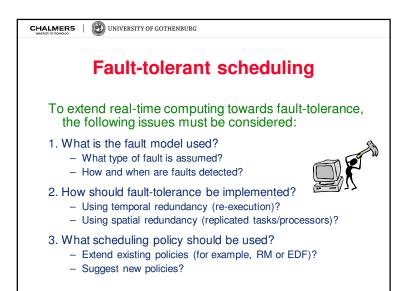


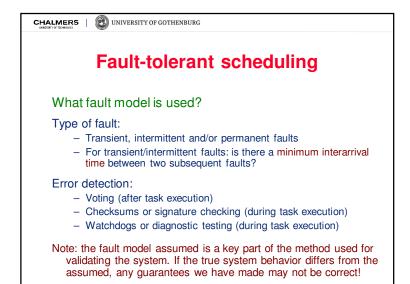


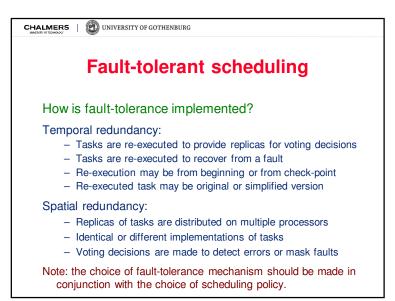
Fault-tolerant techniques

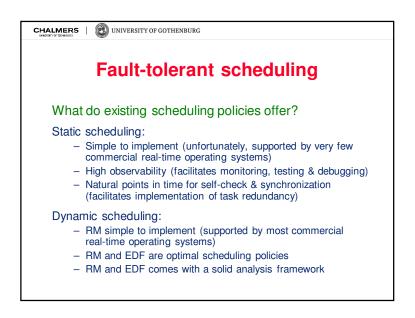
Information redundancy (forward error recovery):

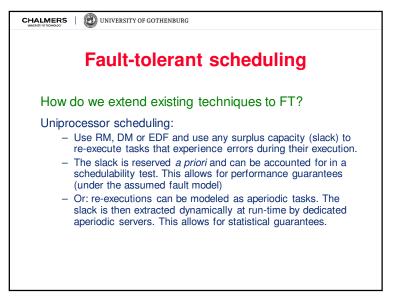
- Duplication:
 - Errors are detected by duplicating each data word
- Parity encoding:
 - Errors are detected/corrected by keeping the number of ones in the data word odd or even
- Checksum codes:
 - Errors are detected by adding the data words into sums
- Cyclic codes:
 - Errors are detected/corrected by interpreting the data bits as coefficients in a polynomial and deriving redundant bits through division of a generator polynomial

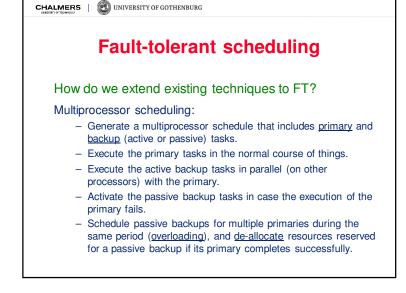


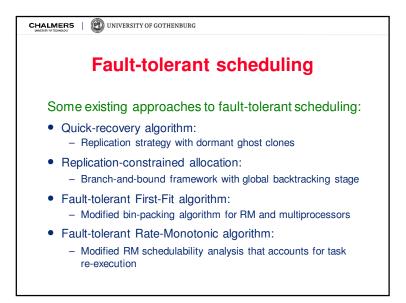


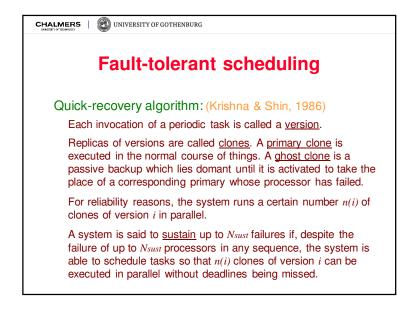


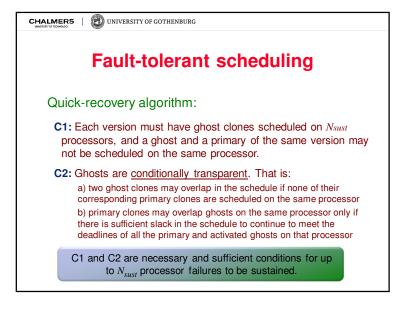


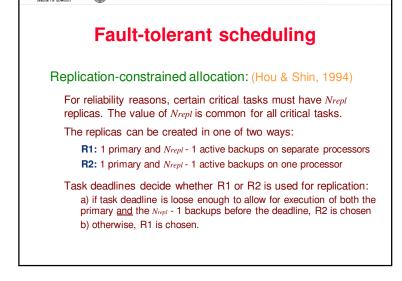






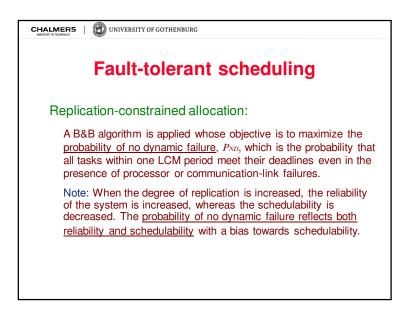


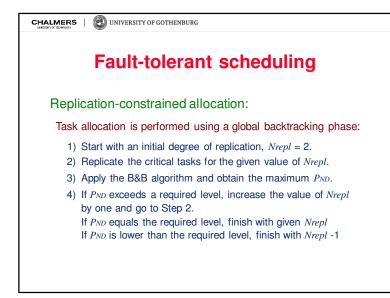


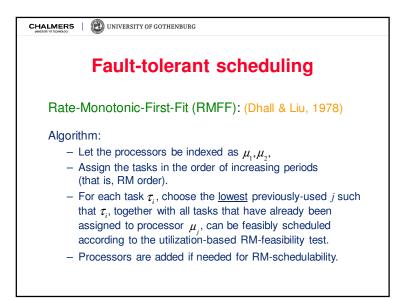


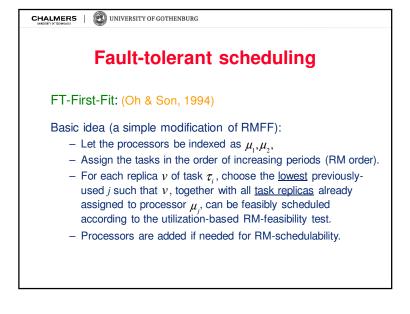
UNIVERSITY OF GOTHENBURG

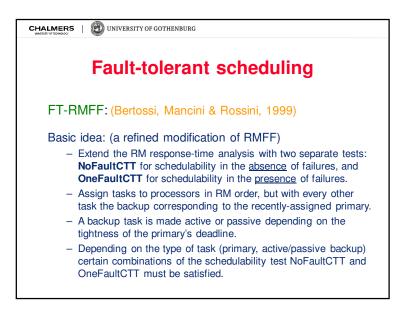
CHALMERS











UNIVERSITY OF GOTHENBURG

CHALMERS

CHALMERS | UNIVERSITY OF GOTHENBURG

Fault-tolerant scheduling

A set of *n* tasks scheduled according to the RM policy always meet their deadlines if

$$U = \sum_{i=1}^{n} \frac{C_i}{T_i} \le U_{LL} = n(2^{1/n} - 1)$$

(Liu & Layland, 1973)

Note: a lower bound can be derived by letting $n \to \infty$.

$$\lim_{n \to \infty} n \left(2^{1/n} - 1 \right) = \ln 2 \approx 0.693$$

Consequence: a task set whose utilization does not exceed ≈ 70% is always schedulable.

Foult tolorent cohoduling

Fault-tolerant scheduling

A set of n tasks scheduled according to the RM policy always meet their deadlines <u>even in the presence of a single fault</u> (using same-priority re-execution) if

$$U \le U_{PM} = 0.5$$

(Pandya & Malek, 1998)

Note: this bound is less pessimistic than the trivial bound:

$$\lim_{n \to \infty} n \left(2^{1/n} - 1 \right) / 2 \approx 0.346$$

CHALMERS UNIVERSITY OF GOTHENBURG

FT-RMA: an example of caution

FT-RMA: (X, Y & Z, 1997)

Make sure there is enough slack in the RM schedule to allow for the re-execution of any task instance if a fault occurs during its execution.

The added slack is distributed throughout the schedule such that the amount of slack available over an interval of time is proportional to the length of that interval.

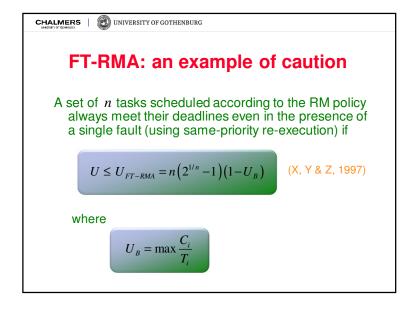
The ratio of slack available over an interval of time is constant and can be regarded as the utilization U_B of a backup task B.

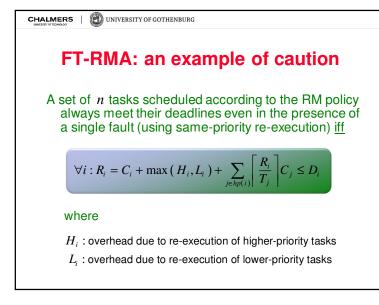
FT-RMA: an example of caution

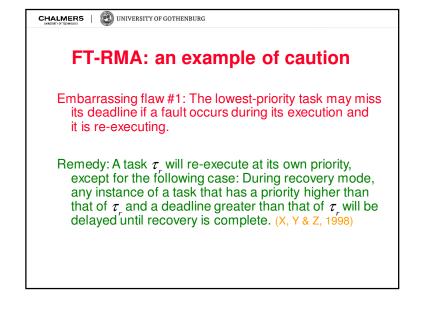
FT-RMA:

A <u>recovery scheme</u> ensures that the slack reserved in the schedule can be used for re-executing a task before its deadline, without causing other tasks to miss their deadlines.

When an error is detected at the end of the execution of some task τ_k , the system enters <u>recovery mode</u>. In this mode, τ_k will execute at its own priority.







9

