

УДК 004.021

АЛГОРИТМ СУБСИДАРНОЙ МЕТАЭВРИСТИКИ**Цветков Виктор Яковлевич¹,***д-р техн. наук, профессор,**e-mail: cvj2@mail.ru,***Козлов Александр Вячеславович¹,***e-mail: av-kozlov@mail.ru,*¹*Российский технологический университет (РТУ МИРЭА), г. Москва, Россия*

В статье исследуется новое направление – «субсидарная метаэвристика». Предложен новый метаэвристический алгоритм, который авторы назвали алгоритмом покрытия. Показано различие между эвристикой и метаэвристикой применительно к субсидарным системам. Метаэвристические алгоритмы являются модифицируемыми по ситуации управления. Общим для эвристики и метаэвристики является итеративный процесс поиска решений. Показано значение ситуации для метаэвристических преобразований. Представлена типовая схема эвристических алгоритмов с фиксированной последовательностью действий. Описан пример формирования алгоритмов субсидарной метаэвристики. Рассмотрена задача покрытия ареалами участка произвольной формы. Сложность задачи заключается в том, что эта площадь неизвестна и определяется итеративно в процессе покрытия. Показано, что субсидарные системы и алгоритмы являются многоцелевыми, выделена группа типичных целей субсидарного алгоритма. Описан алгоритм покрытия площади участка ареалами, связанными с движением робота. Особенность алгоритма метаэвристики состоит в использовании информации от датчиков. Отмечается, что субсидарная система с метаэвристическим алгоритмом управления должна включать механизмы координации в пространстве.

Ключевые слова: метаэвристика, субсидарная система, ситуационное поведение, метаэвристический алгоритм, алгоритм покрытия

THE ALGORITHM OF SUBSIDIARY METAHEURISTICS**Tsvetkov V.Ya.¹,***doctor of technical sciences, professor,**e-mail: cvj2@mail.ru,***Kozlov A.V.¹,***e-mail: av-kozlov@mail.ru,*¹*Russian Technological University (RTU MIREA), Moscow, Russia*

The article explores a new direction “subsidiary metaheuristics”. The article proposes a new metaheuristic algorithm, which the authors called the coverage algorithm. The difference between heuristics and metaheuristics in relation to subsidiary systems is shown. Metaheuristic algorithms are modifiable according to the control situation. The iterative process of finding solutions is common to heuristics and metaheuristics. The significance of the situation for metaheuristic transformations is shown. A typical scheme of heuristic algorithms with a fixed sequence of actions is presented. An example of the formation of algorithms of subsidiary metaheuristics is described. The problem of covering an arbitrary-shaped area with areas is considered. The complexity of the problem lies in the fact that this area is unknown and is determined iteratively during the coating process. It is shown that subsidiary systems and algorithms are multi-purpose, a group of typical goals of a subsidiary algorithm is identified. The algorithm of covering the area of the site with areas associated with the movement of the robot is described. The feature of the metaheuristics algorithm is the use of information from sensors. It is noted that a subsidiary system with a metaheuristic control algorithm should include coordination mechanisms in space.

Keywords: metaheuristics, subsidiary system, situational behavior, metaheuristic algorithm, coverage algorithm

DOI 10.21777/2500-2112-2022-4-65-71

Введение

Субсидиарность как организационный принцип, согласно которому задачи должны решаться на низком или удалённом от центра уровне, на котором их решение возможно и эффективно, проявляется в стохастической среде и при групповом поведении. Субсидиарность также имеет место в интеллектуальных и живых системах [1], в дискретных потоках с прерываниями или внешними воздействиями. В интеллектуальных системах используются алгоритмы субсидиарного поведения и принятия решений. В дискретном потоке субсидиарность возникает при наличии явления перколяции (от лат. *percōlāre* – «просачиваться», «протекать») [2]. Теория перколяции используется для описания возникновения связанных структур (кластеров) в случайных средах, состоящих из отдельных элементов, и находит применение в описании разнообразных систем и явлений.

Можно ввести понятие «отношение субсидиарности». Оно является альтернативой отношению следования [3]. Задача определения отношения субсидиарности возникает, например, при динамической маршрутизации транспортных средств [4]. Динамическая маршрутизация характеризуется тем, что часть или все входные данные неизвестны и обнаруживаются динамически во время прохождения маршрута. При динамической маршрутизации маршруты подвижных объектов переопределяются с помощью субсидиарного алгоритма. Все киберфизические системы являются субсидиарными.

Субсидиарные задачи являются ситуационными, поскольку определяются ситуацией, в которой находится подвижный объект. Субсидиарные алгоритмы используют модели ситуационного поведения. Реализация ситуационного поведения осуществляется разными путями. Одним из традиционных способов является применение многоагентных систем с двумя или тремя алгоритмами поведения [5]. Для реализации ситуационного поведения также широко применяется подход на основе методов эвристики [6] и искусственной нейронной сети с обучением. Метаэвристика как высокоуровневая проблемно-независимая алгоритмическая структура позволяет применять разные алгоритмы эвристической оптимизации, удовлетворяющие заданным ограничениям, и находить близкие к оптимальным решения задачи за приемлемое время [7]. Метаэвристика дает возможность сохранять и комбинировать свойства используемых алгоритмов [8]. Субсидиарная метаэвристика является новым направлением, которое в теории метаэвристики недостаточно исследовано, чем обусловлена актуальность данной работы.

1. Типовая схема эвристических алгоритмов с фиксированной последовательностью действий

Эвристические алгоритмы с фиксированной последовательностью действий являются условно простыми (далее – алгоритм простой эвристики). Формирование алгоритмов простой эвристики происходит в рамках класса решений «X» данной задачи. Реализация алгоритмов простой эвристики определяется допустимым отношением приближения d . Примерами задач простой эвристики является информационный поиск или вычисление сложной функции методом последовательных приближений. Простая эвристика поиска решения имеет итеративную схему.

Обозначим через k номер итерации, A_i – метки алгоритма. Задается функция приближенных вычислений $f(x)$. Рассмотрим два варианта поиска решения в зависимости от допустимого отношения приближения: допустимой точности и допустимого числа вариантов перебора. В первом случае задается точность p , во втором – число итераций N . Представим пошаговый алгоритм:

A1: Определяется исходное решение $x_0 \in X$, $k = 0$;

A2: Вычисляется начальное значение функции $Y_0 = f(x_0)$;

A3: Задается декремент d_x ;

A4: Перевычисляется аргумент $x_1 = x_0 + d_x$ (для первой итерации) или $x_{k+1} = x_k + d_x$ (для последующих итераций);

A5: Вычисляется новое значение функции $Y_1 = f(x_1)$;

A6: Включается счетчик итераций $k = k + 1$;

A7: Проводится сравнение функций и вычисляется расхождение $DY_1 = Y_0 - Y_1$ (для первой итерации) или $DY_k = Y_{k-1} - Y_k$ (для последующих итераций).

Сравнивается расхождение и допустимое отношение:

A8: Если $p \geq DY_k$, то Stop; иначе $k+1$, Go to A4.

Для метода перебора оператор A8 выглядит иначе:

A8*: Если $N = k$, то Stop; иначе $k + 1$, Go to A4.

Существуют модификации метода эвристики с фиксированной последовательностью действий, но принципиальная схема A1-A8 остается основой.

2. Алгоритм «покрытия» субсидиарной метаэвристики

Эвристические алгоритмы с фиксированной последовательностью действий не подходят для субсидиарных систем, поскольку необходима динамическая модификация их выполнения в зависимости от ситуации. Управление субсидиарной системой и киберфизической системой требует специального субсидиарного алгоритма. Наиболее эффективными для решения данной задачи представляются метаэвристические алгоритмы, которые дают возможность сохранять и комбинировать свойства используемых эвристических алгоритмов и находить близкие к оптимальным решения при заданных ограничениях.

Алгоритмы субсидиарных систем имеют разные цели. Основная цель связана со стратегической и/или групповой целью. Локальные цели заключаются в выживаемости системы, реакции на ситуацию, оптимальном поведении системы в конкретной ситуации и др. Эти цели закладываются в алгоритмах субсидиарной метаэвристики.

Рассмотрим алгоритм субсидиарной метаэвристики на примере работы робота-пылесоса. Алгоритм, который реализует данный робот, можно назвать алгоритмом покрытия. Стратегическая цель состоит в прохождении участка заданной формы, то есть в охвате площади S (рисунок 1). Вторая цель заключается в выживаемости системы, то есть работе при температуре меньше допустимой T_d . Вторая цель включает также анализ состояния ресурсов (аккумулятора) между текущим уровнем зарядки R_t и допустимым R_d . Третья цель состоит в реакции на ситуацию, то есть обходе препятствий или уходе от стен. Четвертая цель состоит в попытке оптимизировать движение для конкретной формы помещения. Пятая цель состоит в том, чтобы по завершению работы вернуться к начальной точке, которая называется «базой».

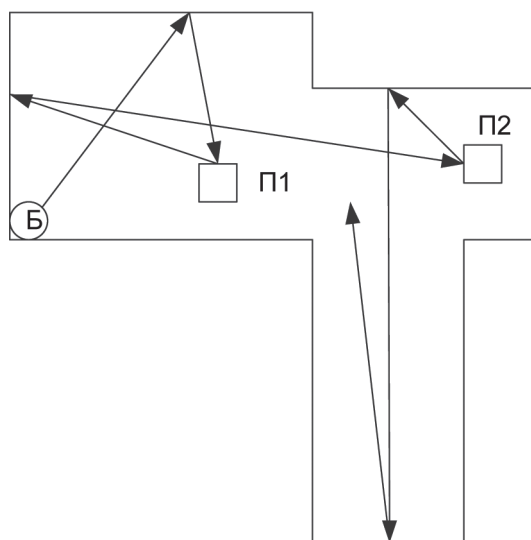


Рисунок 1 – Схема площади и начальные движения робота

На рисунке 1 буквой B обозначено начальное положение робота или база; символами $П_1, П_2$ обозначены препятствия, которые робот должен обходить; линиями со стрелками показаны начальные фазы движения. На самом деле робот охватывает некоторую площадь при движении и линиям соответствуют ареалы, имеющие площадь.

Введем параметр «ширина захвата» S_k . Это ширина, которую охватывает робот при движении. Робот двигается по прямым линиям, которые можно рассматривать как итерации движения. Введем параметр «близость площадей» δS_k , который характеризует расхождение площадей на k и $k-1$ итерации движения робота. Задача робота состоит в покрытии своим маршрутом и полосой захвата площади помещения.

Упрощенно такого робота можно рассматривать как интеллектуального агента, который имеет память M . Дополнительно такой робот обладает способностью к координации.

Алгоритмически субсидиарные вычисления или субсидиарное управление строится на основе следующей парадигмы:

стратегическая задача → информация о реальности → стохастические действия с элементом рациональности → запоминание ситуации → запоминание действий → продолжение действий → анализ задачи → анализ решения → завершение при условной оптимизации.

Сложная метаэвристика поиска имеет итеративную схему.

Рассмотрим вариант задачи – «покрытие». Для него существует отношение площадей $R_s = S/L_s$, где S – площадь покрытия (неизвестная величина); L_s – площадь, определяемая длиной пути робота, умноженная на ширину захвата. Представим пошаговый алгоритм:

A1: Инициализация. Запоминается исходное положение робота на базе в памяти $M(x_0, y_0) \in S$, начальный вектор движения v_0 , номер итерации $k = 0$;

A2: Осуществляется начальное движение $L_k = v_k \times t_k$ ($k = 1$). Если препятствие, то Stop. Длина пути определяется препятствием. Здесь срабатывает физический сенсор препятствия;

A3: С помощью датчика случайных чисел генерируется декремент поворота d_v ;

A4: Задается иррациональный инкремент поворота d_v в соответствии с условиями: $C = d_v \bmod 2 \neq 0 \wedge d_v \bmod 3 \neq 0 \wedge d_v \bmod 5 \neq 0$. Если $C = 1$, то A5, иначе A3. Иррациональность исключает кратность поворота, то есть закливание;

A5: Модифицируется аргумент $v_{k+1} = v_k + d_v$;

A6: Включается счетчик итераций $k = k + 1$;

A7: Осуществляется последующее движение $L_k = v_k \times t_k$, если препятствие, то Stop;

A8: Вычисляется длина пройденного пути $L_{sk} = \sum L_k$, результат запоминается;

A9: Вычисляется покрытая площадь на итерации $Sls(k) = S_k \times L_{sk}$, результат запоминается;

A10: Выполняется три процедуры движения. Если $k < 3$ Go to A3, иначе A12;

A11: Запоминается два последних значения покрытой площади $Sls(k)$, $Sls(k - 1)$;

A12: Вычисляется отношение площадей $R_s(k - 1) = Sls(k - 1)/Sls(k - 1)$, результат запоминается;

A13: Вычисляется отношение площадей $Rsk = Sls(k - 1)/Sls(k)$, результат запоминается;

A14: Вычисляется расхождение отношения площадей $\delta S_k = Rsk - R_s(k - 1)$, результат запоминается;

A15: Если $\delta S_k \geq \Delta$ (нормативный допуск точности площади, рисунок 2), то Go to «база», иначе Go to A3;

A16: Если $R_t \geq R_d$ то Go to «база», иначе Go to A3. Здесь также срабатывает физический датчик.

Типичным для этого примера метаэвристики является использование информации от датчиков и включение ее в алгоритм. На рисунке 2 показана динамика изменения отношения площадей при росте итераций.

По горизонтали показан рост итераций, по вертикали – отношение площадей. Эта величина растет и, в пределе, приближается к единице. Величина Δ характеризует область, при которой дальнейшие итерации не нужны. Величина p_s (рисунок 2) показывает предел итераций.

Особенность данной методики в том, что нет необходимости знать площадь участка S . Данная методика отличается от известных методик метаэвристики типа: метод имитации отжига [9], табу-поиск [10; 11], процедура жадного рандомизированного адаптивного поиска (GRASP¹) [12], метод поиска чередующихся окрестностей VNS² [13], генетические/эволюционные алгоритмы, имитация функционирования муравьиной колонии [14] и др.

¹ GRASP = Greedy Randomized Adaptive Search Procedure.

² Variable Neighborhood Search.

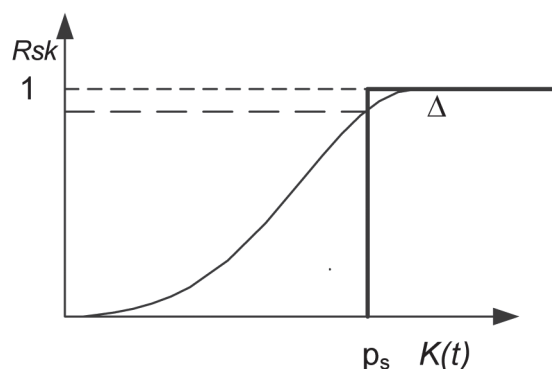


Рисунок 2 – Динамика изменения отношения площадей при росте итераций

Предложенный метод следует назвать «методом покрытия», поскольку его задача – покрыть поверхность ареальными полосами. Следует отметить, что метод является переопределенным, поскольку в процессе работы робот может фрагментарно покрывать части участков повторно. Этот недостаток компенсируется отсутствием знания реальной площади, поскольку основой завершения работы является условие $\lim Rsk \rightarrow 1$.

Данный метод можно применить для автоматизированной аэрофотосъемки при помощи беспилотного летательного аппарата (БПЛА) вертолетного типа. В этом случае физические границы следует заменить аналитическими условиями. В этом случае изменится метка алгоритма А6. БПЛА или центр его управления должны рассчитывать местоположение и при достижении границ участка задавать изменение траектории движения. Если такой БПЛА полностью автономен, то можно говорить о воздушной киберфизической системе [15].

Заключение

Субсидиарная метаэвристика является новым направлением, которое в теории метаэвристики недостаточно исследовано. Субсидиарная метаэвристика использует субсидиарные алгоритмы и применяется в субсидиарных системах. Субсидиарные системы и алгоритмы являются децентрализованными [16] и многоцелевыми. Метаэвристические алгоритмы являются модифицируемыми по ситуации управления либо вычисления. Они эффективны при использовании в субсидиарных системах и при управлении субсидиарным поведением систем. Управление субсидиарной системой и киберфизической системой требует специального субсидиарного алгоритма, наиболее подходящими из которых можно считать метаэвристические алгоритмы. Наряду с алгоритмами существуют субсидиарные вычисления. Они также входят в алгоритмы динамической маршрутизации. Для управления субсидиарными системами обязательным является применение субсидиарных алгоритмов и вычислений. Субсидиарная система с метаэвристическим алгоритмом управления должна включать механизмы координации в пространстве.

Предложенный алгоритм является недетерминированным, поскольку движение робота происходит с нарушением гладкости траектории. Недетерминированность также обусловлена тем, что при повторном запуске робота траектория его движения будет другой, хотя задача в итоге будет решена. Предложенный алгоритм относится к классу решения трудных задач, которые характеризуются неоднозначностью решения и отсутствием единственного алгоритма, который приводит от начальной ситуации к конечной точке.

Механизмы управления субсидиарной системой используют разные метаэвристические алгоритмы. Общий вывод состоит в том, что метаэвристика является потребностью и инструментом функционирования субсидиарных систем.

Список литературы

1. Цветков В.Я., Козлов А.В. Использование моделей живых организмов для анализа эволюции сложных организационно-технических систем // Образовательные ресурсы и технологии. – 2019. – № 4 (29). – С. 68–76.
2. Basu R., Hoffman C., Sly A. Nonexistence of bigeodesics in planar exponential last passage percolation // Communications in Mathematical Physics. – 2022. – Vol. 389, No. 1. – P. 1–30.
3. Кортунов В.В. Формализация логического следования в формальной логике // Сервис plus. – 2018. – Т. 12, № 1. – С. 122–126.
4. Liu Y. An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones // Computers & Operations Research. – 2019. – Vol. 111. – P. 1–20.
5. Розенберг И.Н., Цветков В.Я. Применение мультиагентных систем в интеллектуальных логистических системах // Международный журнал экспериментального образования. – 2012. – № 6. – С. 107–109.
6. Sangaiah A.K. et al. IoT resource allocation and optimization based on heuristic algorithm // Sensors. – 2020. – Vol. 20, No. 2. – P. 539.
7. Elshaer R., Awad H. A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants // Computers & Industrial Engineering. – 2020. – Vol. 140. – DOI 10.1016/j.cie.2019.106242.
8. Воронов В.С., Пересунько П.В., Виденин С.А., Матюхин Н.Е. Последовательная метаэвристика для задачи приоритетной упаковки с несовместимыми категориями // Современные наукоемкие технологии. – 2020. – № 6. – С. 220–227.
9. Metropolis N., Rosenbluth A.W., Rosenbluth M.N., Teller A.H., Teller E. Equation of state calculations by fast computing machines // Journal Chemical Physics. – 1953. – Vol. 21, No. 6. – P. 1087–1091. – DOI 10.1063/1.1699114.
10. Hou N. et al. An efficient GPU-based parallel tabu search algorithm for hardware/software co-design // Frontiers of Computer Science. – 2020. – Vol. 14, No. 5. – P. 1–18.
11. Alinaghian M. et al. An augmented Tabu search algorithm for the green inventory-routing problem with time windows // Swarm and Evolutionary Computation. – 2021. – Vol. 60. – DOI 10.1016/j.swevo.2020.100802.
12. Resende M.G.C., Ribeiro C.C. Greedy randomized adaptive search procedures: advances and extensions // Handbook of metaheuristics. – Springer, Cham, 2019. – P. 169–220.
13. Hansen P. et al. Variable neighborhood search // Handbook of metaheuristics. – Springer, Cham, 2019. – P. 57–97.
14. Tsvetkov V.Ya. Incremental Solution of the Second Kind Problem on the Example of the Living System // Biosciences biotechnology research Asia. – 2014. – Vol. 11 (Spl. Edn.). – P. 177–180.
15. Цветков В.Я. Управление с применением киберфизических систем // Перспективы науки и образования. – 2017. – № 3 (27). – С. 55–60.
16. Козлов А.В. Субсидиарные и децентрализованные системы и алгоритмы // Образовательные ресурсы и технологии. – 2020. – № 2 (31). – С. 69–76. – DOI 10.21777/2500-2112-2020-2-69-76.

References

1. Cvetkov V.Ya., Kozlov A.V. Ispol'zovanie modelej zhivykh organizmov dlya analiza evolyucii slozhnykh organizacionno-tekhnicheskikh sistem // Obrazovatel'nye resursy i tekhnologii. – 2019. – № 4 (29). – S. 68–76.
2. Basu R., Hoffman C., Sly A. Nonexistence of bigeodesics in planar exponential last passage percolation // Communications in Mathematical Physics. – 2022. – Vol. 389, No. 1. – P. 1–30.
3. Kortunov V.V. Formalizaciya logicheskogo sledovaniya v formal'noj logike // Servis plus. – 2018. – Т. 12, № 1. – S. 122–126.
4. Liu Y. An optimization-driven dynamic vehicle routing algorithm for on-demand meal delivery using drones // Computers & Operations Research. – 2019. – Vol. 111. – P. 1–20.
5. Rozenberg I.N., Cvetkov V.Ya. Primenenie mul'tiagentnykh sistem v intellektual'nykh logisticheskikh sistemah // Mezhdunarodnyj zhurnal eksperimental'nogo obrazovaniya. – 2012. – № 6. – С. 107–109.
6. Sangaiah A.K. et al. IoT resource allocation and optimization based on heuristic algorithm // Sensors. – 2020. – Vol. 20, No. 2. – P. 539.
7. Elshaer R., Awad H. A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants // Computers & Industrial Engineering. – 2020. – Vol. 140. – DOI 10.1016/j.cie.2019.106242.

8. *Voronov V.S., Peresun'ko P.V., Videnin S.A., Matyuhin N.E.* Posledovatel'naya metaevristika dlya zadachi prioritetnoj upakovki s nesovmestimymi kategoriyami // *Sovremennye naukoemkie tekhnologii.* – 2020. – № 6. – S. 220–227.
9. *Metropolis N., Rosenbluth A.W., Rosenbluth M.N., Teller A.H., Teller E.* Equation of state calculations by fast computing machines // *Journal Chemical Physics.* – 1953. – Vol. 21, No. 6. – P. 1087–1091. – DOI 10.1063/1.1699114.
10. *Hou N. et al.* An efficient GPU-based parallel tabu search algorithm for hardware/software co-design // *Frontiers of Computer Science.* – 2020. – Vol. 14, No. 5. – P. 1–18.
11. *Alinaghian M. et al.* An augmented Tabu search algorithm for the green inventory-routing problem with time windows // *Swarm and Evolutionary Computation.* – 2021. – Vol. 60. – DOI 10.1016/j.swevo.2020.100802.
12. *Resende M.G.C., Ribeiro C.C.* Greedy randomized adaptive search procedures: advances and extensions // *Handbook of metaheuristics.* – Springer, Cham, 2019. – P. 169–220.
13. *Hansen P. et al.* Variable neighborhood search // *Handbook of metaheuristics.* – Springer, Cham, 2019. – P. 57–97.
14. *Tsvetkov V.Ya.* Incremental Solution of the Second Kind Problem on the Example of the Living System // *Biosciences biotechnology research Asia.* – 2014. – Vol. 11 (Spl. Edn.). – P. 177–180.
15. *Cvetkov V.Ya.* Upravlenie s primeneniem kiberfizicheskikh sistem // *Perspektivy nauki i obrazovaniya.* – 2017. – № 3 (27). – С. 55–60.
16. *Kozlov A.V.* Subsidiarnye i decentralizovannye sistemy i algoritmy // *Obrazovatel'nye resursy i tekhnologii.* – 2020. – № 2 (31). – S. 69–76. – DOI 10.21777/2500-2112-2020-2-69-76.