

Starting point for home pc setup:

- **limit\_rollout\_depth = 6** soon (reduces value-net thrashing)
  - **sims\_per\_move = 300** is fine
  - **c\_puct = 2.5** is good for weak nets
  - **temperature = 1.0** early training
  - **dirichlet\_alpha = 0.3, eps = 0.25** are solid defaults
  - **learning\_rate = 1e-3 → 5e-4** after ~200 games
  - **ReplayBuffer = 10k–30k** positions is ideal
  - **epsilon schedule** only matters if you use random exploration
- 



## MCTS Parameters

Parameter	What it Controls	Increase → Effect	Decrease → Effect
<b>limit_rollout_depth</b>	How many plies MCTS explores before using the value net	More stable Q, less value-net noise, slower	More cutoff hits, faster, but unstable early training
<b>_focused_legal</b>	Restricts moves to a radius around last moves	Smaller search space, faster, but may miss tactics	Larger search space, slower, more exploration
<b>sims_per_move</b>	Number of MCTS simulations per move	Stronger play, slower training	Weaker play, faster training
<b>c_puct</b>	Exploration vs. exploitation balance	More exploration, wider search	More exploitation, narrower search
<b>temperature</b>	Softens the final policy distribution	More randomness, more diverse training data	More deterministic, less exploration
<b>dirichlet_alpha</b>	Strength of Dirichlet noise added to root priors	More exploration in early moves	Less exploration, more deterministic openings
<b>dirichlet_eps</b>	How much Dirichlet noise mixes with policy priors	More randomness in openings	More stable but less diverse openings

---



## Neural Network Training Parameters

Parameter	What it Controls	Increase → Effect	Decrease → Effect
<b>temperature (NN output)</b>	Softness of policy head during training	More exploration in self-play	More deterministic play
<b>learning_rate</b>	Step size for weight updates	Faster learning but unstable	Slower learning but stable
<b>gamma</b>	Discount factor for value targets	Longer-term planning	Short-term focus
<b>weight_decay</b>	L2 regularization strength	Less overfitting, but underfits if too high	More overfitting risk
<b>ReplayBuffer capacity</b>	How many past games	More diverse training	Faster forgetting, less

Parameter	What it Controls	Increase → Effect	Decrease → Effect
	are stored	data	stability
<b>epsilon</b>	Starting probability of random move (if used)	More exploration early	Less exploration early
<b>epsilon_end</b>	Final epsilon value	More randomness late	More deterministic late
<b>epsilon_decay_steps</b>	How fast epsilon decays	Longer exploration phase	Faster convergence to deterministic play
<b>epsilon_step</b>	Step size for epsilon decay	Faster or slower decay depending on value	Opposite effect

## 💡 Diagnostic Checklist for $\pi$ Entropy, Q Stability, and Cutoff Rates

### 🎯 1. $\pi$ (policy) Entropy — What It Means

Entropy tells you how *spread out* the final MCTS policy is.

#### Healthy Ranges

Entropy Interpretation	What It Means
<b>0.0 – 1.0</b> Very low	MCTS found a <i>forced</i> or <i>clearly superior</i> move
<b>1.0 – 3.0</b> Moderate	MCTS has a few good candidates
<b>3.0 – 5.0</b> High	MCTS is uncertain, exploring widely
<b>&gt;5.0</b> Very high	Nets are weak, priors flat, MCTS unfocused

#### Red Flags

- **Entropy = 0.0 repeatedly**  
→ MCTS collapsing to one move too often (pruning too strong or value net thrashing)
- **Entropy > 4.5 early in the game**  
→ Nets are clueless, MCTS exploring too widely

#### What to Adjust

- Too low entropy → reduce pruning or increase rollout depth
- Too high entropy → increase sims\_per\_move or radius, or improve nets

### 🎯 2. Q Stability — What It Means

Q is the average value estimate at the root after MCTS.

#### Healthy Behavior

Pattern	Interpretation
Q drifts slowly over moves	Nets are learning stable evaluations
Q stays near 0 early	Balanced game, no tactical swings
Q spikes only in tactical positions	MCTS found a real advantage

## Red Flags

- **Q flips sign every move**
  - Value net unstable + shallow rollouts  
(very common in games 1–100)
- **Q jumps from +0.8 to -0.7**
  - Value net noise dominating MCTS
- **Q stays near ±1.0 for many moves**
  - Value net overconfident or miscalibrated

## What to Adjust

- Q oscillates wildly → increase rollout depth
- Q too confident → reduce learning rate or add weight decay
- Q too flat → increase sims\_per\_move or reduce pruning

---

## 🎯 3. Cutoff Rate — What It Means

Cutoff hits = how often MCTS stops early and uses the value net.

### Healthy Ranges

Cutoff %	Interpretation
0–10%	MCTS doing most of the work (safe early)
10–25%	Balanced (ideal for early–mid training)
25–40%	Value net is being trusted a lot
>40%	Too many cutoffs → unstable Q, noisy training

## Red Flags

- **Cutoff > 50%**
  - Value net dominating search too early
  - Increase rollout depth immediately
- **Cutoff < 5% for many games**
  - Value net not being trained enough
  - Increase sims\_per\_move or reduce rollout depth slightly

## What to Adjust

- Cutoff too high → increase rollout depth
- Cutoff too low → reduce rollout depth or increase sims\_per\_move



## 4. Combined Interpretation (Most Useful Part)

### Case A — Low entropy + Q stable + low cutoff

→ MCTS is confident and value net is stable

**This is ideal mid-training behavior.**

### Case B — Low entropy + Q unstable + high cutoff

→ MCTS collapsing because value net is noisy

**Increase rollout depth.**

### Case C — High entropy + Q unstable + moderate cutoff

→ Nets are weak, MCTS exploring widely

**Normal in first 50–200 games.**

### Case D — High entropy + Q stable + low cutoff

→ Nets are starting to guide MCTS

**This is the transition to stronger play.**

### Case E — Alternating low/high entropy every move

→ Player perspective flip + unstable value net

**Expected early; stabilizes with deeper rollouts.**



## 5. Quick Actions Based on What You See

Symptom	Fix
$\pi$ entropy = 0.0 too often	Reduce pruning or increase rollout depth
Q flips sign every move	Increase rollout depth
Cutoff > 40%	Increase rollout depth or reduce sims_per_move
$\pi$ entropy > 4.5 early	Increase sims_per_move or radius
Q stuck near $\pm 1.0$	Reduce learning rate
$\pi$ entropy moderate but Q unstable	Increase rollout depth
$\pi$ entropy high but Q stable	Nets improving — keep training