

# ABSTRACT MACHINES

In essence, The small-step semantics we have already seen.

Introduced via. various models of evaluation for MinTTS:

- ① M-machine: basic SSS  $\Rightarrow$  inefficient, requires recursive descent.
- ② C-machine: M-machine + stacks  $\Rightarrow$   $O(1)$  state transitions  
 $\Rightarrow$  Function application eval'd by substitution, still inefficient.
- ③ E-machine: M-machine + environments and closures  
 $\Rightarrow$  Roughly assignment 1, but also take 7.

Stacks: Store partially evaluated expressions to avoid recursion, as recursion  $\cong$  stacks roughly. Consists of frames:

(Plus  $\square$   $e_2$ ), (Plus  $v_1$   $\square$ ) etc.  
 $\uparrow$  1st operand is being evaluated  
 $\uparrow$  evaluated 1st operand, now evaluating the 2nd

A stack is represented as a tower of frames:

$f_1 \triangleleft f_2 \triangleleft f_3 \triangleleft \dots \triangleleft \circ$   $\swarrow$  Empty stack.

A judgment in The C-machine now looks like

$s \triangleright \textcircled{e}$   
evaluation  
mode

or

$s \triangleleft \textcircled{v}$   
return mode

Ex. Language:  $\{P, A, PP, AP, PPAP\}$

$$\frac{}{P \Downarrow P} H_1, \quad \frac{}{A \Downarrow A} H_2, \quad \frac{a \Downarrow P \quad b \Downarrow A}{a+b \Downarrow AP} U_1, \quad \frac{a \Downarrow P \quad b \Downarrow P}{a+b \Downarrow PP} U_2, \quad \frac{a \Downarrow AP \quad b \Downarrow PP}{a+b \Downarrow PPAP} U_3$$

Small-step semantics with stacks to eliminate recursion:

Stacks:  $(\square + b)$  or  $(v + \square)$

Transitions:

- ①  $s \succ P \mapsto s \prec P$
- ②  $s \succ A \mapsto s \prec A$
- ③  $s \succ (a+b) \mapsto (\square + b) \triangleright s \succ a$
- ④  $(\square + b) \triangleright s \prec P \mapsto (P + \square) \triangleright s \succ b$
- ⑤  $(\square + b) \triangleright s \prec AP \mapsto (AP + \square) \triangleright s \succ b$
- ⑥  $(P + \square) \triangleright s \prec A \mapsto s \prec AP$
- ⑦  $(P + \square) \triangleright s \prec P \mapsto s \prec PP$
- ⑧  $(AP + \square) \triangleright s \prec PP \mapsto s \prec PPAP$