Лабораторная часть
Экстремумы ФНП

Мовчан Игорь Евгеньеви

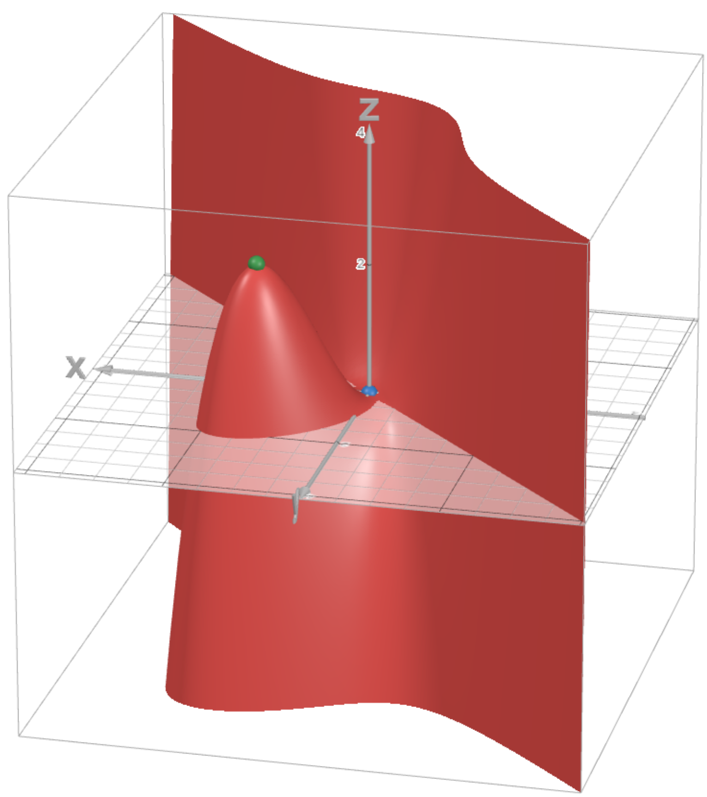
19 ноября 2023 г.

1 Аналитический метод (продолжение)

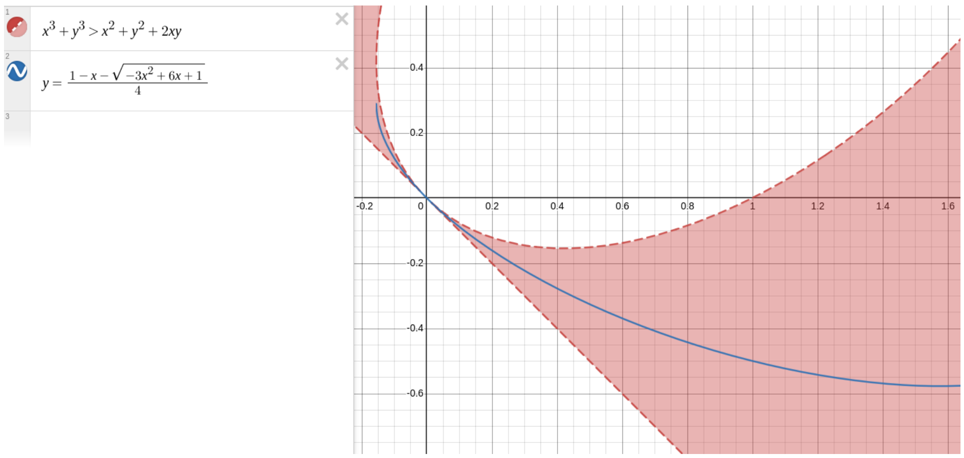
Функция задана следующим соотношением (8 вариант заданяи)

$$z(x, y) = x^2 + y^2 - x^3 - y^3 + 2xy.$$

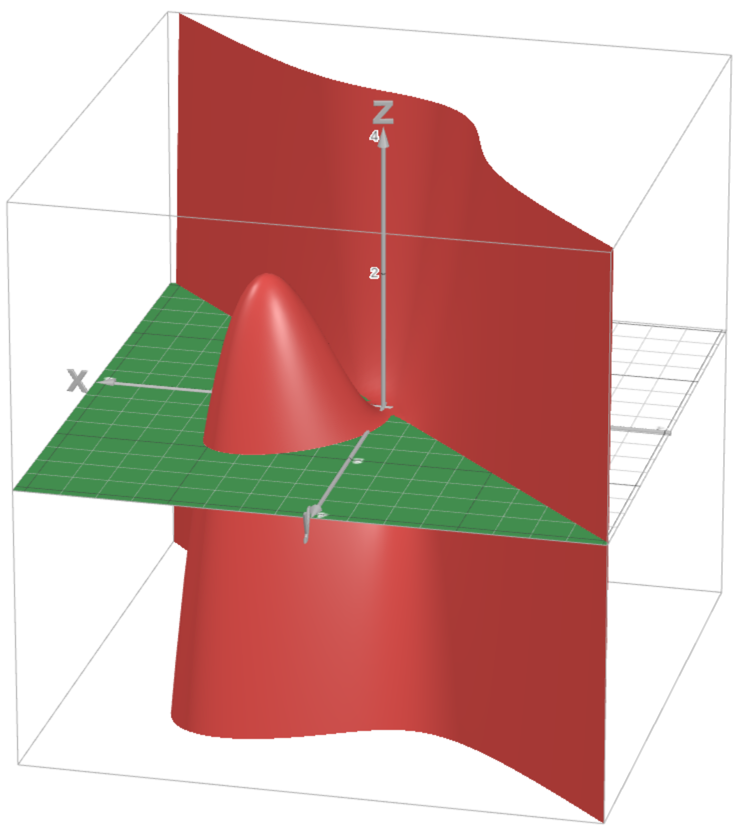
Её график вместе с найденными стационарными точками $(\frac{4}{3}, \frac{4}{3})$ и $(0, 0)$, выглядит, соответственно, следующим образом:



Визуализация неравенства $x^3 + y^3 > x^2 + y^2 + 2xy$ представлена ниже:



В пространстве оно полностью покрывает ту часть графика, которая находится ниже z = 0, чего и хотелось достичь:



2 Численный метод

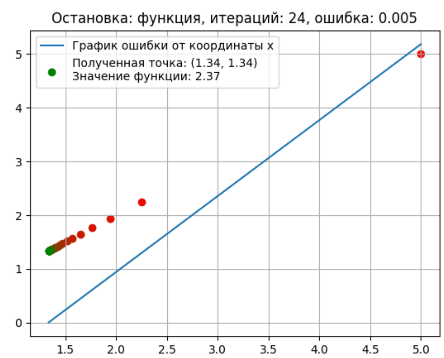
Выберем начальную (стартовую) точку $(5, 5) = (x_0, y_0)$, на каждом шаге будем вычислять значение функции, приращение функции и градиент в текущей точке (x_k, y_k), определять следующую точку по формуле $(x_{k+1}, y_{k+1}) = (x_k, y_k) + a_k \mathrm{grad} f(x_k, y_k)$. Суммируем, так как градиент показывает наискорейший рост функции, а мы как раз-таки ищем локальный максимум функции.

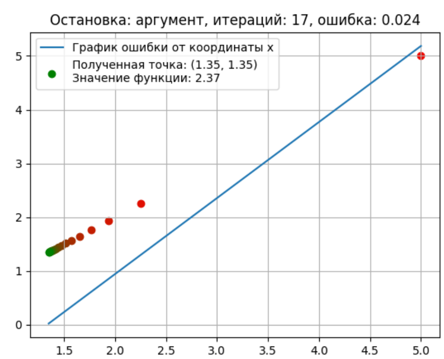
Градиент функции будет задаваться вектором

$$\mathrm{grad} f(x, y) = \begin{bmatrix} 2x - 3x^2 + 2y \\ 2y - 3y^2 + 2x \end{bmatrix}$$

Посмотрим теперь на графики. Во всех ситуациях чем точка зеленее, тем она дальше ушла по итерации, а чем краснее, тем, соответственно, ближе к началу вычислений. Также на графиках будет прописываться критерий остановки, если «функция» - по малости приращений значения функции, то есть $|\Delta f| < \varepsilon = 0.001$, а если «аргумент», значит, по малости приращения аргумента $||(\Delta x, \Delta y)|| < d = 0.001$. Рассматриваемая норма для расчёта приращения аргумента была стандартной теоремой Пифагора.

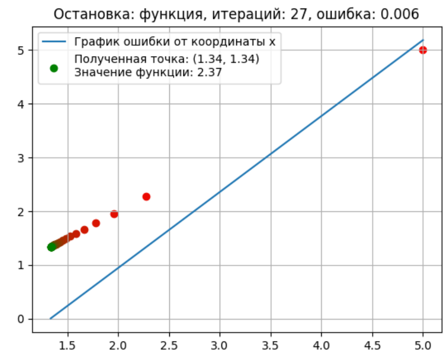
Рассмотрим ситуацию $a_k = 0.05$ (постоянное, малое значение, последнее нужно, чтобы значения не улетали на небеса):

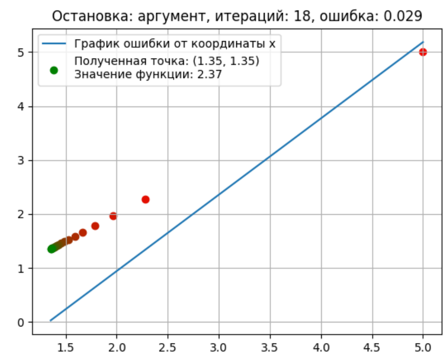




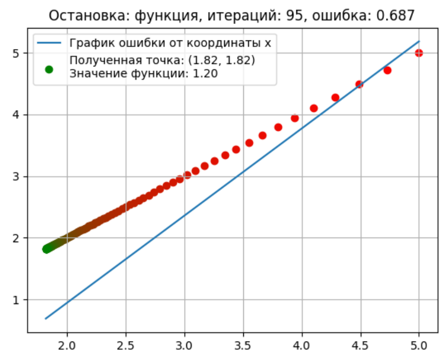
Видим, что значения достаточно близки к найденным вручную, а значит, программа хотя бы работает. Также графики показывают, что вычисления через функцию и аргумент сильно близки и при достойном a_k оба дают хорошие приближения.

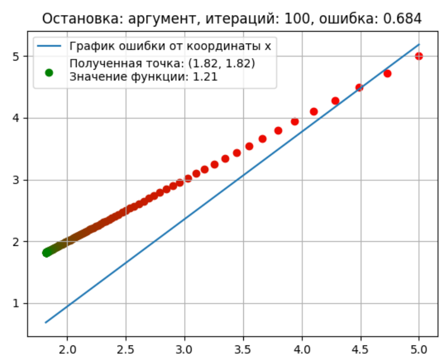
Поэкспериментируем с a_k, рассмотрим «линейное убывание». Достаточно приближенные к исходным точки значения получаем при $a_k = \frac{1}{20}(1 - \frac{k}{100})$:





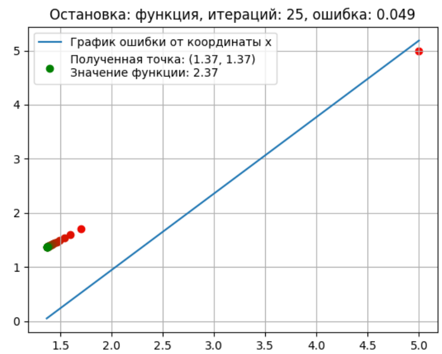
В ином же случае получаем либо взлёт по значениям, а это означает, что шаг выбран слишком большим, либо непохожие на реальность результаты (ниже представлены графики для $a_k = \frac{1}{200}(1 - \frac{k}{100})$):

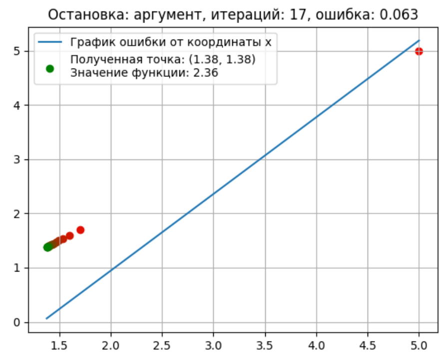




Это говорит нам о том, что a_k не должны быть как слишком малыми, так и слишком большим. И в том, и в том случае результаты работы программы нас будут неудовлетворять. Стоит отметить, что важным критерием подбора малости было количество итераций. Их малое количество свидетельствует о слишком малом a_k, то есть условие остановки срабатывает слишком рано. В идеале (исходя из опыта) их должно быть не меньше 15. Их также не должно быть слишком много (как мы увидели из примера с линейной a_k), ведь это тоже даёт нам неудовлетворительный результат.

Наконец, изучим ситуацию, когда a_k представлена в виде убывающей геометрической прогрессии. Возьмём $a_k = \frac{1}{15}(\frac{9}{10})^k$, где $\frac{1}{15}$ можно назвать нормировочный коэффициентом, нужным для того для того, чтобы функция не улетала на большие значения в самом начале, когда показатель в степени k ещё не слишком большой. В итоге получаем следующие графики:





Полученное полностью согласуется с полученными в аналитической части результатами.

Код программы вместе со всеми графиками: github link