Implementation of Process Networks in Java

Arnab Basu

&

H.P. Vijay Kishen
Process Networks

Concurrent computation model, excellent for use in Computation Intensive Real time applications in Signal and Image processing.

Kahn PN -

- Represented as Directed graphs. Nodes (processes) are connected by a set of Arcs (FIFO queues). Processes’ are seen as Mapping functions of data from the I/P to the O/P queue.
- Blocking reads. Shown to be determinate, run in infinite memory.
- Deadlock occurs when all nodes are blocked on read.
- Model proves that the deadlock occurs irrespective of schedule chosen.
Dataflow PN -

• Leads to bounded memory execution eventually, if one exists.
• Additional constraints of queues with token limit and blocking writes.
• Artificial Deadlocks resolved by increasing the capacities of the queues.

Computation Graphs -

• Have a threshold level($T_p$) associated with each queue which decide when the associated sink function can execute.
• Due to the restrictions placed on the model, it is proved that the graphs terminate and that queue lengths are bounded.
• It is also proved that irrespective of the sequence of firings of the nodes the output tokens that are produced remain the same.
Java

Features

- Object oriented, Strongly typed.
- Support for threads, Exception Handling and Runtime checking.
- Garbage collection.
- Embedded Java™ allows API’ to be configured as needed.
- Tools to provide efficient translation of java into embedded code.