Распараллеливание итерационных методов решения вариационных неравенств*

Д.Н. Запорожец

Омский государственный технический университет

Рассмотрены эффективные методы решения вариационных неравенств: градиентный метод с постоянным шагом, одношаговый и двухшаговый экстраградиентные методы. Предложен итерационный метод с памятью как подход к распараллеливанию итерационных методов решения вариационных неравенств. Проведено сравнение эффективности полученных методов на тестовых задачах.

1. Введение

Вариационные неравенства являются удобным инструментом для решения многих задач. В терминах вариационных неравенств можно сформулировать задачи из различных областей, таких как: математическая физика, механика, экономика и другие.

Универсальным способом решения вариационных неравенств являются итерационные методы, в частности – градиентные. Однако для сходимости градиентных методов требуется выполнение условий сильной монотонности оператора вариационного неравенства или компактности исходного множества. Ослабить эти условия, а значит, расширить класс решаемых задач, позволяют одношаговый экстраградиентный метод, независимо предложенный Г. М. Корпелевич [1] и А.С. Антипиным [2] и двухшаговыйэкстраградиентный метод, предложенный Н. В. Меленьчуком [3, 4]. Экстраградиентные методы применимы при условии монотонности оператора вариационного неравенства и замкнутости исходного множества, а также эти методы обладают важным свойством сходимости из любой начальной точки.

Общим недостатком итерационных методов является строгая последовательность процесса и использование на каждой итерации только той информации, что получена на этой же итерации, невозможность начать вычисление очередного шага, пока не закончится вычисление предыдущего. Предложенные автором статьи итерационные методы с памятью [5] запоминают информацию о направлении на предыдущей итерации и используют её при вычислении очередной итерации. В связи с этим, в итерационных методах с памятью возможно использование параллельных вычислений на каждой итерации, что позволит сократить время решения задачи.

В данной статье этот подход применяется к градиентному методу, к одношаговому экстраградиентному методу и к двухшаговому экстраградиентному методу решения вариационных неравенств, проводится сравнение эффективности этих методов.

2. Постановка задачи и описание методов решения

Решить *вариационное неравенство* – значит найти такой вектор z^*, удовлетворяющий условиям:

$$(G(z^*), z - z^*) \geq 0, \forall z \in \Omega,$$

где $G(z): R^n \to R^n$, Ω – выпуклое, замкнутое множество.

Рассмотрим методы решения вариационных неравенств: градиентный метод с постоянным шагом, одношаговый экстраградиентный метод и двухшаговыйэкстраградиентный метод.

Градиентный метод с постоянным шагом имеет следующую вычислительную схему:

$$x_k = P_\Omega(x_{k-1} - \alpha G(x_{k-1})).$$

Здесь α– длина шага метода, G – оператор вариационного неравенства.

* Исследование выполнено при поддержке Министерства образования и науки Российской Федерации, соглашение № 14.B37.21.1123 и при поддержке РФФИ, проект № 12-07-00326

Одношаговый экстраградиентный метод задается следующими рекуррентными соотношениями:

$$\dot{x}_k = P_\Omega(x_{k-1} - \alpha G(x_{k-1})),$$
$$x_k = P_\Omega(x_{k-1} - \alpha G(\dot{x}_k)).$$

Основное отличие одношагового экстраградиентного метода от градиентного с постоянным шагом заключается в вычислении прогнозной точки $\dot{x}_k$. Для вычисления очередной точки используется направление из прогнозной точки.

Двухшаговыйэкстраградиентный метод описывается рекуррентными выражениями:

$$\dot{x}_k = P_\Omega(x_{k-1} - \alpha G(x_{k-1})),$$
$$\ddot{x}_k = P_\Omega(x_k - \alpha G(\dot{x}_k)),$$
$$x_k = P_\Omega(x_{k-1} - \alpha G(\ddot{x}_k)).$$

В отличие от одношагового экстраградиентного метода двухшаговый метод для вычисления очередной точки использует две прогнозных точки $\dot{x}_k$ и $\ddot{x}_k$.

Легко заметить, что все вышеперечисленные схемы строго последовательны. Вычисление очередной точки происходит в два этапа. Сначала вычисляется направление движения из текущей точки, затем делается шаг метода из текущей точки по вычисленному направлению. При этом на старте каждой итерации используется только точка, полученная на предыдущем шаге.

Итерационные методы с памятью запоминают направление движения метода на каждом шаге и используют его для вычисления очередной точки на следующем шаге. Рассмотрим их схему. Пусть итерационный процесс имеет вид

$$x_{k+1} = F(x_k),$$

где F – некоторая процедура, последовательность действий, необходимая для получения следующей точки.

Определение. Вектором движения итерационного процесса на k итерации называется разность точек, полученных на k и k–1 итерациях.

$$r_k = x_k - x_{k-1}.$$

Используя введенное определение можно на k итерации построить вспомогательную точку $\hat{x}_k$ путем прибавления вектора r_k, то есть

$$\hat{x}_k = x_k + r_k.$$

Тогда вычислительная схема метода с памятью выглядит следующим образом:

$$x_{k+1} = \begin{cases} F(\hat{x}_k), \|F(x_k) - \hat{x}_k\| \leq \|F(x_k) - x_k\| \dfrac{(F(x_k) - \hat{x}_k, F(x_k) - x_k)}{\|F(x_k) - \hat{x}_k\| \times \|F(x_k) - x_k\|} \\ F(x_k), \text{иначе} \end{cases}$$

Выполнение условия

$$\|F(x_k) - \hat{x}_k\| \leq \|F(x_k) - x_k\| \frac{(F(x_k) - \hat{x}_k, F(x_k) - x_k)}{\|F(x_k) - \hat{x}_k\| \times \|F(x_k) - x_k\|}$$

будем называть критерием одного направления.

Суть предлагаемого подхода заключается в том, что используя вычисленное ранее направление r_k, можно за линейное время вычислить вспомогательную точку $\hat{x}_k$. Затем вычислительный процесс можно распараллелить и на одном процессоре вычислить $F(x_k)$, в то время, как на другом –$F(\hat{x}_k)$. И если $F(x_k)$ и $\hat{x}_k$ оказываются достаточно близки, а именно, выполняется критерий одного направления, то в качестве следующей точки следует брать $F(\hat{x}_k)$. Геометрическая схема метода представлена на рисунке 1.

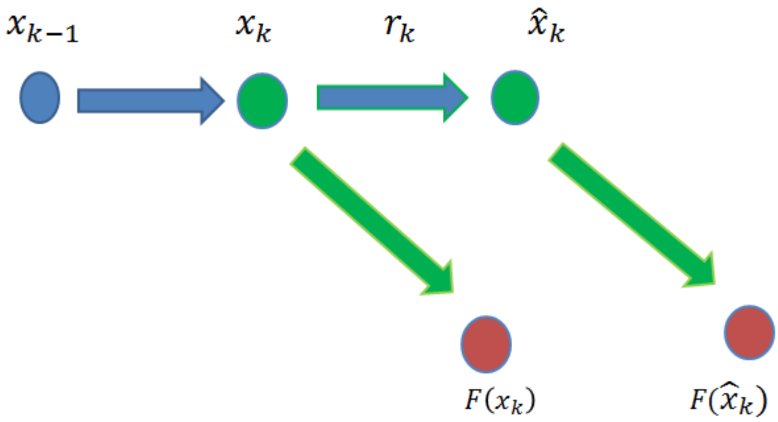

Рис. 1. Геометрическая схема итерационного метода с памятью

Данный подход позволяет существенно сократить число итераций в задачах, где вектор движения итерационного процесса меняется не часто. Также возможно его применение совместно с распараллеливанием трудоемких операций в вычислительной процедуре F. Применение предлагаемого подхода к градиентному методу с постоянным шагом, одношаговому экстраградиентному методу и двухшаговому экстраградиентному методу позволяет получить новый градиентный метод с постоянным шагом с памятью, одношаговый экстраградиентный метод с памятью и двухшаговый экстраградиентный метод с памятью.

3. Программная реализация итерационных методов с памятью

Методы реализованы на языке программирования C++. Так как степень параллелизма итерационных методов с памятью равна двум, то для их реализации достаточно системы с общей памятью, поэтому использовалась библиотека OpenMP 2.0.

Ниже на рисунке 2 приведен код двухшагового экстраградиентного метода с памятью. В коде переменная *dAlfa* отвечает за длину шага метода. Функция *AddVectors* складывает два вектора, которые передаются первым и вторым аргументами, результат сложения записывается в вектор, передаваемым 3 аргументом. Функция *Calc2StepIter* делает одну итерацию двухшаговым экстраградиентным методом. В качестве первого параметра функция принимает начальный вектор, вторым параметром передается вектор, который будет содержать конечную точку и третьим параметром передается длина шага метода. Функция *SubVectors* находит вектор, являющийся разностью двух других вектором. Первым параметром передается уменьшаемое, вторым параметром - вычитаемое, а разность будет содержаться в третьем параметре. Функция *NormSubVectors* вычисляет норму разности двух векторов без непосредственного вычисления вектора разности. Функция *MultyScalarVectors* вычисляет скалярное произведение векторов. Для реализации итерационного метода с памятью применительно к другому методу необходимо изменить правила вычисления константы *dAlfa* и заменить функцию *Calc2StepIter*.

```
doubledAlfa = 1.0/(1.05*sqrt(3.0)*L);
doubleCountIter22Step= 0;
do{
   AddVectors(&uk,&remdir,&vk);
   #pragma omp parallel if (proc_num > 1) num_threads(proc_num)
   {
      #pragma omp sections
      {
      #pragma omp section
            Calc2StepIter(&uk,&uk1, dAlfa);
      #pragma omp section
            Calc2StepIter(&vk,&vk1,dAlfa);
      }
   }
   Vector v(uk.GetSize());
   SubVectors(&uk1, &uk, &v);
   if ((remdir.Norm() != 0.0) && (NormSubVectors(&uk1,&vk) < MultyScalarVec-
tors(&v, &remdir) /remdir.Norm()))
   {
      uk1.SetEqual(&vk1);
      SubVectors(&vk1,&vk,&remdir);
   }
   Else
      SubVectors(&uk1,&uk,&remdir);

   dNorm = NormSubVectors(&uk1,&uk);
   uk.SetEqual(&uk1);
   CountIter22Step++;
}
      while (dNorm > eps);
```

Рис. 2. Двухшаговый экстраградиентный метод с памятью

4. Результаты распараллеливания итерационных методов

Рассмотрим, как применение итерационных методов с памятью влияет на решение задач. В качестве тестовых примеров возьмем некоторые примеры из монографии И. В. Коннова [6], а так же классические для методов оптимизации функции Розенброка и Химмельблау.

Линейная задача дополнительности (ЛЗД) заключается в том, что необходимо найти вектор u^* из R^n_+ и Mu^*+q из R^n_+, являющийся решением

$$< u^*, Mu^* + q > = 0$$

Матрицу M зададим следующим образом

$$M = \begin{cases} 2, i < j \\ 1, i = j, \\ 0, i > j \end{cases}$$

а вектор q возьмем равным $q = (-1,\dots,-1)$. В качестве допустимой области Ω выступает R^n_+. Решением такой задачи является $u^*=(0,\dots,0,1)$. В качестве начальной точки выберем $x_0 = (0,\dots,0)$.

Результаты вычислений представлены в таблице 1. Векторное поле оператора H и траектории методов показаны на рисунке 3. Штриховой линией показана траектория одношагового экстраградиентного метода, штрих-пунктирной линией – двухшагового экстраградиентного метода.

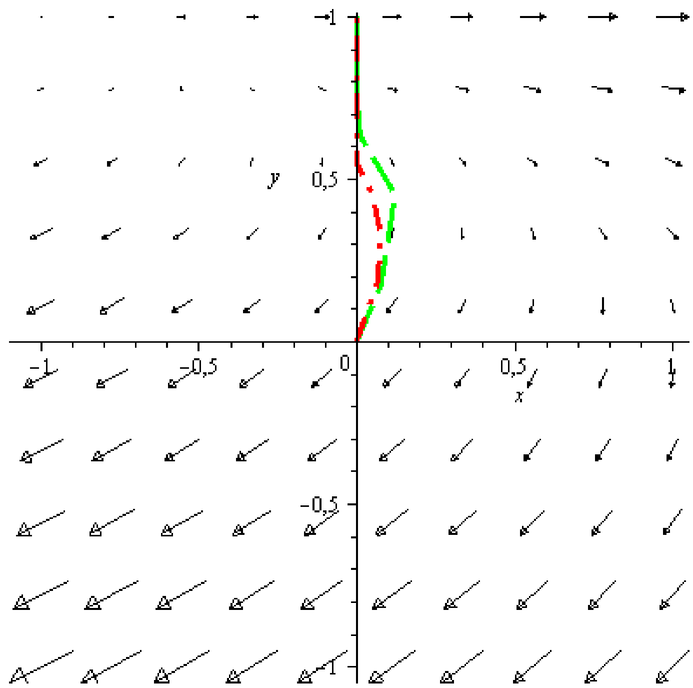

Рис. 3. Траектории экстраградиентных методов при решении ЛЗД

Таблица 1. Результаты решения ЛЗД

Размерность задачи	Исходный метод	Количество итераций исходным методом / с памятью	Эффективность
10	Градиентный	212/145	1.46
10	Одношаговый	302/203	1.49
10	Двухшаговый	371/195	1.9
100	Градиентный	1768/1192	1.48
100	Одношаговый	2406/1625	1.48
100	Двухшаговый	2879/1515	1.9
500	Градиентный	7194/4842	1.49
500	Одношаговый	9666/6498	1.49
500	Двухшаговый	11474/5995	1.91

Результаты показывают, что для приведенной линейной задачи дополнительности градиентный метод с памятью эффективнее градиентного на *46%*, одношаговый экстраградиентный метод с памятью эффективнее одношагового экстраградиентного метода на *49%*, а двухшаговый экстраградиентный метод с памятью эффективнее двухшагового экстраградиентного метода на *90%.* Такая закономерность обуславливается тем, что векторное поле оператора H меняется незначительно, и как следствие, в траекториях методов много шагов, идущих в одном направлении.

Решим нелинейное вариационное неравенство (НВН) с оператором

$$G(u) = \begin{pmatrix} (1+u_n)^3 u_1 \\ (1+u_{n-1})^3 u_2 \\ \dots \\ (1+u_1)^3 u_n \end{pmatrix}$$

и допустимой областью Ω, которая задается условием

$$h(u) = max(h_1(u), h_2(u)),$$

где

$$h_1(u) = (u_1 - 3)^2 + \sum_{i=2}^{n} u_i^2 - 9$$

$$h_2(u) = (u_1+1)^2 + \sum_{i=2}^{n} u_i^2 - 25$$

Решением вариационного неравенства является $u^*=(0, 0, \ldots, 0)$. В качестве стартовой точки выберем $x_0 = (3, 3, \ldots, 3)$. Результаты вычислений представлены в таблице 2. Векторное поле оператора вариационного неравенства и траектории методов при решении нелинейного вариационного неравенства размерности 2 показаны на рисунке 4. Штриховой линией показана траектория одношагового экстраградиентного метода, штрих-пунктирной линией – двухшагового экстраградиентного метода.

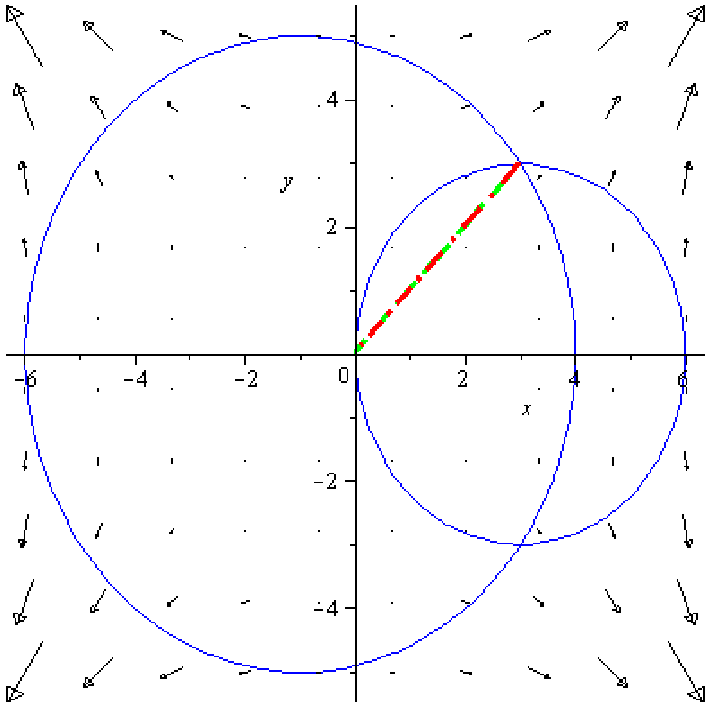

Рис. 4. Траектории экстраградиентных методов при решении НВН

Таблица 2. Результаты решения НВН

Размерность задачи	Исходный метод	Количество итераций исходным методом / с памятью	Эффективность
10	Градиентный	2379/1585	1.5
10	Одношаговый	3264/2175	1.5
10	Двухшаговый	3926/2099	1.59
100	Градиентный	2309/1538	1.5
100	Одношаговый	3163/2108	1.5
100	Двухшаговый	3801/2037	1.86
1000	Градиентный	2316/1544	1.5
1000	Одношаговый	3174/2116	1.5
1000	Двухшаговый	3814/2044	1.86

Результаты решения нелинейного вариационного неравенства подтверждают эффективность применения итерационных методов с памятью. Градиентный метод с памятью эффективнее градиентного на 50%, одношаговый экстраградиентный метод с памятью эффективнее одношагового экстраградиентного метода на 50%, а двухшаговый экстраградиентный метод с памятью эффективнее двухшагового экстраградиентного метода на 86%. Примечательно, что двухшаговый экстраградиентный метод с памятью решает задачу быстрее, чем одношаговый экстраградиентный метод с памятью, притом, что на той же задаче одношаговый экстраградиентный метод эффективнее двухшагового экстраградиентного метода.

Рассмотрим задачу поиска минимума функции Розенброка. Функция Розенброка — функция, используемая для оценки производительности алгоритмов оптимизации, предложенная Ховардом Розенброком в 1960 году. Функция Розенброка для двух переменных определяется как:

$$f(x,y) = (1-x)^2 + 100(y-x^2)^2$$

Известно, что функция достигает своего глобального минимума 0 в точке (1,1). Оптимизационную задачу поиска минимума функции Розенброка можно записать в виде вариационного неравенства, для этого в качестве оператора вариационного неравенства *H* нужно взять вектор

$$\begin{pmatrix} -400x(y-x^2)+2x-2 \\ 200(y-x^2) \end{pmatrix}$$

Функция Розенброка не является выпуклой. Граница области выпуклости походит вдоль кривой $y = x^2 + \frac{1}{200}$. Ниже этой кривой функция строго выпукла, а значит, имеет только одну точку минимума в этой области. Выше границы выпуклости функция Розенброка вогнута, а значит, не может иметь точек минимума. В качестве начальной точки возьмем (-1*,* *2*). Результаты решения задачи представлены в таблице 3.

Таблица 3. Результаты решения функции Розенброка

Размерность задачи	Исходный метод	Количество итераций исходным методом / с памятью	Эффективность
2	Градиентный	298307/198867	1.5
2	Одношаговый	373296/248861	1.5
2	Двухшаговый	423501/271004	1.56

В задаче поиска минимума функции Розенброка итерационные методы с памятью так же демонстрируют свою эффективность. Градиентный метод с памятью эффективнее градиентного на 50%, одношаговый экстраградиентный метод с памятью эффективнее одношагового экстраградиентного метода на 50%, а двухшаговый экстраградиентный метод с памятью эффективнее двухшагового экстраградиентного метода на 56%.

Другой классической тестовой функцией для методов оптимизации служит функция Химмельблау. Функция задается следующим образом:

$$f(x,y) = (x^2 + y - 11)^2 + (x + y^2 - 7)$$

Известно, что 0 является глобальным минимумом функции и достигается в четырех точках (-3.78; -3.28), (-2.8; 3.13), (3.58; -1.85), (3;2).

Задачу поиска минимума функции Химмельблау можно записать в виде вариационного неравенства, для этого в качестве оператора вариационного неравенства необходимо взять

$$H = \begin{pmatrix} 4x(x^2+y-11)+2(x+y^2-7) \\ 2(x^2+y-11)+4y(x+y^2-7) \end{pmatrix},$$

а в качестве допустимой области будем рассматривать первую четверть координатной плоскости. Тогда (3, 2) .будет единственной точкой минимума. Функция Химмельблау не является выпуклой. На рисунке 5 показаны области выпуклости и вогнутости функции. В таблице 4 показаны результаты решения задачи.

Таблица 4. Результаты решения функции Химмельблау

Размерность задачи	Исходный метод	Количество итераций исходным методом / с памятью	Эффективность
2	Градиентный	169/112	1.5
2	Одношаговый	247/163	1.52
2	Двухшаговый	308/162	1.9

В задаче поиска минимума функции Химмельблау, так же как и для функции Розенброка, итерационные методы с памятью оказываются эффективными. Градиентный метод с памятью эффективнее градиентного на 50%, одношаговый экстраградиентный метод с памятью эффективнее одношагового экстраградиентного метода на 52%, а двухшаговый экстраградиентный метод с памятью эффективнее двухшагового экстраградиентного метода на 90%.

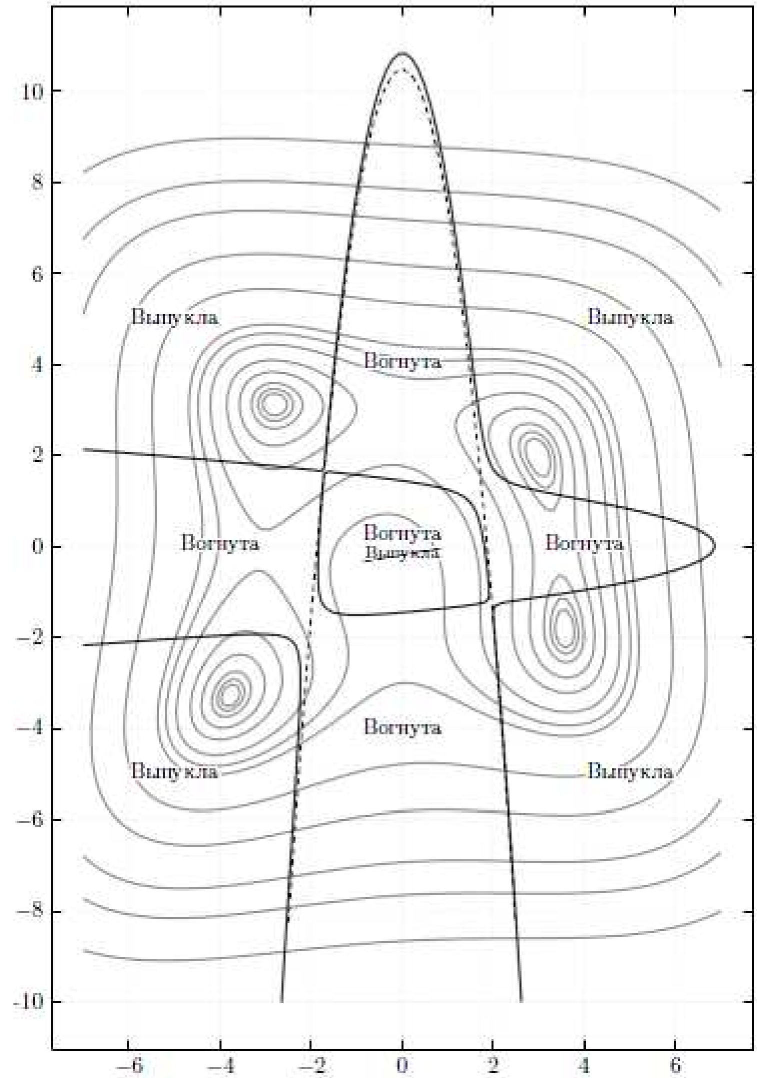

Рис. 5. Линии уровня и области выпуклости функции Химмельблау

5. Заключение

Повышение эффективности вычислительных методов решения вариационных неравенств актуально, поскольку с помощью вариационных неравенств могут быть сформулированы задача теории игр, задача о седловой точке, задача линейного программирования, линейная задача дополнительности и многие другие.

Рассмотрены следующие методы решения вариационных неравенств: градиентный метод, одношаговый и двухшаговый экстраградиентные методы и итерационный метод с памятью. Экстраградиентные методы по сравнению с градиентным обладают тем достоинством, что сходятся из любой точки и позволяют решать более широкий класс задач, за счёт использования прогнозных точек.

Применяя к любому вычислительному процессу итерационный метод с памятью можно существенно увеличить скорость получения решения за счёт использования информации о направлении на предыдущем шаге и применения параллельных вычислений.

В рассмотренных примерах применение итерационного метода с памятью к градиентному методу с постоянным шагом уменьшило количество итерации на46%, для одношагового экстраградиентного метода на49%, а для двухшагового экстраградиентного метода – на56%.

В дальнейших исследованиях планируется рассмотреть другие итерационные методы, а также расширить круг тестовых задач.

Литература

1. Корпелевич Г. М. Экстраградиентный метод для отыскания седловых точек и других задач // Экономика и математические методы. 1976. Т. 12, №4. С. 747–756.
2. Антипин А.С. Экстрапроксимальный метод решения равновесных и игровых задач // Журнал вычислительной математики и математической физики. 2005. Т.45, №11,12. С. 1969-1990, 2102-2111.
3. Зыкина А. В., Меленьчук Н. В. Двухшаговый экстраградиентный метод для вариационных неравенств. // Известия вузов. Математика. 2010. №9. С. 82–85.
4. Зыкина А.В., Меленьчук Н. В, Запорожец Д.Н. Сравнительный анализ экстраградиентных методов решения вариационных неравенств для некоторых задач. // Автоматика и телемеханика. – 2012. – № 4. – С. 32–46.
5. Запорожец Д.Н. Экстраградиентные методы с памятью. // XV Байкальская международная школа-семинар "Методы оптимизации и их приложения". Т. 2: Математическое программирование. Иркутск: РИО ИДСТУ СО РАН, 2011. С. 92 – 95.
6. Konnov I. V. Combined relaxation methods for variational inequalities. Berlin etc.: Springer, 2001