

Точная формулировка задания

Каждый должен реализовать свою программу решения ЗКВ с помощью МВГ. При этом рассматриваем такие варианты заданий.

А. Размерность задачи:

A1: 21 и 35;

A2: 22 и 34;

A3: 23 и 33;

A4: 24 и 32;

A5: 25 и 31.

Б. Исходная функция плотности вероятности для случайных данных:

B1: $c_1 \cdot (1+x^4)$, $x \in [0,1]$;

B2: $c_2 \cdot (1+x^2)$, $x \in [0,1]$;

B3: $c_3 \cdot (1+x^{1/2})$, $x \in [0,1]$;

B4: $c_4 \cdot (1+x^{1/4})$, $x \in [0,1]$

(во всех случаях надо подобрать константу c_i , чтобы на отрезке $[0,1]$ такой плотностью вероятности задавалось бы всё распределение, т.е. нужный интеграл от 0 до 1 давал бы значение 1; я вычислять не буду, это задача начала 2-го курса)

(при этом можно сказать, что «обычное» равномерное распределение имеет плотность c , и не будет ошибкой сказать $c \cdot (1+x^0)$, $x \in [0,1]$).

В. Разделяющий элемент выбирать:

V1: так, как в книге Гудмана – Хидетниemi (ну или в моих лекциях);

V2: полными вычислениями – чтобы была максимальной разность между увеличением границы левой и правой подзадач (считается гораздо дольше, чем V1).

Таким образом, возможны $5 \times 4 \times 2 = 40$ вариантов работы (зависящие от исходных данных и алгоритма выбора разделяющего элемента) – конкретные варианты по студентам см. ниже.

Общее описание задачи.

1. Фиксируем размерность. (Повторю, что для всех трёх размерностей вашего варианта надо всё это сделать.)

2. Для сгенерированных данных запускается МВГ. На 10 *тестовых* запусках считается *среднее время выполнения*. (Здесь желательно употреблять V1 – даже если в вашем варианте V2. Все значения времени будем считать с помощью `clock()`. Причём здесь и далее: если задача не досчитывается за «естественное» время, мы считаем «попытку неудачной» и приступаем к следующей попытке; впрочем, для указанных размерностей, максимум 35, это маловероятно.) Считаем это полученное среднее время T характеристикой размерности.

3. Далее запускаем МВГ с *ограниченным временем выполнения* (truncated branch-and-bound method), причём 5 вариантов: $0.01 \cdot T$, $0.03 \cdot T$, $0.1 \cdot T$, $0.3 \cdot T$ и $0.5 \cdot T$ (т.е. 1%, 3%, 10%, 30% и 50% от времени, *предварительно вычисленного для этой размер-*

ности). Каждый из этих вариантов запускается 2 способами: с применением вызова последовательности правых задач (ППЗ, см. материал лекции) и без его применения. Таким образом, всего у каждого из вас будут такие разные задачи:

- 2 варианта размерности;
- 5 вариантов времени;
- 2 варианта вызова ППЗ –

итого $2 \times 5 \times 2 = 20$ вариантов; *эти 20 разных задач и будут общими результатами вашей работы.*

4. Для каждого запуска фиксируем *результат* (значение целевой функции), полученный к окончанию выделенного времени. После этого досчитываем задачу до конца – и тоже получаем значение целевой функции. Отношение оптимального значения к значению целевой функции, полученному к окончанию выделенного времени, назовём *качеством* решения. Усредняем эти значения качества на 10 *новых* запусках – это будет критерием качества этой *одной из 20 задач, указанных в пункте 3.*

5. При этом само усреднение можно делать разными способами (не только как среднее арифметическое) – и в качестве небольшого дополнительного задания (которое даст дополнительные баллы) – надо будет получить описание усреднения, применить его и включить в ваш отчёт.

Все действия (начиная от подбора константы для пункта «Б» и заканчивая «осмыслением» полученных результатов) нужно представить в файле-отчёте (в виде pdf). При этом желательно включить в pdf и текст функций на Си++ – но обязательно с комментариями, отражающими ваш текст отчёта! Весь текст программы представляйте в одном отдельном файле (cpp; можно 2 файла, h + cpp).

Вот возможные конкретные варианты по студентам (каждый выбирает 1 строчку, без повторения номера в группе):

1. A1 B1 V1
2. A4 B2 V1
3. A1 B1 V2
4. A4 B4 V2
5. A1 B2 V1
6. A5 B1 V1
7. A1 B2 V2
8. A5 B1 V1
9. A2 B1 V1
10. A5 B1 V2
11. A3 B4 V2
12. A5 B2 V1
13. A2 B2 V1
14. A5 B2 V2
15. A2 B2 V2
16. A4 B3 V1
17. A3 B1 V1
18. A4 B3 V2
19. A3 B1 V2
20. A5 B3 V1

21. A3 B2 B1
22. A5 B3 B2
23. A3 B2 B2
24. A5 B4 B1
25. A4 B1 B1
26. A5 B4 B2
27. A4 B1 B2
28. A3 B4 B1