

Применение метода роя частиц в задачах оптимизации

Е. М. Казакова

Институт прикладной математики и автоматизации –
филиал Кабардино-Балкарского научного центра Российской академии наук
360000, Россия, Нальчик, ул. Шортанова, 89 А

Аннотация. Оптимизация на основе роя частиц (particle swarm optimization – PSO) – это эвристический метод глобальной оптимизации, первоначально предложенный Дж. Кеннеди и Р.К. Эберхартом в 1995 году. Сегодня PSO является одним из самых известных и широко используемых алгоритмов роевого интеллекта из-за его простоты и возможности использования в широком диапазоне приложений. В данной статье представлен обзор приложений PSO в различных областях: электротехнике, системах автоматического управления, медицине, биологии, химии и т.д.

Ключевые слова: оптимизация, метаэвристический алгоритм, рой частиц, PSO

Поступила 26.09.2022, одобрена после рецензирования 03.10.2022, принята к публикации 10.10.2022

Для цитирования. Казакова Е. М. Применение метода роя частиц в задачах оптимизации // Известия Кабардино-Балкарского научного центра РАН. 2022. № 5 (109). С. 48–57. DOI: 10.35330/1991-6639-2022-5-109-48-57

MSC: 68T27

Review article

Application of particle swarm method in the optimization problems

E.M. Kazakova

Institute of Applied Mathematics and Automation –
branch of Kabardino-Balkarian Scientific Center of the Russian Academy of Sciences
360000, Russia, Nalchik, 89 A Shortanov street

Annotation. Particle swarm optimization (PSO) is a heuristic global optimization method originally proposed by J. Kennedy and R.K. Eberhart in 1995. Today, PSO is one of the best known and widely used swarm intelligence algorithms due to its simplicity and ability to be used in a wide range of applications. This paper provides an overview of PSO applications in various fields: electrical engineering, automatic control systems, medicine, biology, chemistry, etc.

Key words: optimization, metaheuristic algorithm, Particle Swarm Optimization

Submitted 26.09.2022, approved after reviewing 03.10.2022, accepted for publication 10.10.2022

For citation. Kazakova E.M. Application of particle swarm method in the optimization problems. *News of the Kabardino-Balkarian Scientific Center of RAS*. 2022. No. 5 (109). Pp. 48–57. DOI: 10.35330/1991-6639-2022-5-109-48-57

ВВЕДЕНИЕ

Метод оптимизации роем частиц (PSO) был предложен и первоначально разработан инженером-электриком Расселом К. Эберхартом и социальным психологом Джеймсом Кеннеди. Метод был описан в статьях [1, 2], написанных ими в соавторстве и опубликованных в 1995 г. Алгоритм PSO был инспирирован социальным и биологическим

поведением птичьих стай, ищущих источники пищи. В этом алгоритме частицы летают через пространство поиска для нахождения наилучшего глобального положения, которое минимизирует (или максимизирует) определенную задачу. В конце концов, весь рой, приближается к оптимуму целевой функции со скоростью сходимости, которая сильно зависит от выбранного варианта PSO и значений параметров [3].

Метод PSO основан на предпосылке, что знание заключается в социальном обмене информацией не только между поколениями, но и между элементами одного и того же поколения. Хотя PSO имеет индивидуальные характеристики, этот алгоритм также обладает некоторым сходством с характеристиками, обнаруженными в других вычислительных моделях, основанных на популяции, таких как генетические алгоритмы (GA) и другие эволюционные вычислительные методы. Преимущество метода PSO заключается в том, что он относительно прост, и его алгоритм сравнительно легко описать и реализовать.

Его простота и очевидная компетентность в нахождении оптимальных решений в сложных пространствах поиска привели к тому, что алгоритм PSO стал хорошо известен в научном сообществе, что способствовало его изучению и совершенствованию. Таким образом, было предложено множество подходов и протестированы различные приложения, особенно за последнее десятилетие.

Чтобы охватить научный вклад в отношении PSO, было выполнено несколько обзоров и метаанализов по PSO в различных областях применения. Бэнкс и др. [4] предложили краткий обзор в двух частях различных улучшений исходного метода PSO и его возможностей, а также проблем, связанных с алгоритмом PSO. Кроме того, в обзоре основное внимание уделялось исследованиям, касающимся адаптации для параллельной реализации, конфигурации алгоритма и динамических сред. В своем обзоре А. П. Карпенко [5] рассматривает различные модификации PSO со статической и динамической топологией соседства частиц, а также многороевый метод с миграцией частиц и параллельные методы роя частиц.

В обзоре представлены приложения PSO в различных областях, включая здравоохранение, промышленность и прочее.

КАНОНИЧЕСКИЙ АЛГОРИТМ PSO

Описание алгоритма

Метод PSO направлен на итеративную оптимизацию задачи, начиная с набора или популяции возможных решений, называемых в данном контексте роем частиц, в котором каждая частица знает глобальное наилучшее положение в рое, а также его индивидуальное наилучшее положение, найденное до сих пор в процессе поиска в пространстве решения задачи.

На каждой итерации на скорость и положение каждой частицы в рое, представленные D -мерными векторами, влияют индивидуальные и коллективные знания, направляющие повторяющиеся полеты частиц над пространством возможных решений задачи в поисках оптимума, пока не будет удовлетворен подходящий критерий остановки.

Как и в других метаэвристических алгоритмах [6], в PSO производится несколько основных этапов, включая инициализацию роя, вычисление фитнес-функции для каждой частицы, стратегию обновления и оценку приспособленности.

Процесс инициализации роя

Инициализация роя: частицы разбрасываются случайным образом на начальном этапе создания роя без определенного критерия. При последующих итерациях, обозначенных $t = 0, 1, \dots, T$, PSO начинается со случайно сгенерированного роя N , D -мерных частиц. Проще

говоря, с векторами положения с действительными значениями, представляющими начальные решения-кандидаты, путем инициализации положения каждой частицы x_i^0 (на итерации $t = 0$) в случайную позицию в пространстве поиска как

$$x_i^0 \sim U(x_{min}, x_{max})^D, \quad (1)$$

где D обозначает размерность пространства поиска или размер решаемой задачи, а x_i частица имеет вектор начального положения x_i^0 со значениями, выбираемыми случайным образом из равномерно распределенного диапазона $U(x_{min}, x_{max})^D$, где x_{min} и x_{max} – границы пространства поиска. В общем случае для представления вектора положения i -й частицы на каждой последующей итерации $t \geq 0$ можно использовать следующие обозначения:

$$x_i^t = (x_{i,1}^t, x_{i,2}^t, \dots, x_{i,D}^t). \quad (2)$$

Переменные решения (т. е. элементы вектора положения) задачи часто связаны с физическими мерами или компонентами, которые имеют естественные границы. Длина, вес и масса являются примерами таких границ, где значения переменных решения не должны быть отрицательными, поэтому следует определить диапазон $[x_{min}, x_{max}]$ значений, чтобы независимо ограничить значение каждой переменной решения в этом диапазоне.

Инициализация скорости: все частицы перемещаются в пространстве поиска со скоростью (размером шага), которая отражает эмпирические знания частиц, а также социальный обмен информацией о перспективных областях, посещенных в пространстве поиска, тем самым направляя процесс оптимизации в наиболее подходящие области. Аналогично инициализации положения скорость v_i каждой частицы может быть инициализирована как

$$v_i^t = (v_{i,1}^t, v_{i,2}^t, \dots, v_{i,D}^t), \quad (3)$$

где v_i^t – вектор скорости i -й частицы на итерации t . $v_{i,1}^t, v_{i,2}^t, \dots, v_{i,D}^t$ являются параметрами скорости, каждый из которых инициализируется случайным образом (на итерации $t = 0$) в пределах установленного диапазона $[x_{min}, x_{max}]$ или $[-v_{max}, v_{max}]$, чтобы частицы не выходили из пространства поиска [7]. v_{max} обозначается как максимально допустимый размер шага в любом измерении. v_{max} необязателен, если правильно задать параметры управления, то он вообще не нужен.

Оценка пригодности

Частицы роя проверяются в конце каждой итерации на индивидуальное лучшее положение, а также глобальное лучшее положение. Предполагается, что каждая i -я частица имеет уникальное значение пригодности $f(x_i^t)$ на каждой итерации t , которое вычисляется посредством оценки целевой функции. PSO запоминает индивидуальное лучшее решение (кандидат на лучшее глобальное решение), которое когда-либо встречалось каждой частице до текущей итерации t , как

$$p_{best}^t = (p_{best_1}^t, p_{best_2}^t, \dots, p_{best_N}^t), \quad (4)$$

где p_{best}^t представляет индивидуальные лучшие позиции всех частиц в рое на итерации t . В зависимости от типа задачи оптимизации (минимизация или максимизация) индивидуальная лучшая позиция $p_{best_i}^t$, которую i -я частица посетила до текущей итерации, скажем t , вычисляется как:

$$p_{best_i}^t = x_i^t | f(x_i^t) = \min/\max_{k=0,1,\dots,t} (\{f(x_i^k)\}), \quad (5)$$

где l – тот же индекс, что и k -я итерация, в которой i -я частица нашла наилучшую позицию до текущей итерации t . В конце каждой итерации все решения-кандидаты в p_{best}^t сортируются и первое ранжированное решение выбирается как глобальное (или соседнее) наилучшее решение или позиция g_{best}^t , которую посетила i -я частица-сосед:

$$g_{best}^t = p_{best_m}^t | f(p_{best_m}^t) = \min/\max_{k=1,2,\dots,N}(\{f(p_{best_i}^t)\}), \quad (6)$$

где m – тот же индекс, что и i -я частица, занимающая в целом наилучшее положение среди частиц роя.

Проверка пригодности: после инициализации роя и скорости у каждой частицы в рое проверяется ее начальное индивидуальное лучшее положение $f(p_{best_i}^0)$ вместе с начальным глобальным (или окрестным) наилучшим положением $f(g_{best}^0)$, используя уравнения (5) и (6) соответственно. На этапе инициализации индивидуальному наилучшему решению $p_{best_i}^0$ i -й частицы присваивается начальное положение x_i^0 .

Обновление скорости

На каждой текущей итерации, скажем $t + 1$, скорость i -й частицы v_i сначала регулируется путем направления ее параметров в сторону увеличения в положительном или отрицательном направлении в зависимости от сходимости текущей позиции, притягивая частицу к позициям в пространстве поиска, которые известны как хорошие из прошлого индивидуального опыта, а также из опыта других частиц в окрестности частицы. Исходный PSO [1, 2] был реализован для двух различных топологий соседства, глобального наилучшего ($gbest$) PSO и локального наилучшего ($lbest$) PSO.

$Gbest$ PSO – это базовая версия алгоритма, представленного Кеннеди и Эберхартом [1, 2]. В поисках оптимального решения каждая частица движется: на каждой последующей итерации $t \geq 0$ – к своему более раннему индивидуальному положению p_{best}^t и глобальному лучшему положению g_{best}^t в рое путем добавления вектора скорости к положению частицы на предыдущей итерации

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1}, \quad (7)$$

где $v_{i,j}^0 = 0$ и $v_{i,j}^{t+1}$ – вектор скорости i -й частицы на итерации $t + 1$ в j измерении

$$v_{i,j}^{t+1} = \omega v_{i,j}^t + c_1 r_{1,i,j}^{t+1} (p_{best,i,j}^t - x_{i,j}^t) + c_2 r_{2,i,j}^{t+1} (g_{best,j}^t - x_{i,j}^t), \quad (8)$$

где:

инерциальный компонент $\omega v_{i,j}^t$, также называемый направлением полета, обеспечивает частице соответствующий импульс для перемещения по пространству поиска без резкого изменения направления на основе истории предыдущих направлений полета. Эта часть содержит параметр управления ω (*inertia weight* или IW);

когнитивный компонент $c_1 r_{1,i,j}^{t+1} (p_{best,i,j}^t - x_{i,j}^t)$ обозначает память о предыдущем индивидуальном лучшем положении, в котором она получала свою лучшую результативность в прошлом;

социальный компонент $c_2 r_{2,i,j}^{t+1} (g_{best,j}^t - x_{i,j}^t)$ определяет текущую производительность, связанную с лучшими глобальными решениями, найденными до сих пор.

$j \in 1, 2, \dots, D$ обозначает размеры (или компоненты) i -й частицы. Новые параметры, известные как коэффициенты познания и социального ускорения c_1 и c_2 , вместе с новым инерционным весом ω , которые были добавлены в инерционную версию PSO, представленную в [8], показали более высокую производительность, чем те, которые были с фиксированными параметрами. Параметры c_1 и c_2 , также называемые «коэффициентами

ускорения», являются положительными константами, обычно используемыми для определения ускорения скорости познания i -й частицы в направлении p_{best_i} и g_{best} соответственно. ω был первоначально разработан для решения проблемы быстрого роста скорости в исходной версии PSO 1995 года [1, 2]. Кроме того, ω в сочетании с c_1 и c_2 можно использовать для баланса глобального исследования и локальной эксплуатации. $r_{1,i,j}^{t+1}$ и $r_{2,i,j}^{t+1}$ представляют собой два независимых случайных числа на текущей итерации $t + 1$, которые равномерно распределены в диапазоне $[0, 1]$ и выбираются независимо для каждой i -й частицы в каждом j измерении, чтобы поддерживать разнообразие роя. $p_{best,i,j}^t$ – наилучшее индивидуальное положение частицы (или p_{best}), наилучшее положение, с которым сталкивается i -я частица (на итерации t) в j измерении, $g_{best,j}^t$ – наилучшее глобальное положение (или g_{best}) среди всех частиц в рое. В конце каждой итерации p_{best} и g_{best} оцениваются с помощью уравнений (5) и (6) соответственно, чтобы к концу алгоритма было получено глобальное наилучшее решение g_{best} .

lbest PSO похож на *gbest* PSO. Разница в том, что для *lbest* PSO

$$c_2 r_{2,i,j}^{t+1} (n_{best,j}^t - x_{i,j}), \quad (9)$$

где $n_{best,j}^t$ – наилучшее положение i -й частицы в своей окрестности в j -м измерении, наиболее подходящее p_{best} не всего роя, как в *gbest* PSO, а в подмножестве роя, образующего окрестности i -й частицы. Для *lbest* PSO окрестность каждой i -й частицы состоит из двух других частиц: $i - 1$ и $i + 1$. Топология соседства *lbest* PSO называется топологией кольца. Окрестность каждой частицы в *gbest* PSO – это просто весь рой; такая топология соседства называется полносвязной топологией. С момента появления в 1998 году параметр ω был включен в большинство недавно введенных вариантов PSO. По этой причине алгоритм с этим улучшением называется стандартным, каноническим или инерционным PSO (ω -PSO).

Канонический алгоритм PSO (ω -PSO):

создать и инициализировать N D -мерный рой

repeat

foreach *particle* $i = 1, 2, \dots, N$ **do**

if $f(x_i) < f(p_{best_i})$ **then**

$p_{best_i} = x_i$;

end

if $f(p_{best_i}) < f(g_{best})$ **then**

$g_{best} = p_{best_i}$;

end

end

foreach *particle* $i = 1, 2, \dots, N$ **do**

обновить частицу скорости, используя (8);

обновить позицию частицы, используя (7);

end

until *maximum iteration is reached or stopping condition is true*

ПРИЛОЖЕНИЯ PSO

За последние два десятилетия параллельно с исследованиями в области алгоритмов исследователи PSO также уделяли особое внимание исследованиям, связанными с PSO, для конкретных приложений. Каждая область применения PSO влечет за собой некоторые

проблемы, которые необходимо устранить, чтобы найти успешные решения, которые затем сделают PSO более эффективным и действенным для применения в реальных условиях.

В работе [9] для решения задачи восприятия поисково-спасательными роботами различных газов на объектах химической промышленности была предложена гибридная система обнаружения запахов для этих роботов, основанная на оптимизации роем частиц. С целью повышения устойчивости и точности предсказания системы предлагается метод PSO для оптимизации весовых коэффициентов интегрированной нейронной сети. Для оптимизации весов и порогов нейронной сети был введен улучшенный алгоритм PSO. Оптимизированная сеть используется в системе обнаружения, что снижает ошибку обнаружения системы. Система проанализировала ответные сигналы 4 газовых смесей массива датчиков, экспериментальные результаты показывают, что алгоритм нейронной сети, основанный на оптимизации роем частиц, применяется для обучения количественной идентификации газовой смеси, и скорость сходимости выше, а точность обнаружения выше, чем у алгоритма обратного распространения ошибки.

Проблема распределения задач между несколькими роботами (the multi-robot task allocation – MRTA) в последнее время стала ключевой темой исследований. Распределение задач – это проблема сопоставления задач с роботами, так что выбирается наиболее подходящий робот для выполнения наиболее подходящей задачи, что приводит к оптимальному выполнению всех задач. Увеличение количества задач и роботов может привести к тому, что сотрудничество между роботами станет более жестким. Поскольку этот процесс требует большого вычислительного времени, в статье [10] описывается метод, который уменьшает размер исследуемого пространства состояний путем разделения задач на кластеры. В реальных задачах обычно отсутствует информация о количестве кластеров. Динамическая кластеризация благоприятна для разделения задач на соответствующее количество кластеров. В работе для решения задачи MRTA предлагается новый автоматизированный и эффективный алгоритм кластеризации задач роботов, основанный на оптимизации динамически распределенным роем частиц (the dynamic distributed PSO – D²PSO). Подход состоит из двух этапов: этап I группирует задачи в кластеры с использованием алгоритма D²PSO, а этап II распределяет роботов по кластерам. Назначение роботов кластерам представлено в виде множественных задач коммивояжера (multiple traveling salesman problems – MTSP). Были проведены вычислительные эксперименты, чтобы доказать эффективность подхода с точки зрения времени кластеризации, стоимости и времени MRTA по сравнению с распределенным PSO (the distributed PSO – dPSO) и генетическим алгоритмом (genetic algorithm – GA). Благодаря алгоритму D²PSO можно избежать проблем стагнации и локальных оптимумов за счет добавления разнообразия в популяцию без ущерба скорости сходимости PSO.

Целью исследования статьи [11] является сравнение трех разных контроллеров, настроенных с помощью PSO для управления траекторией робота. По этой причине PID-регулятор (пропорционально-интегрально-дифференцирующий регулятор) дробного порядка (Fractional order PID – FOPID), настроенный с помощью PSO, был изучен в первую очередь для планарного робота с двумя степенями свободы. Поскольку параметры контроллера FOPID были оптимизированы PSO для заданной траектории, для лучшего сравнения использовались три различные функции стоимости. Чтобы сравнить производительность оптимизированного контроллера FOPID с другими контроллерами, контроллер с нечеткой логикой (the fuzzy logic controller – FLC) и PID-регулятор также были настроены с помощью PSO. Для проверки надежности настроенных регуляторов параметры модели и заданная траектория были изменены, а в систему был добавлен белый шум. Все результаты моделирования траектории показали, что контроллер FOPID, настроенный с помощью PSO, имеет хорошие характеристики и лучше, чем FLC и PID, настроенные с

помощью PSO. Кроме того, результаты теста указывают на более высокую надежность контроллера FOPID по сравнению с другими.

Компьютерная диагностика (computer-aided diagnosis – CAD) является широко используемым методом для обнаружения и диагностики таких заболеваний, как опухоли, рак, отеки и т.д. Некоторые критические заболевания сетчатки, такие как диабетическая ретинопатия (diabetic retinopathy – DR), гипертоническая ретинопатия (hypertensive retinopathy – HR), дегенерация желтого пятна, пигментный ретинит (retinitis pigmentosa – RP), в основном анализируются на основе наблюдения изображений глазного дна. Необработанные изображения глазного дна имеют низкое качество, поэтому они не могут непосредственно отображать незначительные изменения. Для обнаружения и анализа незначительных изменений в сосудистой сети сетчатки или для применения усовершенствованных алгоритмов обнаружения заболеваний изображение глазного дна должно быть достаточно улучшено, чтобы визуализировать сосудистое скопление. Производительность моделей глубокого обучения для диагностики этих критических заболеваний сильно зависит от точной сегментации изображений. В частности, для сегментации сосудов сетчатки точная сегментация изображений глазного дна очень сложна из-за низкого контраста сосудов, различной ширины, ветвления и пересечения сосудов. Для повышения контрастности в различных методах сегментации сосудов сетчатки в качестве этапа предварительной обработки применяется повышение контрастности изображения, что может внести шум в изображение и повлиять на обнаружение сосудов. В последнее время во многих исследованиях применялось адаптивное выравнивание гистограммы с ограниченным контрастом (contrast limited adaptive histogram equalization – CLAHE) для повышения контрастности, но со значениями по умолчанию для контекстной области и ограничения клипа. В исследовании [12] цель состоит в том, чтобы улучшить производительность как контролируемых, так и неконтролируемых моделей машинного обучения для сегментации сосудов сетчатки путем применения модифицированной оптимизации роем частиц (modified particle swarm optimization – MPSO) для настройки параметров CLAHE с особым акцентом на оптимизацию предела отсечения и контекстуальных областей.

В работе оцениваются возможности оптимизированной версии CLAHE с применением стандартных метрик оценки. Используются изображения с контрастным усилением, полученные с помощью CLAHE на основе MPSO, чтобы продемонстрировать его реальное влияние на производительность модели глубокого обучения для семантической сегментации изображений сетчатки. Достигнутые результаты оказали положительное влияние на чувствительность моделей машинного обучения с учителем, что очень важно.

При использовании SVM (метода опорных векторов, или support vector machine) для решения практических задач выбор функции ядра и ее параметров играет важную роль в хороших или плохих результатах, и нужно только выбрать соответствующую функцию ядра и параметры, чтобы получить классификатор SVM с хорошей способностью к обобщению. Функция ядра RBF (ядро радиальной базисной функции, или radial basis function kernel) используется наиболее широко, и есть только два параметра: C и γ . В статье [13] обсуждается метод выбора параметров PSO и поиска по сетке соответственно. Метод поиска по сетке требует длительного поиска, за это время PSO легко попадает в локальный оптимум, из-за этих недостатков в данной работе предлагается усовершенствованный метод, сочетающий PSO и метод поиска по сетке. Сравнительный эксперимент по результатам ORL (оптические возвратные потери, или optical return loss) показывает, что предлагаемый метод имеет более высокую скорость и более высокую точность распознавания, чем метод поиска по сетке. Этот метод имеет более высокую точность распознавания, чем метод с одним PSO, и может эффективно избежать попадания в локальный оптимум. В статье [14] предлагается структура для диагностики болезни

Альцгеймера, которая состоит из предварительной обработки МРТ-изображений, выделения признаков, анализа главных компонент и модели SVM.

Для случаев с подозрением на COVID-19 крайне важно точно и быстро диагностировать заболевание, чтобы больных можно было изолировать и оказать им необходимую медицинскую помощь. Самообучающаяся автоматическая модель будет полезна для диагностики человека с подозрением на COVID-19 с помощью рентгена грудной клетки. Недавно для обнаружения COVID-19 были предложены конструкции на основе искусственного интеллекта, в которых используется рентген грудной клетки. Однако эти подходы либо используют закрытую базу данных, либо имеют сложную структуру. В исследовании [15] была предложена новая структура для обнаружения пациентов с коронавирусом в режиме реального времени без ручного вмешательства. В этой структуре содержится трехэтапный процесс, в котором первоначально выполняется кластеризация K-средних, а извлечение признаков выполняется как этап предварительной обработки данных. На втором этапе выбранные функции оптимизируются с помощью нового подхода к оптимизации функций, основанного на алгоритме гибридной дифференциальной эволюции и PSO. Затем оптимизированные функции перенаправляются в классификатор SVM. Эмпирические результаты показывают, что предложенная модель способна достичь точности 99,34%. Это показывает, что модель надежна и устойчива в диагностике человека, инфицированного COVID-19.

Многоцелевая оптимизация (multi-objective optimization – MOO) охватывает многие реальные проблемы оптимизации. Из-за внутренней противоречивости целей, подлежащих оптимизации, решение этих задач представляет собой сложную задачу. Задачи многоцелевой оптимизации решались с использованием различных методов вычислительного интеллекта. В статье [16] представлен PSO с комбинированными нормализованными целями для решения задач многоцелевой оптимизации, выбирающих оптимальные значения для ключевых технологических параметров процесса электролитической обработки, таких как скорость подачи инструмента, расход электролита, приложенное напряжение. Приложенное напряжение играет важную роль в оптимизации показателя производительности процесса. Предлагаемый PSO оценивает производительность скорости съема материала и шероховатости поверхности регрессионной модели и подтверждается с использованием экспериментальных результатов электрохимической обработки алюминиевых композитных материалов, а также проверочных испытаний. Предложенный алгоритм в сочетании с интеллектуальным методом производства приводит к снижению производственных затрат и времени, а также к большему увеличению гибкости выбора параметров обработки.

Задача оптимального управления генерирующим потребителем с возобновляемыми источниками энергии в интеллектуальной сети с распределенной генерацией и двусторонними потоками энергии рассмотрена в статье [17]. Решение данной задачи оптимизации выполнено алгоритмом PSO и другими роевыми алгоритмами, а также алгоритмом градиентного спуска. Полученные результаты показали высокую эффективность роевых алгоритмов, которые продемонстрировали надежную и быструю сходимость к глобальному экстремуму задачи оптимизации при различных вариантах взаимодействия и различных параметрах генерирующих потребителей.

ЗАКЛЮЧЕНИЕ

В заключение можно сказать, что PSO продемонстрировал преимущество во многих областях применения по сравнению со многими другими алгоритмами оптимизации. Некоторые недостатки, выявленные в алгоритме, были уменьшены различными модификациями канонического ω -PSO. Данный метод оптимизации является

универсальным, что еще больше подчеркивает его эффективность. Такая гибкость сделала PSO надежным оптимизатором в самых разных, но сложных сценариях оптимизации.

REFERENCES

1. Eberhart R., Kennedy J. Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks*, 4, IEEE, 1995. DOI: 10.1109/ICNN.1995.488968. Pp. 1942–1948.
2. Eberhart R., Kennedy J. A new optimizer using particle swarm theory. *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, IEEE, 1995. DOI:10.1109/MHS.1995.494215. Pp. 39–43.
3. Cleghorn C. W., Engelbrecht A. P. Particle swarm convergence: An empirical investigation. *2014 IEEE Congress on Evolutionary Computation (CEC) (2014)*. DOI: 10.1007/978-3-319-09952-112. Pp. 2524-2530.
4. Banks A., Vincent J., Anyakoha C. A review of particle swarm optimization. *Part I: background and development*, *Nat. Comput.*, 2007. DOI: 10.1007/s11047-007-9049-5. Vol. 4. No. 6. Pp. 467–484.
5. Karpenko A.P., Seliverstov E.Yu. A review of particle swarm methods for the global optimization problem (Particle Swarm Optimization). *Mashinostroyeniye i komp'yuternyye tekhnologii* [Mechanical Engineering and Computer Technologies]. 2009. No. 3. P. 2 (in Russian)
- Карпенко А. П., Селиверстов Е. Ю. Обзор методов роя частиц для задачи глобальной оптимизации (Particle Swarm Optimization) // *Машиностроение и компьютерные технологии*, 2009. № 3. С. 2.
6. Houssein E.H., Saad M.R., Hashim F.A., Shaban H., Hassaballah M. Levy flight distribution: a new metaheuristic algorithm for solving engineering optimization problems. *Eng. Appl. Artif. Intell.*, 2020. Vol. 94. Pp. 103731. DOI: 10.1016/j.engappai.2020.103731.
7. Cazzaniga P., Nobile M.S., Besozzi D. The impact of particles initialization in PSO: parameter estimation as a case in point. *IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, IEEE, 2015. Vol. 94. Pp. 1–8. DOI: 10.1109/CIBCB.2015.7300288.
8. Farooq M.U., Ahmad A., Hameed A. Opposition-based initialization and a modified pattern for inertia weight (IW) in PSO. *IEEE International Conference on Innovations in Intelligent Systems and Applications (INISTA)*, IEEE, 2017. Pp. 96–101. DOI: 10.1109/INISTA.2017.8001139.
9. Shang J. [et al.] Hybrid Odor Detection System for Search and Rescue Robot Based on PSO. *Chemical Engineering Transactions*, 2018. Vol. 68. Pp. 151–156.
10. Asma A., Sadok B. PSO-based dynamic distributed algorithm for automatic task clustering in a robotic swarm. *Procedia Computer Science*. 2019. Vol. 159. Pp. 1103–1112. DOI: 10.1016/j.procs.2019.09.279.
11. Bingul Z., Karahan O. Tuning of fractional PID controllers using PSO algorithm for robot trajectory control. *IEEE International Conference on Mechatronics*, 2011. Pp. 955–960. DOI: 10.1109/ICMECH.2011.5971254.
12. Aurangzeb K., Aslam S., Alhussein M., Naqvi R.A., Arsalan M., Haider S. I. Contrast Enhancement of Fundus Images by Employing Modified PSO for Improving the Performance of Deep Learning Models. *IEEE Access*. Vol. 9. Pp. 47930–47945. 2021. DOI: 10.1109/ACCESS.2021.3068477.
13. Taijia Xiao, Dong Ren, Shuanghui Lei, Junqiao Zhang, Xiaobo Liu. Based on grid-search and PSO parameter optimization for Support Vector Machine. *Proceeding of the 11th World Congress on Intelligent Control and Automation*, 2014. Pp. 1529–1533. DOI: 10.1109/WCICA.2014.7052946.

14. Zeng N. [et al]. A new switching-delayed-PSO-based optimized SVM algorithm for diagnosis of Alzheimer's disease. *Neurocomputing*, 2018. Vol. 320. Pp. 195–202. DOI: 10.1016/j.neucom.2018.09.001.

15. Dixit A., Mani A., Bansal R. CoV2-Detect-Net: Design of COVID-19 prediction model based on hybrid DE-PSO with SVM using chest X-ray images. *Information sciences*. 2021. Vol. 571. Pp. 676–692. DOI: 10.1016/j.ins.2021.03.062.

16. Prakash S.O., Jeyakumar M., Gandhi B.S. Parametric optimization on electro chemical machining process using PSO algorithm. *Materials Today: Proceedings*. 2022. DOI: 10.1016/j.matpr.2022.04.141.

17. Manusov V.Z., Matrenin P.V., Nasrullo H. Application of swarm intelligence algorithms in the management of a generating consumer with renewable energy sources. *Sistemy analiza i obrabotki dannykh* [Systems of Analysis and Data Processing]. 2019. Vol. 76. No. 3. Pp. 115–134. DOI: 10.17212/1814-1196-2019-3 -115-134 (in Russian)

Манусов В. З., Матренин П. В., Насрулло Х. Применение алгоритмов роевого интеллекта в управлении генерирующим потребителем с возобновляемыми источниками энергии // Системы анализа и обработки данных. 2019. Т. 76. № 3. С. 115–134. DOI: 10.17212/1814-1196-2019-3-115-134.

Информация об авторе

Казакова Елена Мусовна, мл. науч. сотр. отдела нейроинформатики и машинного обучения, Институт прикладной математики и автоматизации – филиал Кабардино-Балкарского научного центра РАН;

360000, Россия, Нальчик, ул. Шортанова, 89 А;

shogenovae@inbox.ru, ORCID: <https://orcid.org/0000-0002-5819-9396>

Information about the author

Kazakova Elena Musovna, Junior Researcher of the Department of Neuroinformatics and Machine Learning, Institute of Applied Mathematics and Automation – branch of Kabardino-Balkarian Scientific Center of the Russian Academy of Sciences;

360000, Russia, Nalchik, 89 A Shortanov street;

shogenovae@inbox.ru, ORCID: <https://orcid.org/0000-0002-5819-9396>