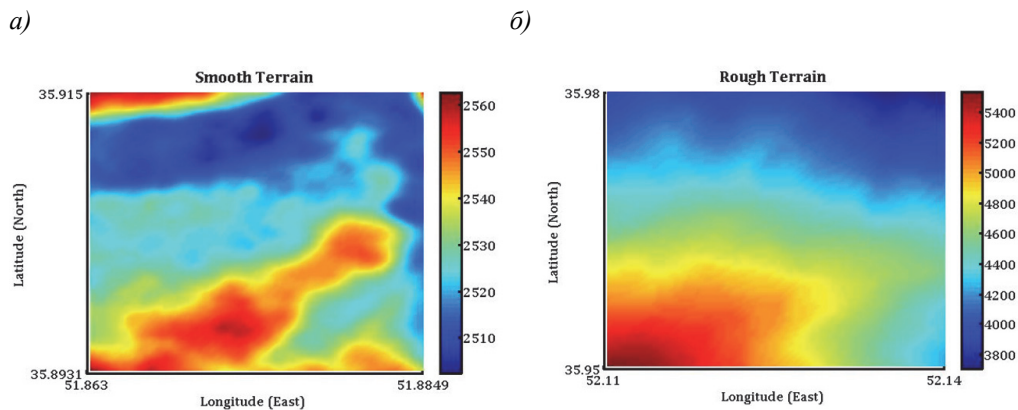


Рис. 4. Колебания высоты для гладкого (а) и изрезанного (б) рельефа

Чтобы снизить вычислительные затраты, для моделирования выбраны ограниченные области, но предлагаемый метод можно использовать и для более широких областей. В рассматриваемых примерах в районах с гладким рельефом перепад высот между самой низкой и самой высокой точкой равен 49 м, а в районах с изрезанным рельефом – 1676 м. На втором этапе проводится анализ устойчивости системы к шуму измерений. Во всех случаях летательный аппарат с ГГ летит со скоростью 50 м/с. Чтобы оценить качество работы системы, использовалось среднее значение и среднеквадратическое отклонение (СКО) ошибок горизонтальных координат (24), а также 50%-ное круговое отклонение (т.е. медиана корня из суммы квадратов ошибок). Высота полета отсчитывалась относительно средней высоты местности.

$$RSSPE = \sqrt{error_{latitude}^2 + error_{longitude}^2} \quad (24)$$

Эксперимент 1

В этом разделе продемонстрирована работа системы при полете над гладкой и изрезанной местностью на высоте 3000 и 12000 м выше средней высоты

района. К сигналам ГГ не добавлялось никакого шума. Отсчеты ГГ гравитационных градиентов имеют частоту 1 Гц. Корень из суммы квадратов ошибок положения в двух районах с гладким и изрезанным рельефом на высоте 3000 м до оптимизации параметров показан на рис. 5. В табл. 1 приведены количество нечетких правил, сформированных по гравитационным данным, и точностные характеристики работы системы.

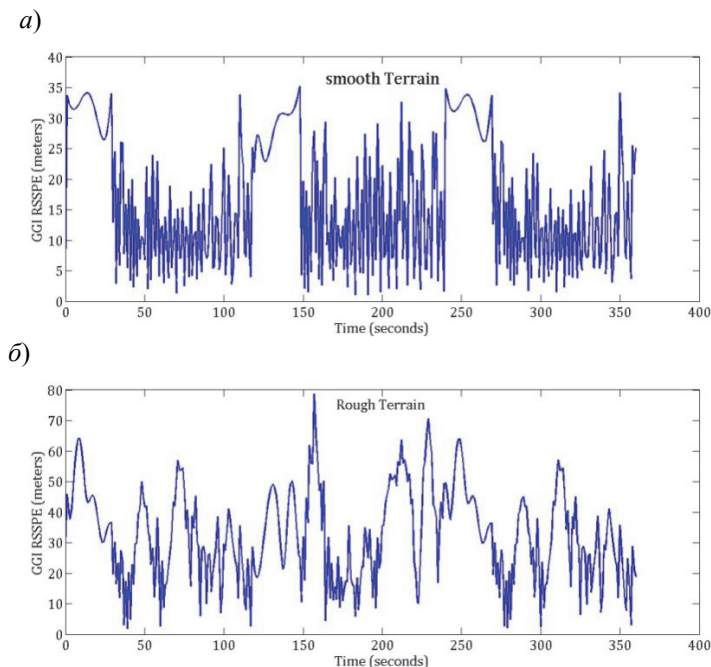


Рис. 5. Точность определения горизонтальных координат на высоте 3000 м без оптимизации параметров нейро-нечеткой сети: а – гладкий рельеф, б – изрезанный рельеф

Т а б л и ц а 1

Характеристики системы позиционирования на различных высотах полета до и после оптимизации сети

1	2	3	4	5	6	7	8	9
Высота (км)	12				3			
Рельеф	Гладкий		Изрезанный		Гладкий		Изрезанный	
Оптимизация параметров	Нет	Да	Нет	Да	Нет	Да	Нет	Да
Кол-во сформированных правил	7	7	9	9	6	6	5	5
СКО корня из суммы квадратов ошибок положения (м)	19,18	8,926	25,23	9,670	10,49	9,469	15,28	9,986
Средний корень из суммы квадратов ошибок положения (м)	40,677	19,845	43,99	25,507	16,503	14,467	32,22	20,202
Медиана (м)	38,207	16,538	55,486	21,256	17,314	16,223	22,92	16,835

Далее мы оптимизируем параметры разработанной нейро-нечеткой системы позиционирования с помощью нашего гибридного обучающего алгоритма с заранее заданным количеством итераций. Точностные характеристики системы, показаны в столбцах 3,5,7 и 9 табл. 1

Эксперимент 2

В эксперименте исследуется устойчивость улучшенной системы позиционирования к шумам датчиков. В процессе полета данные ГГ подвергаются искажениям за счет добавления шума, который, по утверждению производителей градиентометров, является гауссовским белым шумом с нулевым средним [5]. В соответствии с [4] современные градиентометры имеют СКО шума на уровне 1 Э, что уже продемонстрировано в испытаниях, а на будущее строится оптимистический прогноз – уже через 10 лет шум снизится до 0,1 Э. Таким образом, к сигналам ГГ добавляется СКО шума 0,01, 0,1, 0,5 и 1 Э. Численные результаты эксперимента показаны в табл. 2. Следует отметить, что скорость и продолжительность полета сохранились неизменными по сравнению с прошлым экспериментом. Высота полета принималась равной 3000 м. Из таблицы видно, что использование нейро-нечеткой модели системы позиционирования позволяет достичь высокой устойчивости к шуму ГГ в районах, где наблюдаются большие изменения гравитационных градиентов.

Таблица 2

Критерии работы (эксперимент 2)

СКО шума (Э)	1		0,5		0,1		0,01	
Рельеф	Гладкий	Изрезанный	Гладкий	Изрезанный	Гладкий	Изрезанный	Гладкий	Изрезанный
СКО корня из суммы квадратов ошибок положения, м	315.6	23.574	158.6	12.965	30.166	10.008	9.654	10.29
Средний корень из суммы квадратов ошибок положения, м	772.81	39.637	359.49	24.256	72.3127	16.879	18.007	26.245
Медиана, м	348.17	31.633	175.36	22.719	37.854	13.162	12.936	11.947

4.2 Применение системы для коррекции дрейфа ИНС: интегрированная навигация

Очевидно, что значение местоположения, рассчитанное ИНС, имеет тенденцию к дрейфу относительно истинного положения. В данной работе систему позиционирования предлагается использовать в качестве средства коррекции для ИНС. Комплексирование данных от системы коррекции и ИНС выполняется дискретным линейным фильтром Калмана. В уравнениях фильтра Калмана не предусмотрено корреляции между шумами системы и измерений,

оба шума имеют среднее нулевое и гауссово распределение. Согласно [4], с приходом кольцевых лазерных гироскопов, скорость дрейфа ИНС удалось понизить до 0,2 морских миль в час, поэтому выбирается скорость дрейфа 2 км/ч. При моделировании к значениям градиентов на карте и сигналам ГГ добавляется шум с СКО 0,1 и 0,01 Э соответственно. Частота коррекции положения, т.е. частота подачи данных системы позиционирования в фильтр Калмана, составляет 1 Гц. Скорость полета 150 км/ч.

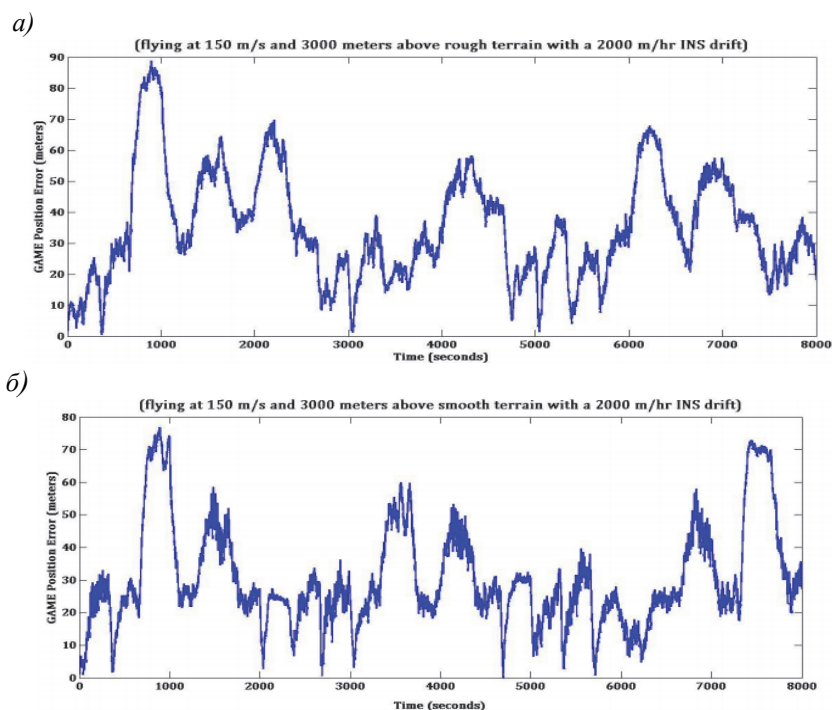


Рис. 6. Ошибка местоположения интегрированной системы (полет со скоростью 150 м/с на высоте 3000 м над изрезанным (а) и гладким (б) рельефом при скорости дрейфа ИНС 2000 м/ч)

Как видно из рисунка, с помощью предлагаемого средства навигационной коррекции удалось решить задачу ограничения скорости дрейфа ИНС во время полета.

5. Выводы

Предложено использование градиентов гравитационного поля Земли для определения местоположения и коррекции ИНС при помощи нейро-нечеткой модели изменений градиентов гравитационного поля. Метод обеспечивает данные о местоположении с приемлемой точностью для пользователя и для коррекции ИНС. Трудность состоит в том, что вычислительные затраты на построение и оптимальную настройку нейро-нечеткой модели изменений гравитационного градиента растут соразмерно площади рассматриваемой области.

ЛИТЕРАТУРА

1. **Bachrach, A., Prentice, S., He, R., and Roy, N.** RANGE—Robust autonomous navigation in GPS-denied environments, *J. Field Robotics*, 2011, vol. 28, pp. 644-666.
2. **Golden, J.P.** Terrain contour matching (TERCOM): a cruise missile guidance aid, in 24th Annual Technical Symposium, 1980, pp. 10-18.
3. **Hagen, O.K.** Terrain navigation principles and application, Geodesi-og Hydrografidagene, Honefoss, Norway, 2005.
4. **DeGregoria, A.** Gravity gradiometry and map matching: an aid to aircraft inertial navigation systems, DTIC Document 2010.
5. **Rogers, M.M.,** An investigation into the feasibility of using a modern gravity gradient instrument for passive aircraft navigation and terrain avoidance, DTIC Document 2009.
6. **Kiamehr, R. and Eshagh, M.** EGMLab, a scientific software for determining the gravity and gradient components from global geopotential models, *Earth Science Informatics*, 2008, vol. 1, pp. 93-103.
7. **Zhu, L.** Gradient modelling with gravity and DEM, PhD Thesis, The Ohio State University, 2007.
8. **Jekeli, C.** Statistical analysis of moving-base gravimetry and gravity gradiometry, The Ohio State University, Report, vol. 446, 2003.
9. **Chi-Yuan, Y., Jeng, W.R., and Shie-Jue, L.** Data-based system modeling using a type-2 fuzzy neural network with a hybrid learning algorithm, *IEEE Transactions on Neural Networks*, 2011, vol. 22, pp. 2296-2309.
10. **Celikyilmaz, A. and Burhan Turksen, I.** Enhanced fuzzy system models with improved fuzzy clustering algorithm, *IEEE Transactions on Fuzzy Systems*, 2008, vol. 16, pp. 779-794.
11. **Zadeh, L.A.,** Fuzzy sets, *Information and Control*, 1965, vol. 8, pp. 338-353.
12. **Cho, S.-Y. and Chow, T.W.S.,** Training multilayer neural networks using fast global learning algorithm - least-squares and penalized optimization methods, *Neurocomputing*, 1999, vol. 25, pp. 115-131.
13. **Poggio, T. and Girosi, F.** Networks for approximation and learning, *Proceedings of the IEEE*, 1990, vol. 78, pp. 1481-1497.
14. **Gori, M. and Tesi, A.** On the problem of local minima in backpropagation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, vol. 14, pp. 76-86.
15. **Bello, M.G.** Enhanced training algorithms, and integrated training/architecture selection for multilayer perceptron networks, *IEEE Transactions on Neural Networks*, 1992, vol. 3, pp. 864-875.
16. **Shie-Jue, L. and Chen-Sen, O.** A neuro-fuzzy system modeling with self-constructing rule generation and hybrid SVD-based learning, *IEEE Transactions on Fuzzy Systems*, 2003, vol. 11, pp. 341-353.
17. **Lina, L. and Yang, Y.** Self-adaptive modeling method based on T-S fuzzy RBF NN and its application, 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), 2010, pp. V4-48-V4-52.
18. **Amiri, B., Fathian, M., and Maroosi, A.** Application of shuffled frog-leaping algorithm on clustering, *The International Journal of Advanced Manufacturing Technology*, 2009, vol. 45, pp. 199-209.
19. **Thai-Hoang, H.** A modified shuffled frog leaping algorithm for optimal tuning of multivariable PID controllers, *IEEE International Conference on Industrial Technology*, 2008, pp. 1-6.

Abstract. This paper proposes a novel method for position determination by using neuro-fuzzy modeling and gravity gradient instrument data, which also can serve as a navigation aid to inertial navigation system using a Kalman filter. Since great majority of changes in gravity gradients are due to terrain, terrain elevation data are just used to model the gravity gradients at test location. To demonstrate the potential performance of this method, two cases including rough and smooth terrain are investigated, and impressive navigation accuracy is produced. Also the suitability of the proposed method for the use in different altitudes is compared.

Материал поступил 13.05.13