

Лекции по МВС

Рубаненко Евгений

2017

1 Лекция 11.10.17

1.1 Системы приоритетов задач пользователей

Проблемы:

1. Могут быть срочные задачи относительно "фона"
2. "Переключение" на более срочную задачу (поддержка контрольных точек)
 - (a) Если задача выполнялась, а затем ее убили, то ее прогресс полностью теряется. Контрольные точки позволяют сохранять какой-то объем данных
 - (b) Иногда выставляются сами, но обычно выставляет сам программист
3. Роли участников в выставлении приоритетов
4. Некорректное использование многопроцессорной системы
 - (a) Правильное реагирование на "тяжелые задачи"

1.2 "Рыночные" алгоритмы планирования

Каждому выдаются "фишки", а затем происходит аукцион. Почти как обычный, только списывается не вся сумма, а минимальная достаточная.

1.3 Интерконнект в кластере

Характеристики:

1. Латентность
 - (a) Время между получением и отправкой
2. Скорость передачи (пропускная способность, $\frac{byte}{s}$)
3. Темп выдачи сообщений
4. Устойчивость
 - (a) Время работы от случая к случаю меняется незначительно

2 Лекция 18.10.17

2.1 Способы соединения узлов кластера

Топология соединений узлов кластера

1. Составные части
 - (a) Вычислительная сущность
 - (b) Switch
 - (c) Свойства switch и вычислительной сущности
2. Способы
 - (a) Звезда
 - (b) Линия
 - (c) Кольцо
 - (d) Решетка
 - (e) Тор
 - (f) Гиперкуб
 - (g) Деревья

2.2 Что влияет на латентность

1. Максимальный транзит (диаметр графа)
2. Надежность (минимальное количество ребер, которое нужно удалить, чтобы граф стал несвязным)
3. Стоимость (отношение числа ребер к числу вершин)
4. Степень вершины
5. Ширина бисекции (минимальное число ребер, которое нужно удалить, чтобы граф разбился на два несвязных между собой множества, с одинаковым числом вершин)
6. Масштабируемость (насколько легко добавить узел // число ребер, которое нужно добавить, чтобы сохранить свойства топологии при добавлении одного узла)
7. Можно еще считать различные средние величины, но однозначного не ясно, будет ли эта информация иметь какую-либо ценность

2.3 Когда хорошо применять какую-либо из топологий

1. Линия - небольшой кластер; конвейерная организация вычислений (например, какая-то работа военных - поиск чего-то в непрерывном потоке данных)
2. Кольцо - передача данных по оптоволокну (если надо передать что-то на далекое расстояние, то временные затраты будут не такими большими)
3. Решетка - очень популярная модель (например, расчет прочности моста или обтекаемости какого-то предмета)

4. Гиперкуб - когда надо максимизировать транзит
5. Дерево - когда интересует сбор информации в одном приемнике

2.4 Задача

Посчитать максимальный транзит в решетке из N узлов.

Решение: Так как $N = n \cdot n$, то необходимо $2n - 2$ перехода. Тогда в общем виде ответ есть $2\sqrt{N} - 2$.

2.5 Маршрутизация

Бывают направленные решетки - из одного узла можно переходить только в какие-то конкретные узлы.

1. Статическая
2. Динамическая

2.6 InfinyBand

Важное свойство - перед тем как что-либо добавить в сеть, происходит исследование канала, которым соединены два узла. Затем для каждого узла добавляется такая величина, как фишки пропускной способности. Затем устройство добавляется в сеть. Subnet manager выдает этому устройству GID и перенастраивает таблицы маршрутизации этой сети. После этого устройство появляется в сети.

2.7 Сети Петри