# **Computational core (constraints & goals — the what/why)**

1. **Stay near-critical (Ψ-band) while performing**

max⁡π E ⁣[∑tγtrt]s.t.ΩΨ(π)⏟time in Ψ-band ≥ θ.\max\_{\pi}\ \mathbb{E}\!\Big[\textstyle\sum\_t \gamma^t r\_t\Big] \quad\text{s.t.}\quad \underbrace{\Omega\_{\Psi}(\pi)}\_{\text{time in Ψ-band}}\ \ge \ \theta .

1. **Non-zero free-energy principle (don’t collapse uncertainty)** Maintain **target, non-zero** predictive free energy FF (or entropy/uncertainty proxy) to preserve exploration capacity:

Fmin⁡ ≤ E[Ft] ≤ Fmax⁡withFmin⁡>0.F\_{\min}\ \le\ \mathbb{E}[F\_t]\ \le\ F\_{\max}\quad\text{with}\quad F\_{\min}>0 .

1. **Risk discipline under heavy tails** Bound downside when shocks are fat-tailed:

ESq(ΔF) ≤ c,\mathrm{ES}\_q(\Delta F)\ \le\ c,

where ESq\mathrm{ES}\_q is Expected Shortfall (CVaR) of performance residuals ΔFt\Delta F\_t.

1. **Competence growth without destabilisation**

E[Δηt] ≥ α >0andPr⁡(Ψ-exit) ≤ ε.\mathbb{E}[\Delta \eta\_t]\ \ge\ \alpha \ > 0 \quad\text{and}\quad \Pr(\text{Ψ-exit})\ \le\ \varepsilon .

1. **Precision/effort budget** Keep control effort bounded (a soft “conservation” of precision/energy):

∑k∈{b,T,λ} ⁣ ⁣ ⁣E ⁣[Δkt2] ≤ B.\sum\_{k\in \{b,T,\lambda\}}\!\!\!\mathbb{E}\!\Big[\Delta k\_t^2\Big]\ \le\ B .

A compact Lagrangian you can optimise online:

L(π)=E ⁣[∑tγtrt]−λΨ[θ−ΩΨ]+−λES[ESq(ΔF)−c]+−λF Φ(Fmin⁡,Fmax⁡)+λη(α−E[Δηt])+,\mathcal{L}(\pi)=\mathbb{E}\!\Big[\textstyle\sum\_t\gamma^t r\_t\Big] -\lambda\_\Psi[\theta-\Omega\_\Psi]\_+ -\lambda\_{\mathrm{ES}}[\mathrm{ES}\_q(\Delta F)-c]\_+ -\lambda\_F\,\Phi(F\_{\min},F\_{\max}) +\lambda\_\eta(\alpha-\mathbb{E}[\Delta \eta\_t])\_+ ,

with Φ\Phi any barrier that penalises FF outside [Fmin⁡,Fmax⁡][F\_{\min},F\_{\max}].

# **Algorithmic/representational core (state spaces & update rules — the how in principle)**

Minimal state: st=(ϕt,Ft∗,bt,Tt,λt,ηt,χt)s\_t=(\phi\_t,F^\*\_t,b\_t,T\_t,\lambda\_t,\eta\_t,\chi\_t).  
 Order parameter(s) for Ψ: e.g., synchrony/metastability mtm\_t and band indicator 1Ψ,t\mathbf{1}\_{\Psi,t}.

1. **Policy with Ψ-sieve + tail penalty**

πt(a∣st) ∝ exp⁡ ⁣(Qt(a)−λr CVaRq[Lt(a)]Tt)  1{a∈AΨ(st)},\pi\_t(a\mid s\_t)\ \propto\ \exp\!\Big(\tfrac{Q\_t(a)-\lambda\_r\,\mathrm{CVaR}\_q[L\_t(a)]}{T\_t}\Big)\; \mathbf{1}\{a\in\mathcal{A}\_\Psi(s\_t)\},

where AΨ={a:Pr⁡(Ψ-exit∣st,a)≤ε}\mathcal{A}\_\Psi=\{a:\Pr(\text{Ψ-exit}\mid s\_t,a)\le \varepsilon\}.  
 (Use QQ or −GEFE-G\_{\rm EFE}; λr\lambda\_r rises only on tail alarms.)

1. **Difficulty set-point servo (keep at the branch)**

Ft+1∗=Ft∗+κF (Et−Ft∗)−ρF ∂Φ/∂F∗(project to keep F∈[Fmin⁡,Fmax⁡]).F^\*\_{t+1}=F^\*\_t+\kappa\_F\,(\mathcal{E}\_t-F^\*\_t)-\rho\_F\,\partial\Phi/\partial F^\*\quad \text{(project to keep }F\in[F\_{\min},F\_{\max}]).

1. **Temperature / uncertainty controller**

Tt+1=Tt+κT (χt−χ\\*)−ρT Tt,T\_{t+1}=T\_t+\kappa\_T\,(\chi\_t-\chi^\\*)-\rho\_T\,T\_t,

with caps Tmin⁡≤Tt≤Tmax⁡T\_{\min}\le T\_t\le T\_{\max}; χ\\*\chi^\\* is your target meta-uncertainty.

1. **Stability–flexibility bias**

bt+1=bt−κb (mt−m\\*)−ρb bt,b\_{t+1}=b\_t-\kappa\_b\,(m\_t-m^\\*)-\rho\_b\,b\_t,

pulling toward a metastability set-point m\\*m^\\* (avoid lock-in or fragmentation).

1. **Representation/map update with bounded step (non-zero F)**

ϕt+1=arg⁡min⁡ϕ{ F(ϕ;st)⏟variational free energy  +  β DKL(pϕt ∥ pϕ)⏟trust region},\phi\_{t+1} =\arg\min\_{\phi}\Big\{\,\underbrace{F(\phi;s\_t)}\_{\text{variational free energy}}\; +\;\beta\,\underbrace{D\_{\mathrm{KL}}\big(p\_{\phi\_t}\,\|\,p\_{\phi}\big)}\_{\text{trust region}}\Big\},

so each learning step reduces FF **a bit** but never to zero (trust-region keeps you in-band).

1. **Mode arbitration (creative vs control)**

mt={Creativeif χt>χhi and ∣ΔF^t∣<δ,Controlotherwise,ΔF^t=κ(Et−Ft∗).m\_t=\begin{cases} \text{Creative} & \text{if } \chi\_t>\chi\_{\rm hi}\ \text{and}\ |\widehat{\Delta F}\_t|<\delta,\\ \text{Control} & \text{otherwise}, \end{cases} \quad \widehat{\Delta F}\_t=\kappa(\mathcal{E}\_t-F^\*\_t).

1. **Tail-alarm reflex (k-step schedule)** If ζ<ζ\\*\zeta<\zeta^\\* or J>J\\*J>J^\\* or ESq>c\mathrm{ES}\_q>c: for next kk steps,

T ⁣↑,λ ⁣↓,b ⁣→bmid,T\!\uparrow,\quad \lambda\!\downarrow,\quad b\!\to b\_{\rm mid},

then decay back via #3–#4 when alarms clear.

five computational constraints (what/why) + seven lean update rules (how-in-principle). They’re modular: you can run 1–3 + 6–7 for a tiny agent, or plug all of them into a single online Lagrangian optimiser.