CONCURRENT COMPUTATION : A FRAMEWORK

MODELLING CONCURRENCY BY NONDETERMINISM AND FAIRNESS

- Concurrency is all about “concurrent execution of a system of process.”
- It is modelled by nondeterministic arrangement of atomic actions of the individual process.
- Semantics of a concurrent program is given by computation tree.

Contd…….

- How Fairness is related to concurrent computation ?
- Concurrency = Nondeterminism + Fairness.

ABSTRACT MODEL OF CONCURRENT COMPUTATION

Abstract concurrent program is a triple (M, $Ø_{start,}$ Φ) , where

- M is a temporal structure
- $Ø_{start}$ is an atomic proposition corresponding to a distinguished set of starting states in M
- Φ is a fair scheduling constraint

Contd.....

- IMPARTIALLY : Iff every process is executed infinitely often during computation.
- WEAK FAIRNESS :Iff every process is enabled almost every where is executed infinitely often.
- STRONG FAIRNESS :Iff every process enabled infinitely often is executed infinitely often.

CONCRETE MODELS OF CONCURRENT COMPUTATION

Can be obtained by refining abstract models of concurrent computation by refining it in various ways

- Providing structure for the global state space
- Defining instructions which each process can execute to manipulate the state space, and
- Providing concrete domains for each global state space.

CONCRETE MODELS OF PARLLEL COMPUTATION BASED ON SHARED VARIABLES

It can be refined further by

- By imposing appropriate restrictions on the way instructions can access and manipulate the data.
- By imposing restrictions on which process are allowed which kind of access to which variables.
- By specifying domain to the variables.

CONCRETE MODELS OF PARLLEL COMUTATION BASED ON MESSAGE PASSING

The communication primitives are

- B;e!α :send the value of expression e along α, provided that guard predicate B is enabled and there is corresponding receive command ready.
- B;v?α :receive a value along channel α and store it in variable v, provided that guard predicate B is enabled and there is corresponding send command ready.

CONNECTING THE CONCURRENT COMPUTATION FRAMEWORK WITH TEMPORAL LOGIC

- In the linear time framework :$(M, \O_{start,}\ \Phi)|= p$ iff $\forall x$ in M such that $M, x |= \O_{start}$ and $M, x |=\Phi$ and $M, x |= p$
- In the branching time framework :$(M, \O_{start,}\ \Phi)|= p$ iff $\forall s$ in M such that $M, s |= \O_{start}$ we have $M, s |= p_{\Phi,}$ where p_{Φ} is the branching time formula obtained from p by relativizing all path quantification to scheduling constraint Φ.

QUESTIONS ?

He discussed only some of the very common fair constraints, Can we find out any other?