

Krassimir Markov, Vitalii Velychko, Oleksy Voloshin
(editors)

**Information Models
of
Knowledge**

**ITHEA[®]
KIEV – SOFIA
2010**

Krassimir Markov, Vitalii Velychko, Oleksy Voloshin (ed.)

Information Models of Knowledge

ITHEA®

Kiev, Ukraine – Sofia, Bulgaria, 2010

ISBN 978-954-16-0048-1

First edition

Recommended for publication by The Scientific Council of the Institute of Information Theories and Applications FOI ITHEA
ITHEA IBS ISC: 19.

This book maintains articles on actual problems of research and application of information technologies, especially the new approaches, models, algorithms and methods for information modeling of knowledge in: Intelligence metasynthesis and knowledge processing in intelligent systems; Formalisms and methods of knowledge representation; Connectionism and neural nets; System analysis and synthesis; Modelling of the complex artificial systems; Image Processing and Computer Vision; Computer virtual reality; Virtual laboratories for computer-aided design; Decision support systems; Information models of knowledge of and for education; Open social info-educational platforms; Web-based educational information systems; Semantic Web Technologies; Mathematical foundations for information modeling of knowledge; Discrete mathematics; Mathematical methods for research of complex systems.

It is represented that book articles will be interesting for experts in the field of information technologies as well as for practical users.

General Sponsor: Consortium FOI Bulgaria (www.foibg.com).

Printed in Ukraine

Copyright © 2010 All rights reserved

© 2010 ITHEA® – Publisher; Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org ; e-mail: info@foibg.com

© 2010 Krassimir Markov, Vitalii Velychko, Oleksy Voloshin – Editors

© 2010 Ina Markova – Technical editor

© 2010 For all authors in the book.

® ITHEA is a registered trade mark of FOI-COMMERCE Co., Bulgaria

ISBN 978-954-16-0048-1

C/o Jusautor, Sofia, 2010

ПАРАЛЛЕЛИЗАЦИЯ КООПЕРАТИВНЫХ МОДЕЛЕ-ОРИЕНТИРОВАННЫХ МЕТАЭВРИСТИК

Сергей Сиренко

Аннотация: Рассматривается кооперативный метаэвристический подход к решению задач комбинаторной оптимизации на основе модели-ориентированных алгоритмов. Его особенностью является решение задачи путем поиска (оптимизации) в пространстве моделей, который проводится на основе частных моделей, сформированных базовыми (составными) алгоритмами. Предлагается схема распараллеливания кооперативного метода, использующая присущий ему естественный параллелизм. Проведено исследование эффективности предлагаемой схемы параллелизации на основе анализов результатов вычислительного эксперимента.

Ключевые слова: комбинаторная оптимизация, модели-ориентированные методы, кооперативные метаэвристики, параллельные алгоритмы, оптимизация муравьиными колониями.

ACM Classification Keywords: G.1.6 [NUMERICAL ANALYSIS] Optimization, I.2.8 [ARTIFICIAL INTELLIGENCE]: Problem Solving, Control Methods, and Search – Heuristic methods, General Terms: Algorithms.

Введение

В работе рассматривается метод построения кооперативных метаэвристик на основе модели-ориентированных алгоритмов [Гуляницкий, 2009]. Особенностью этого подхода к построению кооперативных методов решения задач комбинаторной оптимизации (ЗКО) является наличие управляющего алгоритма агрегирования (оптимизации) частных моделей, сформированных базовыми (составными) алгоритмами. В следующем разделе описана общая схема таких методов, далее излагаются детали разработанного подхода к реализации таких алгоритмов на многопроцессорных вычислительных системах. Результаты и обсуждение проведенного исследования эффективности предлагаемого способа распараллеливания на основе вычислительного эксперимента приведены в предпоследнем разделе.

Построение кооперативных гибридных метаэвристик

Рассмотрим проблему построения гибридных кооперативных метаэвристик, которые базируются не на одном, а на нескольких модели-ориентированных алгоритмах; пусть K – число таких алгоритмов, которые назовем базовыми. Будем считать, что каждый такой алгоритм оперирует со своей моделью $M_k, k = 1, \dots, K$, а в результате их деятельности формируется один или несколько вариантов решения.

Рассмотрим одну из возможных реализаций. Предположим, что работа алгоритмов может осуществляться асинхронно, а их взаимодействие на итерации h происходит путем формирования вышестоящей процедурой агрегированной модели (метамодели) M^{h+1} как с учетом отдельных моделей M_1^h, \dots, M_K^h и сформированной на предыдущей итерации модели M^h , так и, возможно, сгенерированных алгоритмами решений. Таким образом, кооперативная метаэвристика оперирует несколькими моделями и осуществляет оптимизацию в пространстве моделей, с целью предоставления возможности базовым алгоритмам (или некоторым из них) генерировать наилучшие решения ЗКО.

Предлагается методология создания метаэвристических алгоритмов решения ЗКО, общая схема которых состоит из этапов, представленных на Рис. 1 [Гуляницкий и Сиренко; 2009]. Здесь для простоты изложения при формировании агрегированной модели (метамодели) не отражен явно фактор учета решений, сгенерированных базовыми алгоритмами.

Опишем ключевые этапы кооперативного алгоритма. После этапа инициализации осуществляется запуск всех базовых алгоритмов, каждый из которых итерационно и независимо генерирует решения и обновляет свою собственную модель. При выполнении условий обмена, текущие модели базовых алгоритмов (и, возможно, найденные ими варианты решений) используются для формирования новой агрегированной модели (метамоделей). При этом может также использоваться агрегированная модель с предыдущей итерации обмена, а сам процесс формирования может быть представлен в виде оптимизационной проблемы поиска наилучшего элемента в пространстве моделей.

```

procedure БазовыйАлгоритм( $k, M$ )
  Инициализация;
  while (не завершена работа) do
    ГенерированиеВариантовРешения ( $M_k, M$ );
    ОбновлениеМодели ( $M_k$ );
  end while
  return НаилучшееНайденноеРешение;
end procedure
procedure УправляющаяМетаэвристика()
  Инициализация;
  for  $k := 1$  to  $K$  do
    Запустить(БазовыйАлгоритм( $k, M^1$ ));
  end for
   $h := 1$ ;
  while (не выполняется условие завершения) do
    if (выполняется условие обмена) then
      for  $k := 1$  to  $K$  do
         $M_k^h :=$  ПолучитьМодель( $M_k$ );
      end for
       $M^{h+1} :=$  СформироватьМодель ( $M^h, M_1^h, \dots, M_K^h$ );
      for  $k := 1$  to  $K$  do
        РазослатьМодель( $k, M^{h+1}$ );
      end for
       $h := h + 1$ ;
    end if
  end while
  ЗавершитьРаботуАлгоритмов();
  return НаилучшиеНайденныеРешения;
end procedure

```

Рис. 1 Схема кооперативной модели-ориентированной метаэвристики

После этого агрегированная модель рассылается базовым алгоритмам, где может, как использоваться в сочетании с их собственной моделью, так и заместить ее полностью. При выполнении условий завершения работы кооперативная метаэвристика возвращает один или несколько наилучших из найденных базовыми алгоритмами вариантов решения.

В качестве базовых алгоритмов могут выступать отдельные экземпляры одного алгоритма, в этом случае схема называется гомогенной. Экземпляры алгоритмов могут иметь различные значения параметров. В случае гетерогенной схемы (когда базовые алгоритмы не основываются на одном алгоритме/методе), модели базовых алгоритмов должны быть согласованы, т.е. должен существовать способ преобразования значений параметров и/или структур моделей между собой.

Управляющая метаэвристика может формировать на одной итерации не одну, а сразу несколько разных агрегированных моделей (метамоделей). Это целесообразно как в случае гетерогенной схемы, когда

базовые алгоритмы могут использовать модели разной структуры, так и когда требуется исследовать несколько различных моделей одним алгоритмом.

При использовании данной методологии необходимо определить:

1. тип(ы) базовых модели-ориентированных алгоритмов;
2. способ использования базовыми алгоритмами агрегированной модели (метамодели);
3. условия обмена (схема коммуникации);
4. способ формирования агрегированной модели (метамодели);
5. условия завершения работы кооперативной метаэвристики.

Предполагается, что применение разработанной методологии позволит диверсифицировать работу базовых алгоритмов, что уменьшает вероятность завершения процесса поиска в областях, не содержащих глобально-оптимальное решение. Обмен информацией между базовыми алгоритмами позволяет надеется на повышение эффективности процесса поиска за счет проявления синергетических эффектов. Таким образом, выбор конкретной схемы обмена информацией (способа кооперирования) определяет баланс между интенсификацией и диверсификацией поиска.

Разработанная схема может также служить основой для классификации гибридных алгоритмов КО, например, на основе параметризации таких показателей, как синхронность выполнения и схемы взаимодействия базовых алгоритмов, их однотипность, количество используемых моделей и методы формирования агрегированных моделей, степень использования предыстории.

В этой работе исследуется применение в качестве процедуры агрегирования схемы взвешенного суммирования значений параметров (феромонных значений) моделей базовых алгоритмов τ_1, \dots, τ_K (для простоты изложения будем считать, что значения параметров нормированы). Веса вычисляются на основе значений целевой функции на текущих наилучших вариантах решений $f_1^{opt}, \dots, f_K^{opt}$ и средних значений целевой функции $f_1^{avg}, \dots, f_K^{avg}$ по найденным базовыми алгоритмами решениям на последней завершенной итерации по следующему соотношению:

$$\omega_k = \frac{0.2}{K} + 0.6 \frac{f_{\max}^{opt} - f_k^{opt}}{\sum_{i=1}^K (f_{\max}^{opt} - f_i^{opt})} + 0.2 \frac{f_{\max}^{avg} - f_k^{avg}}{\sum_{i=1}^K (f_{\max}^{avg} - f_i^{avg})}, k = 1, \dots, K; \quad (1)$$

$$f_{\max}^{opt} = \max_{k=1, \dots, K} f_k^{opt}, f_{\max}^{avg} = \max_{k=1, \dots, K} f_k^{avg}.$$

Поскольку исследовалось влияние наличия обмена информацией в кооперативной схеме на эффективность, то в качестве условия завершения было выбрано выполнение базовыми алгоритмами определенного количества итераций IT_{\max} .

Распараллеливание кооперативной метаэвристической схемы

Эффективное распараллеливание метаэвристик является нетривиальной задачей и стало отдельной областью исследования [Parallel Metaheuristics, 2006]. Представленная кооперативная схема обладает естественным параллелизмом – она включает набор подчиненных алгоритмов выполняющихся одновременно со сравнительно низким обменом информации между ними. В общем случае наиболее целесообразным представляется не разбивать базовые алгоритмы между процессорами, чтобы минимизировать накладные расходы коммуникации. Поэтому мы предлагаем синхронную схему «главный-подчиненный» для распараллеливания кооперативной схемы, использующий представило агрегирования (1) (Рис. 2). Каждому процессору присваивается один или несколько базовых алгоритмов. Один процессор выделяется как главный и дополнительно выполняет операций управляющей метаэвристики.

Предлагается согласованная схема распределения K базовых алгоритмов на P процессорах ($K \geq P$), который использует выполнение базовыми алгоритмами заданного количества IT итераций, как условие

выполнения обмена и агрегирования. В предлагаемой схеме все базовые модели берутся для агрегации с одной и той же итерации базовых алгоритмов.

Процедура агрегирования также частично распараллелена. Подчиненные процессоры отправляют главному информацию об их текущих лучших и средних значениях целевой функции. Главный процессор вычисляет веса и отправляет их подчиненным. Они нормализуют и суммируют свои базовые модели (если им присвоено больше одного основного алгоритма) и отправляют результат главному процессору. Он завершает процесс агрегирования и передает агрегированную модель подчиненным процессорам.

Вычислительный эксперимент

Поскольку теоретическое исследование метаэвристических алгоритмов решения ЗКО крайне редко позволяет получать практически применимые результаты, принято анализировать показатели эффективности путем проведения вычислительных экспериментов. С этой целью обычно используют "классические" модели комбинаторной оптимизации – такие, например, как задача коммивояжера (ЗК) [Hoos and Stützle, 2005], которая состоит в поиске минимального гамильтонового цикла в полном взвешенном графе.

Проведенный эксперимент состоял в исследовании эффективности разработанной схемы распараллеливания для кооперативного алгоритма на базе алгоритмов оптимизации муравьиными колониями [Гуляницкий и Сиренко; 2009]. В качестве базовых алгоритмов при реализации были использованы экземпляры известного алгоритма Max-Min Ant System (MMAS) с реинициализацией феромонных значений [Stützle and Hoos, 2000], реализация которых взята из пакета ACOTSP [Stützle, 2004]. Феромонные значения реинициализируются, если решения сгенерированные на одной итерации достаточно близки друг к другу и на протяжении заданного количества итераций алгоритмом не было найдено улучшения. Модель в этих алгоритмах ОМК для ЗК представлена в виде квадратной матрицы, элементами которой являются феромонные значения. В алгоритме MMAS используются динамические ограничения τ_{\max}, τ_{\min} на значения элементов феромонной матрицы (параметров модели), поэтому перед агрегированием матриц осуществлялось масштабирование элементов с приведением их значений к отрезку $[0, 1]$: $\tau_i^{norm} = (\tau_i - \tau_{\min}) / (\tau_{\max} - \tau_{\min})$.

```

procedure ПодчиненныйПроцесс(pr_id)
     $K_{pr\_id} = K \text{ div } P$ ;
    if ( $P \text{ mod } K \geq pr\_id$ ) then
         $K_{pr\_id} = K_{pr\_id} + 1$ ;
    end if
    Инициализация( $M, M_1, \dots, M_{pr\_id}$ );
    while (не завершена работа) do
        for  $i := 1$  to  $IT$  do
            for  $k := 1$  to  $K_{pr\_id}$  do
                ГенерированиеВариантовРешения( $M_k, M$ ); ОбновлениеМодели( $M_k$ );
            end for
            end for
            ОтправитьГлавномуПроцессу( $f_1^{opt}, f_1^{avg}, \dots, f_{K_{pr\_id}}^{opt}, f_{K_{pr\_id}}^{avg}$ );
            ПолучитьВеса( $\omega_1, \dots, \omega_{K_{pr\_id}}$ );
             $M^{temp} :=$  НормироватьИАгрегировать( $M_1, \omega_1, \dots, M_{K_{pr\_id}}, \omega_{K_{pr\_id}}$ );
            ОтправитьГлавномуПроцессу( $M^{temp}$ );
            ПолучитьМодель( $M$ );
        end while

```

```

end procedure
procedure ГлавныйПроцесс()
     $K_1 = K \text{ div } P$ ;
    Инициализация ( $M, M_1, \dots, M_{pr\_id}$ );
     $h := 1$ ;
    while (не выполняется условие завершения) do
        for  $i := 1$  to  $IT$  do
            for  $k := 1$  to  $K_1$  do
                ГенерированиеВариантовРешения ( $M_k, M$ ); ОбновлениеМодели ( $M_k$ );
            end for
        end for
        ПолучитьЗначения( $f_{K_1+1}^{opt}, f_{K_1+1}^{avg}, \dots, f_K^{opt}, f_K^{avg}$ );
        ВычислитьВеса( $\omega_1, \dots, \omega_K$ );
        РаспределитьМеждуПодчиненнымиПроцессами( $\omega_{K_1+1}, \dots, \omega_K$ );
         $M_1^{temp} :=$  НормироватьИАгрегировать( $M_1, \omega_1, \dots, M_{K_1}, \omega_{K_1}$ );
        ПолучитьЧастичныеМодели ( $M_2^{temp}, \dots, M_K^{temp}$ );
         $M^{h+1} :=$  Агрегировать( $M^h, M_1^{temp}, \dots, M_K^{temp}$ );
        РазовлатьМодель ( $M^{h+1}$ );
         $h := h + 1$ ;
    end while
    ЗавершитьРаботуАлгоритмов();
end procedure

```

Рис. 2 Схема параллельной кооперативной модели-ориентированной метаэвристики

Параметры базовых алгоритмов устанавливались равными стандартными значениям, рекомендуемыми в литературе для ЗК [Dorigo and Stützle, 2004], т.е. дополнительная их оптимизация для конкретного набора тестовых ЗК не проводилась. Количество муравьев составляло 25, коэффициент испарения феромона $\rho = 0.5$, в псевдослучайном пропорциональном правиле выбора, которое применяли муравьи при построении решений, были такие значения параметров: $\alpha = 1$, $\beta = 2$. В качестве деятельности демона в алгоритмах ОМК ко всем построенным муравьями решениям применялся алгоритм простого локального поиска 3-opt [Hoos and Stützle, 2005], реализация которого также взята из пакета ACOTSP [Stützle, 2004]. Формирование управляющей метаэвристикой агрегированной модели осуществлялось после каждых 60 итераций, выполненных базовыми алгоритмами.

Тесты проводились на кластере СКИТ-3 Института кибернетики им. В.М. Глушкова. Была использована часть кластера с узлами, состоящими из двух четырех-ядерных процессоров Intel Xeon 5345 EM64T (2,2 ГГц на ядро) с 16 Гб оперативной памяти на узел. Алгоритмы были запрограммированы на языке C++. Для организации обмена сообщениями в параллельной версии использовалась реализация стандарта MPI версии OpenMPI 1.2.4.

Исследовались кооперативные алгоритмы, состоящие из 2, 5, 10 и 25 копий алгоритма MMAS. Три ЗК att532, u1432, pr2392 размерностей 532, 1432 и 2392 городов соответственно были отобраны, и замерялось среднее время необходимое алгоритмам для того, что бы достигнуть точности 0,75% относительно известного оптимального решения. Средние значения вычислялись на основе 30 попыток.

В табл. 1 приведены эффективности алгоритмов на 2, 5, 10 и 25 процессорах (ядер). Тут K – число базовых алгоритмов, а P – число задействованных процессоров. Конфигурации, для которых $K < P$ не рассматривались. Эффективности вычисляются на основе времени относительного ускорения (ускорение параллельного кода относительно того же кода запущенного на одном ядре).

Таблица 1 Результаты решения ЗК

Задача	att532				u1432				pr2392				
	<i>K</i>	2	5	10	25	2	5	10	25	2	5	10	25
<i>P</i>													
2		0.88	1.10	0.89	1.00	0.86	1.07	0.91	0.98	0.85	1.05	0.97	0.95
5		–	0.81	0.84	0.87	–	0.67	0.70	0.87	–	0.48	0.53	0.64
10		–	–	0.67	0.97	–	–	0.59	0.94	–	–	0.40	0.61
25		–	–	–	0.65	–	–	–	0.52	–	–	–	0.34

В целом ускорение параллельного алгоритма близко к линейному для меньших значений количества процессоров и базовых алгоритмов. Однако при возрастании размерности наблюдается тенденция к значительному снижению эффективности параллельного алгоритма. Также отметим, что более целесообразно присваивать одному процессору (ядру) несколько базовых алгоритмов для снижения расходов ресурсов на обмен сообщениями.

Для 2 ядер и 5 базовых алгоритмов наблюдается сверхлинейное ускорение. Это факт объясняется тем, что в этом случае главному ядру присваивается 2, а подчиненному – 3 базовых алгоритма. И, поскольку накладные расходы в случае двух ядер невелики, это позволяет главному ядру находить решение нужного качества более чем в два раза быстрее по сравнению с последовательным алгоритмом.

Проведенный эксперимент показал, что кооперативная модели-ориентированная схема может быть распараллелена эффективно.

Заключение

В работе исследуются возможности параллелизации кооперативной модели-ориентированной метаэвристической схемы. Рассмотрен подход, в котором каждому процессору присваивается один или более базовых алгоритмов. Экспериментальное исследование на задачах коммивояжера с известными решениями показало, что разработанный параллельный алгоритм позволяет эффективно использовать естественный параллелизм присущий кооперативной схеме.

Благодарности

Статья частично профинансирована в рамках проекта **ITHEA XXI** Института Информационных Теории и Приложений FOI ITHEA и ADUIS Ukraine (www.ithea.org, www.aduis.com.ua).

Список литературы

- [Hoos and Stützle, 2005] H.H. Hoos, T. Stützle. Stochastic Local Search: Foundations and Applications. San Francisco: Morgan Kaufmann Publ, 2005. 658 p.
- [Parallel Metaheuristics, 2006] Parallel Metaheuristics: A New Class of Algorithms (eds. E. Alba). (2006) Wiley. 576 p.
- [Stützle, 2004] T. Stützle. ACOTSP, Version 1.0. <http://www.aco-metaheuristic.org/aco-code>. 2004.
- [Гуляницький, 2009] Л.Ф. Гуляницький. Розробка кооперативних метаевристик. In: Abstract of Int. Conf. "Problems of Decision Making under Uncertainties (PDMU-2009)" (April 27-30, 2009, Skhidnytsia, Ukraine). Kyiv, 2009. pp. 90–91.
- [Гуляницький и Сиренко, 2009] Л.Ф. Гуляницький, С. Сиренко. Кооперативные модели-ориентированные метаэвристики для задач комбинаторной оптимизации. In: Intelligent Support of Decision Making (Eds. Krassimir Markov et al.). Sofia: ITHEA, 2009. pp. 165–172.

Информация об авторе



Сергей Сиренко (Sirenko) – аспирант, Институт кибернетики им. В.М.Глушкова НАН Украины, пр-т Глушкова, 40, Киев, 03680, Украина. e-mail: s.sirenko@gmail.com