Reinforcement learning for heliostat aiming: Improving the performance of Solar Tower plants

J.A. Carballo a,b, J. Bonillaa,b, N.C. Cruzc, J. Fernández-Rechea, J.D. Álvarez b, A. Avila-Marina ,
M. Berenguel b,
*
a CIEMAT - Plataforma Solar de Almería (PSA), Ctra. de Senés km 4.5, E-04200 Tabernas, Almería, Spain
b CIESOL, Solar Energy Research Centre, Joint Institute University of Almería - CIEMAT, Department of Informatics, E-04120, Almería, Spain
c Department of Computer Engineering, Automatics, and Robotics. University of Granada, Granada, Spain

A new approach to heliostat aiming strategies.
While aiming all heliostats at the receiver's center is an option, it risks flux peaks that damage the receiver or shorten its lifespan. Conversely, aiming at the receiver's edges causes energy spillage. Effective control must balance energy efficiency and operational safety while adapting to disturbances like solar variation and optical errors.
Contributions

A real-time, optimized aiming strategy using the Soft Actor–Critic (SAC) Reinforcement Learning (RL) algorithm.

Continuous optimal aiming without predefined points.

Dynamic heliostat field optimization based on prior state and solar position.

Yearly evaluation and strategy comparisons using different reward functions.

Discussion on applicability and future developments.

Literature Review

Existing strategies range from simple fixed-point aiming to advanced AI and optimization methods. Early approaches used grid-based control with predefined points, adjusting based on sensor feedback. Later methods incorporated meta-heuristics (TABU Search, Genetic Algorithms, Ant Colony Optimization, Particle Swarm Optimization) to optimize flux distribution. Some relied on mathematical solvers for theoretical best configurations.

Recent work integrates AI, particularly reinforcement learning, to bypass traditional optimization, enabling real-time decision-making. A notable study applied RL to select predefined aim points, optimizing thermal output. This research advances that approach by introducing a fully dynamic, AI-driven aiming system for heliostat fields.

3. Methodology

Reinforcement Learning (RL) has improved control

tasks across fields, with Soft Actor-Critic (SAC) excelling in continuous control problems. This work applies SAC to optimize heliostat aiming, leveraging TensorFlow and TF-Agents for flexibility and scalability. The approach considers heliostat field state and solar conditions, with a reward function based on receiver power absorption.

Two fully connected neural networks act as the RL agent's critic and actor. A custom environment, modeled with SolarPilot, interfaces the RL algorithm with the CESA-1 system. The setup includes a reset method with constraints and a step method governing system evolution. The trained agent's performance was evaluated over a year against a traditional five-point aiming strategy.

3.1. Learning

SAC learns through episodes—sequences of agent-environment interactions—updating policies based on rewards. Key metrics include Average Episode Length and Reward Return. Training was conducted on PSA's HELIOSUN workstation (200 CPUs), taking ~9 days. Future improvements will leverage GPU computing for faster training, and transfer learning will refine models efficiently.

3.2. Environment

The RL environment encapsulates the CESA-1 field, modeled using SolarPilot's Python API (CoPylot). The selected 114 heliostats, chosen for optimal receiver coverage, align with existing five-point aiming

strategies. The SolAir3000 receiver (5.72 m^2, 94% absorptivity, 900 kW/m^2 peak flux) was modeled at 86 m height. A fixed 950 W/m^2 DNI facilitated initial training; future work will introduce variable DNI for more complex dynamics.

An analytical flux simulation model, chosen for efficiency, could be replaced with ray-tracing. The TensorFlow-defined environment includes reset and step methods. The reset function initializes conditions and enforces constraints:

Solar elevation < 7°

Spillage losses > 60%

Episode duration > 1000 min

Flux peak > 900 kW/m^2

Violations terminate episodes, training the agent to operate safely. Randomized start conditions improve generalization. The step method applies actions, updates the solar position, and returns the next state, reward, and episode status.

3.3. Action

An action in this context refers to a decision made by the agent, determined by the actor network. This network, implemented as a neural network, produces a probability distribution over the action space. The agent

samples actions from this distribution, applies them to the environment, and receives a reward along with other relevant feedback.

In this study, actions are defined as a vector representing the changes in aiming point coordinates along the horizontal and vertical axes for each heliostat or group of heliostats. Each value in the vector corresponds to an adjustment in these coordinates.

3.4. Reward

In reinforcement learning, the reward is the feedback signal that indicates how favorable an action was in a given state. It plays a crucial role in guiding the agent's learning process. The reward function can be deterministic or stochastic and may consider multiple objectives and constraints relevant to the task.

In this study, three different agents were trained using two different reward functions. The first and second tests used a reward function that combined three components:

The power absorbed by the receiver.

The image intercept loss, which penalizes the loss of reflected power that does not reach the receiver.

The cumulative movement factor, which discourages excessive heliostat movement to reduce energy consumption and wear.

The reward function was designed to balance power absorption, loss reduction, and minimal movement, with weight factors manually tuned for efficiency. In the third test, an additional term guided the solar flux distribution toward a target, optimizing both power absorption and receiver longevity. Since flux distribution affects the receiver's lifespan and maintenance, incorporating expert-defined targets refines the strategy. Future enhancements could include factors like thermal stress for improved learning.

State

In SAC, the state represents key environmental data used to guide decision-making. Here, it includes boundary conditions and the configuration of the CESA-1 heliostat field, such as azimuth, elevation, direct normal irradiance (DNI), and aiming point coordinates. This information updates after each step to inform the agent's next move.
Agent

The agent is the decision-making system in reinforcement learning. It consists of:

An actor network, which determines the probability of different actions based on the current state and reward.

A critic network, which estimates the long-term value of each action.

Both networks were implemented using TF-Agents and designed with multiple fully connected layers. The architecture was manually tuned, increasing layers until achieving satisfactory results. Training was set to 10 million iterations with a replay buffer capacity of 100,000, and learning rates were manually set. Future work aims to refine these parameters and explore more advanced network designs.

Results

Three agents were trained using different reward functions and configurations.

Test 1

The first agent aimed to maximize power absorption, minimize spillage losses, and reduce aiming point movement while adhering to a peak power limit of 900 kW/m^2. To evaluate its performance, the agent's strategy was compared to the existing CESA-1 approach using yearly simulations with the SolAir3000 receiver.

CESA-1 currently groups 114 heliostats into five fixed clusters with predefined aiming points. In this test, the agent was only allowed to adjust these fixed points within the clusters. The heliostats were divided into four groups of 23 and one of 22, following a specific sequential order.

Several agents were initially trained for short periods to identify the most promising candidate. The best agent

completed training in nine days, achieving stable performance after 10 million episodes. While the reward function stabilized, episode length showed variability due to the random initialization of each episode, depending on the time of day and season. The results indicate that further training could refine the agent's performance.

The agent's aiming strategy increased absorbed power, especially midday, and extended plant operation at both ends of the solar day, improving overall efficiency at no additional cost. This enhances economic performance and lowers technology costs.

Test 2

This experiment used the same training parameters and reward function but increased the aiming points from 5 to 10, dividing the heliostats into 10 groups. This adjustment tested whether more aiming flexibility improves system performance. To speed up training, transfer learning was applied, refining the previously trained agent with an additional 10 million episodes.

The reward improved quickly but didn't fully stabilize, suggesting the model hadn't reached its maximum potential despite extensive training. The increased flexibility required more training steps. While the 10-point strategy improved annual absorbed power by 8.7% over the 5-point method, it performed worse in later hours and had abrupt fluctuations. Further optimization of aiming points and training could

enhance performance.

Test 3

This test used the same setup as Test 1 but with a reward function designed to shape a specific solar flux distribution while maximizing absorbed power and minimizing movement. The agent concentrated aiming points along a diagonal to reduce spillage. Training was more challenging, with frequent resets due to excessive losses or concentration limits. The new strategy increased yearly absorbed energy by 3.9%, but it had more failures, premature operation stops, and abrupt power changes. While it successfully shaped the desired flux distribution, further optimization is needed for a more effective strategy.