

PARALLELISM

- $T = \left(\frac{N}{P}\right) / F$
- $T_1 = \sum_{i=1}^{\text{nodes}} (\text{work-node}_i)$
- $T_\infty = \sum_i \text{criticalpath}(\text{work-node}_i)$
- $\text{PARALLELISM} = T_1 / T_\infty$
- $\text{speedUp} = T_1 / T_P = \frac{T_1}{(1-P)T_1 + (P T_1 / P)} = \frac{1}{1-P+P/P}$
- $\text{EFFICIENCY} = S_P / P$
- $T_P = \frac{T_1}{P} + T_{\text{overhead}}(P)$
- $T_P \geq \frac{T_1}{P} \wedge T_P \geq T_\infty$
- $T_1 = T_{\text{seq}} + T_{\text{PAR}}$
- $\eta = T_{\text{PAR}} / T_1 = T_{\text{PAR}} / (T_{\text{seq}} + T_{\text{PAR}})$
- $S_\infty = \frac{1}{1-\eta}$
- 2^{22} lines, 2 bits/linia, 2^{23} bits/cache $\Rightarrow 2^{23} / 8 / 2^{20} = 1 \text{MB}$

CONCURRENCIA: DIVISIÓ DEL PROG EN PARTS DISCRETES QUE PUGUIN SER EXECUTABLES PER UN THREAD.

PARALLELISME: EMPRAR MÚLTIPLES PROCESSADORS PER OBTENIR UNA REDUCCIÓ DEL TEXE (PROGRAMA CONCURRENT).

THROUGHPUT: EXECUCIÓ SIMULTÀNEA DE MÚLTIPLES PROGRAMES MITJANÇANT DIVERSOS PROCESSADORS.

PROBLEMES:

DATARACE: MÚLTIPLES TASQUES MODIFIQUEN SIMULTÀNEAMENT UNA MATEIXA VARIABLE (VALOR INCORRECTE).

DEADLOCK: DUES O MÉS TASQUES ESPEREN A QUE UNA DE LES ALTRES REALITZI UNA DETERMINADA ACCIÓ PER A PROSEGUIR.

STARVATION: INCAPACITAT D'UNA TASCA PER A CONTINUAR L'EXECUCIÓ A CAUSA D'UNA DADA RETINGUDA PER UN ALTRE THREAD.

LIVELOCK: CANVIS D'ESTAT CÍCLICS D'UN CONJUNT DONAT DE TASQUES QUE EVITA EL PROGRÉS.

THREAD, SISTEMA OPERATIU, PROCESSADOR, HARDWARE: GENERALMENT TENEN UNA RELACIÓ 4 A 1.

TASK DEPENDENCE GRAPH: GRAF DIRIGIT I ACÍCLIC QUE REPRESENTA LES RELACIONS EXISTENTS ENTRE LES DISTINTES TASQUES QUE FORMEN UN PROGRAMA; ELS PESOS D'AQUESTES.

CRITICAL PATH: CAMÍ EL VALOR RESULTANT DE LA SUMA DELS PESOS DELS NODES ÉS MÉS ELEVAT (T_∞).

GRANULARITAT: TAMANY COMPUTACIONAL DE LES TASQUES.

COARSE-GRAIN DECOMPOSITION: MENOR PARALLELISME i MENOR OVERHEAD.

FINE-GRAIN DECOMPOSITION: MAJOR PARALLELISME i MAJOR OVERHEAD.

FONTS D'OVERHEAD:

DATA SHARING: COMPARTICIÓ DE LES DADES.

IDLENESS: EL THREAD NO TÉ TASQUES A EXECUTAR (DEPENDÈNCIES, LOAD IMBALANCE).

COMPUTATION: COST AFEGIT PER PODER EXECUTAR EN PARALLEL.

MEMORY: MEMÒRIA EXTRA USADA PER OBTENIR UN ALGORISME EN PARALLEL.

CONTENTION: COMPETÈNCIA PER PODER OBTENIR ELS RECURSOS.

EMBARRASSINGLY PARALLEL DECOMPOSITION: DESCOMPOSICIÓ D'UN PROGRAMA COMPLETAMENT PARALLELITZABLE.

LOAD UNBALANCE/IMBALANCE: DIFERÈNCIA EN LA QUANTITAT D'INSTRUCCIONS A EXECUTAR PER LES DIFERENTS TASQUES.

STRONG SCALING: AUGMENT DEL NOMBRE DE PROCESSADORS i MIDA CONSTANT DEL PROBLEMA $\uparrow P \downarrow \text{TEXE}$.

WEAK SCALING: INCREMENT PROPORCIONAL DEL NOMBRE DE PROCESSADORS i LA MIDA DEL PROBLEMA $\uparrow P = \text{TEXE}$.

COHÈNCIA EN MEMÒRIES CENTRALITZADES i COMPARTIDES:

WRITE UPDATE: COMPARTeix LES ESCRITURES.

WRITE INVALIDATE: INVALIDA LES CÒPIES.

SNOOPING: S'EMPRÀ UN BUS COMÚ.

DIRECTORY: CADA BLOC té ASSIGNAT UN PUNT D'ORDENACIÓ.

MSI: UNA LÍNIA A CACHE POT ESTAR EN TRES ESTATS:

MODIFIED (M): CÒPIA BRUTA.

SHARED (S): CÒPIA NETA.

INVALID (I): CÒPIA ERRÒNIA / INEXISTENT.

TÉ DOS ESDEVENIMENTS DE CPU:

PrRd: PROCESSOR READ.

PrWr: PROCESSOR WRITE.

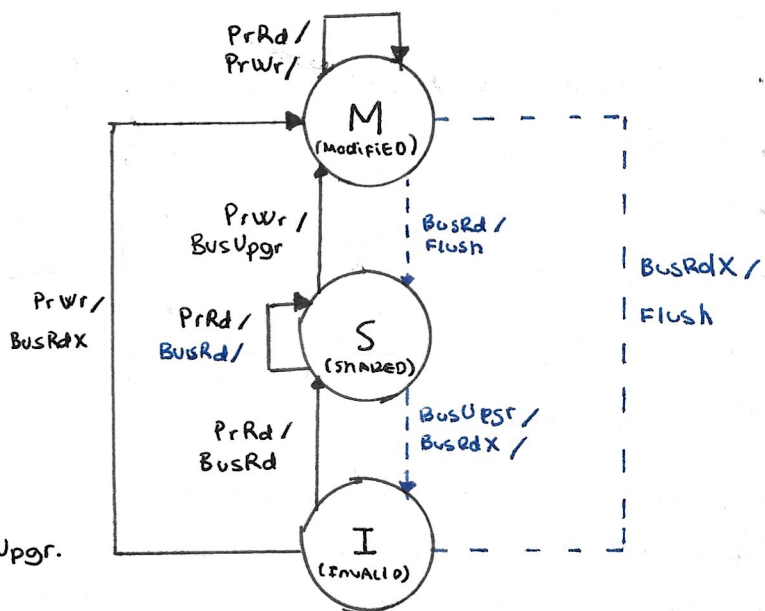
I QUATRE ESDEVENIMENTS DE BUS:

BusRd: READ.

BusRdX: READ & WRITE.

BusUpgr: INVALIDA LA DESTA DE CÒPIES.

Flush: LA LÍNIA ES ENVIADA PEL BUS.



MBI: EXCLUSIVE, EL CANVI D'E A M NO REQUEREIX BusUpgr.

MULTICORE: CHIPS AMB DIVERSOS PROCESSADORS.

SOCKET: PROCESSADOR MULTICORE CONNECTAT A MEMÒRIA.

UPI/QPI: PORTS EMPRATS PER INTERCONNECTAR SOCKETS PER GARANTIR LA COHERÈNCIA DE LA MEMÒRIA.

MSU: DIRECTORY-BASED CACHE COHERENCY: UNA PART DEL DIRECTORI ASSOCIAT A CADA NODE DE MEMÒRIA, UNA ENTRADA PER CADA LÍNIA DE MEMÒRIA - TÉ TRES ESTATS:

MODIFIED (M): CÒPIA BRUTA.

SHARED (S): UN O MÉS NODES TENEN CÒPIA.

UNCACHED (U): NO HI HA CÒPIES DE LA LÍNIA A MEMÒRIA).

TÉ UNA LLISTA DE SHARERS, QUE CONTROLA AQUELLS NODES QUE TENEN UNA CÒPIA DE LA LÍNIA. S'IMPLEMENTA AMB BIT STRINGS, 1 bit per node. Si 64B bloc: 12.5% overhead (64 nodes), 50% (256 nodes), 200% (4096 nodes).

TRUE SHARING: DATA SHARING IS UNAVIDABLE IN PARALLEL COMPUTING. COHERENCE MECHANISMS ARE THERE TO ALLOW THIS DATA SHARING; SYNCHRONIZATION ALLOWS TO SHARE APPROPRIATELY.

FALSE SHARING: CACHE LINE MAY ALSO INTRODUCE ARTEFACTS: MORE THAN 1 DISTINCT DATA OBJECT, OR ALSO MULTIPLE ELEMENTS OF THE SAME OBJECT MAY RESIDE IN THE SAME LINE. OCCURS WHEN DIFFERENT PROCESSORS MAKE REFERENCES (rd, wr) TO DIFFERENT OBJECTS WITHIN THE SAME LINE.