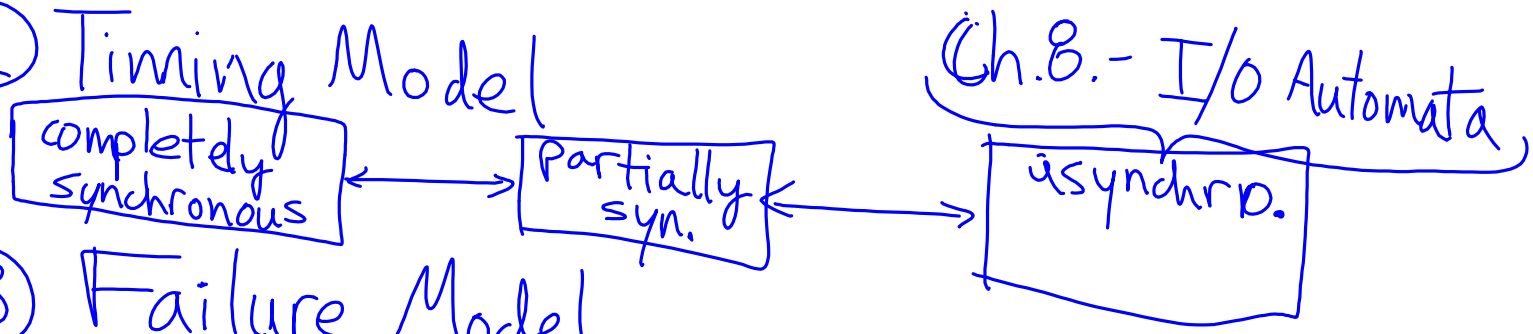
LPE Reading Group Meeting
26 April 2001

dhcp-228.cds.caltech.edu

Attributes of Distr. Algor.

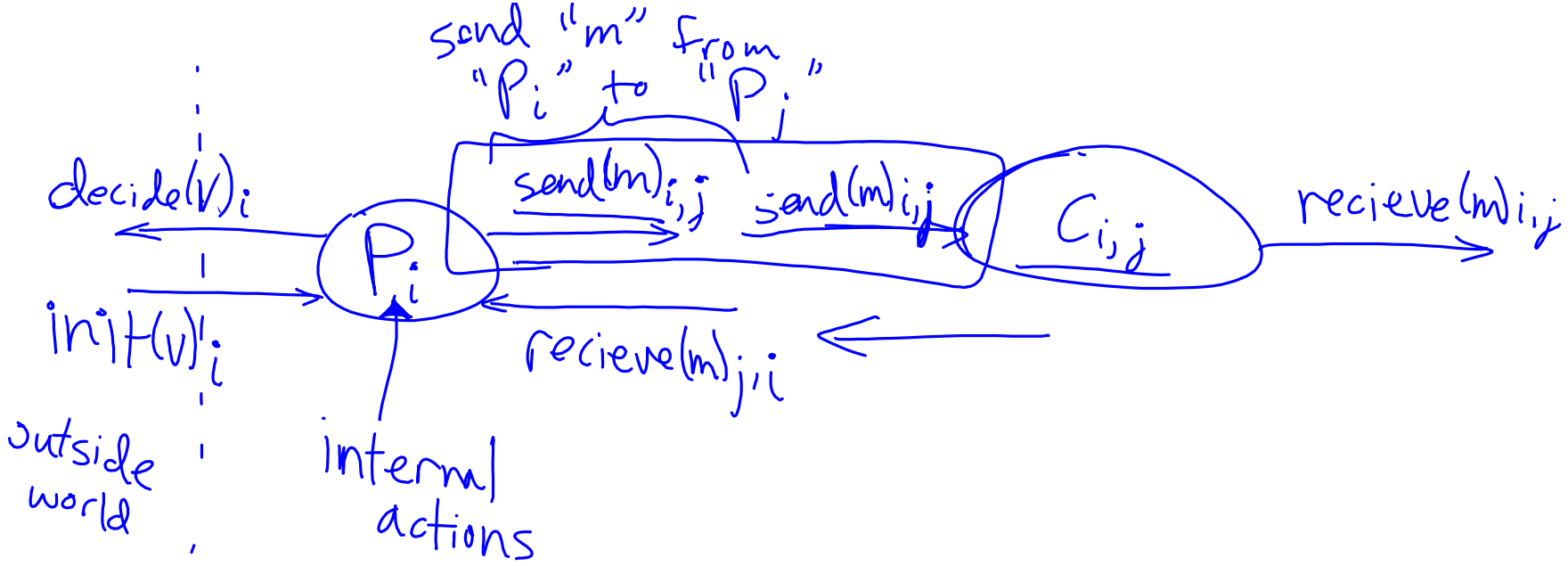
1. IPC
2. Timing Model



3. Failure Model
4. Problems Addressed

I/O Automata

- General – gives precise way of reasoning about system components
- Fairness – each process gets infinite chances on an infinite time scale


send "m" from "P_i" to "P_j"
$send(m)_{i,j}$
$send(m)_{i,j}$
$C_{i,j}$
$recieve(m)_{i,j}$
$decide(v)_i$
P_i
$init(v)_i$
$recieve(m)_{j,i}$
outside world
internal actions

I/O Autom. – 5 components (A)

① sig(A) – S = (in(A), out(A), int(A))

② states(A)

③ start(A) (req. ≠ ∅)

④ trans(A) ⊆ states(A) × act(sig(A)) × states(A)

⑤ tasks(A); equivalence relation on local(sig(A)) (countably many)

local(A)

(s, π, s')

① aids in error ID.

② helps theory

input-enabled

Example - channel t/o $C_{i,j}$

M - fixed message alphabet

① sig $(C_{i,j})$:

Input: $send(m)_{i,j}$, $m \in M$

Output: $receive(m)_{i,j}$, $m \in M$

②,③ states $(C_{i,j})$:

FIFO queue (elements of M), initially empty

④ trans $(C_{i,j})$

1 trans.

$send(m)_{i,j}$

Effect:

add "m" to queue

$receive(m)_{i,j}$

Precondition: m is 1st on queue

Effect: remove m

⑤ tasks $(C_{i,j})$

$\{receive(m)_{i,j} : m \in M\}$

Example P_i (n processes), $f : V^n \to V$ V: fixed value set and $\phi \notin V$

from outside world → in_i

decide ← sent to world

from → receive local $P_j, j \neq i$

send ← sent to particular $P_j, j \neq i$