# Open Face Chinese Poker Solver: One-Page Overview

> **AI experiment disclosure:** The entire project - implementation, experiments, analysis, and this writeup - was produced by AI under user direction. It was a quick experiment using Codex Goal mode and 5.6 Sol with only a few prompting examples to test the system's capabilities. The work has not been independently peer reviewed; results should be treated as experimental rather than authoritative poker strategy.

## Project

This repository implements a complete solver and research environment for two-player classic Open Face Chinese Poker (OFCP). Each player places five opening cards and then eight single cards into a 3-card front, 5-card middle, and 5-card back. A completed board fouls unless `back >= middle >= front`. The engine implements row scoring, the three-point scoop bonus, and standard royalties.

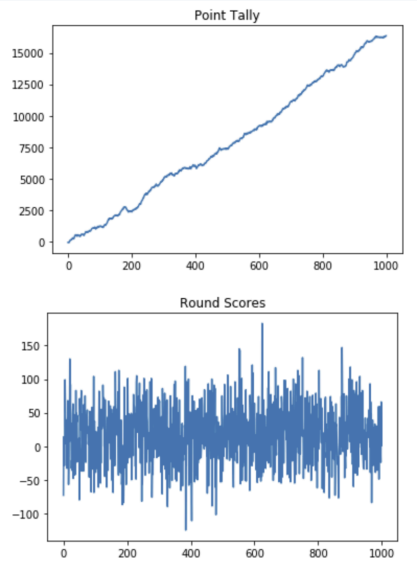
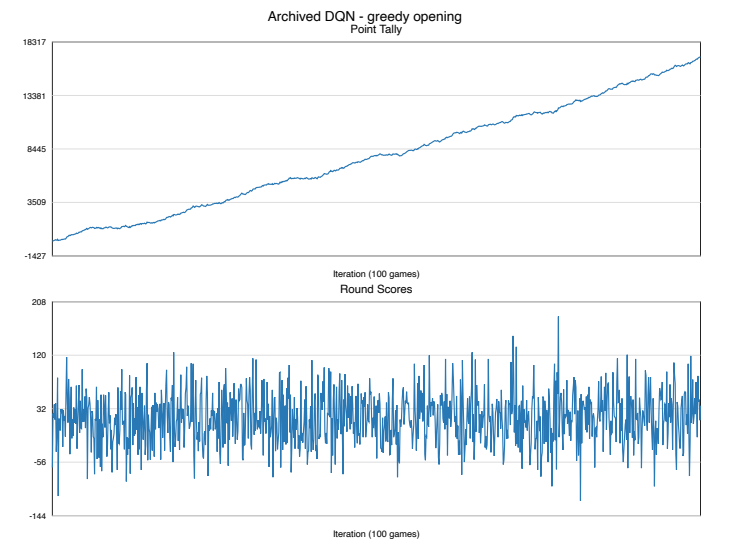
The starting point was Tan and Xiao's 2018 DQN study.[1] Their deleted implementation was recovered from public Git object history, ported, tested, and separated into two modes: `archived` preserves observable source behavior, including historical defects, while `paper` implements the corrected nine-decision algorithm described by the equations. A final profile adds a Bellman-consistent multi-sample target.

Four neural networks were trained for 100,000 simulated games each against a random opponent. Every network sees both boards plus the current card and predicts the value of placing that card in the front, middle, or back. The corrected profiles sample four possible hidden-card futures ( `M=4` ).

## Replication

Figure 4 in the paper reports a highly variable naive run ending at -3,301 and an updated greedy run ending visually near +16,500 after 100,000 games. The exact naive numeric trace was recovered. The updated numeric trace and complete initialization seeds were not published, so exact point-for-point replay is impossible from the surviving artifacts.

| Figure 4 target | Paper | Local source-compatible run | Result |
|---|---:|---:|---|
| Naive final training tally | -3,301 | +3,238 | Volatility reproduced; exact stochastic path not reproduced |
| Greedy final training tally | approximately +16,500 | +16,955 | Endpoint within about 2.8%; similar rising curve |

| Paper Figure 4 updated model | Local archived-greedy run |
| --- | --- |
|  |  |





The corrected DQN reached +0.412 points per hand against random play; the improved target reached +0.477. These are held-out, 10,000-game, balanced-seat evaluations. The improved network nevertheless lost to the deterministic heuristic at -1.885 points per hand and fouled 66.1% of hands, so it should be viewed as a paper-replication baseline rather than the best player.

## Best Result

The strongest tested policy is a separate hidden-card rollout search. At each decision it evaluates eight promising placements across eight independently sampled unseen-card futures, without reading the real future deck. In 1,000 balanced hands against the heuristic it scored **+1.534 points per hand**, with a **95% confidence interval of +1.164 to +1.904**, while fouling **24.5%** of hands versus the heuristic's 43.5%. This search policy is the default for recommendations.

The repository includes the game engine, exact 13-card arrangement, heuristic and search solvers, four model checkpoints, training plots, raw hand-level results, confidence intervals, and automated tests. The full findings and limitations are in `REPORT.md`.

---

1. Andrew Tan and Jarry Xiao, "Mastering Open-face Chinese Poker by Self-play Reinforcement Learning," September 2018, authors' PDF. ↩