Белорусский государственный технологический университет

Кафедра «Информационных систем и технологий»

Лабораторная работа № 11

**Оптимизация процессов и систем**

Выполнил студент

3 курса 2 группы

Процукович К.М.

Проверил

Колесников В. Л.

Минск 2019

1.Цель работы

Цель данной лабораторной работы ознакомиться с методами оптимизации, заняться вычислением целевой функции и провести вычислительный эксперимент.

2. Описание объекта исследования

2.1 Сущность

*Задачей оптимизации* в математике, информатике и исследовании операций называется задача нахождения экстремума (минимума или максимума) целевой функции в некоторой области пространства, ограниченной набором линейных и/или нелинейных равенств и/или неравенств.

Мы часто сталкиваемся с проблемой, когда из нескольких вариантов требуемся сделать выбор. Выбор предусматривает:

* анализ ситуации;
* сравнение вариантов по тому или иному критерию;
* выбор наилучшего.

Варианты, из которых производится выбор – *альтернативы*.

При решении задачи оптимизации мы решаем для себя следующие проблемы:

1. Что нужно улучшить? (определяем предмет оптимизации)
2. Что значит «лучше»? (увеличить или уменьшить?)
3. За счет чего можно добиться улучшения? (выбор параметров)
4. В каких пределах можно изменять выбранные параметры? (ОДЗ, ограничения).

Отвечая на все поставленные вопросы, мы формулируем «критерий оптимизации». Что позволяет выразить количественное выражение величины, зависящее от параметров? Конечно функция, представляющая собой некоторую математическую модель оптимизируемого процесса. Такую функцию называют критериальной (или целевой). Как всякая функция, целевая функция имеет свои параметры – параметры оптимизации, где f - имя критериальной функции, а x, y, z – параметры оптимизации.

По количеству независимых переменных различают задачи одномерной оптимизации и многомерной оптимизации.

## **2.2 Классификация**

Методы оптимизации классифицируют в соответствии с задачами оптимизации:

* *Локальные методы*: сходятся к какому-нибудь локальному экстремуму целевой функции.
* *Глобальные методы*: имеют дело с многоэкстремальными целевыми функциями.

Существующие в настоящее время методы поиска можно разбить на три большие группы:

* детерминированные;
* случайные (стохастические);
* комбинированные.

По критерию размерности допустимого множества, методы оптимизации делят на методы **одномерной** оптимизации и методы **многомерной** оптимизации.

По виду целевой функции и допустимого множества, задачи оптимизации и методы их решения можно разделить на следующие классы:

* Задачи оптимизации, в которых целевая функция и ограничения являются линейными функциями, разрешаются так называемыми методами линейного программирования.
* В противном случае имеют дело с задачей нелинейного программирования и применяют соответствующие методы.

Помимо того, оптимизационные методы делятся на следующие группы:

* аналитические методы (например, метод множителей Лагранжа и условия Каруша-Куна-Таккера);
* численные методы;
* графические методы.

Способ нахождения экстремума полностью определяется классом задачи. Но перед тем, как получить математическую модель, нужно выполнить 4 этапа моделирования:

1. Определение границ системы оптимизации
2. Выбор управляемых переменных
3. Определение ограничений на управляемые переменные
4. Выбор числового критерия оптимизации

При решении конкретной задачи оптимизации исследователь прежде всего должен выбрать математический метод, который приводил бы конечным результатам с наименьшими затратами на вычисления или же давал возможность получить наибольший объем информации об искомом решении. Выбор того или иного метода в значительной степени определяется постановкой оптимальной задачи, а также используемой математической моделью объекта оптимизации.

В настоящее время для решения оптимальных задач применяют в основном следующие методы:

· методы исследования функций классического анализа;

· методы, основанные на использовании неопределенных множителей Лагранжа;

· вариационное исчисление;

· динамическое программирование;

· принцип максимума;

· линейное программирование;

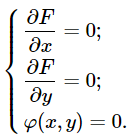
· нелинейное программирование.

Мы остановимся только на изучении численных методов оптимизации, поскольку это наиболее распространенные и эффективные методы. И хотя все численные методы оптимизации носят итерационный характер, а, следовательно, достаточно трудоемки, они очень алгоритмичны. Именно это свойство позволяет исключить ручные расчеты и полностью их автоматизировать (программировать).

**2.3 Метод множителей Лагранжа**

Метод множителей Лагранжа состоит в том, что для отыскания условного экстремума составляют функцию Лагранжа F(x,y)=f(x,y)+λφ(x,y) (параметр λ называют множителем Лагранжа).

Необходимые условия экстремума задаются системой уравнений, из которой определяются стационарные точки:



2.4 Алгоритмы

2.4.1 Алгоритм с возвратом при неудачном шаге

1. Задать начальный шаг *h*0, число проб *s*≤ *n* и точность *ε.*
2. Сгенерировать или задать начальную точку X0 и вычислить в ней функцию *f*; положить *i=*1 (*i* – счетчик проб).
3. Сгенерировать случайный вектор направления ⋅https://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-aKI9o6.png.
4. На направленииhttps://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-14ZINT.png определить точку X*k*+1 = X*k* + *hk*⋅Ξ*k* и вычислить в ней функцию *f*.
5. Проверить: а) если *f*(X*k*+1)<*f*(X*k*), положить *k*=*k+*1, *i=*1 и вернуться на 3; б) если *f*(X*k*+1)≥*f*(X*k*) и *i*<*s*, положить *i*=*i+*1 и вернуться на 3; в) если*f*(X*k*+1)≥*f*(X*k*), *i*=*s*и *hk*>*ε*, положить *hk*=*hk*/2, *i=*1 и вернуться на 3. г) если *f*(X*k*+1)≥*f*(X*k*), *i*=*s*и *hk*≤*ε*, поиск закончить, приняв X*k* за точку минимума.▲

Таким образом, поиск останавливается, если в текущей точке*s* направлений, сгенерированных подряд, оказались неудачными при шаге, меньшем заданной точности. На рис. 2.1 показан характер движения при поиске по данному алгоритму (жирной линией выделены успешные шаги).

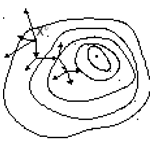


Рис. 2.1 – Характер движения

**2.4.2 Алгоритм с обратным** **шагом**

Он построен путем модификации предыдущего алгоритма с целью уменьшить число розыгрышей направлений. Внесено следующее изменение: если сгенерированное направление неудачное, то после возврата в исходную точку Xk делается шаг в противоположном направлении; если же и оно неудачное, то генерируется новое направление из точки Xk. Логика действия очевидна: если в данном направлении функция ухудшается, то можно ожидать, что в противоположном направлении она улучшится. Понятно, что эффект модернизации алгоритма тем выше, чем ближе функция в текущей области к линейной.

2.4.3 Алгоритм наилучшей пробы

Здесь используются два шага: пробный *α* и рабочий *h*. Величина пробного шага соответствует необходимой точности. Задается также число пробных шагов *m*, меньшее числа переменных *n*, причем разница между *m*и *n* увеличивается с ростом *n*.

В текущей точке генерируются *m* направлений Ξ*j*и на них делаются пробные шаги. Вычисляются изменения функции

https://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-ZYGraI.png.

В направлении с наименьшим (отрицательным) приращением Δ*fj*выполняется рабочий шаг:

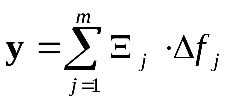
https://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-0L22t3.png.

Поиск заканчивается, еслиhttps://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-Ve6FQi.png.

По аналогии с предыдущим алгоритмом можно рассматривать и неудачные направления: вычислить . Если максимум соответствует положительному приращениюΔ*fj*, то рабочий шаг делается в противоположном направлении.

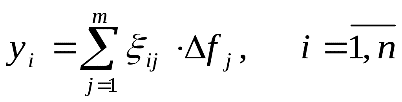
2.4.4 Алгоритм статистического градиента

Все параметры и начальные действия такие же, как в предыдущем алгоритме. Отличие состоит в выборе направления движения. Оно определяется с учетом всех разыгранных направлений:

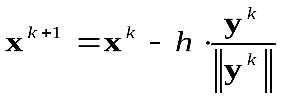
. (2.1)

Направление **y**называется *статистическим градиентом*. В пределе (*m*◊∞) он стремится к градиенту **∇***f*. Но определение **y**требует меньше вычислений, чем **∇***f*, и тем в большей степени, чем сильнее неравенство *m*<*n*.

Согласно (2.1) каждый компонент вектора **у**вычисляется по формуле

.

Новая точка находится перемещением на рабочий шаг *h* в направлении статистического антиградиента:

.

Поиск завершается при выполнении условия https://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-8PxQEQ.pngилиhttps://studfiles.net/html/710/197/html_A43b7MC95Q.knmu/img-1kKuua.png▲

1. **Ход работы**

Вся работа будет выполнятся в программном средстве «Optim».

Для начала работы нам необходимо создать переменные во всех предложенных вкладках.

В первой вкладке, как видно на рисунке 3.1, мы можем увидеть, что объявлено три переменные, выбранные еще в предыдущей лабораторной работе, с заданными границами:

* расход полимера (30-90),
* расход волокна (30-150),
* степень помола (20-40).

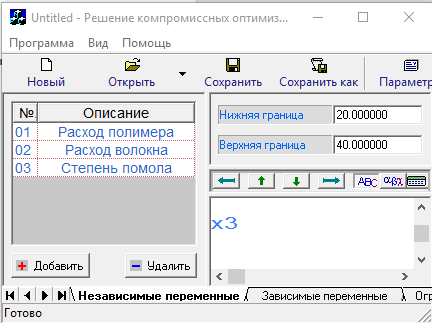


Рисунок 3.1 – Независимые переменные

После создания независимых переменных мы переходим к созданию зависимых, что можно увидеть на рис. 3.2. В нашем случае это будет прочность, пластичность и влагопрочность.

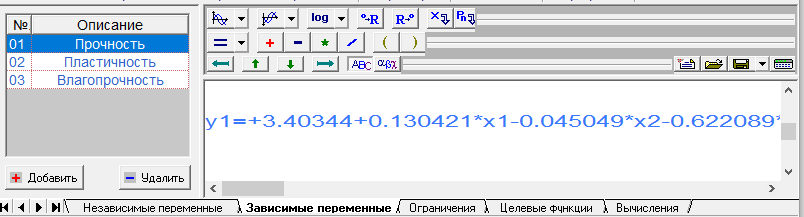


Рисунок 3.2 – Зависимые переменные

Третью вкладку, что мы посетим будет «ограничения», там мы впишем ограничения нашим зависимым переменным, что видно на рис. 3.3. По сути формируются требования к результатам.

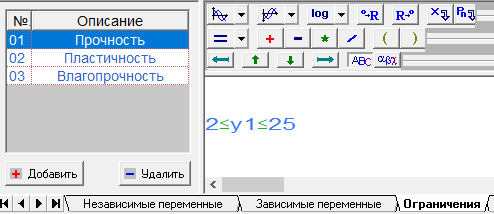


Рисунок 3.3 – Ограничения

И последняя вкладка с для создания переменных это «целевые функции», что мы видим на рис. 3.4. Там мы записываем целевую функцию, которая будет использоваться в дальнейшем.

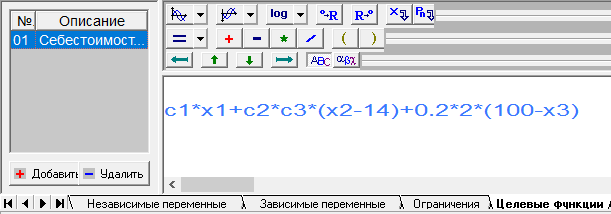


Рисунок 3.4 – Целевые функции

После задания параметров переходим во вкладку вычисления и нажимаем кнопку «Вычислить». Здесь мы можем использовать два метода, метод скользящего допуска и случайного локального поиска, использовать будем оба, после чего сформируем таблицы и проанализируем их.

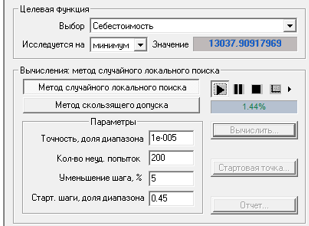


Рисунок 3.5 – Вычисление

Таблица 1 – Вычислительный эксперимент методом случайного локального поиска

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | Заданная  точность,  доля диапазона | Число неудачных  Попыток | Уменьшение  шага, % | Стартовый  шаг, доля  диапазона | Диапазон изменений результатов  решения задачи, max-min | | | | | |
| X1  (Расход полимера) | X2  (Расход волокна) | X3  (Степень помола) | Y1  (Прочность) | Себестоимость, Fx | Время расчета (с) |
| 1 | 0.000001 | 200 | 5 | 0.13 | 100.000000 | 25.000006 | 75.000089 | 0.626244 | 12678.00000000 | 10.7 |
| 2 | 0.00001 | 200 | 5 | 0.14 | 100.000071 | 25.000084 | 75.000075 | 0.626244 | 12665.00585938 | 10.8 |
| 3 | 0.0001 | 200 | 5 | 0.18 | 100.000844 | 25.001205 | 75.000071 | 0.626784 | 12650.09375000 | 8.1 |
| 4 | 0.001 | 200 | 5 | 0.19 | 100.005526 | 25.001173 | 75.018895 | 0.625500 | 12677.85937500 | 4.9 |
| 5 | 0.01 | 200 | 5 | 0.50 | 100.009890 | 25.006849 | 75.092512 | 0.624463 | 12654.06347656 | 4.2 |
| 6 | 0.000001 | 100 | 10 | 0.17 | 100.000000 | 25.000010 | 75.000787 | 0.627846 | 12690.00097656 | 8.5 |
| 7 | 0.00001 | 100 | 10 | 0.11 | 100.000018 | 25.000042 | 75.000038 | 0.625119 | 12650.00781250 | 8.2 |
| 8 | 0.0001 | 100 | 10 | 0.17 | 100.000196 | 25.000315 | 74.000084 | 0.627817 | 12650.06542969 | 6.6 |
| 9 | 0.001 | 100 | 10 | 0.15 | 100.001178 | 25.002594 | 72.016983 | 0.625948 | 12651.10156250 | 5.7 |
| 10 | 0.01 | 100 | 10 | 0.50 | 100.017886 | 25.045616 | 75.179230 | 0.633624 | 12159.87207031 | 2.1 |
| 11 | 0.000001 | 50 | 15 | 0.13 | 100.000000 | 25.000008 | 75.000000 | 0.625128 | 12650.00000000 | 9.8 |
| 12 | 0.00001 | 50 | 15 | 0.14 | 100.000018 | 25.000216 | 75.000107 | 0.678982 | 13650.02148438 | 7.8 |
| 13 | 0.0001 | 50 | 15 | 0.15 | 100.001024 | 25.000311 | 78.001198 | 0.675971 | 12650.19140625 | 6.8 |
| 14 | 0.001 | 50 | 15 | 0.20 | 100.002330 | 25.011873 | 75.008278 | 0.625178 | 12651.14453125 | 4.4 |
| 15 | 0.01 | 50 | 15 | 0.50 | 100.037363 | 25.093063 | 75.195465 | 0.627817 | 12655.85546875 | 2.9 |
| 16 | 0.000001 | 100 | 20 | 0.13 | 100.000782 | 25.000038 | 74.000015 | 0.625948 | 12750.00195313 | 5.5 |
| 17 | 0.00001 | 100 | 20 | 0.14 | 100.000078 | 25.000168 | 74.000053 | 0.725922 | 12650.01171875 | 7.7 |
| 18 | 0.0001 | 100 | 20 | 0.14 | 100.000014 | 25.001873 | 74.000763 | 0.626369 | 12650.12890625 | 6.7 |
| 19 | 0.001 | 100 | 20 | 0.20 | 100.000912 | 25.004395 | 75.000000 | 0.626272 | 12650.95312500 | 5.6 |
| 20 | 0.01 | 100 | 20 | 0.50 | 100.020129 | 25.059793 | 72.016983 | 0.639841 | 12656.86914063 | 2.2 |
| 21 | 0.000001 | 200 | 25 | 0.13 | 100.000730 | 25.000006 | 75.179230 | 0.625948 | 12680.00097656 | 8.6 |
| 22 | 0.00001 | 200 | 25 | 0.14 | 100.000017 | 25.000065 | 75.000000 | 0.627817 | 12650.01171875 | 6.7 |
| 23 | 0.0001 | 200 | 25 | 0.15 | 100.000487 | 25.000856 | 75.000107 | 0.325948 | 12650.11035156 | 5.1 |
| 24 | 0.001 | 200 | 25 | 0.20 | 100.008112 | 25.015192 | 75.026703 | 0.628768 | 12651.91796875 | 4.2 |
| 25 | 0.01 | 200 | 25 | 0.14 | 100.024193 | 25.032524 | 75.175797 | 0.627430 | 12660.37304688 | 2.3 |
| 27 | 0.00001 | 50 | 30 | 0.14 | 100.000053 | 25.000092 | 75.115524 | 0.621245 | 12613.70898438 | 9.7 |
| 28 | 0.0001 | 50 | 30 | 0.17 | 100.000018 | 25.000223 | 77.268692 | 0.275043 | 12850.74609375 | 6.1 |
| 29 | 0.001 | 50 | 30 | 0.22 | 100.009890 | 25.008333 | 75.010689 | 0.627829 | 12652.26660156 | 3.7 |
| 30 | 0.01 | 50 | 30 | 0.595 | 100.034599 | 25.094189 | 75.385506 | 0.634178 | 12641.50781250 | 3 |
| 31 | 0.000001 | 200 | 50 | 0.13 | 100.000018 | 25.000006 | 75.000023 | 0.625642 | 12650.00195313 | 9.1 |
| 32 | 0.00001 | 200 | 50 | 0.14 | 100.000912 | 25.000069 | 75.000099 | 0.625954 | 12650.01660156 | 8.7 |
| 33 | 0.0001 | 200 | 50 | 0.15 | 100.000465 | 25.001070 | 75.003679 | 0.546055 | 12650.23046875 | 5.6 |
| 34 | 0.001 | 200 | 50 | 0.20 | 100.000196 | 25.000183 | 75.004906 | 0.625866 | 12650.75488281 | 6.2 |
| 35 | 0.01 | 200 | 50 | 0.50 | 100.001178 | 25.266590 | 75.005779 | 0.475821 | 12655.58886719 | 2.1 |
| 36 | 0.000001 | 100 | 75 | 0.13 | 100.017886 | 25.000137 | 79.019798 | 0.420830 | 12778.64062500 | 7.5 |
| 37 | 0.00001 | 100 | 75 | 0.14 | 100.005526 | 26.546869 | 77.431427 | 0.888213 | 12805.20605469 | 6.9 |
| 38 | 0.0001 | 100 | 75 | 0.15 | 100.009890 | 27.646431 | 75.962255 | 0.850933 | 12813.17285156 | 5.2 |
| 39 | 0.001 | 100 | 75 | 0.20 | 100.000000 | 25.961687 | 78.407219 | 0.642941 | 12809.05761719 | 4.7 |
| 40 | 0.01 | 100 | 75 | 0.50 | 100.000018 | 25.009205 | 75.347656 | 0.614223 | 12663.76953125 | 1.5 |

Выделим из таблицы лучший и худший результат. Лучший результат – Y1 = 0.888213 при себестоимости 12805.20605469, худший – Y1=0.275043 при себестоимости 12850.74609375. Среднее значение Y1 = 0.633624, себестоимость - 12850.74609375.

Таблица 2 – Вычислительный эксперимент методом скользящего допуска

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № | Заданная  точность,  доля диапазона | Число неудачных  попыток | Диапазон изменений результатов  решения задачи, max-min | | | | | |
| X1  (Расход полимера) | X2  (Расход волокна) | X3  (Степень помола) | Y1  (Прочность) | Себестоимость, Fx | Время расчета (с) |
| 1 | 0.000001 | 200 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 7.8 |
| 2 | 0.00001 | 200 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 7.4 |
| 3 | 0.0001 | 200 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 7.8 |
| 4 | 0.001 | 200 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | **8.2** |
| 5 | 0.01 | 200 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 7.7 |
| 6 | 0.000001 | 100 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 5.6 |
| 7 | 0.00001 | 100 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 5.5 |
| 8 | 0.0001 | 100 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 5.2 |
| 9 | 0.001 | 100 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 5.7 |
| 10 | 0.01 | 100 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 5.6 |
| 11 | 0.000001 | 50 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 4.3 |
| 12 | 0.00001 | 50 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 4.8 |
| 13 | 0.0001 | 50 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 4.4 |
| 14 | 0.001 | 50 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | **4.2** |
| 15 | 0.01 | 50 | 101.678593 | 27.569912 | 71.185655 | 0.411252 | 11789.78912354 | 4.5 |

В данном эксперименте определили лучшее и худшее время. Наилучшим временем стало 4.2 секунд, наихудшим – 8.2 секунд.

**Выводы**

В данной работе были освоены методы решения оптимизационных задач при помощи программного средства «Решение компромиссных оптимизационных задач».

Оптимизация производственного процесса представляет собой основную задачу для любого человека, управляющего предприятием. И применение данных методов делает этот процесс легким в понимании, интерпретации, так как из всех методов системного анализа, данный метод показал себя как наиболее понятный с точки зрения формулирования рекомендации по заданию технологического режима производства.

В сочетании с моделированием производственных процессов, решение оптимизационных задач является наиболее мощным и результативным методом системного анализа из всех изученных ранее.