

предназначена для настройки менеджеров процессов на текущем компьютере и других компьютерах сети. Для этого она подсоединяется к менеджерам процессов на выбранных компьютерах, читает имеющиеся у них настройки, сообщает им новые настройки, если нужно.

В поле «hosts» надо вписать имена всех компьютеров кластера и создать общий сетевой ресурс, доступный всем компьютерам кластера. Для запуска MPI-программ использовалась входящая в комплект MPICH2 программа Wmpiexec, которая представляет собой оболочку вокруг соответствующей утилиты командной строки Mpiexec.

Ниже приведен график зависимости времени счета  $T$  программы от количества компьютеров  $N$  и количества процессов  $P$ . Программа использовалась для расчета интегралов

вида:  $\int_0^{10} \frac{dx}{ax^2 + b}$ , где  $a=1,2..5$ ,  $b=1,2..5$  – изменяющиеся параметры, с числом шагов  $10^9$ .

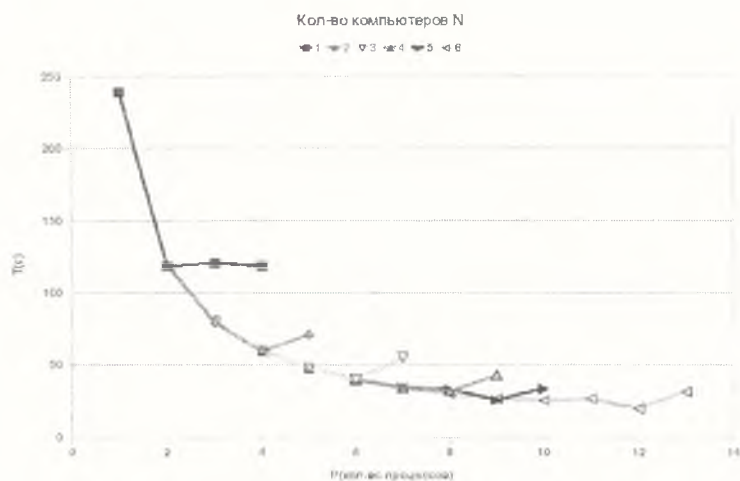


Рисунок 1

Как видно из графика (рисунок 1), наименьшее время выполнения соответствует использованию шести компьютеров с двенадцатью процессами, по 2 на каждый компьютер. То есть, при  $N=6$  и  $P=12$  производительность увеличилась в 12 раз ( $238,9(c)/19,939(c)=11,986$ ).

Таким образом, реализованный кластер показал хороший прирост производительности, что позволяет использовать его для численного решения сложных физических задач даже на базе обыкновенного дисплейного класса, без особых материальных затрат.

## УСЛОВИЯ СУБНОРМАЛЬНОСТИ ДЛЯ ПОДГРУПП КОНЕЧНЫХ ГРУПП

*В. А. Ковалева (УО «ГГУ им. Ф. Скорины»)*

*Научн. рук. А. Н. Скиба,*

*доктор физ.-мат. наук, профессор*

Все рассматриваемые группы являются конечными.

Пусть  $A$ ,  $K$  и  $H$  – подгруппы группы  $G$  и  $K \leq H \leq G$ . Тогда мы говорим, что  $A$  покрывает пару  $(K, H)$ , если  $AH = AK$ ;  $A$  изолирует пару  $(K, H)$ , если  $A \cap H = A \cap K$  [1]. Пара  $(K, H)$  группы  $G$  называется максимальной, если  $K$  – максимальная подгруппа в  $H$ .

В данном сообщении, основываясь на теории покрытия и изолирования максимальных пар, мы приводим новые условия субнормальности подгрупп.

**Определение.** Пусть  $A$  – подгруппа группы  $G$  и  $\Sigma = \{G_0 \leq G_1 \leq \dots \leq G_n\}$  – некоторый ряд подгрупп из  $G$ . Тогда мы говорим, что  $A$  является  $\Sigma$ -вложенной в  $G$ , если  $A$  либо покрывает, либо изолирует каждую такую максимальную пару  $(K, H)$ , что  $G_{i-1} \leq K < H \leq G_i$ , для некоторого  $i$ .

**Теорема 1.** Если подгруппа  $E$  группы  $G$  является  $\{C_p \leq G\}$ -вложенной в  $G$  для всякой подгруппы простого порядка  $C_p$  из  $G$ , то  $E$  субнормальна в  $G$ .

Теорема 2. Подгруппа  $E$  группы  $G$  субнормальна в  $G$ , если  $E$  является  $\{E^x \leq G\}$ -вложенной в  $G$  для всякого элемента  $x$  из  $G$ .

Теорема 3. Если каждая силовская подгруппа группы  $G$  является  $\{H \leq G\}$ -вложенной в  $G$ , то подгруппа  $H$  субнормальна в  $G$ , причем каждый такой композиционный фактор  $T/L$  группы  $G$ , что  $H \leq L$ , является абелевым.

## ОБЪЕКТНЫЙ ПОДХОД C++ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ОПРЕДЕЛЕНИЯ ОСАДОК ПЛИТНОГО ФУНДАМЕНТА

*В. Ю. Коноплев (УО «ГГУ им. Ф. Скорины»)*

*Научн. рук. Л. А. Цурганова,*

*канд. физ.-мат. наук, доцент*

Механизмы C++ помогают использовать значительно меньшее количество памяти и ускорить работу программы. При передаче больших объемов между средствами расчета и отображения не создаются лишние копии данных, а просто предоставляется ограниченный доступ к уже выделенной памяти.

Задача определения осадок плитного фундамента в самой общей постановке аналитического решения не имеет и может быть решена только численно на основе метода конечных элементов [1]. Компьютерное моделирование осадок плитного фундамента включает два основных аспекта: математическое моделирование определения осадок плитного фундамента и разработку автоматизированного рабочего места, представляющего удобный интерфейс ввода-вывода, быстрый переход от одного варианта задачи к другому. Для реализации объектного подхода C++ созданы следующие классы: **Rect** описывает физические характеристики узловых точек (такие как координаты, нагрузка, номер, тип) и методы для управления этими значениями; **FinalElement** описывает выбранный тип конечных элементов – треугольник, вершинами которого являются указатели на объекты класса **Rect** и для конечного элемента определены характеристики грунта модуль упругости  $E$  и коэффициент Пуассона  $\mu$ ; **RigidityMatrix** описывает матрицу жесткости, хранит наборы узловых точек и конечных элементов, позиционирует указатели и производит основные расчеты; **Slau** содержит различные методы для решения систем линейных уравнений.

Пользователь приложения на этапе выполнения может изменять соотношение размеров представления данных и результатов. Заполнение исходных данных максимально упрощенно – необходимо лишь выделить мышкой нужный элемент и задать для него характеристики. Элемент, для которого были указаны характеристики, будет визуально отличаться от остальных. Поиск номера выделенного элемента ведется быстрым алгоритмом определения положения точки относительно прямых. Каждый элемент умеет отрисовывать себя на указанном визуальном элементе, что позволяет быстро менять способ представления результатов.

### Литература

1 Быховцев, В. Е. Компьютерное объектно-ориентированное моделирование нелинейных систем деформируемых твердых тел : монография / В. Е. Быховцев. – Гомель : ГГУ им. Ф. Скорины, 2007. – 219 с.

## РАЗРАБОТКА МУЛЬТИМЕДИЙНОГО УЧЕБНОГО РЕСУРСА ПО ОСНОВАМ РАБОТЫ В ОПЕРАЦИОННОЙ СИСТЕМЕ LINUX

*Н. В. Коноплич (УО «ГГУ им. Ф. Скорины»)*

*Научн. рук. А. В. Варуев,*

*канд. техн. наук, доцент*

Формирование единого информационного пространства невозможно без внедрения информационных технологий в процесс обучения, развития средств обучения сферы