

ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ, БИОИНФОРМАТИКА

УДК 004.021

МЕТОД СБОРКИ КОНТИГОВ ГЕНОМНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ИЗ ПАРНЫХ ЧТЕНИЙ С ОШИБКАМИ ВСТАВКИ И УДАЛЕНИЯ НА ОСНОВЕ СОВМЕСТНОГО ПРИМЕНЕНИЯ ГРАФОВ ДЕ БРЁЙНА И ГРАФОВ ПЕРЕКРЫТИЙ

А.В. Александров, С.В. Казаков, А.А. Сергушичев
Научный руководитель – ассистент **Ф.Н. Царев**

Многие современные задачи биологии и медицины требуют знания геномов живых организмов, которые состоят из нескольких нуклеотидных последовательностей молекул дезоксирибонуклеиновой кислоты (ДНК). В связи с этим возникает необходимость в дешевом и быстром методе секвенирования, т.е. определения последовательности нуклеотидов в образце ДНК.

Существующие секвенаторы – устройства для чтения ДНК – не позволяют считать за один раз всю молекулу ДНК. Вместо этого они позволяют читать фрагменты генома небольшой длины. Длина фрагмента может быть разной, она является важным параметром секвенирования – от нее напрямую зависит стоимость секвенирования и время, затрачиваемое на чтение одного фрагмента: чем больше длина считываемого фрагмента, тем выше стоимость чтения и тем дольше это чтение происходит. Современные высокопроизводительные секвенаторы основаны на разных подходах со своими достоинствами и недостатками. Одним из возможных недостатков, с которым необходимо справляться, является наличие ошибок вставки и удаления – в прочитанных фрагментах в некоторых местах возможно удаление или вставка символа, например, одной из распространенных ошибок является добавление или удаление одного или нескольких из серии одинаковых подряд идущих нуклеотидов. Часто для сборки используют парные чтения, для которых известно примерное расстояние между ними. Это позволяет получить больше информации о геноме при сравнительно небольших затратах.

Задачей сборки генома является восстановление последовательности ДНК (ее длина составляет от миллионов до миллиардов нуклеотидов у разных живых существ) на основании информации, полученной в результате секвенирования. Этот процесс делится, как правило, на следующие этапы:

1. исправление ошибок в данных секвенирования;
2. сборка контигов – максимальных непрерывных последовательностей нуклеотидов, которые удалось восстановить;
3. построение скэффолдов – последовательностей контигов, разделенных промежутками, для длины которых найдены верхние и нижние оценки.

Одной из наиболее часто используемых при сборке генома математических моделей является так называемый граф де Брёйна. На его использовании основаны следующие программные средства: Velvet, Allpaths, AbySS, SOAPdenovo, EULER.

Одним из недостатков, которым обладают перечисленные программные средства, является то, что эти программы были оптимизированы для работы с секвенаторами компании Illumina и не поддерживают работу с большим числом ошибок вставки и удаления.

В работе предлагается метод, лишенный указанных недостатков. Построение скэффолдов в настоящей работе не рассматривается.

Алгоритм исправления ошибок состоит из четырех этапов.

1. Подсчет частотной характеристики k -меров. Подсчитывается, сколько раз встречается каждый k -мер в наборе чтений, после чего все k -меры делятся на надежные (те, что встречаются много раз) и ненадежные (те, что встречаются мало раз).
2. Разбиение k -меров на корзины согласно их префиксам.
3. Исправление ошибок в k -мерах одной корзины. Для каждого k -мера перебираются все возможные исправления, которые можно осуществить – на каждой можно заменить, удалить или добавить символ. Максимальное число ошибок, исправляемых внутри одного k -мера, является параметром алгоритма. Внутри используемого при разбиении на корзины префикса ошибки не исправляются, благодаря чему исправленные k -меры лежат в той же корзине, что и исправляемые. Если ненадежный k -мер единственным образом превращается в надежный, найденное исправление записывается.
4. Применение исправлений. Для каждого чтения выписываются все k -меру, содержащиеся в нем. Те, к которым найдены исправления, меняются на соответствующие надежные k -меры. Далее для каждой позиции по правилу консенсуса выбирается символ.

Сборка контигов в предлагаемом методе выполняется в два этапа.

1. Сборка квазиконтигов из чтений геномной последовательности. Квазиконтигами называются последовательности, по длине большие чтений, но не являющихся контигами в смысле невозможности наращивания вправо и влево. Этот этап осуществляется с использованием графа де Брёйна.
2. Сборка контигов из квазиконтигов. Осуществляется с использованием графа перекрытий и метода *Overlap-Layout-Consensus*.

Экспериментальные исследования проводились на библиотеке чтений бактерии *p_stutzeri*, геном которой имеет длину около 4 миллионов нуклеотидов. Данные получены секвенатором Ion Torrent. Всего использовалось 3 библиотеки, одна из которой содержала парные чтения со средним расстоянием между чтениями 1000 нуклеотидов. Общий объем исходных данных составлял 2,9 Гб.

В результате исправления ошибок число ненадежных k -меров уменьшилось с 486 миллиона до 400 миллиона, т.е. примерно на 20%. Сборка квазиконтигов из исправленных чтений собирает квазиконтиги из 14% парных чтений, тогда как сборка квазиконтигов из неисправленных чтений – из 7% парных чтений. В результате было получено 534 контига. Значение метрики N50 – 2616, длина максимального контига – 7621, что на 10% выше, чем у сборщика *AbySS*.

В дальнейшем планируется реализовать алгоритм сборки квазиконтигов с учетом ошибок вставки и замены, а также улучшить алгоритм исправления ошибок.

УДК 004.021

МЕТОД ОЦЕНКИ РАССТОЯНИЯ МЕЖДУ КОНТИГАМИ НА ОСНОВЕ ПРИНЦИПА МАКСИМАЛЬНОГО ПРАВДОПОДОБИЯ

А.А. Ахи, А.А. Сергушичев

Научный руководитель – ассистент Ф.Н. Царев

Многие современные задачи биологии и медицины требуют знания геномов живых организмов, которые состоят из нескольких нуклеотидных последовательностей молекул дезоксирибонуклеиновой кислоты (ДНК). В связи с этим возникает необходимость в дешевом и быстром методе секвенирования, т.е. определения последовательности нуклеотидов в образце ДНК.

Задачей сборки генома является восстановление последовательности ДНК (ее длина составляет от миллионов до миллиардов нуклеотидов у разных живых существ) на основании информации, полученной в результате секвенирования. Задача восстановления

геномной последовательности по ее чтениям, как правило, решается в три этапа: исправление ошибок в данных секвенирования, сборка контигов – непрерывных частей генома, и сборка скэффолдов – наборов упорядоченных контигов. Для построения скэффолдов используют парные чтения, полученные из фрагментов геномной последовательности длиной в несколько тысяч нуклеотидов.

Одним из этапов сборки скэффолдов является оценка расстояний между контигами на основе парных чтений. Для этого сначала выполняется картирование чтений на контиги. Затем на основе результатов картирования происходит вычисление искомого расстояния. Большая часть существующих алгоритмов оценки этих расстояний используют предположение о том, что распределение расстояний между парными чтениями является нормальным. Это приводит к использованию среднего арифметического в качестве наиболее вероятного расстояния, что не всегда является хорошей оценкой. Сложность задачи сборки скэффолдов обуславливается следующими факторами: наличие ошибок, как в чтениях, так и в собранных контигах; неопределенность расстояния между парными чтениями; наличие повторов в геномной последовательности; большой объем обрабатываемых данных.

Для повышения точности нахождения расстояний между контигами в лаборатории «Алгоритмы сборки геномных последовательностей» НИУ ИТМО был разработан метод оценки расстояния между контигами, основанный на принципе максимального правдоподобия. Этот метод, как и многие известные, использует предположение о том, что распределение расстояний между парными чтениями является нормальным, но использует измененную функцию правдоподобия. В функции правдоподобия учитываются не только вероятность получить конкретные чтения, но также вероятность того, что случайное парное чтение свяжет пару контигов, которая зависит от длин контигов.

В качестве входных данных в вычислительных экспериментах использовались 600000 парных чтений, полученных секвенатором компании Illumina из генома бактерии *E. Coli*, и 502 контига, собранные из чтений. Эти данные и референсная геномная последовательность выложены в сети Интернет по адресу <http://genome.ifmo.ru/node/17>. Результат работы предложенного метода был сравнен с результатом метода среднего арифметического (SOPRA, <http://www.biomedcentral.com/1471-2105/11/345>). Были рассмотрены пары контигов, находящихся друг от друга на расстоянии, превышающем среднее расстояние между парными чтениями не более чем на порядок. Разработанный метод в 67% случаев оказывается лучше метода среднего арифметического, оценивая расстояние величиной, более близкой к реальной. При этом в среднем расстояние до правильной величины сокращается на 36%. В 15% случаев результаты работы методов совпадают. В 16% результат оказывается хуже, при этом происходит увеличение расстояния до правильной величины на 14% в среднем. Средняя ошибка составляет –3 и –99 для предложенного и среднего арифметического методов соответственно, при этом стандартное отклонение ошибки составляет 489 и 1881.

УДК 004.4'242

ПОСТРОЕНИЕ АВТОМАТОВ УПРАВЛЕНИЯ ОБЪЕКТАМИ СО СЛОЖНЫМ ПОВЕДЕНИЕМ ПО ТЕСТАМ С УЧЕТОМ НЕПРЕРЫВНЫХ ВОЗДЕЙСТВИЙ

И.П. Бужинский, В.И. Ульянов

Научный руководитель – ассистент Ф.Н. Царев

Постановка задачи. Автоматное программирование – парадигма, в рамках которой программы проектируются в виде систем автоматизированных объектов управления, каждый из которых состоит из объекта управления и конечного автомата. Автоматное программирование целесообразно применять для задач управления объектами со сложным поведением. В настоящем исследовании рассматривается задача построения автоматов

управления объектами с непрерывными управляющими параметрами (органами управления) по обучающим примерам, или тестам.

Каждый тест представляет собой две последовательности: последовательность входных воздействий, описывающих состояние объекта управления или среды, в которой он находится, а также «эталонную» последовательность значений управляющих параметров (положений органов управления). Тесты описывают желаемое поведение объекта управления и являются входными данными для рассматриваемой задачи. В качестве событий для автоматов, являющихся решениями задачи, могут рассматриваться значения предикатов – величин, которые могут быть вычислены по входным воздействиям, а в качестве выходных воздействий – изменения значений управляющих параметров.

Цель работы. В работе [2] предложен метод построения автоматов, управляющих рассматриваемыми объектами, на основе генетических алгоритмов. Метод был опробован на задаче управления моделью беспилотного самолета и позволил построить автомат, под управлением которого самолет выполнял «мертвую петлю». К недостаткам метода можно отнести невысокую производительность (для построения автомата требуется несколько часов), а также не всегда приемлемое качество получающихся автоматов. Целью настоящей работы является улучшение существующего метода путем устранения указанных недостатков.

Базовые положения исследования. Исследования по улучшению метода построения автоматов производились в следующих направлениях:

- для построения автоматов применяются не только генетический алгоритм, но и другие алгоритмы поисковой оптимизации: эволюционная стратегия и одна из разновидностей [1] муравьиного алгоритма;
- предлагается новый способ представления автоматов, имеющих непрерывные выходные воздействия. В рамках этого способа для представления функции переходов используется метод сокращенных таблиц. Ключевым компонентом для представления функции выходов являются вещественные переменные – обобщение предикатов. Выходным воздействием автомата на каждом такте является линейная комбинация значений вещественных переменных с коэффициентами, расставленными при помощи алгоритма, аналогичному предложенному в [2] алгоритму расстановки выходных воздействий;
- функция приспособленности автомата модифицирована для возможности получения автоматов, совершающих небольшое число переходов между состояниями. Уменьшение числа совершаемых переходов между состояниями упрощает понимание физического смысла различных состояний автомата.

Основные результаты. Предложенные модификации метода построения автоматов опробованы на задаче управления моделью беспилотного самолета. Были записаны три набора тестов, описывающих выполнение различных фигур пилотажа: «мертвой петли», «бочки» и разворота самолета в горизонтальной плоскости на 180° . Построение автомата, выполняющего разворот, стало возможным только в результате одной из предложенных модификаций метода, а качество построенных автоматов для остальных фигур пилотажа улучшилось по сравнению с первоначальным методом. Время построения автомата для каждой фигуры пилотажа на персональном компьютере не превышало 20 минут.

Литература

1. Chivilikhin D., Ulyantsev V. Learning Finite-State Machines with Ant Colony Optimization // Lecture Notes in Computer Science. – 2012. – V. 7461/2012. – P. 268–275.

2. Александров А.В., Казаков С.В., Сергушичев А.А., Царев Ф.Н., Шалыто А.А. Генерация конечных автоматов для управления моделью беспилотного самолета // Научно-технический вестник СПбГУ ИТМО. – 2011. – № 2(72). – С. 3–11.

УДК 004.023:519.161

ГЕНЕРАЦИЯ СЛОЖНЫХ ТЕСТОВЫХ ДАННЫХ ДЛЯ ЖАДНОГО АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ О МИНИМАЛЬНОЙ ОБЩЕЙ НАДСТРОКЕ

М.В. Буздалов, Ф.Н. Царев

Научный руководитель – д.т.н., профессор А.А. Шалыто

Введение. Задача о наименьшей общей надстроке имеет множество применений в задачах вычислительной биологии (например, в сборке генома [1]). Задача является NP-трудной [2], однако известны различные эвристические алгоритмы, позволяющие получать приемлемые по оптимальности решения. Для одного из этих алгоритмов (GREEDY [3]) имеется следующая теоретическая оценка оптимальности: длина надстроки, полученной этим алгоритмом, не может превышать $3,5N$, где N – длина наименьшей надстроки [4]. Однако неизвестны входные данные, где длина такой надстроки равняется или превышает $2N$.

Построение сложных тестовых данных для задачи о наименьшей общей надстроке, как правило, производится вручную. В работе предлагается подход к построению таких тестовых данных с помощью эволюционных алгоритмов. Вводятся новые критерии качества тестовых данных. Тесты, сгенерированные с помощью эволюционного алгоритма, по указанным критериям превосходят известные тесты, построенные вручную.

Описание предлагаемого подхода. Предлагается искать сложные тестовые данные в виде совокупности шаблонов строк определенного вида – π -строка. π -строка имеет вид $a_1^{b_1k+c_1} a_2^{b_2k+c_2} \dots a_N^{b_Nk+c_N}$, где a_i – символы алфавита, $b_i k+c_i$ – степени символов, k – целочисленный параметр, стремящийся к бесконечности. Длина π -строки – линейная функция от k , равная сумме степеней всех символов.

При работе с несколькими π -строками параметр k считается общим для всех строк. В этом случае над π -строками можно определить операции конкатенации, сравнения длин и максимального перекрытия, которые необходимы для реализации алгоритмов поиска наименьшей общей надстроки.

Поиск сложных тестовых данных сводится к задаче оптимизации. В данной работе рассматриваются два подхода к заданию оптимизируемых величин – однокритериальный и многокритериальный.

При использовании однокритериального подхода единственным оптимизируемым критерием является асимптотика отношения длины надстроки $(Ak+B)$, полученной алгоритмом GREEDY, к длине минимальной надстроки $(ak+b)$ при стремлении значения параметра x к бесконечности. Таким образом, значение критерия равно A/a . Для оптимизации по указанному критерию используется генетический алгоритм.

При использовании многокритериального подхода используется дополнительный критерий – разность между длиной надстроки, полученной алгоритмом GREEDY, и длиной минимальной надстроки, умноженной на A/a . Значение этого критерия равно $\frac{aB-Ab}{a}$. Для оптимизации по обоим указанным критериям используется алгоритм NSGA-II [5].

Результаты. В случае однокритериального подхода генетический алгоритм достаточно быстро находит отношение $A/a = 2,0$. Отношение, превышающее 2,0, найдено не было. На рис. 1, а) показан пример графика функции приспособленности. На рис. 1, б) построена

гистограмма распределения числа поколений, необходимых для достижения отношения 2,0, для 97% лучших запусков из 10 000 проведенных.

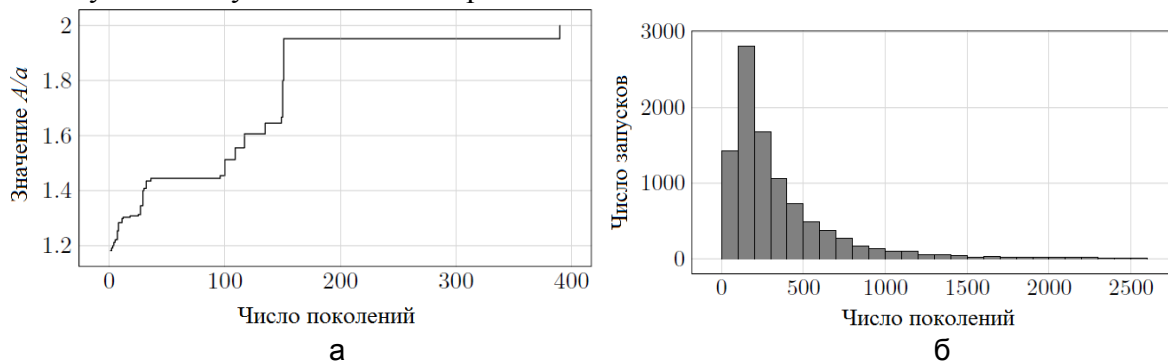


Рис. 1. Результаты экспериментов, однокритериальный подход

В случае многокритериального подхода максимальное значение отношения A/a также составило 2,0. Среди всех тестов со значением $A/a = 2,0$ наилучшее значение второго критерия составило $-3,0$, что превосходит аналогичные значения у известных тестов.

Приближение Парето-фронта для данных двух критериев, полученное в ходе экспериментов, приведено на рис. 2.

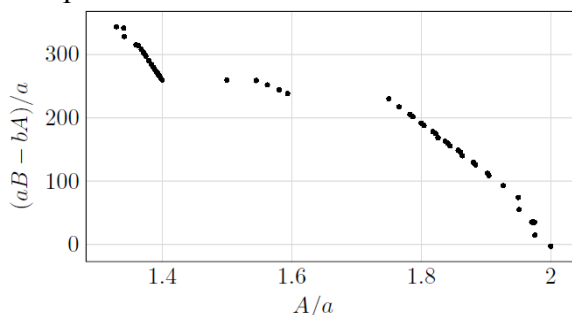


Рис. 2. Результаты экспериментов, многокритериальный подход

Литература

1. Bockenhauer H.-J., Bongartz D. Algorithmic Aspects of Bioinformatics (Natural Computing Series). – Springer, 2007. – 404 p.
2. Gallant J., Maier D., Storer J.A. On finding minimal length superstrings // Journal of Computer and System Sciences. – 1980. – V. 20. – P. 50–58.
3. Frieze A., Szpankowski W. Greedy algorithms for the shortest common superstring that are asymptotically optimal. – 1997.
4. Kaplan H., Shafirir N. The greedy algorithm for shortest superstrings // Information Processing Letters. – 2005. – V. 93. – P. 13–17.
5. Deb K., Pratap A. et al. A fast elitist multi-objective algorithm: NSGA-II // IEEE Transactions on Evolutionary Computation. – 2000. – V. 6. – P. 182–197.

ПРИМЕНЕНИЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ К ГЕНЕРАЦИИ ТЕСТОВ ДЛЯ ОЛИМПИАДНЫХ ЗАДАЧ ПО ПРОГРАММИРОВАНИЮ

А.С. Буздалова, М.В. Буздалов

Научный руководитель – д.т.н., профессор А.А. Шалыто

Введение. В работе рассматривается одно из практических применений метода $EA+RL$ [1, 2], предложенного авторами ранее и позволившего получить высокие результаты при решении ряда модельных задач. Метод основан на обучении с подкреплением [3] и предназначен для повышения эффективности эволюционных алгоритмов [4]. Эволюционные алгоритмы могут быть использованы для решения сложных задач оптимизации. В частности, в работе [5] был предложен способ генерации тестов для олимпиадных задач по программированию с помощью эволюционных алгоритмов. В ходе настоящего исследования метод $EA+RL$ был применен совместно с этим способом.

Целью работы является применение метода $EA+RL$ к генерации тестов для олимпиадной задачи «Ships. Version 2» [5]. Ожидается, что метод позволит повысить эффективность работы соответствующего эволюционного алгоритма. Эффективность эволюционного алгоритма определяется как число поколений, необходимое для генерации теста, на котором время работы тестируемого решения олимпиадной задачи превышает заданное ограничение.

Описание предлагаемого подхода. Тесты генерируются в ходе выполнения генетического алгоритма [4], максимизирующего время работы тестируемого решения t . Таким образом, функция t является целевой функцией приспособленности. Метод $EA+RL$ позволяет ускорить процесс оптимизации целевой функции приспособленности путем подстановки наиболее эффективных вспомогательных функций в качестве текущей функции приспособленности особи [1, 2]. Вспомогательные функции выбираются из заранее подготовленного набора, в который входят следующие функции:

- число итераций алгоритма решения i ;
- суммарное число вызовов рекурсивной функции на каждой итерации r ;
- упорядоченная пара $p = (i, r')$, где r' – число вызовов рекурсивной функции на последней итерации.

Описание эксперимента. В ходе эксперимента измерялась эффективность работы как обычных генетических алгоритмов, использующих различные функции приспособленности, так и генетических алгоритмов, настраиваемых с помощью метода $EA+RL$. Также был рассмотрен алгоритм Random, который случайным образом выбирал функцию приспособленности для каждого поколения генетического алгоритма. Каждый алгоритм запускался по 100 раз.

Использовались генетические операторы, аналогичные операторам, описанным в [5]. В качестве алгоритмов обучения с подкреплением применялись алгоритмы Q-learning [3] и DelayedQ-learning [6]. Значения параметров алгоритмов обучения составляли $\varepsilon = 0,3$, $\alpha = 0,4$, $\gamma = 0,01$ для алгоритма Q-learning и $m = 5$, $\varepsilon = 0,001$, $\gamma = 0,1$ для алгоритма DelayedQ-learning. Алгоритм DelayedQ-learning перезапускался каждые 50 поколений.

Результаты. Результаты проведенного эксперимента представлены в таблице. Алгоритмы отсортированы в порядке возрастания среднего числа поколений, необходимого для генерации искомого теста. Можно видеть, что оптимизация по целевой функции приспособленности t является неэффективной, в то время как оптимизация по функциям,

назначаемым с помощью Delayed Q-learning, приводит к наиболее быстрому получению результата.

Таблица. Результаты эксперимента. ГА – генетический алгоритм

Алгоритм	Используемые ФП	Доля успешных запусков	Среднее число поколений	Среднеквадратичное отклонение
Delayed Q-learning+ГА	i, r, p, t	90%	4411	4352
ГА	p	89%	4497	4716
Q-learning + ГА	i, r, p, t	85%	4580	5186
Random + ГА	i, r, p, t	76%	6074	6788
ГА	i	74%	8182	7459
ГА	r	51%	15030	13969
ГА	t	1%	990319	994987

Результаты показывают, что метод $EA+RL$ позволяет автоматизировать процесс выбора вспомогательных функций приспособленности таким образом, что эффективность генерации тестов для рассмотренной олимпиадной задачи значительно возрастает по сравнению с показателями, достигаемыми при использовании традиционной функции приспособленности.

Литература

1. Buzdalova A., Buzdalov M. Increasing Efficiency of Evolutionary Algorithms by Choosing between Auxiliary Fitness Functions with Reinforcement Learning // Proceedings of the Eleventh International Conference on Machine Learning and Applications, ICMLA 2012. – Boca Raton: IEEE Computer Society, 2012. – V. 1. – P. 150–155.
2. Буздalова А.С., Буздalов М.В. Метод повышения эффективности эволюционных алгоритмов с помощью обучения с подкреплением // Научно-технический вестник информационных технологий, механики и оптики. – 2012. – № 5(81) – С. 115–119.
3. Sutton R.S., Barto A.G. Reinforcement Learning: An Introduction. – MIT Press, Cambridge, MA, 1998. – 322 p.
4. Eiben A. E., Smith J. E. Introduction to Evolutionary Computing. – Springer-Verlag, Berlin, Heidelberg, New York, 2007. – 304 p.
5. Буздalов М.В. Генерация тестов для олимпиадных задач по программированию с использованием генетических алгоритмов // Научно-технический вестник СПбГУ ИТМО. – 2011. – № 2(72) – С. 72–77.
6. Strehl A.L., Li L., Wiewora E., Langford J., Littman M.L. PAC Model-Free Reinforcement Learning // Proceedings of the 23rd International Conference on Machine learning. ICML'06. – 2006. – P. 881–888.

УДК 004.75

РАЗРАБОТКА РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ С УЗЛАМИ, ИНТЕГРИРОВАННЫМИ В СОЦИАЛЬНУЮ СЕТЬ

В.К. Горбунов

Научный руководитель – магистрант В.И. Ульянов

Краткое введение. В настоящее время вычислительная сложность исследовательских задач постоянно повышается. Для их решения необходимы ресурсы, сопоставимые с вычислительной мощностью суперкомпьютера, стоимость которых велика. Наиболее гибким и дешевым решением видится организация вычислительного кластера, доступного для

использования нашими молодыми учеными, развернутого на уже доступных вычислительных ресурсах, таких как домашние рабочие станции.

Постановка задачи. Задача состоит в том, чтобы реализовать и развернуть систему, состоящую из следующих частей;

1. серверная часть – управляет выполнением задач: раздает клиентам задачи и сохраняет результаты вычислений;
2. клиентская часть, которая состоит из:
 - менеджера задач – модуля, обеспечивающего коммуникацию с сервером и управление остальными компонентами;
 - специализированных модулей – программного обеспечения (ПО), разрабатываемого исследователями для каждой задачи отдельно. Загружаются менеджером задач, и решают поставленные задачи;

Для упрощения шага развертывания клиентского ПО на рабочих станциях предлагается выполнять вычислительные задачи в браузере и реализовать интеграцию с социальной сетью.

Обзор существующих решений. В настоящий момент наиболее широкое распространение получили систему, такие как BOINC (разработка Калифорнийского Университета в Беркли открытой платформы) и Globus Toolkit (разработка международной ассоциации Globus Alliance). Ни в одной из них не реализовано клиентского модуля с требуемыми характеристиками.

Цель работы. Разработать необходимое программное обеспечение, протестировать его на реальной исследовательской задаче и внедрить в научный процесс.

Промежуточные результаты. Клиент спроектирован в виде Java-апплета, что дает следующие преимущества:

- кроссплатформенность – может запускаться на любой операционной системе с установленным браузером и виртуальной Java-машиной;
- запуск клиента сводится к открытию странички в браузере;
- отсутствует необходимость обновления программного обеспечения на стороне клиента, так как оно подгружается динамически;
- поддерживается интеграция с социальной сетью vkontakte.

В настоящий момент ведется подбор компонентов и реализация системы.

УДК 004

ПОСТРОЕНИЕ АВТОМАТНОЙ МОДЕЛИ ПО ДАННЫМ ТРАССИРОВКИ ПРОГРАММЫ

А.П. Данильченко

Научный руководитель – ассистент Ф.Н. Царев

Введение. Ключевым элементом парадигмы автоматного программирования является построение управляющего конечного автомата. Использование автоматной модели позволяет производить автоматическое построение тестов и верификацию, а также обладает рядом других достоинств [1]. Однако построение управляющего автомата является нетривиальной задачей, и хотя иногда может быть выполнено вручную, для большинства реальных задач является весьма трудоемким процессом. Во многих случаях уже существует программа, решающая необходимую задачу. Такая программа, написанная не в автоматном стиле, не

предоставляет автоматной модели сама по себе, но может быть использована для построения управляющего автомата.

Целью работы является построение управляющего конечного автомата путем динамического анализа программы, т.е. на основе данных трассировки программы.

Описание предлагаемого подхода. В основе метода лежит подход, предложенный в работе [2]. Его суть состоит в построении управляющего автомата по тестовым сценариям с помощью сведения к задаче CSP (задача удовлетворение ограничений) и запуска сторонней программы, решающей эту задачу. Сценарием называется последовательность троек $\langle e, f, A \rangle$, где e – входное событие, f – булева формула от входных переменных, задающая охранное условие, A – последовательность выходных воздействий. В настоящей работе предлагается способ автоматического построения тестовых сценариев по данным трассировки программы, написанной на языке Java. При построении сценариев используются базовые знания о структуре программы (иерархия классов, методы и поля классов) и данные трассировки, получаемые с помощью инструментирования байт-кода (порядок вызова методов, значения полей и параметров при входе в метод, значения полей и возвращаемое значение на выходе из метода). Для проведения трассировки и обработки данных, полученных в ходе этого процесса, используется фреймворк Daikon [3].

Результаты. Разработан и реализован метод автоматического построения автоматной модели по существующей программе на языке Java, используя данные трассировки.

Литература

1. Поликарпова Н.И., Шалыто А.А. Автоматное программирование. – СПб: Питер, 2010. – 176 с.
2. Ульянов В.И., Царев Ф.Н. Применение методов решения задачи удовлетворения ограничениям для построения управляющих конечных автоматов по сценариям работы. // Всероссийская научная конференция по проблемам информатики (СПИСОК-2012). СПб: ВВМ. СПбГУ. – 2012. – С. 444–445.
3. Michael D. Ernst, William G. Griswold, Yoshio Kataoka, and David Notkin. Dynamically discovering pointer-based program invariants / University of Washington Department of Computer Science and Engineering technical report. UW-CSE-99-11-02, (Seattle, WA), November 16. – 1999. Revised March 17. – 2000.

УДК 004.021

ВОССТАНОВЛЕНИЕ ФРАГМЕНТОВ ПАРНЫХ ЧТЕНИЙ ПРИ СБОРКЕ ТРАНСКРИПТОМА

В.О. Долганов

Научный руководитель – ассистент Ф.Н. Царев

Многие современные задачи биологии и медицины требуют знания транскриптома живых организмов, который состоит из различных транскриптов и их изоформ. В связи с этим возникает необходимость в дешевом и быстром методе секвенирования, т.е. определения последовательности нуклеотидов в образце РНК.

Существующие секвенаторы – устройства для чтения РНК – не позволяют считать за один раз все молекулы РНК полностью. Вместо этого они позволяют читать фрагменты транскриптов небольшой длины. Длина фрагмента может быть разной, она является важным параметром секвенирования – от нее напрямую зависит стоимость секвенирования и время,

затрачиваемое на чтение одного фрагмента: чем больше длина считываемого фрагмента, тем выше стоимость чтения и тем дольше это чтение происходит. Современные высокопроизводительные секвенаторы основаны на разных подходах со своими достоинствами и недостатками. Часто для сборки используют парные чтения, для которых известно примерное расстояние между ними. Это позволяет получить больше информации о транскриптом при сравнительно небольших затратах.

Задачей сборки транскриптома является восстановление последовательностей кДНК (соответствующих РНК, отвечающих транскриптам) на основании информации, полученной в результате секвенирования. Чем длиннее исходных чтения, тем более качественной получается итоговая сборка. В этой работе была поставлена задача восстановления фрагментов транскриптов, отвечающих парным чтениям (восстановления последовательности между концами парных чтений). Алгоритм основан на использовании графа де Брёйна и частоты встречи k -меров для поиска пути.

Алгоритм восстановления фрагментов транскриптов реализован на языке программирования Java. Для запуска программ использовался компьютер с 32 ГБ оперативной памяти (из них 20 GB было использовано) и двумя 4-ядерными процессорами. Время работы программы составило менее суток. Опишем результаты выполнения программы.

Исходные данные были получены из анализа организма *Mus musculus* при помощи системы Illumina Genome Analyzer II. Полученные данные представлены в виде 90 миллионов парных чтений. Среднее расстояние между концами чтения составило 300 нуклеотидов. В результате работы программы было однозначно восстановлено 32% от всех чтений.

УДК 519.677

О МЕТОДЕ ПОСТРОЕНИЯ ВОССТАНАВЛИВАЮЩИХ АГРЕГАТОВ НА БАЗЕ ЯДЕР СТЕКЛОВА ДЛЯ РЕШЕНИЯ ЗАДАЧИ ПЕРЕДИСКРЕТИЗАЦИИ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ

А.В. Масальских

Научный руководитель – д.т.н., профессор В.Г. Парфёнов

Методы цифровой обработки изображений подразумевают оперирование дискретными изображениями. Дискретное изображение представляет собой набор отсчетов с известными дискретизированными значениями интенсивности в них. При цифровой обработке изображений часто является недостаточным имеющееся количество отсчетов. Существует два решения для получения недостающих для работы алгоритма данных: произвести переоцифровку аналогового изображения с нужными для алгоритма параметрами дискретизации или же рассчитать значения в недостающих отсчетах на базе имеющихся. Нас будут интересовать алгоритмы, позволяющие рассчитывать значения в произвольных точках на базе информации об интенсивности в исходных отсчетах. Многоцветные дискретные изображения, по сути, являются набором значений интенсивности компонентов (в зависимости от применяемой схемы, например, компоненты Red, Green, Blue), заданных для каждого отсчета (или пикселя). Не уменьшая общности, будем рассматривать изображения с одним значением интенсивности на пиксель.

Алгоритмы, позволяющие рассчитывать значения интенсивности на основе имеющихся отсчетов делятся на решетчатые и непрерывные. Решетчатые позволяют рассчитать значения только в определенных точках, непрерывные могут восстановить значение в точках с любыми координатами в пределах изображения. В свою очередь непрерывные реконструкторы (агрегаты восстановления) делятся на две большие группы:

интерполяционного и аппроксимационного типа. Восстанавливаемая поверхность реконструкторами первого типа проходит через изначально известные точки отсчетов, а поверхность, получаемая реконструкторами второго типа, может не проходить через исходные точки. Методы восстановления делятся так же на конечные и бесконечные, но так как цифровая обработка изображений подразумевает только конечные множества отсчетов, практический интерес представляют только конечные реконструкторы. Конечные реконструкторы бывают локальные и не локальные. От степени локальности реконструктора зависит количество ближайших исходных отсчетов, значения в которых будут использованы для получения нового значения в рассчитываемой точке.

Наиболее часто используются методы восстановления, основанные билинейной, бикубической интерполяцией, а так же на основе диаграмм Вороного. Метод ближайшего соседа (построенный на основе диаграммы Вороного), восстанавливает значение поверхности интенсивности в любой точки, принимая значение в ней равным значению интенсивности в ближайшем известном отсчете. Метод билинейной интерполяции является расширением линейной интерполяции до функции двух переменных и при расчете значения интенсивности в точке, использует взвешенные значения интенсивности в 4 ближайших отсчетах. Метод бикубической интерполяции является расширением кубической интерполяции для функции двух переменных. Он использует значения интенсивности в ближайших 16 отсчетах. Метод, построенный на базе бикубической интерполяции позволяет восстанавливать гладкую поверхность интенсивности, имеющую непрерывные частные производные первого порядка и непрерывную смешанную частную производную первого порядка.

Более сложные методы, построенные на базе кусочно-полиномиальной интерполяции и интерполяции с использованием кардинального синуса (фильтр Ланцоша). Интерес представляет построение кусочно-полиномиальной интерполяции с изменяемыми параметрами гладкости и локальности агрегата восстановления.

Для восстановления непрерывной поверхности интенсивностей использовать агрегат, построенный на основе работ [1, 2]. Агрегат восстановления получает непрерывную поверхность интенсивности в два этапа. Сначала на базе известных значений интенсивности в имеющихся отсчетах, строится аппроксимирующая функция простого вида, затем применяется сглаживающий оператор интегрального типа. В операторе интегрального типа используются ядра, соответствующие функциям Стеклова.

Подбор ядер Стеклова нужного порядка позволяет получать поверхность интенсивности необходимой гладкости. Благодаря финитности ядер Стеклова получены простые вычислительные формулы для восстановления непрерывной поверхности интенсивности на ограниченной области. Построенный восстанавливающий агрегат является непрерывным реконструктором аппроксимационного типа, строящим кусочно-полиномиальную поверхность интенсивности. Агрегат обладает свойством локальности, степень которой может регулироваться параметром, полученным естественным образом.

Метод апробирован на цветных цифровых изображениях. С использованием построенного агрегата восстановления успешно решалась задача увеличения количества отсчетов в дискретном изображении в 100 и 10000 раз. Метод показал ожидаемые практические результаты и свойства.

Литература

1. Додонов Н.Ю., Жук В.В. О равномерном приближении непериодических функций, заданных на всей оси // Проблемы математического анализа. – 2004. – № 29. – С. 25–35.
2. Додонов Н.Ю. О приближении функций в пространствах $L_p(R)$ посредством сдвигов и сжатий одной функции // Проблемы математического анализа. – 2005. – № 30. С. 47–59.

ОПРЕДЕЛЕНИЕ ПОЛОЖЕНИЯ ТОЧКИ СЪЕМКИ С ИСПОЛЬЗОВАНИЕМ 2D- И МАЛОГО ЧИСЛА 3D-СООТВЕТСТВИЙ

А.П. Минюк

(Санкт-Петербургский национальный исследовательский университет информационных технологий механики и оптики)

Научный руководитель – к.т.н. В.А. Сухов
(ЗАО «Транзас Экспресс»)

Введение. В последние годы стала активно развиваться беспилотная авиация. Одним из ее применений является аэрофотосъемка, с последующим преобразованием аэрофотоснимков в трехмерную модель местности. Трехмерная модель местности используется при решении различных задач, таких как: планирование и контроль строительства, установление границ муниципальных территорий, оперативная оценка обстановки и много другое. Один из этапов построения трехмерной модели местности – определение взаимного расположения точек съемки.

Постановка задачи. Для определения взаимного расположения точек съемки существуют два основных современных метода. Первый предназначен для восстановления взаимного положения точек съемки двух фотоснимков, он использует соответствующие двумерные соответствия между фотоснимками (соответствующие друг другу точки на снимках). Данный метод определяет поворот камеры и направление сдвига камеры между фотоснимками, но не длину сдвига (метод не определяет масштаб). Второй предназначен для того, что бы определить точку съемки очередного фотоснимка относительно группы точек съемки, с известным взаимным расположением (масштаб будет сохранен относительно группы), он использует трехмерные соответствия (соответствие между точкой трехмерного пространства и ее проекцией на фотоснимок). Для такого метода необходимо наличие большой общей зоны на фотоснимках между фотоснимком, точку съемки которого необходимо определить, и хотя бы двумя фотоснимками, взаимное положение которых уже определено, – зоны тройного перекрытия изображений.

Частой проблемой при проведении аэросъемки является ветер. При прямом пролете по ветру, зона тройного перекрытия становится очень мала или вовсе исчезает. Из-за чего использование стандартных подходов определения точки съемки становится не надежным. Поэтому необходимо сделать пролет в обратную сторону против ветра, либо. Однако это требует дополнительного времени, что крайне не выгодно при необходимости оперативной оценки обстановки. Таким образом, необходима разработка устойчивого метода определения точек съемки при малых зонах тройного перекрытия.

Практические результаты. В работе предлагается подход, использующий как трехмерных, так и двумерные соответствия, после того как определено взаимное расположение первых двух камер. Для каждой последующей точки съемки сначала определяется ее ориентация, так как она лучше обусловлена, а после координаты съемки. На первом этапе, используя двумерные соответствия, определяется поворот камеры относительно всех точек съемки, со снимками которых имеются двумерные соответствия, и, положения которых известны. На основе этих поворотов и ориентаций камер, положения которых известны, с использованием фильтраций от выбросов, определяется ориентация точки съемки. На втором этапе при известной ориентации камеры координаты точки съемки определяются как координаты, при которых среднеквадратичное отклонение проекции трехмерной точки от соответствующей ей двумерной точки на плоскости фотоснимка минимально при минимальном числе выбросов.

Предложенный метод позволил существенно сократить размер зоны тройного перекрытия, необходимый для качественного определения точки съемки. Однако предложенный метод требует значительной зоны двойного перекрытия изображений. Минимальное необходимое количество трехмерных соответствий для определения точки съемки, при условии отсутствия выбросов – два.

УДК 004.912

ПРИМЕНЕНИЕ АЛГОРИТМОВ ФОНЕТИЧЕСКОГО АНАЛИЗА ДЛЯ ПОИСКА ДАННЫХ В СОЦИАЛЬНЫХ СЕТЯХ

А.М. Надточий

Научный руководитель – к.т.н., доцент Г.А. Корнеев

Постановка задачи. В настоящее время социальными сетями пользуется почти половина населения России. В такой большой аудитории эффективный поиск людей, групп и любого другого контента становится все более актуальной задачей. Кроме того, многие пользователи при написании своих персональных данных используют транслитерацию, сленг или намеренно ошибочное написание, а также различные знаки эмоджона («смайлики»). Как следствие, использование формального подхода к задаче сравнения строк становится сильно затрудненным. Аналогичная проблема имеет место при поиске групп, музыки и городов в социальных сетях.

Целью работы является разработка методов для поиска контента в русскоязычных социальных сетях, которые удовлетворяют следующим требованиям: позволяют искать информацию по точному написанию, искаженному написанию, написанию с ошибками и с использованием латиницы, а также учитывают синонимы. При этом разработанные методы должны возвращать числовое значение, количественно описывающее релевантность найденного результата поданному запросу.

Описание предлагаемого подхода. Ключевым требованием к разрабатываемому алгоритму является необходимость определения численной меры схожести сравниваемых строк по фонетическому написанию. Существующие фонетические алгоритмы поиска (SoundEx, Metaphone) не возвращают никакой численной информации, а также не умеют сравнивать транслитное написание слова с кириллицей. Для решения этой задачи хранение и фонетическое сравнение строк было реализовано в латинице, при этом для каждой пары символов была задана цена замены одного символа на другой. Численная мера схожести, возвращаемая алгоритмом, есть суммарная цена замен, которые необходимо сделать для превращения одной строки в другую. При превышении некоторого порога, зависящего от длины сравниваемых строк, они считаются непохожими. Для расширения поля поиска была также определена возможность замены в строке слова на синоним с соответствующим уменьшением меры схожести на некоторую константу. Опечатки учитывались на основании определения расстояния Левенштейна. В зависимости от вида поиска (люди, группы и т.д.) были заданы дополнительные веса, определяющие приоритет совпадений между собой.

Результаты. Разработанный алгоритм был апробирован на сайте «Одноклассники». По статистике, собираемой на сайте, число переходов на страницы результатов с низкой релевантностью при использовании разработанного алгоритма уменьшилось на 20%, что подтверждает его эффективность. В процессе апробации были выявлены особенности написания имен и названий в различных видах поиска (группах, музыке, городах и др.), которые были учтены в последующих версиях алгоритма. В настоящее время разработанный

алгоритм используется при поиске людей, групп, городов и музыкальных файлов. Планируется дальнейшее расширение использования алгоритма для поиска видео, сообщений и т.д.

УДК 004.434

ПРИМЕНЕНИЕ АВТОМАТНОГО ПРОГРАММИРОВАНИЯ В КОНТРОЛЛЕРАХ ARDUINO

С.О. Попов

Научный руководитель – д.т.н., профессор В.Г. Парфенов

В настоящее время цена контроллера Arduino позволяет покупать их широкому кругу разработчиков и начинающим командам. Контроллеры Arduino имеют достаточно дружелюбный интерфейс для начинающих программистов, а также базируются на промышленных контроллерах Atmel. Что позволяет применять их, например, в системах домашней автоматизации. Следует отметить, что при формировании нового продукта на рынке очень сильное влияние оказывает его качество. Поэтому помимо хороших контроллеров командам разработки следует обеспечить максимально возможную отказоустойчивость программного обеспечения самого контроллера. Потому, что ошибка в контроллерах стоит больших денег и репутации при замене или перепрограммировании на месте установки. Одним из наиболее признанных решений проблем качественного программного обеспечения во встроенных системах, является автоматное программирование. Оно более наглядное, требует меньше памяти и может быть верифицировано.

Основная **цель работы**, это выработка решений по практической реализации автоматных программ на контроллерах Arduino.

Особенностями программирования контроллеров является: Ограниченность памяти, отсутствие механизмов отладки, отсутствие механизмов ведения журналов. В поставленные перед контроллером задачи входило обеспечение: индикации работы контроллера, работа с шиной данных RS-485, интерпретация и передача данных получаемых по шине на сервера в интернете. Было принято решение реализовать библиотеки на языке C++ с использованием директив препроцессора, что позволило пользоваться стандартными средствами разработки.

В результате работы была реализована библиотека ведения журнала и передачи данных с устройств в любой поток, как в последовательный порт, так и в TCP соединение с сервером. Была реализована библиотека для более удобного и наглядного создания автоматов с автоматическим ведением журнала. На основе этой библиотеки были созданы автоматы по соединению с серверами приема данных. А также были созданы несколько автоматов-драйверов для получения данных с внешних устройств. Например, для сбора информации о температуре, показаниях счетчиков воды и электричества.

Стоит отметить, что удалось реализовать параллельную работу автоматов на одном ядре контроллера.

Апробация на готовых устройствах показала безотказность работы автоматов и библиотек. В настоящее время все обнаруженные ошибки относятся к отказу аппаратных частей контроллера или программного обеспечения не связанного с автоматами. Например, зависанию микросхемы WizNet5100 или неправильной отработки срабатывания сторожевого таймера загрузчиком контроллера.

В дальнейшем планируется развивать набор библиотек и инструментов. В том числе переписывать стандартные библиотеки контроллера с использованием автоматного подхода. А также решить проблему взятия блокировок на общие для автоматов ресурсы контроллера.

ПРИМЕНЕНИЕ БАЙЕСОВСКИХ КЛАССИФИКАТОРОВ ДЛЯ ОБНАРУЖЕНИЯ КОММЕРЧЕСКИХ ССЫЛОК

С.С. Поромов

Научный руководитель – к.т.н., доцент А.С. Станкевич

Поисковая оптимизация является одним из важных факторов продвижения сайта с целью повышения количества пользователей сайта. Применение различных методов поисковой оптимизации приводит к тому, что продвигаемый сайт оказывается выше в выдаче основных поисковых систем в интернете, и таким образом все большее число пользователей переходит на сайт с этих поисковых систем.

Одним из факторов, учитываемых поисковыми системами, является количество ссылок на сайт с других ресурсов интернета – индекс цитирования. Для увеличения этого фактора, поисковые оптимизаторы покупают внешние ссылки на других ресурсах и размещают там ссылки с некоторым оптимизированным текстом, ведущие на продвигаемые страницы. Этот метод похож на обычную рекламу, но своей целью имеет не повышение интереса у пользователей, увидевших рекламу, а накрутку индекса цитирования для поисковых систем.

Для борьбы с такими методами нечестного искусственного продвижения сайтов, необходимо разработать систему обнаружения купленных, т.е. коммерческих ссылок, а также сайты, продвигаемые с помощью такого метода и страницы, которые продают ссылки. После выявления таких ссылок, данный фактор можно учитывать при формировании поисковой выдачи с целью повышения ее качества. При этом надо учитывать, что коммерческие ссылки вполне могут встречаться вместе с естественными ссылками и на авторитетных ресурсах.

Основной проблемой является отделение коммерческих ссылок от обычных, поскольку поисковые оптимизаторы пытаются замаскировать такие ссылки и стараются сделать их наиболее естественными. Как показало исследование, многие коммерческие ссылки отличаются от обычных своим текстом, т.е. в оптимизированном тексте ссылок вероятность присутствия некоторых ключевых слов, например: «продажа, магазин, бесплатно» выше, чем в обычных ссылках.

Для оценки вероятности текста ссылки быть коммерческим, предлагается использовать байесовский классификатор [1]. Байесовский классификатор считает вероятность текста быть оптимизированным на основе частоты встречаемости отдельных слов текста в обычных текстах, а также в оптимизированных текстах. Чем больше в тексте слов, специфических для оптимизированных текстов, тем больше вероятность текста быть таковым. Для обучения данного классификатора необходимо построить два обучающих множества – тексты оптимизированных ссылок, а также обычные тексты. По этим обучающим множествам строятся вероятности встречаемости слов. Поскольку слово может встречаться в своих различных словоформах, для приведения слов к основе используется стемминг Портера [2].

Результатом работы текстового классификатора является оценка вероятности того, что текст ссылки является коммерческим. При таком подходе используется только информация о тексте ссылок. Для улучшения результата применяется анализ графа коммерческих ссылок, основанный на двух фактах. Первый - поисковые оптимизаторы покупают ссылки на многих различных ресурсах, второй - ресурсы продают ссылки разным поисковым оптимизаторам. В результате получается, что граф коммерческих ссылок имеет высокую связность и для поиска коммерческих ссылок можно применять аналогичный индекс цитирования, например, алгоритм HITS [3] (аналог PageRank, используемого поисковой системой Google [4]). Также, на данном этапе можно использовать следующую особенность сайтов, продающих ссылки. В то время как ссылки с обычных сайтов обычно имеют одну тематику, ссылки с сайтов, продающих их, могут иметь много разных тем. Для определения тематики ссылок также используются байесовские классификаторы, обученные на текстах различных тем, часто

встречающихся в коммерческих ссылках.

Совместное применение этих методов позволяет получать хорошую оценку вероятности ссылки быть коммерческой, а также позволяет находить сайты, продвигаемые с помощью такого метода поисковой оптимизации. Построенный классификатор при определении коммерческих ссылок имеет полноту и точность более 80%. Разработанные методы внедряются в поисковую систему «Поиск Mail.Ru».

Литература

1. Маннинг К.Д., Рагхаван П., Шютце Х. Введение в информационный поиск. – Вильямс. – 2010. – 528 с.
2. Porter M.F. An algorithm for suffix stripping // Program. – 1980. – V. 14. – № 3. – P. 130–137.
3. Kleinberg J. Authoritative sources in a hyperlinked environment // Journal of the ACM. – 1999. – V. 46(5). – P. 604–632.
4. Brin S., Page L. The Anatomy of a Large-Scale Hypertextual Web Search Engine. – 1998.

УДК 004.4'242

ПОСТРОЕНИЕ УПРАВЛЯЮЩИХ КОНЕЧНЫХ АВТОМАТОВ С ПОМОЩЬЮ МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ УДОВЛЕТВОРЕНИЯ ОГРАНИЧЕНИЙ

В.И. Ульянов

Научный руководитель – д.т.н., профессор А.А. Шалыто

Введение. При применении парадигмы автоматного программирования для реализации сущности со сложным поведением выделяется система управления и объект управления. На начальном этапе проектирования программы выделяются события, входные переменные и выходные воздействия. После этого проектирование программы может идти разными путями. Один из них состоит в написании сценария работы программы, по которому далее эвристически строится автомат.

К автоматной программе, как правило, предъявляются два требования:

- непротиворечивость – не должно быть двух переходов, исходящих из одного состояния управляющего автомата и одновременно выполнимых при некоторой комбинации события и входных переменных;
- полнота – любой комбинации события и входных переменных должен соответствовать переход в каждом состоянии.

В работе проводится экспериментальное исследование метода построения автоматных программ, удовлетворяющих требованию непротиворечивости, но не удовлетворяющих требованию полноты.

Цель работы. Ранее авторами был разработан метод автоматизированного построения управляющих конечных автоматов по сценариям работы [1]. Целью работы является разработка и экспериментальное исследование метода построения управляющих автоматов с гарантией не только непротиворечивости системы переходов, но и ее полноты.

Базовые положения исследования. Сценарием работы программы будем называть последовательность троек $\langle e, f, A \rangle$, где e – входное событие, f – булева формула от входных переменных, задающая охранное условие, A – последовательность выходных воздействий. На вход разрабатываемому алгоритму подается набор сценариев работы. Построение управляющего автомата осуществляется в пять этапов.

1. Построение дерева сценариев.

2. Построение графа совместимости вершин дерева сценариев.
3. Построение ограничений на целочисленные переменные, задающей требования к раскраске построенного графа и выражающей непротиворечивость и полноту системы переходов результирующего автомата.
4. Запуск сторонней программы, решающей задачу удовлетворения построенным ограничениям (constraint satisfaction problem, CSP).
5. Построение автомата по найденной выполняющей подстановке.

Основной результат. Разработан метод автоматизированного построения управляющих конечных автоматов по сценариям работы. Этот метод основан на сведении указанной задачи к задаче выполнимости ограничений. Работоспособность метода проверена на специально сгенерированных случайных входных данных. Искомый автомат содержал от четырех до десяти состояний, а суммарная длина тестов составляла от 800 до 1200 элементов сценариев работы. Результаты экспериментов показали наличие большого числа входных данных, для которых ранее предложенный метод находил управляющий автомат, не удовлетворяющий требованию полноты, в то время как разработанный метод справлялся с задачей, пусть и с меньшей производительностью.

Литература

1. Ulyantsev V., Tsarev F. Extended Finite-State Machine Induction using SAT-Solver // Proceedings of the Tenth International Conference on Machine Learning and Applications, ICMLA 2011, Honolulu, HI, USA. IEEE Computer Society. – 2011. – V. 2. – P. 346–349.

УДК 004.4'242

ЭВОЛЮЦИОННЫЕ СТРАТЕГИИ С АДАПТИВНЫМ ПАРАМЕТРОМ НА ОСНОВЕ СВОЙСТВ ЛАНДШАФТА ФУНКЦИИ ПРИСПОСОБЛЕННОСТИ

Д.С. Чивилихин

Научный руководитель – ассистент Ф.Н. Царев

В области эволюционных вычислений вводится понятие ландшафта функции приспособленности (fitness landscape). В простейшем случае двухпараметрического пространства поиска ландшафт функции приспособленности (ФП) можно представить как поверхность, задающую график ФП, где абсцисса и ордината обозначают значения параметров особи, а аппликата соответствует значению ФП этой особи. Оценка свойств ландшафта ФП может проводиться как для оценки сложности решения задачи эволюционным алгоритмом, так и для настройки параметров эволюционных алгоритмов.

Существует несколько стандартных методов оценки сложности задач, основанных на ландшафте ФП. В методе fitness-distance correlation (*fdc*) строится случайная выборка решений из пространства поиска и оценивается расстояние между каждым из этих решений и априорно известным глобально оптимальным решением. Далее вычисляется коэффициент корреляции между набором значений ФП и набором расстояний до глобально оптимального решения. В зависимости от значения этого коэффициента можно определить, насколько сложно будет эволюционному алгоритму, начав работу от некоторого случайного решения, достичь до глобально оптимального решения. Хотя данный метод и позволяет достаточно адекватно оценивать сложность задач, его использование для настройки алгоритмов невозможно, так как для оценки коэффициента корреляции требуется знание глобально оптимального решения.

Второй из наиболее распространенных методов, negative slope coefficient (*nsc*), основан на понятии облаков фитнеса (fitness clouds). При построении облаков фитнеса

рассматривается некоторая выборка решений из пространства поиска. Для каждого решения из выборки вычисляется его значение ФП f_i . С помощью оператора мутации из решения получают несколько других и выбирают из их числа решение с наибольшим значением ФП f_i' . Для построения облака фитнеса на плоскость наносятся точки с координатами (f_i, f_i') . Точки облака разделяются по абсциссам на несколько непересекающихся множеств, для каждого множества вычисляется точка его центра масс. Наконец, центры масс соседних множеств соединяются отрезками. Коэффициент nsc пропорционален сумме уклонов всех отрезков с отрицательной производной. Задачи, для которых nsc отрицателен, считаются сложными для эволюционных алгоритмов.

Целью работы является разработка метода оценки свойств ландшафта ФП для динамической настройки параметров эволюционных алгоритмов. В данной работе описанные методы fdc и nsc применяются для исследования сложности построения конечных автоматов для известной задачи об «Умном муравье». Показывается, что коэффициент nsc для рассматриваемой задачи указывает на то, что задача является простой для эволюционных алгоритмов, в то время как коэффициент fdc и экспериментальные исследования утверждают обратное. Также предлагается новый метод оценки свойств ландшафта ФП, позволяющий использовать вычисленные оценки для динамической настройки параметров эволюционных алгоритмов.

Основные положения исследования. Основой предлагаемого метода является построение распределения $M_s(f_0)$ числа успешных мутаций особей по значениям ФП. Для этого перед запуском эволюционного алгоритма генерируется случайная выборка решений из пространства поиска. Для каждого решения с помощью операторов мутации строится множество всех его соседей и рассматривается значение ФП исходного решения f_c и значения ФП каждого из соседей f_i . Если $f_i > f_c$, то значения $M_s(f_c)$ и $M(f_c)$ увеличиваются на единицу ($M(f_c)$ – число мутаций особей со значением ФП, равным f_c), в противном случае увеличивается только значение $M(f_c)$. В процессе работы эволюционного алгоритма распределения $M_s(f_c)$ и $M(f_c)$ обновляются аналогичным образом.

Результаты. Полученные распределения могут быть использованы для динамического выбора значений некоторых параметров эволюционных алгоритмов в процессе их работы. В качестве примера рассматривается $(1, \lambda)$ -эволюционная стратегия, являющаяся одним из простейших эволюционных алгоритмов. В этом алгоритме популяция состоит из единственного решения. На каждой итерации из него строится λ соседей, лучший из которых заменяет текущее решение. В предложенной адаптивной модификации эволюционной стратегии значения параметра λ выбираются пропорционально отношению $M_s(f_c)$ и $M(f_c)$, где f_c – значение ФП текущего решения.

Тестирование предложенного алгоритма проводилось на примере построения конечных автоматов для задачи об «Умном муравье». Результаты запусков адаптивной эволюционной стратегии сравнивались с результатами запусков эволюционных стратегий с различными фиксированными значениями λ . Эксперименты показали, что ЭС с адаптивным параметром позволяет найти решение в среднем быстрее, чем ЭС с фиксированным параметром.

МЕТОД ПОСТРОЕНИЯ КОНЕЧНЫХ АВТОМАТОВ НА ОСНОВЕ МУРАВЬИНОГО АЛГОРИТМА

Д.С. Чивилихин, В.И. Ульянов

Научный руководитель – ассистент Ф.Н. Царев

В парадигме автоматного программирования выделяется система управления, представляемая в виде конечного автомата, и объект управления. В некоторых случаях автомат управления может быть построен эвристически, однако для большинства систем со сложным поведением построение автоматов вручную является весьма трудоемким. Для автоматического построения конечных автоматов применяются такие алгоритмы, как метод имитации отжига, эволюционные стратегии, генетические алгоритмы и обучение с подкреплением.

Конечный автомат задается множеством состояний S , стартовым состоянием s_0 , множеством входных событий Σ и множеством выходных воздействий Δ , а также функцией переходов $\delta: S \times \Sigma \rightarrow S$ и функцией действий $\lambda: S \times \Sigma \rightarrow \Delta$.

При решении задач управления вводится так называемая функция приспособленности автомата, значения которой тем больше, чем ближе поведение автомата управления к желаемому поведению. Следовательно, для получения автомата с желаемым поведением необходимо найти автомат, значение функции, приспособленности которого не меньше некоего наперед заданного числа.

В последние два десятилетия для решения самых разнообразных комбинаторных задач было разработано множество новых алгоритмов из класса алгоритмов роевого интеллекта, работа которых основана на моделировании поведения коллективов живых существ. Одним из подклассов алгоритмов роевого интеллекта являются муравьиные алгоритмы оптимизации, основанные на поведении колоний муравьев в процессе поиска пищи. Авторами был разработан новый алгоритм построения конечных автоматов, основанный на муравьином алгоритме.

Целью работы является экспериментальное исследование разработанного метода построения конечных автоматов, основанного на муравьином алгоритме, на хорошо известных задачах и сравнение характеристик его производительности с другими алгоритмами, такими, как генетические алгоритмы и эволюционные стратегии.

Постановка задачи. Пусть задано число состояний автомата N , множество входных событий Σ , множество выходных воздействий Δ и вещественная функция приспособленности f , заданная на множестве всех автоматов с параметрами (N, Σ, Δ) . Требуется найти автомат x такой, что значение функции приспособленности этого автомата $f(x)$ не меньше заданного барьерного значения b .

Основные положения исследования. Для решения поставленной задачи сведем ее к задаче поиска оптимальной вершины в некотором графе, вершинами в котором являются конечные автоматы, а ребрами – мутации конечных автоматов. Мутация конечного автомата – небольшое изменение в его структуре. В предлагаемом алгоритме используется два типа мутаций автоматов: для случайно выбранного перехода в автомате меняется либо состояние, в которое ведет этот переход, либо действие, которое на нем выполняется.

В муравьиных алгоритмах решения строятся набором агентов-муравьев, называемых муравьиной колонией. Каждый муравей переходит по ребрам графа от вершины к вершине, пользуясь неким стохастическим алгоритмом для выбора следующего ребра.

На каждом ребре графа задаются две величины, эвристическая информация и значение феромона. Эвристическая информация на ребре определяется как разность значений

функции приспособленности вершин, которые связывает ребро, в то время как значения феромона изменяются муравьями в процессе поиска.

Можно выделить следующие основные этапы работы алгоритма, которые повторяются, пока не будет найдено подходящее решение или не выполнится одно из условий останова:

1. Размещение муравьев по вершинам графа.

Каждый муравей помещается в одну из вершин графа, из которой он начнет строить решения. Одной из лучших стратегий является размещение всех муравьев в вершину, соответствующую текущему лучшему решению.

2. Построение решений муравьями.

Каждый муравей, независимо от других, строит решения, пока не израсходует выделенные ему ресурсы. На каждом шаге муравей с некоторой вероятностью выбирает следующее ребро из числа ребер, инцидентных текущей вершине, руководствуясь эвристической информацией и значениями феромона на этих ребрах. Иначе муравей выполняет некоторое число мутаций текущего решения и переходит во вновь построенную вершину, имеющую наибольшее значение функции приспособленности.

3. Обновление значений феромона.

Сначала значения феромона на ребрах графа, посещенных каждым муравьем, увеличиваются пропорционально значению функции приспособленности лучшего из решений, найденных этим муравьем. Затем происходит испарение феромона – значения феромона на всех ребрах графа уменьшаются в одинаковое число раз.

Результаты. Экспериментальное исследование разработанного алгоритма проводилось для широко известной задачи об «Умном муравье». Рассматривалось несколько конфигураций задачи при различных ограничениях на число ходов муравья и для различных размеров целевого автомата. Разработанный алгоритм сравнивался с генетическим алгоритмом, генетическим программированием и эволюционными стратегиями. В результате было установлено, что во всех случаях муравьиному алгоритму требуется меньше вычислений функции приспособленности для получения идеальных решений, что позволяет говорить о высокой эффективности разработанного метода.

УДК 004.4'242

МИНИМАЛЬНАЯ МОДИФИКАЦИЯ АВТОМАТНЫХ ПРОГРАММ ПРИ ИЗМЕНЕНИИ СЦЕНАРИЕВ ИХ РАБОТЫ

А.В. Шестаков

Научный руководитель – ассистент Ф.Н. Царев

Введение. Одним из подходов к программированию объектов со сложным поведением является использование парадигмы автоматного программирования [1] или «программирование с явным выделением состояний» [2]. При использовании этого подхода поведение программы задается в виде управляющих конечных автоматов. Состояния каждого автомата соответствуют состояниям управляемого объекта, а переходы между состояниями осуществляются при выполнении заданных условий.

Автоматное программирование хорошо подходит для создания событийно-ориентированных приложений [3, 4]. По каждому событию автомат в зависимости от состояния генерирует определенное выходное воздействие, а последовательность событий порождает последовательность выходных воздействий. Такая цепочка пар «событие – воздействие» называется сценарием работы автоматной программы.

Изменение сценариев работы требует модификации автоматной программы. Метод, предложенный в данной работе, позволяет изменить программу в соответствии с новыми

сценариями. Использование этого метода так же гарантирует минимальность изменений. Существуют методы, позволяющие построить автомат на основе определенного множества сценариев, например, с помощью генетических алгоритмов [5]. Предлагаемый метод, в отличие от разработанных ранее, не использует эволюционные алгоритмы.

Цель работы заключается в создании метода модификации автоматной программы, удовлетворяющего требованиям:

1. модифицированная программа должна удовлетворять измененному набору сценариев работы;
2. модифицированная программа должна минимально отличаться от исходной.

Метрика изменения автомата задается как $\mu = E + W \cdot S$, где E – число добавленных переходов; S – число добавленных состояний; W – вес добавления состояния.

Описание предлагаемого подхода. Суть метода заключается в сведении исходной задачи модификации автомата к хорошо изученной задаче поиска кратчайшего пути в графе, для чего по исходному автомату и добавляемому сценарию работы строится взвешенный граф. Кратчайший путь в этом графе будет соответствовать такому минимальному изменению автомата, что полученный автомат будет содержать требуемый сценарий.

Для построения графа рассматривается список воздействий добавляемого сценария. Каждому воздействию сопоставляется список состояний автомата, при входе в которые выполняется это воздействие. Далее список состояний, соответствующий отдельному воздействию сценария, называется слоем. Для того чтобы добавить сценарий, необходимо построить непротиворечивый путь через эти слои, причем разрешается использовать только одно состояние из слоя.

Если кратчайший путь найден, то остается добавить в автомат новые состояния, соответствующие вершинам, через которые проходит путь, и переходы, соответствующие ребрам пути ненулевого веса.

Результаты. В настоящей работе исследована проблема изменения автоматной программы при модификации множества ее сценариев работы. Поставлена конкретная задача достижения минимального изменения автоматной программы при добавлении сценария в рабочее множество. Для решения поставленной задачи предложен метод, который также может быть использован для построения автомата по набору сценариев работы.

Литература

1. Шалыто А.А. Парадигма автоматного программирования // Научно-технический вестник СПбГУ ИТМО. – 2008. – № 8 (53). – С. 3–24.
2. Поликарпова Н.И., Шалыто А.А. Автоматное программирование. – СПб: Питер, 2010. – 176 с.
3. Шалыто А.А. SWITCH-технология. Алгоритмизация и программирование задач логического управления. – СПб: Наука, 1998. – 628 с.
4. Шалыто А.А. Алгоритмизация и программирование для систем логического управления и «реактивных» систем // Автоматика и телемеханика. – 2001. – № 1. – С. 3–29.
5. Царев Ф.Н. Метод построения управляющих конечных автоматов на основе тестовых примеров с помощью генетического программирования // Информационно-управляющие системы. – 2010. – № 5. – С. 31–36.

**ПАРАЛЛЕЛЬНЫЕ И РАСПРЕДЕЛЕННЫЕ ТЕХНОЛОГИИ ДЛЯ ПОСТРОЕНИЯ
И ДЕКОДИРОВАНИЯ КМПЧ****И.А. Манушин****Научный руководитель – д.т.н., профессор Б.Д. Кудряшов**

Код с малой плотностью проверок на четность (LDPC-код от англ. Low-density parity-check code, LDPC-code, низкоплотностный код) – частный случай блочного линейного кода (далее - КМПЧ). Его основной особенностью является малая плотность значимых элементов проверочной матрицы, за счет чего достигается относительная простота реализации средств декодирования. Идея декодирования сводится к тому, чтобы производить вычисления только для ненулевых значений матрицы декодирования. На основании этого строится декодер, в котором, при фиксированной скорости кода, объем вычислений декодера линейно зависит от длины кодового слова. Декодирование строится в виде итеративного процесса, в котором на каждой итерации происходит вычисление вероятностей правильности битов в исходной последовательности. В конце итерации входное слово обновляется и вычисляется синдром. Если синдром не стал нулевым после определенного числа итераций (задается как параметр алгоритма), то слово считается не декодированным.

Особый интерес представляют коды большой длины, начиная от нескольких десятков тысяч бит во входной последовательности. Они обладают лучшими параметрами декодирования, но для их использования и исследования необходимы большие мощности, в том числе из-за памяти, так как изначально требуемый объем памяти пропорционален квадрату длины кода (при фиксированной скорости кода).

Одним из способов реализации декодирования кодов с большой длиной блока является использование видеопроцессора (GPU), в виду большого количества потоков и локального решения задачи декодирования. При разработке приложений, использующих ресурсы видеопроцессора, требуется модифицировать исходный алгоритм. Основная причина в том, что процессоры устройства не являются независимыми. Все операции одновременно выполняются группой процессоров (на современных видеокартах: от 48 процессоров в одной рабочей группе и больше). Выполнение инструкций и обращения к памяти приходится координировать в соответствии с этой архитектурой.

В результате работы была получена эффективная реализация алгоритма декодирования на видеопроцессоре. Средняя скорость декодирования для слов длины более 12000 бит была увеличена более чем в 10 раз, сохраняя возможность обработки нерегулярных кодов.

**ПОСТРОЕНИЕ ОПТИМАЛЬНОЙ СТРАТЕГИИ В ПОВТОРЯЮЩЕЙСЯ ДИЛЕММЕ
ЗАКЛЮЧЕННОГО ПРОТИВ ФИКСИРОВАННОГО МНОЖЕСТВА ПРОТИВНИКОВ
С ДЕТЕРМИНИРОВАННЫМИ СТРАТЕГИЯМИ С ПОМОЩЬЮ КОНЕЧНЫХ
АВТОМАТОВ НА ОСНОВЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ****М.Я. Пинский****Научный руководитель – доцент Д.В. Степанов**

Постановка задачи. Повторяющаяся дилемма заключенного (ПДЗ) – одна из ключевых задач теории игр, исследующая механизм возникновения кооперативных связей между индивидуумами, преследующими эгоистичные цели. Игра состоит из серии раундов, в каждом из которых два игрока осуществляют выбор между «предательством» и «кооперацией». Выигрыш игрока зависит как от его собственных действий, так и от действий

противника, при этом собственно «дилемма» состоит в том, что очки назначаются так, что взаимная кооперация выгоднее обоюдного предательства, но для каждого отдельного игрока рациональным выбором будет предательство. Если число раундов конечно и известно заранее, то рациональным выбором игрока будет постоянное предательство. Все, однако, меняется как только число раундов становится случайным, – множество рациональных стратегий существенно расширяется.

Весной 2012 года В. Прессом и Ф. Дайсоном (W.H. Press, F.J. Dyson «Iterated Prisoner's Dilemma contains strategies that dominate any evolutionary opponent») был предложен принципиально иной класс стратегий, основанных на интерпретации игры с помощью Марковских цепей и названных ZD-стратегиями (zero-determinant strategies). Предложенные стратегии вносят существенные изменения в суть игры: игрок, который использует ZD-стратегию, связывает свой выигрыш с выигрышем оппонента, придерживающегося эволюционной стратегии, определяя пропорцию, в которой выигрышные очки будут разделены между ним и противником.

Недостатком ZD-стратегий является то, что они не являются эволюционно устойчивыми – т.е. проигрывают сами себе и вытесняются из популяции. Единственный вариант при котором ZD-стратегия может закрепиться в популяции – это наличие у стратегии «теории ума» – способности распознавать в противнике игрока, придерживающегося ZD-стратегии для последующей кооперации с ним.

Цель работы. Создать эволюционно устойчивую стратегию для повторяющейся дилеммы заключенного в популяции, составленной из игроков с фиксированным набором стратегий. Такая стратегия должна правильно реагировать на действия противника: с разными стратегиями нужно играть по-разному, обеспечивая эволюционную стабильность.

В качестве стратегий, представленных в популяции, были выбраны а) всегда кооперирующийся игрок; б) всегда предающий; в) действующий случайно; г) повторяющий действия противника (стратегия «око за око»); а также д) ZD-стратегия.

Базовые положения исследования. Так как инструментом достижения цели работы является нахождение стратегии, распознающей стратегию противника, для описания искомой стратегии была выбрана концепция конечных автоматов, позволяющая корректировать выходное воздействие в зависимости от входного. Конечные автоматы строились с помощью генетических алгоритмов, описывающих эволюционный отбор наиболее успешных стратегий при заданной функции приспособленности.

Промежуточные результаты. Для решения поставленной задачи был реализован генетический алгоритм, который строит искомый конечный автомат. Функция приспособленности была определена как сумма средних выигрышей в попарных играх со всеми противниками, а так же с самим собой.

Полученный автомат умеет определять, с каким противником он играет и действует таким образом, чтобы получить максимальное количество очков. В случае, когда такой автомат играет сам против себя, он всегда кооперируется, обеспечивая тем самым наиболее выгодный для обоих игроков результат.

Основной результат, практические результаты. Реализована стратегия для повторяющейся дилеммы заключенного, максимизирующая число очков при игре со всеми противниками из заданного множества, включая и саму искомую стратегию, тем самым обеспечивая эволюционную стабильность.

ЭФФЕКТИВНАЯ ОБРАБОТКА ИНФОРМАЦИИ, ЗАДАННОЙ НА ЭЛЕМЕНТАХ ЕВКЛИДОВА ПРОСТРАНСТВА, ОТ РАЗЛИЧНЫХ ПРОВАЙДЕРОВ

А.А.Шевченко

Научный руководитель – к.т.н. Д.Д. Захаров

Краткое вступление, постановка проблемы. Испокон веков информация играла важную роль в жизни людей, но в последнее время роль ее, несомненно, возросла. Много усилий было потрачено на то, чтобы максимально быстро доставлять информацию, и человечество достигло громадного прорыва в этой области за последние десятилетия. Но, чаще всего, скорость доставки – не единственное требование к информации. При этом остается открытым вопрос точности и полноты полученной информации. Насколько точна информация, полученная от провайдера данных (провайдером может быть компания, веб-сервис, датчик и, вообще говоря, любой объект, генерирующий информацию и способный ее передать в соответствии с установленным протоколом)? Сомнения в этой области обычно устраняются путем добавления дополнительных провайдеров данных и последующей сверкой информации, получаемой из разных источников, между собой. Таким образом, возникает еще один уровень в процессе передачи информации – тот, на котором сверяется информация, полученная от разных провайдеров, и далее передается уже готовый продукт – три раза перепроверенная и дополненная информация.

Целью работы ставится оптимизации работы этого промежуточного уровня, впрочем, ограничившись лишь определенным типом передаваемой информации – информацией, привязанной к элементам Евклидова пространства. В современных условиях оптимизация работы этого уровня с необходимостью включает в себя рассмотрение возможности распараллеливания данной задачи.

Базовые положения исследования. На сегодняшний день написано достаточно много работ в сфере «Information fusion», в которых исследуется вопрос о том, каким именно образом сверять и перепроверять информацию, полученную из разных источников, с тем, чтобы увеличить ее достоверность и полноту. Но эта работа лежит в совершенно другой плоскости – она посвящена оптимизации слияния информации независимо от конкретного используемого способа.

На данный момент множество провайдеров данных на элементах Евклидова пространства небезосновательно считается гетерогенным, так что формат выходных данных у каждого провайдера чуть ли не уникальный. Мы сознаем этот факт, но полагаем, что это не норма, а всего лишь временно сложившаяся ситуация. В работе предлагается унифицированный формат выходных данных, так что для применения ее результатов потребуется, по крайней мере, написать адаптеры для существующих провайдеров данных.

Зафиксировав формат выходных данных провайдеров, мы исследуем возможность разбиения обрабатываемых данных для обеспечения равномерной нагрузки на задействованные процессоры.

Основной результат. В результате работы предложен экономичный способ представления информации, заданной на элементах Евклидова множества. Также разработан алгоритм слияния такой информации от нескольких провайдеров, работающий за $O(n/p)$, где n – количество элементов из Евклидова пространства, для которых предоставлена какая-то информация; p – количество процессоров. И, наконец, предлагается структура данных для хранения результатов обработки данных, поддерживающая быстрое выполнение вероятных запросов от клиента.