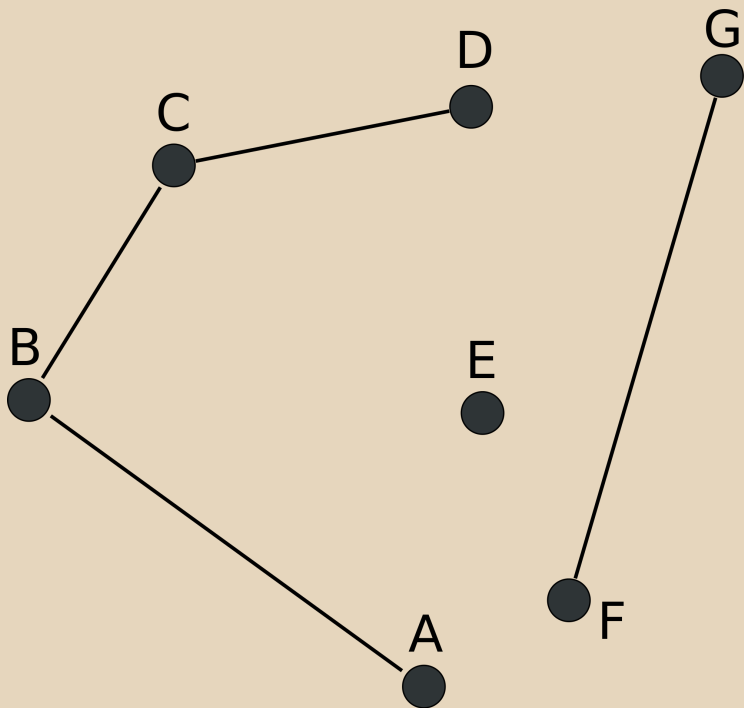


Поиск компонент связности графа на мультипроцессоре

студент: Комаров С.О.
науч. рук.: Калгин К.В.

Компонента связности



- **Вершина w** графа $G(V, E)$ **достижима** из вершины v , если либо $v = w$, либо существует путь из v в w .
- Граф называется **связным**, если любая его вершина достижима из любой дугой.
- **Компонентой связности** графа называется его связный подграф, не являющийся собственным подграфом никакого другого связного подграфа.

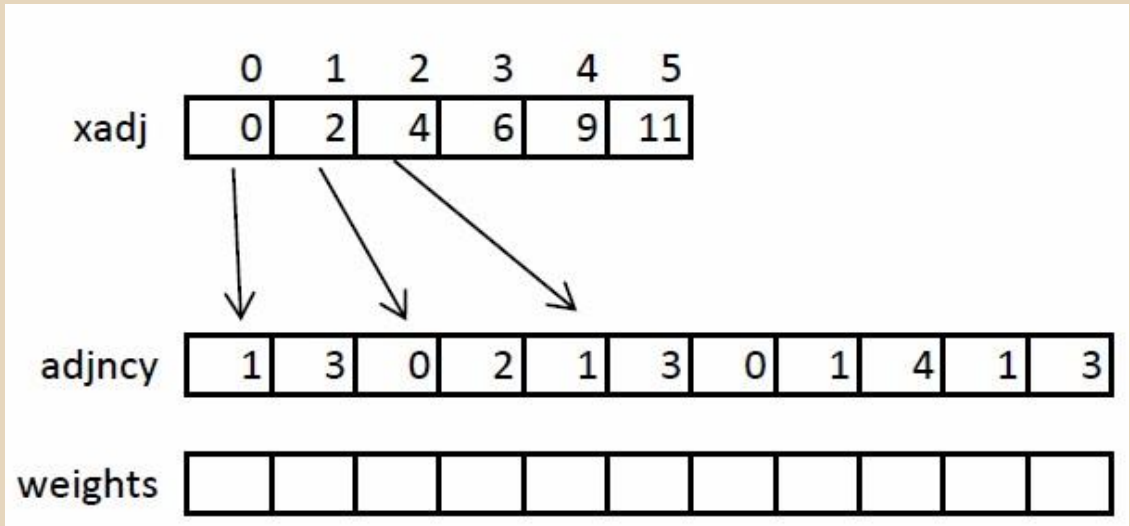
Представление графа

Граф $G(V, E)$ задается в сжатом CSR формате.

$||xadj|| = N + 1$ [N - число вершин]

$||adjncy|| \ \& \ ||weights|| = 2 * M$ [M - число ребер (для неориентированных нужно 2 дуги)]

Принцип: весь список соседей вершины i находится в массиве `adjncy` с индекса `xadj[i]` до `xadj[i+1]`, не включая его. Аналогично хранятся веса



Обход в ширину

Пока все вершины графа $G(V, E)$ не покрашены, выполнять:

- 1) Выбрать не покрашенную вершину, покрасить и поместить ее в изначально пустую очередь Q .
- 2) Пока очередь Q не пуста:
 - a) Извлечь из очереди Q вершину v
 - b) Добавить в очередь Q и покрасить все смежные вершины v , которые ещё не покрашены

Сложность: $O(N)$

Обход в глубину

Пока все вершины графа $G(V, E)$ не покрашены, выполнять:

- 1) Выбрать не покрашенную вершину, покрасить и поместить ее в изначально пустой стек S
- 2) Пока стек S не пуст:
 - a) Извлечь из стека S вершину v
 - b) Добавить в стек S и покрасить первую смежную вершину, которая ещё не покрашена (если таковые имеются)

Сложность: $O(N)$

Алгоритм Shiloach-Vishkin(SV)

Массив $D[1..n]$, где $D[i]$ - номер компоненты, к которой принадлежит “ i ”.
Изначально $D[i] = i$.

В бесконечном цикле:

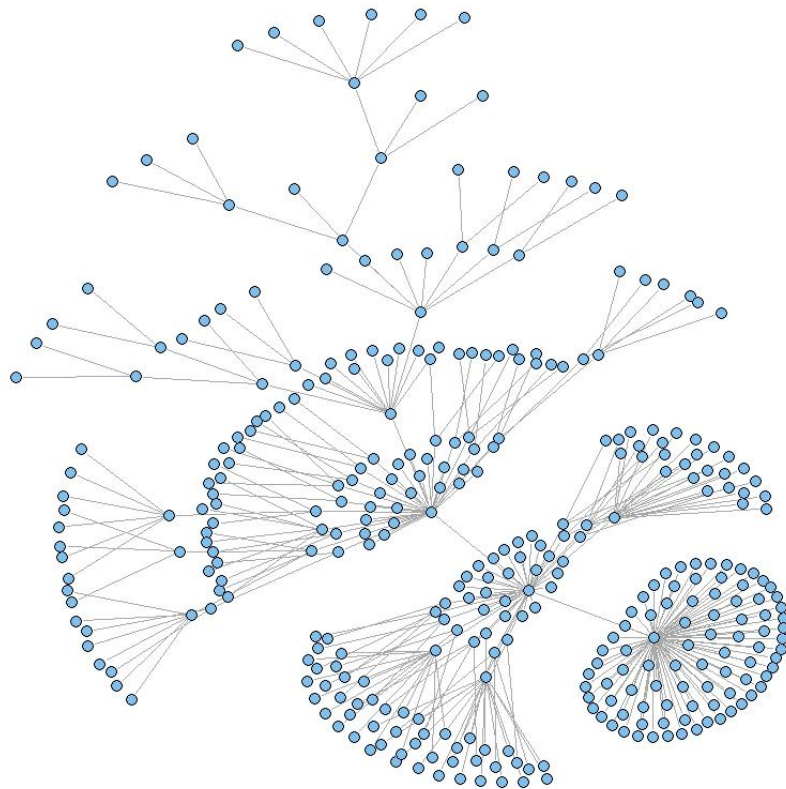
- 1) Пробегаем по всем ребрам $(i - j)$
если $((D[i] < D[j]) \ \&\& \ (D[j] == D[D[j]]))$, то $D[D[j]] = D[i]$;
- 2) Пробегаем по всем вершинам
 - а) пока $(D[i] != D[D[i]])$
 делать $D[i] = D[D[i]]$;
- 3) Если мы ни разу не обновляли D , то КОНЕЦ.

Сложность: $O(E \cdot \log(V) + V^2)$ [худший случай]

Примеры графов

RMAT-n

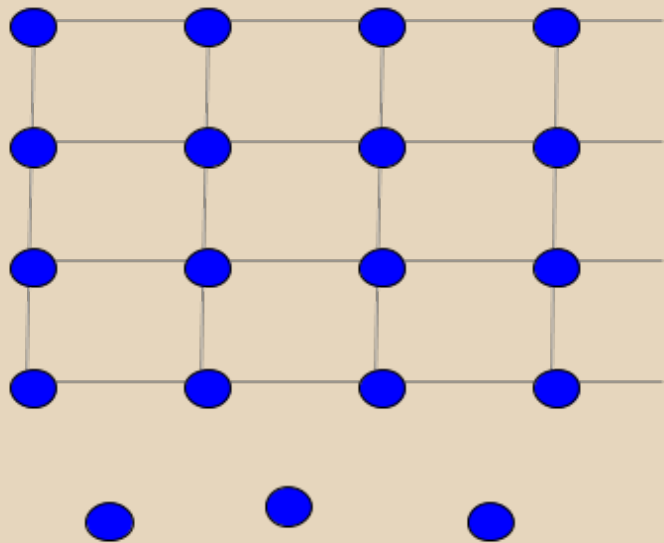
моделирует социальный граф (по степеням вершин) с 2^n вершинами



Примеры графов

Сеточный граф

набор графов-сеток + одиночные вершины



Последовательный вариант

Core i5-2410M (2.30GHz)

2 cpu cores + Hyper-threading

	время	кол-во компонент	кол-во вершин	кол-во ребер	
Grid	обход в ширину	30.43	110000	176990000	702240000
	Shiloach-Vishkin	181.47			

	время	кол-во компонент	кол-во вершин	кол-во ребер	
RMAT	обход в ширину	1.768	57796	4194304	134217728
	Shiloach-Vishkin	4.014			

Алгоритм Shiloach-Vishkin

- + хорошо масштабируется
- + хорошо параллелится
 - а. простые вставки openMP
 - б. отсутствие зависимостей
- часто повторяет действия (переделяет работу)

Свой асинхронный алгоритм

Пул цветов $P[1..n]$, где $P[i]$ - номер компоненты, к которой принадлежит “ i ”.
Изначально $P[i] = i$.

- 1) стандартный обход в ширину
каждый поток красит свой подграф
- 2) редукция таблицы эквивалентности цветов
- 3) раскраска (простой обход)

Сложность: $O(V + V^2)$ [худший случай]

Параллельный вариант

Core i5-2410M (2.30GHz)

2 cpu cores + Hyper-threading

RMAT

	время	потоки	кол-во компонент	кол-во вершин	кол-во ребер
Shiloach-Vishkin	4.014	1	57796	4194304	134217728
асинхронный	1.859	1			
Shiloach-Vishkin	2.371	2			
асинхронный	1.071	2			
Shiloach-Vishkin	2.127	4			
асинхронный	1.079	4			

Параллельный вариант

Core i5-2410M (2.30GHz)

2 cpu cores + Hyper-threading

Grid

	время	потоки	кол-во компонент	кол-во вершин	кол-во ребер
Shiloach-Vishkin	181.47	1	110000	176990000	702240000
асинхронный	42.25	1			
Shiloach-Vishkin	111.03	2			
асинхронный	23.19	2			
Shiloach-Vishkin	103.38	4			
асинхронный	16.86	4			

Планы

- добавятся новые виды графов
- тестирование на многосокетных системах
- обзор малоизвестных параллельных алгоритмов