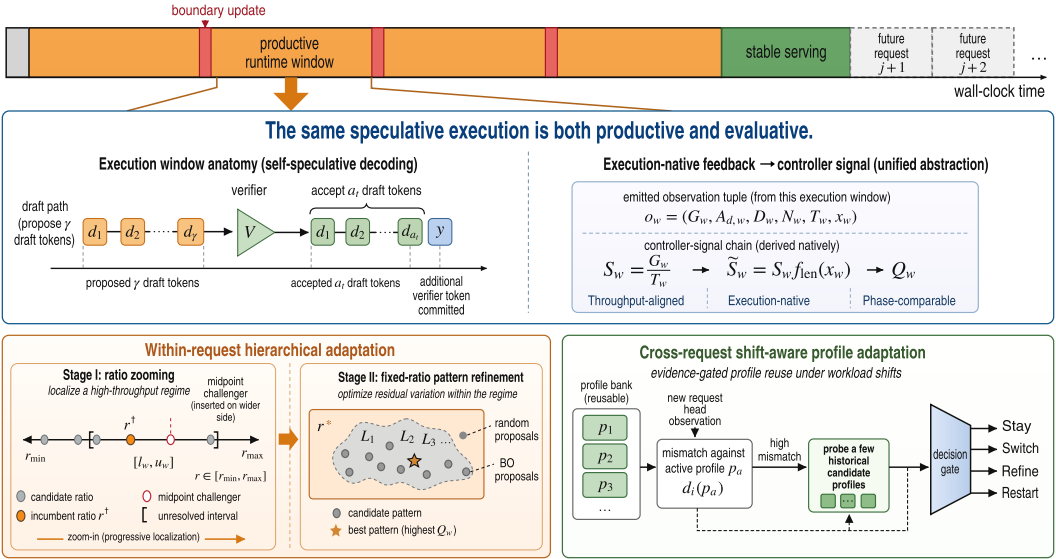

boundary update
productive runtime window
stable serving
future request $j+1$
future request $j+2$
...
wall-clock time
The same speculative execution is both productive and evaluative.
Execution window anatomy (self-speculative decoding)
draft path (propose γ draft tokens)
d_1 d_2 ... d_γ
verifier
V
accept a_t draft tokens
d_1 d_2 ... d_{a_t} y
proposed γ draft tokens
accepted a_t draft tokens
additional verifier token committed
Execution-native feedback → controller signal (unified abstraction)
emitted observation tuple (from this execution window)
$o_w = (G_w, A_{d,w}, D_w, N_w, T_w, x_w)$
controller-signal chain (derived natively)
$S_w = \frac{G_w}{T_w}$ → $\widetilde{S}_w = S_w f_{\text{len}}(x_w)$ → Q_w
Throughput-aligned
Execution-native
Phase-comparable
Within-request hierarchical adaptation
Stage I: ratio zooming
localize a high-throughput regime
midpoint challenger (inserted on wider side)
$r^\dagger$
$r_{\min}$
$r_{\max}$
$[l_w, u_w]$
$r \in [r_{\min}, r_{\max}]$
candidate ratio
midpoint challenger
incumbent ratio $r^\dagger$
unresolved interval
zoom-in (progressive localization)
Stage II: fixed-ratio pattern refinement
optimize residual variation within the regime
r^*
L_1 L_2 L_3 ...
random proposals
BO proposals
candidate pattern
best pattern (highest Q_w)
Cross-request shift-aware profile adaptation
evidence-gated profile reuse under workload shifts
profile bank (reusable)
p_1
p_2
p_3
...
new request head observation
mismatch against active profile p_a
$d_i(p_a)$
high mismatch
probe a few historical candidate profiles
decision gate
Stay
Switch
Refine
Restart