A parallel pebble game with applications to expression evaluation

Lecture 11

Parallel pebble game on binary tree

- Within the game each node $v$ of the tree has associated with it similar node denoted by $\text{cond}(v)$.
- At the outset of the game $\text{cond}(v)=v$, for all $v$.
- During the game the pairs $(v, \text{cond}(v))$ can be thought of as additional edges.
- Node $v$ is “active” if and only if $\text{cond}(v) \neq v$.

Pebbling

- **Pebbling a node** denotes the fact that in the current state of the game the processor associated with that node has sufficient information to evaluate the subtree rooted here.

Three operations active, square and pebble

- **Activate**
  ```
  for all non-leaf nodes $v$ in parallel do
  if $v$ is not active and precisely one of its sons is pebbled then
    $\text{cond}(v)$ becomes the other son
  if $v$ is not active and both sons are pebbled then
    $\text{cond}(v)$ becomes one of the sons arbitrarily.
  ```

- **Square**
  ```
  for all nodes $v$ in parallel do
  $\text{cond}(v) \leftarrow \text{cond}(\text{cond}(v))$
  ```

- **Pebble**
  ```
  for all nodes $v$ in parallel do
  if $\text{cond}(v)$ is pebbled then
    pebble $v$
  ```

Key result

- At the outset of the game only the leaves of the tree are pebbled.
- One composite move of the pebbling game is the sequence of individual operations
  
  (activate, square, square, pebble)

Theorem

Let $T$ be a binary tree with $n$ leaves. If initially only the leaves are pebbled then after $\log_2 n$ moves of the pebbling game the root of $T$ becomes pebbled.

Example

activate

Example
The application of the pebbling game

- Consider the arithmetic expression 
\[((3+(2*2))*3+5)\]
- We assign a processor to each non-leaf node of the tree.

Evaluation of arithmetic expressions

Arithmetic expressions can be evaluated on a PRAM in $O(\log n)$ time using $O(n)$ processors.