УДК 391

О ТРУДНО РЕШАЕМЫХ ЗАДАЧАХ КОМБИНАТОРНОГО ПОИСКА

А.А. Борисенко, *д-р техн. наук, профессор,*
Сумский государственный университет, г. Сумы

В работе исследуются задачи поиска одиночного объекта в условиях полной неопределенности. Определено правило сохранения информации в этих задачах и влияние ее на время решения задачи. Найдены условия, при которых время решения задачи поиска становится экспоненциальным, а при каких полиномиальным.

Ключевые слова: *информация, поиск, комбинаторика.*

У роботі досліджуються задачі пошуку одиночного об'єкта в умовах повної невизначеності. Визначено правило збереження інформації в цих задачах і вплив її на час розв'язання. Знайдені умови, при яких час розв'язання задачі пошуку стає експоненціальним, а при яких - поліноміальним.

Ключові слова: *інформація, пошук, комбінаторика.*

1 ВВЕДЕНИЕ И ПОСТАНОВКА ЗАДАЧИ

Трудно решаемые задачи, или НП-трудные задачи, – это широкий класс комбинаторных оптимизационных задач, сложность которых определяется большим временем их решения [1]. Попытки его уменьшения сталкиваются на сегодня с вопросом о принципиальной возможности такого уменьшения, на который пока что нет окончательного ответа. В данной работе предпринята попытка ответа на этот важный вопрос комбинаторики, основывающаяся на методах решения комбинаторных задач поиска и теории информации. Причем рассматриваются только такие задачи поиска, в которых по исходному предположению среди множества возможных решений предварительно существует искомое решение. К таким задачам можно отнести большое количество комбинаторных задач, и среди них особое место занимают задачи комбинаторной оптимизации, в которых среди других решений находится одно, представляющее искомый минимум или максимум [2,3]. Особое место среди таких задач занимает задача коммивояжера, так как к ней путем перекодирования сводится множество других оптимизационных задач [1]. Поэтому их решение является одновременно и решением задачи коммивояжера, которое вполне обоснованно сводится к задаче поиска, так как в последней задано конечное множество возможных решений (вариантов решения) и требуется среди этого множества найти одно, являющееся минимумом задачи.

Задачи как непрерывного, так и дискретного поиска встречаются во многих математических и технических приложениях, и поэтому их исследованию в научных работах уделяется достаточно большое внимание [4,5]. В данной работе исследуются лишь задачи дискретного поиска одиночного объекта среди конечного множества подобных ему объектов, представляющих собой возможные варианты решения, как, например, поиск фальшивой монеты, отличающейся по весу от остальных. Несмотря на простоту постановки данной задачи поиска, она отнюдь, как убедимся ниже, не тривиальна. Удобство данной задачи состоит в том, что в ее решении можно увидеть решения других более сложных задач поиска. Они вместе с рассматриваемой задачей часто встречаются в практических задачах, таких, как, например, задачи комбинаторной оптимизации или диагностики цифровых устройств. Однако при их решении до настоящего времени остаются открытыми вопросы, связанные с оценкой времени поиска объектов, и поэтому в данной работе предпринята попытка исследования в первую очередь этих вопросов, так как от их ответов зависит общий ответ на вопрос, имеют ли эти задачи реальное (полиномиальное) или реально недостижимое (экспоненциальное) решение. Казалось бы, что обилие работ на эту тему должно было бы уже давно ответить на поставленные вопросы, но это далеко не так, и сегодня вопросы, связанные с экспоненциальным и полиномиальным временем решения, не имеют окончательного ответа. Вообще-то есть гипотеза, по которой считается, что так называемые трудно решаемые или универсальные задачи не имеют полиномиального решения, но она долгое время не находит ни подтверждения, ни отрицания [1]. Не опровергая в целом этой гипотезы, все же можно сказать, что в некоторых частных случаях она неверна, и это подтверждают удачные решения, казалось бы, неразрешимых задач [1]. Конечно, в данной работе не ставится глобальная задача ответа на данную гипотезу. Но это лишь очередная работа автора в серии подобных работ [6], и возможно, что к концу этой серии можно будет дать определенный ответ, хотя бы только в виде некоторого продвижения к окончательному ответу.

В основу данного исследования положена теория информации как наиболее адекватная теория, отражающая сущность задач поиска. Она применима к задачам поиска в первую очередь потому, что эти задачи, как, впрочем, и любые другие математические задачи, предполагают множество возможных вариантов ответа на поставленные в них вопросы, а их решения сводятся к выбору *одного* из них. В процессе решения задачи специальный *тест,* анализируя каждый его вариант, снимает существовавшую *неопределенность* о нем, состоящую, как правило, в одном из ответов - да или нет, и тем самым производит преобразование этой неопределенности в *апостериорную* информацию J, то есть в информацию, получаемую в процессе решения. Отсутствие необходимости анализа и соответственно тестирования хотя бы некоторых вариантов решения говорит о том, что задача поиска в виде *априорной* информации I уже имеет частичное решение. Если же решение можно получить вообще без тестирования вариантов решения, путем только логического анализа, то это значит, что количество априорной информации, содержащейся в задаче, является достаточным для ее полного решения без получения апостериорной информации. Например, если задано множество перестановок, представляющих варианты решения задачи поиска, на которых задана весовая функция, принимающая одно и то же значение, за исключением одной, с другим весовым значением, то, перебирая эти перестановки, рано или поздно при их анализе придем к искомому решению. Это возможно потому, что при переборе и тестировании перестановок будет постепенно накапливаться апостериорная информация об искомой перестановке, пока в конечном итоге она полностью не будет получена. В результате снимается существовавшая неопределенность о решении задачи и взамен появляется соответствующая ей апостериорная информация. Априорная информация I в данном примере содержится в условии (ограничении), что допустимыми вариантами решения задачи могут быть только перестановки, тем самым из перебора исключаются другие комбинаторные объекты, не являющиеся перестановками, что сокращает время перебора. Такую априорную информацию I_d, которая содержится в условиях задачи, будем называть *детерминированной*. Однако априорная информация содержится не только в виде такой информации, а и в виде *вероятностной* информации I_v, вызванной разными вероятностями вариантов решения, представляющими собой вероятностные ограничения. Соответственно величина априорной информации $I = I_d + I_v$ бит. Если вероятности вариантов одинаковы, то это значит, что вероятностная информация в задаче отсутствует. Очевидно, что знание вероятностей вариантов решения задачи, при их наличии, может значительно ускорить процедуру поиска, что в ряде случаев и происходит на практике. Так, если вероятность одного из вариантов равна 1, а остальных – 0, то можно с уверенностью утверждать, что именно этот вариант и есть искомым. Выходит, что априорная информация, по сути, и предопределяет время решения задачи и чем ее больше, тем короче это решение.

Априорная информация – это та же апостериорная информация, только полученная до решения задачи. Поэтому ее наличие в задаче поиска уменьшает необходимое для ее решения количество апостериорной информации, связанное с перебором вариантов, что позволяет их сокращать, причем в ряде случаев довольно существенно. Отсюда следует, что вопрос наличия в условиях той или иной задачи поиска априорной информации является основным для снижения времени решения задачи, так как положительный ответ на него дает возможность в ряде случаев значительно повысить скорость нахождения искомого решения. Ответ на этот вопрос с точки зрения теории информации и является основной *задачей* данной работы.

2 СОХРАНЕНИЕ ИНФОРМАЦИИ В ЗАДАЧАХ ПОИСКА

Простейшей задачей поиска будет задача, состоящая в том, что среди N объектов – вариантов решения задачи поиска, выбираемых равновероятно, требуется найти один искомый объект – решение задачи, удовлетворяющее некоторым наперед заданным условиям. Будем для простоты считать, что каждый из объектов представим словом (кодовой комбинацией) длины n, буквы которого выбираются из алфавита, содержащего два символа – 0 и 1, и что вероятности каждого из объектов быть искомым равны между собой. Так как любой из этих объектов может оказаться искомым, то задача поиска будет обладать максимально возможной энтропией по отношению к этому решению:

$$H_{\max} = \log_2 N = \log_2 2^n = n. \tag{1}$$

Источником информации (апостериорной) в этой задаче выступает тест – логический оператор, проверяющий каждый из вариантов ее решения на принадлежность к искомому объекту. Именно при тестировании генерируется информация, снимающая неопределенность задачи и уменьшающая энтропию поиска. Причем тест в задачах поиска может принимать разные формы, например, тестом в задаче о фальшивой монете выступают показания весов, на которых взвешиваются разделенные на две группы монеты, среди которых в одной из групп находится фальшивая монета с большим (меньшим) весом, чем каждая из имеющих одинаковый вес остальных монет. Она находится в течение $n = \log_2 N$ взвешиваний (шагов решения). В алгоритмах и программах тестом выступает оператор условного перехода, проверяющий некоторое условие, в зависимости от результатов которого происходит разветвление программы в ту или иную сторону. Очевидно, что, последовательно выбирая объекты, рано или поздно, при неограниченном времени их тестирования будет найден искомый объект, обладающий теми или иными свойствами.

Максимальное количество информации, которое может быть получено в процессе тестирования при решении простейшей задачи, то есть при равновероятном поиске одного объекта среди множества из N объектов,

$$J_{\max} = H_{\max} = n. \tag{2}$$

Любая другая задача получается из простейшей задачи введением в нее дополнительных условий (ограничений), как детерминированных, так и вероятностных, и вследствие этого будет содержать $K \le N$ объектов. Соответственно она будет обладать величиной энтропии

$$H = \log_2 K\ , \tag{3}$$

меньшей, чем величина энтропии $H_{\max}$ в простейшей задаче. Эта энтропия в процессе решения задачи с помощью теста преобразуется в апостериорную информацию

$$J = \log_2 K\ . \tag{4}$$

Разница между энтропией простейшей задачи и полученной из нее с помощью ограничений энтропией конкретной задачи определяет величину априорной информации, содержащейся в задаче

$$I = H_{\max} - H\ . \tag{5}$$

Ее наличие позволяет уменьшить число шагов решения задачи, а при значениях $H = 0$ и соответственно $I = H_{\max}$ вовсе их исключить, так как искомый комбинаторный объект становится заранее известным. В общем же случае для полного решения задачи поиска необходимо получить как априорную информацию I, так и получаемую в процессе поиска апостериорную информацию J в суммарном количестве

$$J + I = J + I_d + I_v = H_{max}\ . \tag{6}$$

Если же полученное при поиске суммарное количество апостериорной информации и априорной информации будет меньше энтропии $H_{\max}$, то решение задачи поиска будет частичным, при котором будет сужено множество возможных вариантов решений по сравнению с исходным множеством.

Условие (6) можно сформулировать в виде правила постоянства (сохранения) информации в задачах поиска, утверждающего, что ***суммарная величина необходимой для поиска объекта априорной и апостериорной информации остается постоянной, каким бы способом не происходил этот поиск, и равна*** H_{max}.

Собственно это правило есть не что иное, как проявление закона сохранения информации в задачах поиска, на котором, хотя и в неявном виде, основывал свою теорию информации еще Шеннон. Суть этого закона состоит в утверждении, что уменьшение на определенную величину энтропии о том или ином событии всегда приводит к росту ровно на такую же величину информации о нем, и обратно, уменьшение величины информации о событии увеличивает ровно на такую же величину энтропию о нем [7].

3 АНАЛИЗ НР – ТРУДНЫХ ЗАДАЧ ПОИСКА

Из приведенного выше правила сохранения информации следует важный вывод, что задача поиска не может быть решена со значением априорной информации $I < H_{\max}$ без перебора ее вариантов с целью получения апостериорной информации в количестве $H_{\max} - I$ бит. Причем это количество для одного и того же значения априорной информации I

в соответствии с правилом сохранения информации остается для разных методов поиска неизменным. Особенности же самих методов решения задач поиска проявляются в способах использования априорной информации и тестирования объектов. Очевидно, что могут быть два диаметрально разных случая, когда априорная информация в задаче поиска отсутствует вообще и когда ее значение равно исходной энтропии $H_{\max}$ задачи. В первом случае задача поиска решается исключительно с помощью перебора вариантов решения, число которых растет с ростом длины соответствующих им кодовых последовательностей экспоненциально, а во втором она решается вообще без перебора путем логических умозаключений. Но очевидно, что существует практически неограниченное количество задач поиска с промежуточными соотношениями априорной и апостериорной информации, которые в процессе перебора получают недостающую для своего решения апостериорную информацию. От ее величины зависит время перебора, которое может быть как экспоненциальным, так и полиномиальным. Поэтому сама постановка вопроса к задаче поиска вообще, является ли она полиномиальной или экспоненциальной, не совсем корректна. Ответ может быть как утвердительный, так и отрицательный, так как все зависит от параметров индивидуальной задачи поиска. Это значит, что решение, например, широко известной задачи коммивояжера, в которой ищется перестановка с минимальным весом, может произойти как за экспоненциальное, так и за полиномиальное время.

Действительно, допустим, что весовые значения перестановок в задаче коммивояжера близки к одной и той же величине. Тогда в ней присутствует только детерминированная информация, формирующая перестановки, на которых в процессе перебора ищется оптимум, и количество перестановок, которые надо перебрать, чтобы получить недостающую апостериорную информацию и тем самым найти решение задачи, растет экспоненциально с ростом размерности задачи. Следовательно, в данном случае задача является экспоненциальной. Но может быть и другой противоположный случай, когда среди перестановок с разным весом имеется одна, вес которой близок к нулю. Очевидно, что именно она будет решением задачи коммивояжера. Тогда можно говорить о полиномиальном решении этой задачи. Но если это так, то можно предположить, что существует много промежуточных вариантов, для которых решение задачи коммивояжера близко к экспоненциальному или полиномиальному времени. Все зависит от соотношения количеств априорной и апостериорной информации, которые в совокупности при получении решения задачи поиска в соответствии с правилом сохранения информации должны всегда давать значение, равное $H_{\max}$ бит. Вопрос о возможности решения комбинаторной задачи поиска за полиномиальное время и в том числе задачи коммивояжера сводится к вопросу, имеется ли в ее условиях для этого достаточное количество априорной информации. Если да, то такое решение возможно, если нет, то невозможно. В задачах коммивояжера, как уже говорилось, есть априорная информация в виде условия, что решение задач ищется на перестановках, однако этой информации явно недостаточно для их решения за полиномиальное время. Поэтому нужны другие источники априорной информации, и в качестве такого источника выступают весовые значения перестановок. Очевидно, что они могут привести как к задачам с экспоненциальным временем решения, так и с полиномиальным временем, что как раз и иллюстрируют два приведенных выше примера. Других источников априорной информации, кроме источников, приведенных выше, в задачах коммивояжера не усматривается. То, что это так, подтверждают и практически полученные решения некоторых задач коммивояжера, одни из которых, при большей входной размерности, решаются за полиномиальное время, а другие, при меньшей размерности, за экспоненциальное время [1].

4 МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ПОИСКА

Как было показано выше, задачи поиска так или иначе используют для своего решения перебор и тестирование исследуемых вариантов решения на предмет принадлежности к искомому. Перебор может быть разный – *простой*, когда перебираются отдельные варианты, и *сложный*, когда могут перебираться и группы вариантов, что значительно ускоряет поиск. Алгоритмы сложного перебора разбивают исходное множество вариантов на содержащие больше, чем один вариант, классы эквивалентности (подмножества), обычно на два класса, один из которых тестируется на предмет наличия в нем искомого решения. При этом на каждом шаге разбиения с помощью теста анализируется один из классов на наличие в нем искомого варианта (решения). Класс, в котором отсутствует решение, в дальнейшем исключается из анализа, а класс, его содержащий, разбивается на очередные два класса и снова подвергается анализу на предмет нахождения в одном из них искомого варианта. Появление двух классов, содержащих по одному варианту решения, позволяет говорить об окончании на следующем шаге решения задачи поиска, так как оно представляется одним из двух оставшихся его вариантов. Очевидно, что решение, использующее рассматриваемую процедуру деления на классы эквивалентности, будет получено быстрее, чем при простом переборе. Недостаток такого решения - это усложненная процедура тестирования, требующая анализа не одного выбранного объекта, а целого их класса, что далеко не всегда удается реализовать на практике. Поэтому во многих случаях от сложного перебора приходится отказываться, заменяя его простым. Разбиение на классы эквивалентности вариантов исходного множества, при котором в каждом из классов находится одинаковое количество решений, является оптимальным, так как в этом случае при тестировании одного из двух подмножеств будет вырабатываться максимально возможное количество апостериорной информации – *один* бит. А так как для получения искомого решения в соответствие с правилом сохранения информации требуется n бит информации, то число разбиений, а значит и время поиска, будет минимально возможным, равным n шагов. При других разбиениях на классы эквивалентности с неравным количеством вариантов решений решение задачи поиска будет получено за число, большее, чем n шагов. Однако, как бы не происходило разбиение исходного множества на классы эквивалентности, количество получаемой в процессе тестирования информации, необходимой для поиска искомого объекта, в соответствии с правилом ее сохранения остается неизменным - n бит.

При простом переборе, когда тест обнаруживает искомый объект, выдается такая величина апостериорной информации, которая в сумме с ранее полученной апостериорной информацией и содержащейся в условиях задачи априорной информацией становится равной величине n бит. В противном случае, если тест не признает анализируемый вариант искомым, он отбрасывается, а для тестирования выбирается новый вариант. Это приводит к уменьшению числа тестируемых вариантов на единицу. Рано или поздно при переборе вариантов решений появится вариант, который тест признает искомым решением, и на этом процедура поиска будет закончена. Максимальное число шагов перебора, а значит, и максимальное время поиска будут в случае, когда будут протестированы $N-1$ вариантов решения из возможного числа N вариантов. Оставшийся последний вариант не тестируется, так как информация, полученная при тестировании предпоследнего варианта, позволяет оценить его как искомое решение. Если предпоследний вариант является искомым, значит, последний вариант не является таковым, и обратно, если предпоследний вариант не является искомым, то искомым является последний вариант. Это правило вытекает из условия, что рассматриваемые задачи поиска имеют решение, и, значит, оно находится в одном из возможных вариантов решения.

5 РАВНОВЕРОЯТНЫЙ ПОИСК

Рассмотрим вначале простой поиск, при котором в задаче поиска отсутствует априорная информация вообще, и любой вариант ее решения, взятый из N возможных вариантов, имеет одинаковую вероятность с любым другим стать искомым решением. Такой поиск будем называть *равновероятным*. Максимальное количество шагов перебора и соответственно тестирований при таком поиске будет равняться $N-1$, потому что при тестировании $N-1$ варианта при любом его исходе решение задачи поиска становится известным. При удачном тестировании, когда искомое решение найдено, это будет $N-1$ вариант, а при неудачном, когда тестирование не дает положительного результата, - оставшийся N вариант.

Решение задачи поиска происходит в процессе разбиения исходного множества вариантов на два класса эквивалентности, в одном из которых находится один вариант, а во втором – все остальные варианты, и тестирования класса с одним вариантом, что приводит к двум вероятностям нахождения искомого решения в том или ином классе. Одна из них, для удачного тестирования, равняется $1/N$, а вторая, для неудачного, - $(N-1)/N$. Соответственно количество апостериорной информации, получаемой в случае удачного тестирования, равно $log_2 N - log_2 1 = log_2 N$ бит, а в случае неудачного, - $log_2 N - log_2(N-1)$ бит.

При больших значениях N величина апостериорной информации, получаемой при неудачном тестировании, будет очень мала, однако при тестировании других вариантов она постепенно будет увеличиваться, пока не достигнет при максимальной длине поиска на последнем $N-1$ шаге решения величины в 1 бит, после которого процедура поиска успешно заканчивается. Разность энтропий при втором тестировании определяет следующую порцию апостериорной информации $log_2(N-1) - log_2(N-2)$ бит, которая уже будет больше первой. Последнее возможное тестирование, которое происходит на $N-1$ шаге, даст максимальную величину апостериорной информации $\log_2(N-(N-2)) - \log_2(N-(N-1)) = \log_2 2 - \log_2 1 = 1$ бит.

Если тестирование произошло удачно, то решение задачи поиска получено, и после него процедура поиска должна быть прекращена. Это значит, что суммарная величина апостериорной информации, полученной при удачном тестировании и на всех неудачных предшествующих шагах, в соответствии с правилом сохранения информации должна равняться

$$J = H_{\max} = \log_2 N . \tag{8}$$

Действительно, допустим, что удачный исход тестирования произошел на втором шаге тестирования. Тогда до апостериорной информации $log_2 N - log_2(N-1)$, полученной при первом неудачном шаге тестирования, следует добавить апостериорную информацию, полученную при удачном тестировании на втором шаге. Так как после первого неудачного тестирования количество неисследованных вариантов решения уменьшилось до величины $N-1$, то количество апостериорной информации при втором удачном тестировании стало равным $log_2(N-1)$ бит информации. Тогда суммарное количество апостериорной информации после удачного тестирования на втором шаге решения станет равным сумме значений $\log_2 N - \log_2(N-1)$ + $log_2(N-1)$ = $log_2 N$ бит. Оно остается неизменным и для третьего удачного тестирования $\log_2 N - \log_2(N-1) +$ $\log_2(N-1) - \log_2(N-2)$ + $log_2(N-2)$ = $\log_2 N$ бит. Аналогично можно найти такой же результат удачного тестирования на четвертом шаге, затем на пятом и так далее до $N-1$ шага. На этом последнем шаге в каждом классе эквивалентности будет находиться по $N-(N-1)=1$ варианту, и соответственно вероятность каждого из этих вариантов стать искомым решением будет равной 1/2. Общее же количество апостериорной информации, получаемой при удачном тестировании на $N-1$ шаге решения, остается таким же, как и при любом другом удачном шаге тестирования, - равным $log_2 N - log_2(N-1)+$ $+\log_2(N-1)-\log_2(N-2)$ + ... + $\log_2(N-(N-3))-\log_2(N-(N-2))$ + $+\log_2(N-(N-2))-\log_2(N-(N-1))=log_2 N$.

Приведенная цепочка величин апостериорных информаций, полученная на разных шагах при удачном тестировании на $N-1$ шаге, в силу равенства вероятностей удачного и неудачного тестирования, остается такой же и при неудачном тестировании на нем. Этим как раз объясняется возможность нахождения искомого решения на $N-1$ шаге, даже если тестирование на нем дает отрицательный результат.

Усредним величину апостериорной информации на каждом шаге решения задачи поиска, используя для этого формулу Шеннона для энтропии. Для первого шага энтропия и соответственно апостериорная информация будут равны

$$J = H = \frac{1}{N}\log_2\frac{1}{N} + \frac{N-1}{N}\log_2\frac{N-1}{N}. \qquad (9)$$

На остальных шагах поиска ее количество будет постепенно нарастать, пока на последнем $N-1$ шаге не достигнет величины в 1 бит. В целом же, как показано в [6], суммарная средняя величина полученной в процессе решения апостериорной информации на всех шагах тестирования будет равна $\log_2 N$ бит.

Ниже, на рис. 1, дан пример решения задачи поиска с $N=8$ вариантами. В этом примере показаны вероятности возможных результатов тестирования вариантов решения на различных шагах от первого до седьмого. Из него видно, что вероятности получения искомого варианта при неудачных тестированиях постепенно уменьшаются от величины 7/8 на 1-м шаге поиска до 1/2 на последнем 7-м шаге. Соответственно увеличивается количество получаемой при тестировании апостериорной информации от минимального значения $\log_2 8/7$ до максимального значения в 1 бит на последнем 7-м шаге. Именно на этом шаге будет в обязательном порядке определен искомый вариант, если он не был до этого найден при удачном тестировании на одном из возможных предыдущих шагов.

Недостаток простого перебора очевидный – большое время поиска, так как при его реализации необходимо последовательно перебирать возможные варианты решения до появления искомого. В соответствии с правилом сохранения информации ее суммарное количество, полученное на всех шагах решения с учетом априорной информации I, будет равно n бит, что позволяет найти искомое решение задачи. То, что оно будет найдено, следует из самой сути метода простого перебора, который в процессе последовательного тестирования не пропустит ни одного возможного варианта решения, а значит, рано или поздно на одном из числа $N-1$ шагов выйдет на искомый вариант.

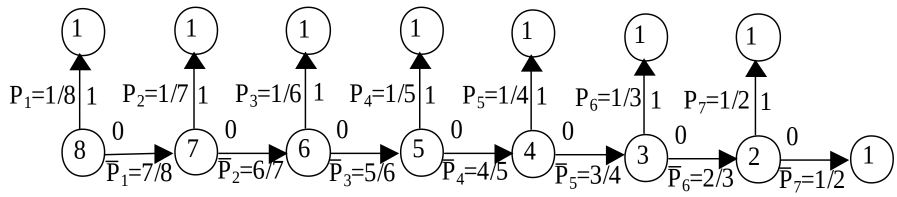

Рисунок 1 – Пример решения задачи поиска с помощью простого перебора

Наличие в переборной задаче априорной информации I приводит к тому, что в соответствии с правилом сохранения информации величина требуемой апостериорной информации J, необходимой для решения задачи поиска, снизится ровно на величину имеющейся в задаче априорной информации. Это снижение приведет к уменьшению времени решения задачи, так как исходное число вариантов решения уменьшается до числа $K \le N$. Следовательно, чем больше величина априорной информации I, тем меньше будет время решения задачи. Действительно, если, например, дана априорная информация I, что искомый объект находится среди двоичных комбинаций длины n, обладающих одним и тем же количеством единиц (весом), то очевидно, что время поиска сузится, так как из процедуры перебора и тестирования исключаются все комбинации с другим числом единиц в двоичной комбинации, чем в заданном.

6 ОЦЕНКА ВРЕМЕНИ ТЕСТИРОВАНИЯ ЗАДАЧ ПОИСКА

Так как число шагов перебора до появления искомого решения при наличии априорной информации может быть равным 1, 2, ... , K – 1, то среднее их число до появления искомого решения

$$K_{cp} = \frac{K}{2}. \tag{10}$$

Следует из того, что сумма шагов перебора

$$1+2+3+\ldots+K-1=\frac{(K-1)K}{2}. \tag{11}$$

При делении на максимальное число шагов поиска K – 1 она даст полученный выше в (8) результат. Очевидно, что при отсутствии в задаче поиска априорной информации, то есть $K = N$, среднее число шагов до ее решения

$$N_{cp} = \frac{N}{2}. \tag{12}$$

При сложном равновероятном поиске, когда получаемые при разбиении классы эквивалентности содержат больше чем один вариант, решение задачи поиска можно найти значительно быстрее, так как вместо анализа одного его варианта проводится анализ сразу нескольких вариантов на предмет наличия среди них искомого.

На рис. 2. показан сложный равновероятный поиск, при котором разбиение происходит на классы, содержащие одинаковое количество вариантов решений.

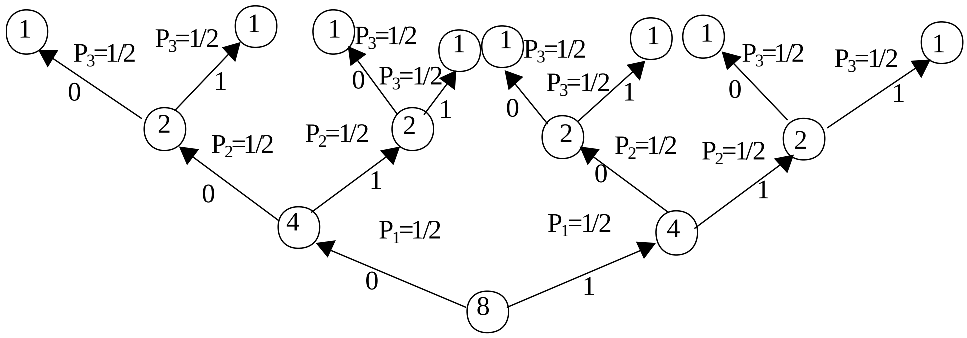

Рисунок 2 – Граф-схема процедуры поиска с помощью разбиения на группы

Это наиболее эффективный поиск, приводящий к решению задачи поиска за минимальное среднее количество шагов

$$N_{ср} = \log_2 N = \log_2 2^n = n. \qquad (13)$$

Очевидно, что для рассматриваемого примера

$$N_{ср} = log_2 8 = log_2 2^3 = 3.$$

Сравнение крайних значений среднего времени простого и сложного равновероятного поиска при отсутствии в соответствующей задаче априорной информации показывает, что коэффициент, определяющий их отношение,

$$k = \frac{N}{2\,log_2 N} = \frac{2^n}{2\,log_2 2^n} = \frac{2^{n-1}}{n} \qquad (14)$$

растет экспоненциально с ростом n. Из этого свойства вытекает закономерный вывод, что всегда, если это возможно, с целью повышения скорости решения задачи поиска следует применять сложный поиск, при котором следует разбивать, по возможности, варианты решения на группы, состоящие из примерно одинакового их количества.

И простой, и сложный поиск можно представить в виде процедуры кодирования, при которой каждому варианту поиска будет соответствовать своя кодовая комбинация, в рассматриваемом случае двоичная. Это позволяет задачу поиска свести к задаче кодирования, что в ряде случаев может облегчить решение исходной задачи. Кодовые комбинации для рассматриваемого примера простого и сложного поиска показаны на рис. 1 и 2 соответственно. Для простого поиска это будут комбинации: 1, 01, 001, 0001, 00001, 000001, 0000001, 0000000, длина двух максимальных из которых равна 7. Очевидно, что представленный код является неравномерным, в котором ни одна комбинация не может быть началом другой. Для сложного поиска код представлен следующими комбинациями: 000, 001, 010, 011, 100, 101, 110, 111. Это уже будет равномерный код с длиной кодовых комбинаций, равной 3. Сравнение этих кодов показывает, что поиск на основе равномерного кода, с точки зрения времени, более предпочтительный, чем на основе неравномерного, хотя неравномерный код дает возможность обнаружить решение задачи уже на первом шаге кодирования, а равномерный всегда только на 3-м шаге.

Не вызывает сомнения, что между двумя крайними рассмотренными способами поиска могут располагаться и другие, в которых разбиения происходят на классы эквивалентности с разным числом от 1 до n вариантов решений. Число этих способов определяется числом соответствующих этим кодам деревьев, но это задача уже следующей статьи, являющейся продолжением данной.

Введение априорной информации в задачу поиска в соответствии с правилом сохранения информации уменьшает величину, необходимой для ее решения апостериорной информации вплоть до нулевого значения, при котором перебор вариантов, как таковой, отсутствует вообще. Результат поиска тогда указывается сразу за один шаг анализа. Если же все-таки перебор в задаче необходим, то это значит, что количество содержащейся в ней априорной информации меньше $log_2 N$ и ее надо добрать с помощью перебора.

7 ВЫВОДЫ

В основе решения комбинаторных задач поиска, в том числе и трудно решаемых, лежат два вида информации – априорная и апостериорная. В сумме эти два вида информации образуют ее общее количество, которое для любого метода решения задачи поиска с одной и той же размерностью постоянно. Апостериорная информация генерируется во время перебора и тестирования вариантов решения задачи поиска, то есть в процессе решения задачи, а априорная находится в ее условиях до решения и может быть использована для сокращения времени перебора и тестирования. От соотношения априорной и апостериорной задачи зависит время решения задачи поиска – чем больше в задаче априорной информации, тем время ее решения будет меньшим, и обратно, чем меньшим будет количество, тем большим будет время ее решения. Никакие существующие на сегодня методы или созданные в будущем не могут уменьшить минимальное время решения, задаваемое априорной информацией задачи.

НП-трудные задачи, такие, как задача коммивояжера, содержат априорную информацию в двух условиях, что в качестве вариантов их решения используются перестановки и что каждая из этих перестановок обладает своим весовым коэффициентом. Именно соотношения весовых коэффициентов в первую очередь могут существенно повлиять на уменьшение времени решения задачи, вплоть до полиномиального времени. Следовательно, решение задачи коммивояжера и любых трудно решаемых задач может произойти в зависимости от условий индивидуальной задачи как за полиномиальное, так и за экспоненциальное время.

8 ЗАКЛЮЧЕНИЕ

Однако описание самого метода нахождения решения в задачах поиска, использующего содержащуюся в них априорную информацию, выходит за рамки данной работы, но будет рассмотрен в одной из последующих статей. Можно лишь сказать, что по идее этот метод должен обеспечить максимально возможную скорость решения задачи, которую ни один из ныне существующих методов не сможет превысить. Кроме того, он должен обладать максимально возможной универсальностью. Недостаток такого метода по сравнению с существующими методами решения задачи коммивояжера – это повышенная сложность его реализации. Существенным требованием к такому методу является его практическая апробация, которая может затянуться на неопределенно долгое время.

SUMMARY

ON DIFFICULT-TO-SOLVE TASKS OF COMBINATORIAL SEARCHING

A.A. Borysenko
Sumy State University, Sumy

The author studies the problem of finding a single object in a state of limbo. There is defined an information storage rule for such problems and its influence on the time of solution. The author defines the conditions under which the time of the solution search becomes exponential, and under which - polynomial.

Key words: *information, searching, combinatorics.*

СПИСОК ЛИТЕРАТУРЫ

1. Рейнгольд Э. Комбинаторные алгоритмы. Теория и практика; пер. с англ./ Э. Рейнгольд, Ю. Нивергельт, Н. Део. – М.: Мир, 1980. – 476 с.
2. Липский В. Комбинаторика для программиста / В. Липский. — М.: Мир, 1988. — 213 с.
3. Андерсон Джеймс. Дискретная математика и комбинаторика = Discrete Mathematics with Combinatorics / Джеймс Андерсон. — М.: Вильямс, 2006. — С. 960.
4. Альсведе Р. Задачи поиска; пер. с нем. / Р. Альсведе, И. Вегенер. – М.: Мир, 1982. - 368 с.
5. Аржененко А.Ю. Оптимальные бинарные вопросники / А. Ю.Аржененко, Б. Н. Чугаев. – М.: Энергоатомиздат, 1989. – 128 с.
6. Борисенко А.А. Об информационной оценкt эффективности решения переборных задач / А.А. Борисенко // Вестник СумГУ. – 2002. - №13(46). - С. 177-184.
7. Шеннон К. Работы по теории информации и кибернетике / К. Шеннон. – М.: Изд-во иностр. лит., 1963.

Поступила в редакцию 29 ноября 2010 г.