Совершенствование методов построения экстремальных систем управления

Д.В. Болдырев, А.И. Колдаев, А.А. Евдокимов, Ю.Н. Кочеров

Северо-Кавказский федеральный университет, г. Невинномысск

Аннотация: Представлен новый подход к повышению эффективности систем экстремального управления путем совершенствования метода поиска экстремума целевой функции. При ее многомерной нелинейной оптимизации вместо традиционного линейного поиска вдоль однократно выбранного направления используется исследующий поиск, направление которого на каждом шаге адаптируется к топологии целевой функции. Это позволяет максимально быстро локализовать экстремум и существенно сократить время его определения. Предложен алгоритм интерполяционного поиска экстремума в найденном интервале. Целевая функция моделируется сегментом кубического сплайна на основе информации о ее векторе-градиенте в граничных точках интервала, в результате чего количество шагов интерполяционного поиска существенно сокращается. Рассмотрена возможность упрощенной негладкой интерполяции сплайнами первого порядка в области экстремума. Результаты численного эксперимента подтверждают высокую эффективность нового метода при решении различных задач.

Ключевые слова: системы экстремального управления, нелинейная оптимизация, ускорение поиска экстремума, квазиньютоновский метод, полиномиальная интерполяция, негладкая интерполяция.

Введение

Самонастраивающиеся системы экстремального управления широко используются в различных областях науки и техники, таких как химическая технология, машинное обучение, анализ данных, обработка сигналов и изображений [1,2]. В условиях постоянного повышения требований к качеству выпускаемой продукции использование таких систем будет только расширяться. Это приводит к росту требований к их эффективности и качеству.

Качество экстремальной системы зависит от быстродействия и точности контура ее самонастройки. Эти показатели определяются совершенством алгоритма поиска экстремума целевой функции.

Разработка эффективных методов оптимизации сдерживается отсутствием универсальной стратегии поиска. Глобальная стратегия предполагает покрытие всей области поиска пробными точками, в которых вычисляется значение целевой функции. Это обеспечивает широту поиска, но снижает его эффективность. Локальная стратегия позволяет найти точное положение экстремума в ограниченном диапазоне параметров целевой функции, который может быть найден в процессе глобального поиска или быть известным априори [3]. Такая стратегия реализуется абсолютным большинством практически важных алгоритмов.

Проблема поиска экстремума целевой функции:

$$\underset{X \in R^n}{extr}\, Q(X),$$

где $Q: R^n \rightarrow R$ непрерывно дифференцируема, является сложной с вычислительной точки зрения, особенно если целевая функция нелинейная и имеет изменяющуюся топологию или большую размерность.

Существующие как детерминированные, так и стохастические методы локальной оптимизации, которые не используют информацию о производных $Q(X)$, генерируют значительное количество пробных точек, что существенно замедляет поиск. В данной статье алгоритмы этой группы рассматриваться не будут.

Большинство алгоритмов нелинейной оптимизации, использующих информацию о производных $Q(X)$, построены по общим принципам: сначала по определенным правилам выбирается направление поиска, затем в выбранном направлении экстремум $Q(X)$ локализуется с помощью линейного поиска. Эти действия повторяются до тех пор, пока не будет найдена стационарная точка функции $Q(X)$. Модернизация алгоритмов оптимизации обычно осуществляется следующими способами:

- изменение правил выбора начального шага поиска для увеличения скорости сходимости метода. Как правило, за основу берется алгоритм сопряженных градиентов [4]. Модификации квазиньютоновских алгоритмов (например, [5]) уделяется меньше внимания, по-видимому, из-за их достаточной с точки зрения авторов эффективности;

- изменение технологии линейного поиска путем адаптации его шага к профилю целевой функции [6,7];
- изменение правил учета ограничений при поиске экстремума [8]. Эту задачу можно рассматривать как частный случай безусловной оптимизации, поэтому алгоритмы этой группы рассматриваться не будут;
- адаптация алгоритмов к решению конкретных задач (таких, как задачи наименьших квадратов [9], задачи квадратичного программирования [10], задачи большой размерности, решение систем нелинейных уравнений, исследование составных оптимизационных моделей [11]).

Важно отметить, что все эти улучшения принципиально не меняют стратегию алгоритмов. В большинстве случаев они приводят к изменению способа выбора приемлемого направления, вдоль которого организуется линейный поиск экстремума. При этом игнорируется тот факт, что выбранное изначально удачное направление перестает быть таковым, если профиль $Q(X)$ меняется. Поскольку основное время работы алгоритмов тратится именно на линейный поиск, придерживаться его в удалении от области экстремума неоправданно.

Традиционно в процессе линейного поиска для снижения трудоемкости вектор-градиент $\nabla Q(X)$ не определяется, что приводит к неоправданным потерям времени на анализ топологии целевой функции. При этом информация о $\nabla Q(X)$, как правило, доступна. Пренебрегать ей неразумно, поскольку градиентные схемы значительно ускоряют определение экстремума с требуемой степенью точности.

В статье описан метод повышения эффективности многомерной нелинейной оптимизации, который сокращает время определения экстремума целевой функции.

Эффективный алгоритм нелинейной оптимизации

В основу алгоритма положена квазиньютоновская схема. Основное внимание уделено максимально быстрой локализации экстремума $Q(X)$. Проблема выбора начального шага поиска и, следовательно, задача аппроксимации обратной матрицы Гессе H не рассматривалась.

Линейный итерационный поиск заменен процедурой исследующего поиска экстремума $Q(X)$ и его последующего уточнения путем интерполяции. Чтобы учесть изменение профиля целевой функции, направление исследующего поиска регулярно корректируется на основе информации о векторе-градиенте $\nabla Q(X)$. Этот вектор определяет только направление, но не длину шага поиска, который остается постоянным или увеличивается. Поиск прекращается после нахождения двух точек, в которых первые производные целевой функции имеют противоположные знаки.

Внутри области экстремума организуется интерполяционный поиск. Функция моделируется сегментом кубического сплайна. Важно отметить, что линейный поиск с использованием кубической интерполяции использовался и ранее (например, в пакете научных подпрограмм IBM). Однако именно использование сплайнов, основанных на информации о значениях $Q(X)$ и $\nabla Q(X)$ в граничных точках области экстремума и учитывающих профиль целевой функции, позволяет сократить общее количество шагов интерполяционного поиска.

Ниже рис. 1 представлен эвристический алгоритм нелинейной оптимизации. Для определенности типом экстремума считается минимумом. Шаг исследующего поиска S предлагается выбирать так, как это принято в квазиньютоновских методах. В многомерных задачах используются скалярные проекции вектора-градиента $\nabla Q_S\left(X\right)$ на направление поиска.

Алгоритм 1: Поиск экстремума с использованием аппроксимации обратной матрицы Гессе

Входные данные: Базовая точка X_0, параметр длины шага λ_0, коэффициент акселерации $\alpha > 1$, точность поиска ε.

Шаг 1. Аппроксимировать обратную матрицу вторых производных целевой функции единичной матрицей $H \leftarrow I$.

Шаг 2. Задать $A \leftarrow X_0$, $\lambda \leftarrow \lambda_0$.

Шаг 3. *Выполнить исследующий поиск*
Определить шаг поиска $S \leftarrow -H\nabla Q(A)$ и выполнить движение в этом направлении $B \leftarrow A + \lambda S$.

Шаг 4. Если $\nabla Q_S(A)\nabla Q_S(B) < 0$, перейти к шагу 6.

Шаг 5. Пересчитать H по схеме квазиньютоновского метода, переназначить $A \leftarrow B$, $\lambda \leftarrow \alpha\lambda$ и перейти к шагу 3.

Шаг 6. *Выполнить интерполяционный поиск*
Уточнить минимум целевой функции на отрезке $[A,\ B]$

$$Q(X_1) \leftarrow \min_{0\le\gamma\le1} Q\left[A + \gamma(B - A)\right].$$

Шаг 7. Если выполняются условия сходимости для точек X_0 и X_1

$$\begin{cases} \left|Q(X_1) - Q(X_0)\right| < \varepsilon\left(1 + \left|Q(X_0)\right|\right), \\ \left\|X_1 - X_0\right\| < \sqrt{\varepsilon}\left(1 + \left\|X_0\right\|\right), \\ \left\|\nabla Q(X_1) - \nabla Q(X_0)\right\| < \sqrt[3]{\varepsilon}\left(1 + \left\|\nabla Q(X_0)\right\|\right), \end{cases}$$

завершить работу алгоритма с результатом X_1.

Шаг 8. Пересчитать H по схеме квазиньютоновского метода, переназначить базовую точку $X_0 \leftarrow X_1$ и перейти к шагу 2.

Рис. 1. – Алгоритм 1: Поиск экстремума с использованием аппроксимации обратной матрицы Гессе

На шаге 6 предлагается применить алгоритм интерполяции кубическими сплайнами на отрезке $[A,\ B]$, представленный на рис. 2.

Алгоритм 2: Сплайн-интерполяция с использованием первых производных целевой функции

Входные данные: Границы отрезка A и B, точность поиска ε.

Шаг 1. Определить базовую точку C как середину отрезка $[A, B]$.

Шаг 2. Определить направление $S \leftarrow B - A$.

Шаг 3. Определить вспомогательные параметры

$$p \leftarrow 2\frac{Q(A)-Q(B)}{\|S\|} + \nabla Q_S(A) + \nabla Q_S(B),$$

$$q \leftarrow 3\frac{Q(B)-Q(A)}{\|S\|} - 2\nabla Q_S(A) - \nabla Q_S(B),$$

Шаг 4. Определить интерполяционный параметр

$$\gamma \leftarrow \frac{\sqrt{q^2 - 3p\nabla Q_S(A)} - q}{3p}.$$

Если $\gamma \notin [0,1]$, переназначить $\gamma \leftarrow 0{,}5$.

Шаг 5. Выполнить интерполяцию $D \leftarrow A + \gamma S$.

Шаг 6. Если $\|D - C\| < \varepsilon$, завершить работу алгоритма с результатом D.

Шаг 7. *Уменьшить отрезок локализации минимума.*
Если $\nabla Q_S(A)\nabla Q_S(D) < 0$, переназначить $B \leftarrow D$,
иначе переназначить $A \leftarrow D$.

Шаг 9. Переназначить базовую точку $C \leftarrow D$ и перейти к шагу 2.

Рис. 2. – Алгоритм 2: Сплайн-интерполяция с использованием первых производных целевой функции

Если функция $Q(X)$ достаточно гладкая, можно использовать упрощенную интерполяцию сплайнами первого порядка, алгоритм которой представлен на рис. 3.

Результаты тестирования алгоритмов

Выполнено сравнение алгоритма № 1 и квазиньютоновского алгоритма BFGS [12]. Оценивалась эффективность локализации экстремума и эффективность интерполяционного поиска, который выполнялся с использованием кубической интерполяции и алгоритмов № 2 и № 3.

Алгоритм 3: Негладкая интерполяция с использованием первых производных целевой функции

Входные данные: Границы отрезка A и B, точность поиска ε.

Шаг 1. Определить базовую точку C как середину отрезка $[A,\ B]$.

Шаг 2. Определить направление $S \leftarrow B - A$.

Шаг 3. Определить интерполяционный параметр

$$\gamma \leftarrow \left(\nabla Q_S(B) - \frac{Q(B) - Q(A)}{\|S\|} \right) \cdot \frac{1}{\nabla Q_S(B) - \nabla Q_S(A)}.$$

Если $\gamma \notin [0,1]$, переназначить $\gamma \leftarrow 0{,}5$.

Шаг 4. Выполнить интерполяцию $D \leftarrow A + \gamma S$.

Шаг 5 *Скорректировать координаты минимума.*

Если $Q(A) < Q(B)$, то, пока $Q(D) > Q(A)$, переназначать $D \leftarrow \dfrac{A + D}{2}$.

Если $Q(A) \geq Q(B)$, то, пока $Q(D) > Q(B)$, переназначать $D \leftarrow \dfrac{B + D}{2}$.

Шаг 6. Если $\|D - C\| < \varepsilon$, завершить работу алгоритма с результатом D.

Шаг 7. *Уменьшить отрезок локализации минимума.*

Если $\nabla Q_S(A) \nabla Q_S(D) < 0$, переназначить $B \leftarrow D$,

иначе переназначить $A \leftarrow D$.

Шаг 9. Переназначить базовую точку $C \leftarrow D$ и перейти к шагу 2.

Рис. 3. – Алгоритм 3: Негладкая интерполяция с использованием первых производных целевой функции

Для тестирования отобраны невыпуклая функция Розенброка [11], функция Пауэлла с сингулярной матрицей Гессе в точке экстремума и мультимодальная функции Вуда [11], которые имеют сложную топологию. Для всех функций поиск начинался из одной и той же начальной точки и проводился с одинаковыми параметрами точности. Кратность испытаний составляла 5...7. Для каждого алгоритма оценивалась скорость уменьшения целевой функции и относительное время поиска (для каждой функции на единицу принималось минимальное значение этого показателя).

В таблице 1 приведено относительное время поиска экстремума различными методами. Изменение целевой функции в процессе работы алгоритмов показано на рис. 4.

Таблица № 1

Относительное время поиска экстремума

Алгоритм оптимизации	Алгоритм исследующего поиска		
	Кубическая интерполяция	Алгоритм 1	Алгоритм 2
Функция Розенброка			
BFGS	1,70	1,32	1,79
Алгоритм	1,32	1,00	1,13
Функция Пауэлла			
BFGS	1,76	1,21	2,03
Алгоритм	1,10	1,56	1,00
Функция Вуда			
BFGS	1,43	1,61	1,88
Алгоритм	1,32	1,00	1,12

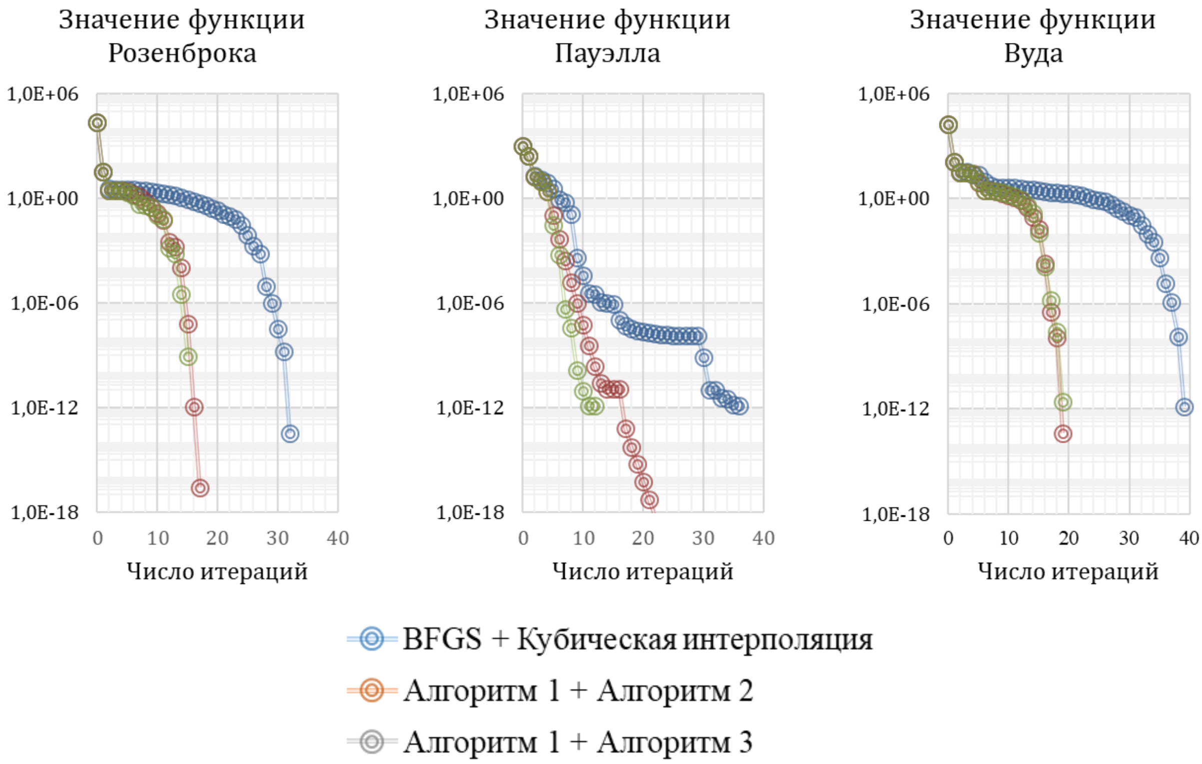

Рис. 4. – Сравнение производительности различных алгоритмов оптимизации

Полученные результаты дают основания утверждать, что регулярная коррекция направления исследующего поиска сокращает время работы алгоритмов на 15-50%, в зависимости от вида $Q(X)$. Некоторое снижение эффективности алгоритма № 1 для функции Пауэлла можно объяснить неудачным выбором направления из-за плохой обусловленности матрицы Гессе. Тем не менее, даже в этом случае на минимальный поиск затрачивается меньше шагов, чем для алгоритма BFGS.

Моделирование целевой функции с помощью кубического сплайна на основе информации о векторе-градиенте сокращает время работы алгоритмов на 10-20% в зависимости от вида $Q(X)$. Если экстремум функции выражен слабо, как в функции Пауэлла, то эффективность алгоритма № 2 снижается.

Алгоритм № 3, несмотря на менее строгий подход к воспроизведению профиля $Q(X)$, оказался достаточно эффективным. Он также более надежен, поскольку, в отличие от методов кубической интерполяции или сплайн-интерполяции, не содержит функций с ограниченной областью определения. Возможная ошибка в оценке положения экстремума устраняется путем коррекции на шаге 5.

Заключение

Основываясь на результатах исследования, можно сделать вывод, что Использование алгоритма № 1 в сочетании с алгоритмами № 2 или № 3 обеспечивает достижение меньшего значения $Q(X)$ и, следовательно, большую точность решения задачи оптимизации. Регулярная адаптация направления исследующего поиска к профилю целевой функции и использование информации о векторе градиента при интерполяционном поиске позволили получить эффективный и надежный алгоритм поиска экстремума.

Литература

1. Островский Г.М. Новый справочник химика и технолога. С.-Пб. АНО НПО «Профессионал», 2004. 848 с.

2. Казанцева А.М., Рыжкова Е.А., Масанов Д.В. К вопросу о распознавании контуров // Инженерный вестник Дона, 2023, № 3. URL: ivdon.ru/ru/magazine/archive/n3y2023/8254.

3. Яхина Е.П., Шаранин В.Ю. Оптимизация на основе смешения методов при решении задач многокритериального выбора // Инженерный вестник Дона, 2023, № 9. URL: ivdon.ru/ru/magazine/archive/n9y2023/8705.

4. Abdelrahman A., Mamat M., Mohd I., Rivaie V., Omer O. A new nonlinear conjugate gradient method // AIP Conference Proceedings. 2015. Vol. 1643(1). pp. 577-586. DOI: 10.1063/1.4907497.

5. Oliphant T.-L., Ali M.M. A trajectory-based method for mixed integer nonlinear programming problems // Journal of Global Optimization. 2018. Vol. 70(3). pp. 601-623. DOI: 10.1007/s10898-017-0570-5.

6. Armand P., Omheni R. A globally and quadratically convergent primal-dual augmented Lagrangian algorithm for equality constrained optimization // Optimization Methods and Software. 2017. Vol. 32(1). pp. 1-21. DOI: 10.1080/10556788.2015.1025401.

7. Sampaio P.R., Toint P.L. Numerical experience with a derivative-free trust-funnel method for nonlinear optimization problems with general nonlinear constraints // Optimization Methods and Software. 2016. Vol. 31(3). pp. 511-534. DOI: 10.1080/10556788.2015.1135919.

8. De Leone R. Nonlinear Programming and Grossone: Quadratic Programming and the role of Constraint Qualifications // Applied Mathematics and Computation. 2016. Vol. 318. DOI: 10.1016/j.amc.2017.03.029.

9. Chen L., Du C., Ma Y. The higher-order Levenberg-Marquardt method with Armijo type line search for nonlinear equations // Optimization Methods and Software. 2017. Vol. 32(3). pp. 516-533. DOI: 10.1080/10556788.2016.1225214.

10. Lu Z., Xiao L. A Randomized Nonmonotone Block Proximal Gradient Method for a Class of Structured Nonlinear Programming // SIAM Journal on Numerical Analysis. 2017. Vol. 55(6). pp. 2930-2955. DOI: 10.1137/16M1110182.

11. Химмельблау Д. Прикладное нелинейное программирование. М. : Мир, 1975. 536 с.

12. Li X., Wang B., Hu W. A modified nonmonotone BFGS algorithm for unconstrained optimization // Journal of Inequalities and Applications. 2017. Vol. 83. DOI: 10.1186/s13660-017-1453-5.

References

1. Ostrovskij G.M. Novyj spravochnik himika i tehnologa [The new handbook of chemist and technologist]. S.-Pb.: ANO NPO «Professional», 2004. 848 p.

2. Kazanceva A.M., Ryzhkova E.A., Masanov D.V. Inzhenernyj vestnik Dona, 2023, № 3. URL: ivdon.ru/ru/magazine/archive/n3y2023/8254.

3. Jahina E.P., Sharanin V.Ju. Inzhenernyj vestnik Dona, 2023, № 9. URL: ivdon.ru/ru/magazine/archive/n9y2023/8705.

4. Abdelrahman A., Mamat M., Mohd I., Rivaie V. AIP Conference Proceedings. 2015. Vol. 1643(1). pp. 577-586. DOI: 10.1063/1.4907497.

5. Oliphant T.-L., Ali M.M. Journal of Global Optimization. 2018. Vol. 70(3). pp. 601-623. DOI: 10.1007/s10898-017-0570-5.

6. Armand P., Omheni R. Optimization Methods and Software. 2017. Vol. 32(1). pp. 1-21. DOI: 10.1080/10556788.2015.1025401.

7. Sampaio P.R., Toint P.L. Optimization Methods and Software. 2016. Vol. 31(3). pp. 511-534. DOI: 10.1080/10556788.2015.1135919.

8. De Leone R. Applied Mathematics and Computation. 2016. Vol. 318. DOI: 10.1016/j.amc.2017.03.029.

9. Chen L., Du C., Ma Y. Optimization Methods and Software. 2017. Vol. 32(3). pp. 516-533. DOI: 10.1080/10556788.2016.1225214.

10. Lu Z., Xiao L. A SIAM Journal on Numerical Analysis. 2017. Vol. 55(6). pp. 2930-2955. DOI: 10.1137/16M1110182.

11. Himmel'blau D. Prikladnoe nelinejnoe programmirovanie [Applied nonlinear programming]. M.: Mir, 1975. 536 p.

12. Li X., Wang B., Hu W. Journal of Inequalities and Applications. 2017. Vol. 83. DOI: 10.1186/s13660-017-1453-5.