Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

How far from energy-optimal are learned quadrotor navigation policies? Flight-calibrated energy modelling, optimal-control bounds, and wind- and budget-aware learning

Iqrar Ali

iqrargabol23@gmail.com

Tianjin Sino-German University of Applied Sciences

Sayeed Omar

Tianjin Sino-German University of Applied Sciences

Ma Guoli

Tianjin Sino-German University of Applied Sciences

Research Article

Keywords: Quadrotor, Deep reinforcement learning, Energy-optimal flight, Optimality gap, Energy model

Posted Date: November 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11749401/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

How far from energy-optimal are learned quadrotor navigation policies? Flight-calibrated energy modelling, optimal-control bounds, and wind- and budget-aware learning

Iqrar Ali[1],*, Sayeed Omar[1] , Ma Guoli[1]

[1] Department of Aviation and Aerospace Engineering, Tianjin Sino-German University of Applied Sciences, Tianjin 300350, China

Abstract Energy-aware deep reinforcement learning for multirotor navigation is usually evaluated by comparing one learned policy with another, which establishes an ordering but leaves the quantity a designer needs unmeasured: how much energy is still being left on the table. We supply the missing reference. A blade-element/momentum energy model with a Thevenin battery, embedded in six-degree-of-freedom dynamics with a geometric inner loop, is calibrated on the cruise regime of 188 instrumented DJI Matrice 100 delivery flights at two payloads; it then predicts cruise power at a held-out payload to 2.3% and the power of two flight regimes that played no part in the calibration – climb and descent – to 6.9% and 1.9%, a pooled held-out error of 3.5% against the 2.1% of the empirical regression it is tested on. The optimal mission energy is bracketed between a rigorous lower bound, the minimum energy per metre of the vehicle times the straight-line distance, and an achievable reference that is planned, given an exactly optimal speed profile by dynamic programming, and then flown in the same simulator as the learned policies; across 200 held-out urban-delivery scenarios the reference solves 99% and uses 1.31 times the lower bound. The ratio of a policy's energy to that reference is therefore a conservative optimality gap: a rigorous lower bound on how far the policy is from optimal. Over four seeds per configuration, soft actor-critic with an energy term in the reward attains a gap of 7.3% (95% CI 4.0–12.3) against 14.8% (8.6–25.0) without it, with a large effect size (Hedges' $g = -0.94$) that four seeds cannot certify after multiplicity correction. Decomposing the excess by solving the minimum-energy speed profile along the path each policy actually flew attributes roughly half of it to speed selection rather than routing, and the speed term is the one the energy signal moves, falling from 11.1% to 4.9%; routing is the part a map would remove, speed is not. Energy awareness is not free: the energy reward costs about nine points of success rate at this budget. We also report what a single seed would have concluded – a 2.8% gap against 26.8% – as a caution for a literature that commonly reports one or two. The bounds and the decomposition are cheap and reusable by any study reporting the energy of a learned flight policy.

Keywords Quadrotor · Deep reinforcement learning · Energy-optimal flight · Optimality gap · Energy model

1 Introduction

An electric multirotor spends almost all of its stored energy holding itself up, so how it flies decides how far it goes. A large literature trains deep reinforcement-learning (DRL) policies for mapless obstacle avoidance and reports that adding an energy term to the reward lowers consumption by some percentage relative to a policy trained without it. Such comparisons establish an ordering between learned policies, but they do not answer the question a designer actually has: how much energy is still being left on the table. A 13% saving over an arbitrary baseline means something very different if that baseline was 20% above the achievable optimum than if it was 200% above it.

Answering that question needs three things that are usually missing.

An energy model whose absolute numbers can be trusted. An optimality gap is a ratio of energies and inherits every bias of the model that produced them. Most DRL navigation studies model the vehicle as a point mass and its power as a constant, as proportional to speed, or as proportional to commanded acceleration. None of these reproduces the characteristic shape of multirotor power against airspeed – a decreasing induced term plus an increasing parasite term – and that shape is exactly what determines the energy-optimal way to fly.

A reference that is optimal rather than merely reasonable. A map-based planner such as A* followed by a tracking controller is the usual baseline, but it is designed for feasibility, not for energy, and its own distance from the optimum is unknown. Comparing a learned policy with it measures a difference, not a gap.

Bounds rather than point estimates. The minimum-energy trajectory problem in a cluttered three-dimensional environment is non-convex, and no solver returns a certified global optimum at the scale needed to evaluate hundreds of scenarios. An honest answer must bracket the optimum and state which side of the bracket each claim rests on.

This paper supplies all three for a quadrotor delivery task and uses them to measure how far learned navigation policies are from energy-optimal.

The energy model (Sect. 3) is a blade-element/momentum description of the rotors with an equivalent-flat-plate airframe, a propulsion chain and a Thevenin battery, embedded in six-degree-of-freedom rigid-body dynamics with a geometric inner loop, so that every manoeuvre a policy makes is charged its true aerodynamic and electrical cost. Three aerodynamic coefficients are calibrated against the regression that Rodrigues et al. [1] fitted to 188 instrumented delivery flights of a DJI Matrice 100, using the cruise regime at two of the three payloads only. The model then predicts, with no further fitting, cruise power at the held-out payload and the power of two flight regimes it never saw.

The optimum is bracketed rather than guessed (Sect. 4). A rigorous lower bound follows from two facts: the energy spent per metre of ground track can never fall below the minimum of the steady-flight energy-per-metre curve, and no path is shorter than the straight line. An achievable upper bound is obtained by planning a trajectory, solving its speed profile exactly by dynamic programming, and then flying it in the same simulator as the learned policies, so that what is compared is a measured energy and not a model of one. Because the reference is achievable, the ratio of a policy's energy to it is a *conservative* estimate of that policy's true optimality gap: the real gap can only be larger.

Within that framework we ask four questions. How large is the optimality gap of standard algorithms, and does it differ between them? Where does the gap come from – flying the wrong route, or flying the right route the wrong way? Do policies discover the minimum-energy cruise speed, and do they adapt it to wind as theory requires? And is an energy budget better imposed as a reward penalty or as an explicit constraint?

Contributions.

1. A quadrotor energy model calibrated on, and validated out of sample against, published in-flight measurements, including two flight regimes that played no part in the calibration.
2. A pair of computable bounds on the optimal mission energy and the resulting conservative optimality gap, a metric that is a rigorous lower bound on how far a policy is from optimal.
3. An additive decomposition of that gap into routing and speed-and-manoeuvre terms, obtained by solving the exact minimum-energy speed profile along the path each policy actually flew.
4. A benchmark of SAC, TD3 and PPO with reward and constraint variants measured against those bounds, with implementations verified by finite-difference gradient checks and validated against published reference scores on a standard continuous-control task.
5. An analysis of wind, in which the energy-optimal ground speed falls and the optimal airspeed rises in a headwind, and a measurement of whether policies reproduce that behaviour with and without an observation of the wind.
6. A constrained formulation in which the mission energy budget is derived from the lower bound and enforced as a constrained-MDP constraint, compared with the usual reward-penalty treatment.

The work is simulation-based and uses published measurements for calibration and validation; no new flight tests were conducted.

2 Related work

Multirotor power and energy models. Momentum theory with a profile-power correction is the standard basis for multirotor power prediction. Zeng et al. [2] give a closed-form propulsion-power expression for level flight that is widely used in communication and path-planning work; Gong et al. [3] extend the treatment to three-dimensional flight and validate it against more than seven thousand in-flight power measurements of a DJI M210. Purely empirical regressions are also common: Rodrigues et al. published a 209-flight instrumented data set for a DJI Matrice 100 [4] and fitted an average-power law per flight regime [1], reporting a 2.1% average absolute error on held-out flights. Stolaroff et al. [5] used a comparable model to compare delivery drones with road vehicles. All of these predict power in *steady* flight. Embedding such a model inside a closed-loop simulator, so that the transients a learned policy produces are charged their true cost, is less common, and is what allows the present study to attribute energy to specific behaviours.

Energy-optimal flight and planning. The existence of a speed that minimises energy per unit distance is classical for rotorcraft [6, 7] and follows from the sum of a decreasing induced term and an increasing parasite term. The accompanying wind result – fly faster through a headwind, slower with a tailwind – is standard aircraft-performance theory. Minimum-energy trajectory optimisation with obstacle

constraints is non-convex; the usual remedy is to separate a geometric path from a speed profile along it [8, 9], which is the structure adopted here for the achievable bound, with the speed profile solved exactly by dynamic programming [10].

DRL for UAV navigation. Deep Q-networks [11], proximal policy optimisation [12, 13] and soft actor-critic [14, 15] are the usual algorithms for mapless obstacle avoidance from range or depth input [16–19], and DRL has reached champion level in drone racing [20]. Energy terms in the reward are frequently reported to reduce consumption, and structural energy harvesting has been proposed as a complement [21–23]. Evaluations are often based on few seeds and on baselines whose own efficiency is unknown; Henderson et al. [24] and Agarwal et al. [25] document how fragile such comparisons are and recommend interquartile means with stratified bootstrap intervals, which we adopt.

Constrained reinforcement learning. Treating a resource limit as a constraint rather than a reward penalty is the constrained-MDP formulation of Altman [26]; Lagrangian relaxations with a learned multiplier are standard [27–29]. We set the constraint threshold from the energy lower bound of Sect. 4.2, which makes it directly interpretable as *finish within a stated factor of the theoretical floor*.

Positioning. Our earlier work on this airframe family [22, 23] established an energy account for structural energy harvesting and found that harvest-seeking reward terms do not help, whereas energy-aware flight does. The present paper takes up that conclusion and asks how good energy-aware flight can be, which requires the model fidelity, the bounds and the decomposition developed here.

3 Vehicle, energy model and validation

3.1 Aircraft

The aircraft is a DJI Matrice 100 in the instrumented configuration of the reference data set [4]: 3.68 kg before payload, four rotors of radius 0.1651 m with two blades of 24 mm mean chord (solidity 0.0925), and a 130 Wh six-cell pack. Appendix A lists every parameter (Table A1).

3.2 Rotor aerodynamics

With total thrust T shared by n rotors, the component of the airspeed along the thrust axis $v_\perp$ (positive in climb) and the in-plane component $v_\parallel$, the induced velocity of one rotor follows momentum theory,

$$v_i = \frac{v_h^2}{\sqrt{v_\parallel^2 + (v_\perp + v_i)^2}}, \qquad v_h = \sqrt{\frac{T/n}{2\rho A}} \tag{1}$$

which is the quartic

$$x^4 + 2kx^3 + (k^2 + m^2)x^2 - 1 = 0, \qquad x = v_i/v_h,\ k = v_\perp/v_h,\ m = v_\parallel/v_h \tag{2}$$

In climb and level flight ($k \geq 0$) the quartic has a single positive root, obtained by a safeguarded Newton iteration that agrees with a direct polynomial solve to a relative error of 1×10^{-15} over 200 random states.

In descent the rotor works in its own wake and the quartic has no admissible root; we use the empirical curve of Johnson [7] for $-2 \le k \le 0$, ramped in over $0.3 \le -k \le 0.7$ because the vortex-ring state occupies roughly $0.5 \le -k \le 1.5$ [6], and the windmill-brake branch below $k = -2$. The descent and quartic branches are blended over $0.5 \le m \le 1.5$, since edgewise flow suppresses the vortex-ring state.

Shaft power is summed over rotors,

$$P_{\mathrm{mech}} = \sum_{j} \left[\kappa\, T_j v_{i,j} + T_j v_{\perp} + \delta/8\, \rho s A \left(\Omega_j R\right)^3 \left(1 + 4.65\, \mu_j^2\right) \right] \tag{3}$$

with $\Omega = \sqrt{(T/(C_{\mathrm{T}}\, \rho\, A))}/R$ and $\mu = v_{\parallel}/(\Omega R)$. The airframe is represented by an equivalent flat-plate area S_{fp}; its drag enters the rigid-body equations as a force rather than as a separate power term, so the work done against it is charged exactly once, through the axial term. Electrical power is $P_{\mathrm{mech}}/\eta + P_{\mathrm{avionics}}$, with no regeneration.

3.3 Battery

The pack is a Thevenin equivalent circuit. A load P at the terminals gives $V_{\mathrm{oc}}(\sigma)I - I^2 R_{\mathrm{int}} = P$, so the terminal voltage sags to $V_{\mathrm{oc}} - I\,R\mathrm{int}$ and the chemical drain is $V_{\mathrm{oc}}\, I$, the load plus the ohmic loss. The open-circuit voltage is monotone in state of charge between 19.8 V and 25.2 V.

3.4 Six-degree-of-freedom dynamics

Position, velocity, attitude (as a rotation matrix) and body rate are integrated at 100 Hz. A geometric attitude controller on SE(3) [30] converts a commanded acceleration into a collective thrust and a desired attitude; the resulting wrench is allocated to four rotors through an X-configuration mixer, each rotor's thrust following a first-order lag of 45 ms, with per-rotor thrust limits and a 32° tilt limit. The policy acts every 100 ms, so each control interval contains ten dynamics steps.

Because the realised per-rotor thrusts and the body attitude drive the power model, a policy that oscillates, over-tilts or climbs unnecessarily pays for it. Verification: hover is an equilibrium to 4×10^{-15} m of drift over 400 steps; the hover power of the closed-loop model equals the steady model to 10^{-6} W; rotation matrices stay orthonormal to 2×10^{-16}; the thrust allocation round-trips to 10^{-10}; and the integrated energy matches the power trace to 10^{-9} J (Appendix B, Table B1).

3.5 Calibration and the tests that follow it

Rodrigues et al. [1] regressed the measured average power of each flight regime of 188 instrumented missions on the induced-power group $m^{1.5}/\sqrt{\rho}$, obtaining $b_1 = 80.4 \pm 2.6$, 68.9 ± 2.0 and 71.5 ± 1.7 and $b_0 = 13.8 \pm 18.9$, 16.8 ± 15.0 and -24.3 ± 12.5 W for take-off, cruise and landing, with R^2 of 0.84, 0.85 and 0.90 and an average absolute relative error (ARE) of 2.1% on 68 held-out flights. Their envelope was 4–12 m/s ground speed, 25–100 m altitude, 0/250/500 g payload and 2–16 kn of wind.

Three aerodynamic coefficients of our model – S_{fp}, the blade profile drag coefficient δ and the induced-power factor κ – were fitted by least squares to the cruise law only, at two of the three payloads (3.68 and 4.18 kg) and five speeds, ten points in all. The propulsion efficiency ($\eta = 0.80$) and the avionics power (30 W) were fixed a priori; Sect. 6.8 shows the results are insensitive to both. A tilt-dependent growth of the drag area was also tried and is not supported by the data, so it was removed.

Four tests then use information that played no part in the fit (Table 1):

Table 1 Out-of-sample tests applied to the calibrated energy model

Test	What it checks	Shares data with the fit?
T1	cruise power at the held-out 250 g payload	no
T2	take-off regime, as a steady 2.5 m/s climb	no – different regime
T3	landing regime, as a steady 1.5 m/s descent	no – different regime
T4	manufacturer hover endurance of the stock aircraft at three masses *below* the calibration range	no – different aircraft configuration

Both the induced and the profile terms scale as $m^{1.5}$ at fixed geometry, which is why a single induced-power group with a near-zero intercept describes the measured data so well, and why the published regression needs no speed term.

4 Energy-optimal reference and bounds

4.1 Task

The vehicle flies from a start point to a delivery point across a field of 6–12 cylindrical buildings of radius 5–10 m and height 15–40 m in a 180 × 120 × 58 m volume; straight-line distances are 150–175 m. Half of the buildings are placed near the direct line. The vehicle may fly around them or over them, which is the central energy trade-off of the task: climbing costs energy now, a detour costs energy later. Each scenario is generated deterministically from an integer seed; training, validation and test scenarios come from disjoint seed ranges.

4.2 A rigorous lower bound

At every instant the energy spent per metre of ground track is at least the minimum over admissible ground speeds of the steady-flight energy-per-metre curve, and any path from start to goal is at least as long as the straight line. Hence

$$E^{\star} \;\geq\; \min_{v}\left[\frac{P(v)}{v}\right]\cdot d_{\mathrm{straight}} \;=:\; E_{\mathrm{lb}} \tag{4}$$

In wind the minimum is taken at the along-track wind component. The bound ignores obstacles, acceleration, deceleration, turning and attitude transients, all of which only add energy, so it is conservative by construction. It is also cheap, which matters because it is evaluated for every scenario at training time to set the budget of the constrained agent.

4.3 An achievable upper bound

An upper bound requires a trajectory that can actually be flown. A three-dimensional A* search [31] on a 4 m grid with obstacles inflated by 2.2 m yields a geometric path, shortened by randomised shortcutting and resampled at 4 m. The minimum-energy speed profile along that fixed path is then solved exactly by dynamic programming over a 61-point speed grid: the cost of entering a segment at one speed and leaving at another is the steady power at the mean ground speed – for the airspeed and climb angle implied by that segment and the wind – times the segment time, plus the kinetic energy that must be added. Transitions are restricted by a 3 m/s² tangential acceleration limit, and the profile begins at rest. The DP is a global optimum for that path up to the speed discretisation.

The path and its profile are then tracked by a pure-pursuit controller in the same simulator used for the learned policies, and the measured electrical energy is recorded. Five scalings of the planned profile are flown and the lowest-energy successful one is kept, which tightens the bound. Since the result is an achieved trajectory, $E^\star \le E_{\mathrm{ref}}$.

4.4 The conservative optimality gap

For a policy that spends E_π on a scenario solved by both the policy and the reference,

$$\eta_{\mathrm{cons}} := \frac{E_\pi}{E_{\mathrm{ref}}} - 1 \;\le\; \frac{E_\pi}{E^\star} - 1 \;\le\; \frac{E_\pi}{E_{\mathrm{lb}}} - 1 =: \eta_{\mathrm{opt}} \tag{5}$$

We report η_{cons} as the headline metric. Whatever the true optimum is, the policy is at least this far above it. The ratio $E_{\mathrm{ref}}/E_{\mathrm{lb}}$ measures the width of the bracket, and therefore how much room is left between the two statements (Fig. 1).

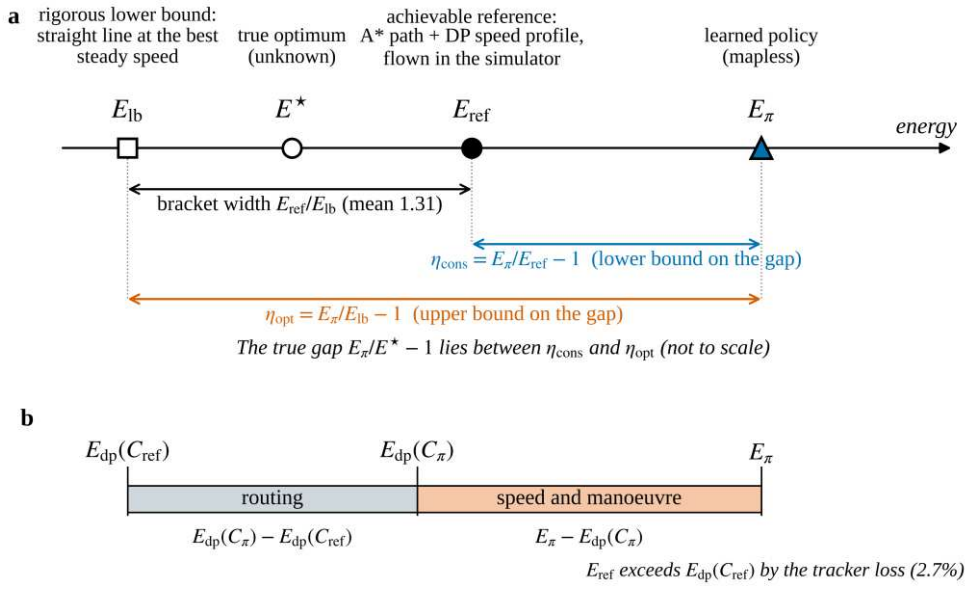


Fig. 1 Bracketing the energy-optimal mission. **a** The rigorous lower bound E_{lb} and the achievable reference E_{ref} enclose the unknown optimum $E^\star$; the energy E_π of a policy gives a conservative gap η_{cons} (against E_{ref}) and an upper bound η_{opt} (against E_{lb}). **b** The excess of a policy over the idealised reference splits additively into a routing term and a speed-and-manoeuvre term (Sect. 4.5). Schematic, not to scale

4.5 Decomposing the gap

The same dynamic programme can be run on the path a policy actually flew. Writing $E_{\mathrm{dp}}(C)$ for the DP optimum along a curve C, and C_π, C_{ref} for the flown policy and reference paths, the excess over the idealised reference splits additively (Fig. 1b):

$$E_\pi - E_{\mathrm{dp}}(C_{\mathrm{ref}}) \;=\; \underbrace{\left[E_\pi - E_{\mathrm{dp}}(C_\pi)\right]}_{\text{speed and manoeuvre}} \;+\; \underbrace{\left[E_{\mathrm{dp}}(C_\pi) - E_{\mathrm{dp}}(C_{\mathrm{ref}})\right]}_{\text{routing}} \tag{6}$$

The first term is what the policy loses by not flying the best speed profile along its own route, including attitude transients and course corrections. The second is what it loses by choosing a worse route. The loss of the reference tracker against its own plan is reported separately.

The separation matters for interpretation. The reference planner is privileged – it holds the obstacle map – so part of the routing term reflects partial observability rather than a defect of the learning algorithm. The speed term does not: the energy-optimal cruise speed is a property of the vehicle, inferable from the policy's own state, and a map would not reveal it.

5 Energy-aware learning

5.1 Markov decision process

The observation has 30 components: the unit vector to the goal and the clipped distance (4); the velocity (3); the body z axis (3); ten range readings – eight horizontal rays at 45° spacing plus up and down, 40 m maximum – as proximities (10); the roof height of the building each horizontal ray strikes, relative to the current altitude (8); the fraction of the energy budget remaining (1); and the fraction of the time limit remaining (1). A three-dimensional range sensor resolves the top of an obstacle as well as its range, and a policy needs that to judge whether climbing over is cheaper than going around. A wind-aware variant appends the measured wind vector (3).

The action is a commanded acceleration in $[-4, 4]^2 \times [-3, 3]$ m/s². An episode ends on reaching the goal within 5 m, on collision with a building or the volume boundary, or after 300 steps (30 s). The reward is

$$r_t = (d_{t-1} - d_t) \;-\; 0.05 \;-\; \lambda_E \frac{E_t}{P_{\mathrm{hover}} \Delta t} \;+\; 20 \cdot \mathbb{1}_{\mathrm{goal}} \;-\; 20 \cdot \mathbb{1}_{\mathrm{crash}} \tag{7}$$

with E_t the electrical energy of the step. Because the progress term telescopes to the initial distance, the route is not rewarded directly: only time and energy distinguish one route from another.

5.2 Energy as a constraint

The constrained variant removes the energy term from the reward ($\lambda_E = 0$) and treats the mission as a constrained MDP with per-step cost $c_t = E_t/B$, where the budget $B = 1.25\, E_{\mathrm{lb}}$ comes from the lower bound and is therefore known before the flight and interpretable as *finish within 25% of the theoretical floor*.

The constraint $\Sigma\, c_t \leq 1$ is relaxed with a learned multiplier on a softplus parameterisation, updated by dual gradient ascent against a separate cost critic [26–29].

Two details matter. The constraint is on the episodic cost measured from the start of a mission, so the cost critic is undiscounted and the dual variable is driven by Q_c evaluated at *initial* states only; evaluating it at replay-buffer states would measure the *remaining* cost and would make the constraint look satisfied when it is not. The remaining budget fraction is part of the observation in every configuration, so the comparison isolates the treatment of the limit rather than the information about it.

5.3 Algorithms, implementation and verification

SAC [14, 15], TD3 [32] and PPO [12, 13] are implemented directly in NumPy from their original formulations, with two hidden layers of 64 units, Adam at 3×10^{-4}, $\gamma = 0.99$ and a batch size of 256; the full list is in Appendix A (Table A2). One gradient step is taken per environment step for the off-policy algorithms.

Every analytic gradient is checked against central finite differences: the dense-network parameter and input gradients, the SAC critic gradients, the reparameterised SAC actor gradient through the tanh squashing including the entropy term, the temperature gradient, the TD3 deterministic policy gradient, the PPO clipped surrogate with respect to both the mean and the log standard deviation, and the generalised-advantage estimator against its explicit definition. The worst relative error over the suite is 1×10^{-5}. The log-determinant of the tanh transform is evaluated in the stable form $\log(1 - \tanh^2 u) = 2[\log 2 - u - \mathrm{softplus}(-2u)]$, whose derivative is exactly $-2 \tanh u$, so the policy gradient needs no stabilising epsilon.

Because the implementations are not taken from a library, they are additionally validated against the scores published for Stable-Baselines3 [33] on Pendulum-v1 under matched budgets. All three match or exceed the published reference: SAC reaches −123.8 against −176.3, TD3 −126.7 against −196.0 and PPO −136.7 against −230.4 (Appendix B, Table B2).

5.4 Protocol

Each configuration is trained with several seeds in 32 parallel environments. Every 40 000 steps the deterministic policy is evaluated on 24 validation scenarios; these curves are reported but never used for model selection. The final policy of each run is evaluated deterministically on the same 200 held-out test scenarios used for the reference.

Metrics are computed on the subset of test scenarios solved by the reference and by every policy entering the comparison, which removes the selection effect that would otherwise flatter a policy that solves only the easy scenarios. Gaps are aggregated with the interquartile mean and a hierarchical bootstrap that resamples seeds and then scenarios within a seed [25]; per-seed means are compared with Welch's *t*-test under Holm correction, with Hedges' *g* as the effect size.

6 Results

6.1 The energy model reproduces measured flight power out of sample

Three aerodynamic coefficients were fitted to the published cruise-power law at two payloads and five speeds. On that training set the average absolute relative error is 2.3%. Four tests then use data that played no part in the fit (Table 2).

Table 2 Held-out validation of the energy model against the published power regression (ARE, average absolute relative error)

Test	What it checks	n	ARE	Max	Inside the 1σ band
calibration	cruise, 3.68 and 4.18 kg, 4–12 m/s	10	2.3%	4.0%	9/10
T1	cruise at the held-out 250 g payload	5	**2.3%**	3.9%	5/5
T2	take-off regime, 2.5 m/s climb	3	**6.9%**	7.6%	0/3
T3	landing regime, 1.5 m/s descent	3	**1.9%**	2.5%	3/3
all held-out	T1 + T2 + T3	11	**3.5%**	7.6%	8/11

For comparison, the purely empirical regression that the model is tested against achieves 2.1% on held-out *flights* (Fig. 2). The model over-predicts the take-off regime by about 7%, which is partly definitional: the take-off segment of the reference missions includes rotor spin-up and the transition to cruise, which a steady 2.5 m/s climb does not represent. The model also reproduces the reported weak dependence of cruise power on speed – 7.1% variation between 4 and 12 m/s against the roughly 5% reported.

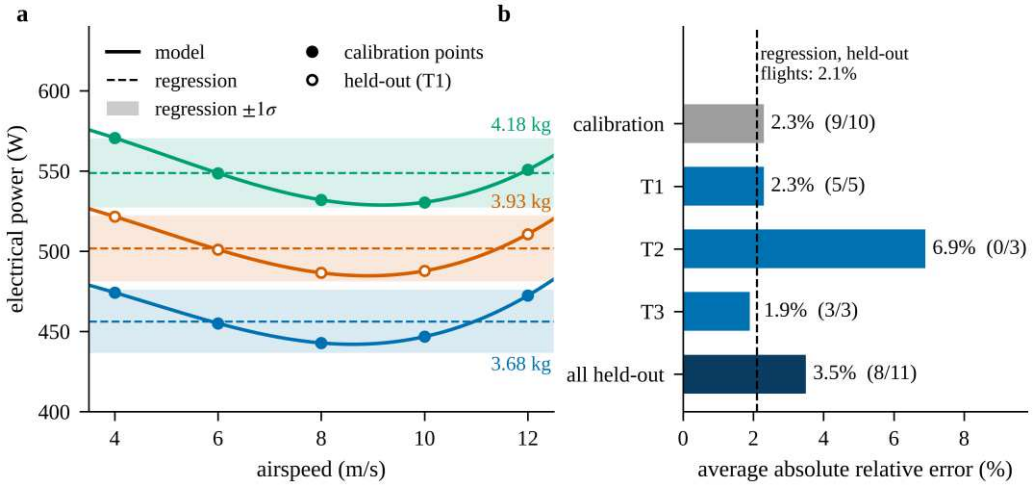


Fig. 2 Validation of the energy model against the published power regression of Rodrigues et al. [1]. **a** Cruise electrical power against airspeed for the two calibration masses (filled markers) and the held-out 250 g payload (open markers); dashed lines and shaded bands are the regression and its 1σ uncertainty. **b** Average absolute relative error of each test, with the number of points inside the 1σ band in parentheses; the dashed line is the 2.1% error of the regression on held-out flights

Two consequences matter for the rest of the paper. First, because both the induced and the profile terms scale as $m^{1.5}$, a single induced-power group explains the measured power well, which is why the published regression needs no speed term. Second, the resulting energy-per-metre curve has an interior minimum at 14.9 m/s with no payload, rising to 15.8 m/s at 500 g. In a 6 m/s headwind the optimal *ground* speed falls to 11.8 m/s while the optimal *airspeed* rises to 17.8 m/s, and in a 6 m/s tailwind the

optimum is pushed against the vehicle's 17 m/s speed limit – the classical result, recovered here from a model calibrated only on measured cruise power (Fig. 3).

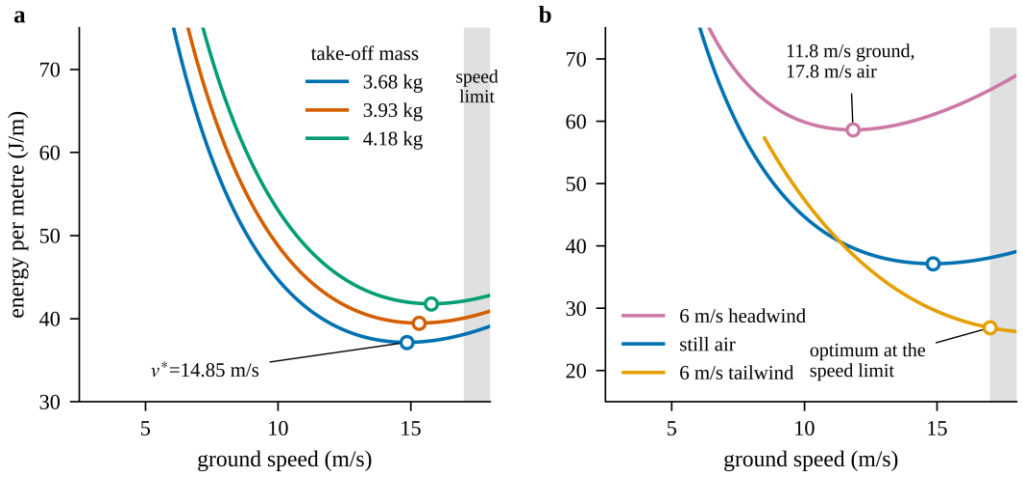


Fig. 3 Energy per metre of ground track from the calibrated model of Sect. 3 (parameters of Table A1). **a** Still air, for three take-off masses; open circles mark the energy-optimal speed v^*. **b** Effect of a 6 m/s head- or tailwind on the 3.68 kg vehicle: in a headwind the optimal ground speed falls while the optimal airspeed rises, and in a tailwind the optimum lies at the 17 m/s speed limit

6.2 The bounds bracket the optimum within about 30%

Across the 200 held-out test scenarios the achievable reference solves 99.0% and uses on average 1.31 times the rigorous lower bound (range 1.18–1.60, median 1.30). A typical reference mission is 164 m of path flown in 14.4 s for 7.9 kJ, against a lower bound of 6.0 kJ.

The bracket is wide because the missions are short. The vehicle starts from rest, climbs over or steers around buildings, and arrives still moving, while the lower bound assumes steady flight at the best speed along the straight line. The reference tracker itself loses about 4% against the dynamic-programming optimum of its own path, which is why the decomposition of Sect. 4.5 is stated against that idealised reference rather than against the flown one.

6.3 How far from optimal are the learned policies?

Four seeds were trained per configuration for 90 000 environment steps. Methods are compared on the 23 of 200 test scenarios that the reference solves and that at least half the seeds of every off-policy method solve, which removes the selection effect that would otherwise flatter whichever method fails on the hard cases. Gaps are interquartile means with 95% intervals from a hierarchical bootstrap over seeds and then scenarios (Table 3).

Table 3 Optimality gap of the learned policies against the achievable reference (four seeds per configuration, 90 000 environment steps; matched subset of 23 test scenarios; gap as interquartile mean with 95% hierarchical-bootstrap interval)

Method	Seeds	Success	E (kJ)	J per m	Gap vs E_{ref} (IQM, 95% CI)	Cruise speed (m/s)	E/E_{lb}
SAC + energy reward	4	37.8 ± 11.3%	8.48	48.5	**7.3% [4.0, 12.3]**	11.29	1.40
SAC, no energy term	4	47.0 ± 23.3%	8.88	50.1	**14.8% [8.6, 25.0]**	10.29	1.48
TD3 + energy reward	4	24.0 ± 15.3%	9.05	49.7	14.0% [10.7, 17.2]	12.76	1.50
SAC + energy constraint	4	36.5 ± 23.4%	8.87	50.0	10.7% [8.6, 14.2]	11.54	1.47
PPO + energy reward	4	0.0 ± 0.0%	–	–	–	–	–

The energy term roughly halves the optimality gap. Energy-aware SAC spends 7.3% more energy than a trajectory we can actually fly; the same algorithm without the energy term spends 14.8% more (Fig. 4a). Because the reference is achievable rather than optimal, both figures are lower bounds on the true gap.

The difference is large but not statistically significant at four seeds. Welch tests on per-seed means give Hedges' $g = -0.94$ for the energy term against no energy term, -1.34 against TD3 and -1.11 against the constrained variant – all large effects – but none survives Holm correction (adjusted $p \geq 0.26$). The obstacle is seed variance in success rate, which reaches ±23 percentage points for two of the configurations. The ordering is consistent across all four seeds; the sample of seeds is simply too small to certify it. This is reported rather than hidden: a single seed of this experiment gave a 2.8% gap against 26.8%, an apparent order-of-magnitude separation that four seeds reduce to roughly a factor of two.

Energy awareness costs success. Energy-aware SAC succeeds on 37.8% of test scenarios against 47.0% without the energy term (Fig. 4b). The energy penalty competes with the crash penalty, and at this training budget the trade is visible. Neither the direction nor the size of that cost is certified by four seeds ($g = -1.12$, adjusted $p = 0.56$).

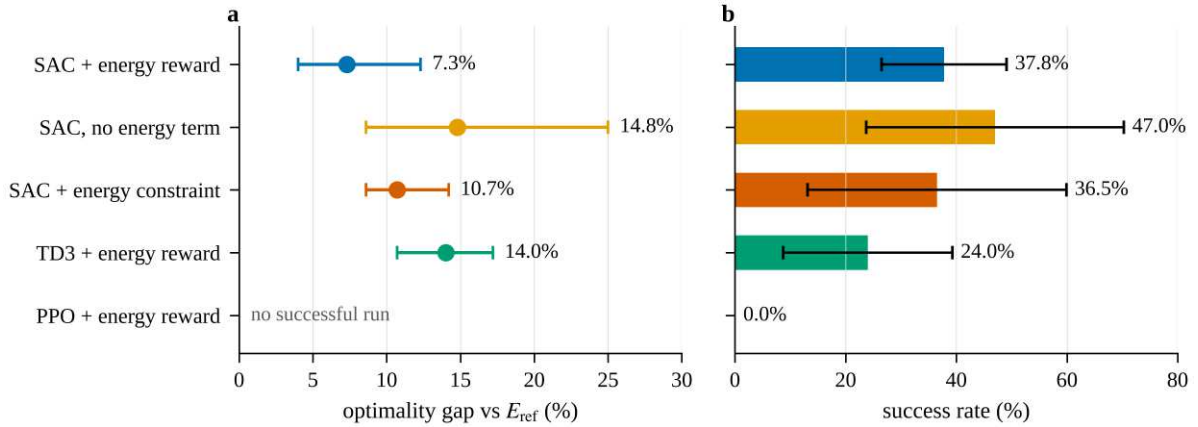


Fig. 4 Learned policies in still air. **a** Optimality gap against the achievable reference (interquartile mean with 95% hierarchical-bootstrap interval) on the matched subset of 23 scenarios. **b** Success rate on the test scenarios (mean ± spread over four seeds). PPO reached no successful run at this budget

PPO does not reach a usable policy at a matched step budget. With the budget matched in environment steps, the on-policy method ends at 0% success on every seed. Because PPO is roughly six times cheaper per environment step here, it was additionally run at five times the budget (450 000 steps, four seeds). That does not rescue it: one seed of the four reaches 34.0% success and a 34.1% gap, while the other three stay at or below 1.0%, for a mean of 8.8 ± 14.6%. We therefore do not claim that PPO's failure is purely a budget effect. It is a statement about this on-policy implementation, with hyperparameters shared across algorithms rather than tuned per algorithm, on this task; PPO is known to need per-task tuning and larger budgets for problems of this kind, and we report the outcome as observed rather than tuning it until it works.

6.4 Where the gap comes from

Running the same dynamic programme on the path each policy actually flew splits the excess into a speed-and-manoeuvre term and a routing term (Table 4, Fig. 5).

Table 4 Decomposition of the excess energy into speed-and-manoeuvre and routing terms (matched subset of Table 3)

Method	vs E_{ref}	Speed and manoeuvre	Routing	Sum vs idealised reference	Path length excess
SAC + energy reward	8.8%	4.9%	6.7%	11.6%	7.5%
SAC, no energy term	18.8%	**11.1%**	10.1%	21.2%	12.0%
TD3 + energy reward	14.5%	8.6%	9.2%	17.8%	10.8%
SAC + energy constraint	15.9%	11.3%	8.2%	19.5%	9.8%

The decomposition is computed on the same matched subset as Table 3, so the two are directly comparable; the reference tracker's own loss against the optimum of its path is 2.7%.

Speed and routing contribute comparably, with speed the larger term for every method that carries a visible gap. For the reward without an energy term the split is 11.1 points of speed against 10.1 of routing; for TD3, 8.6 against 9.2; for the constrained variant, 11.3 against 8.2. Energy-aware SAC has the smallest of both (4.9 and 6.7). Roughly half of the excess energy is therefore attributable to how fast the policy flies rather than to which way it goes.

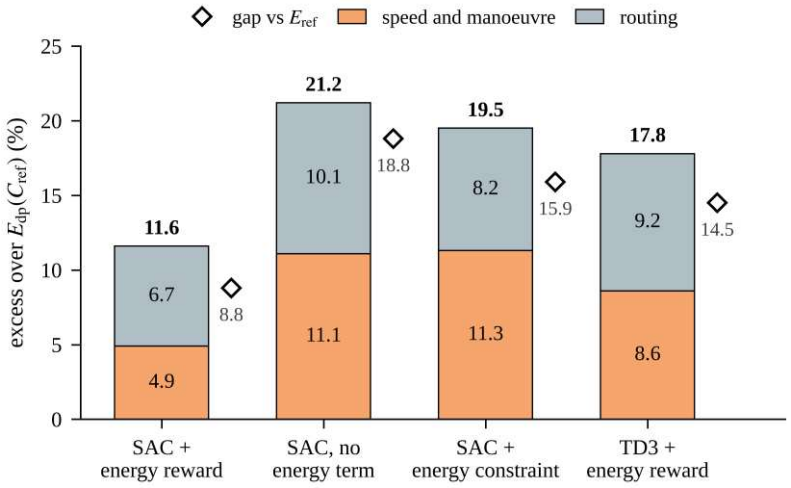


Fig. 5 Decomposition of the energy excess over the idealised reference into speed-and-manoeuvre and routing terms (bars, totals in bold) and the gap against the flown reference E_{ref} (diamonds), on the matched subset of Fig. 4

This matters because the routing term is the part a map would remove. The reference planner is privileged; the policies are mapless, and their paths are 7.5–12.0% longer than the reference's. The speed term carries no such excuse: the energy-optimal cruise speed is a property of the vehicle, inferable from the policy's own state. That makes it the diagnostic worth reporting, and it is the term that responds to the energy signal – it falls from 11.1% to 4.9% when the energy term is added to the reward.

6.5 Do the policies find the energy-optimal speed?

The calibrated model puts the minimum of the energy-per-metre curve at 14.85 m/s in still air. Averaged over seeds the policies cruise at 11.29 m/s (energy reward), 12.76 m/s (TD3), 11.54 m/s (constraint) and 10.29 m/s (no energy term). Every method sits on the left branch of the curve, where energy per metre rises as speed falls, and the method with the lowest gap is not the fastest: TD3 cruises closest to v^* but pays for it in routing and in a 24% success rate (Fig. 6).

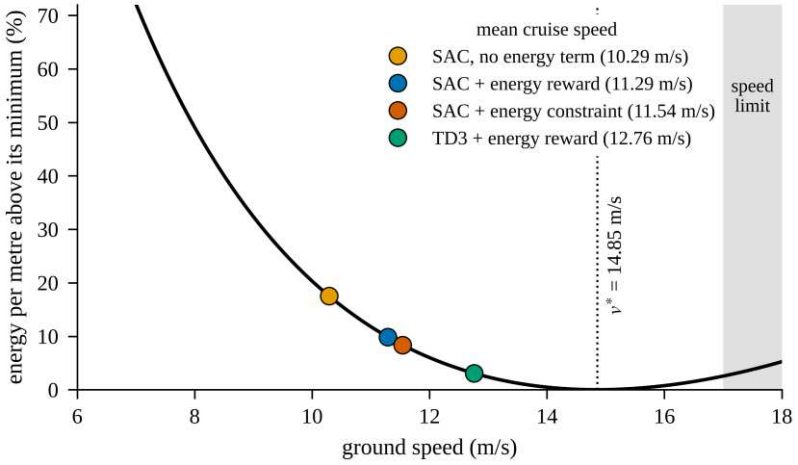


Fig. 6 Mean cruise speeds of the learned policies on the energy-per-metre curve of the calibrated model (3.68 kg, still air). Every policy cruises on the left branch of the curve, below the energy-optimal speed v^*

No method reaches v^*. Flying at the energy optimum means flying close to the vehicle's 17 m/s limit, which raises collision risk in a cluttered field, and every policy trades part of that optimum for safety. That trade is legitimate; what the decomposition shows is its price – at 10.3 m/s the energy-per-metre curve is about 12% above its minimum, which is most of the speed term for the reward without an energy penalty.

6.6 Wind

Policies were re-evaluated in a 6 m/s steady wind field with first-order gusts (sigma = 0.8 m/s), on the 49 scenarios solved by the reference and reliably by the wind-relevant methods (Table 5, Fig. 7).

Table 5 Policies re-evaluated in a 6 m/s steady wind with first-order gusts (49 scenarios)

Method	Success in wind	E (kJ)	Gap vs E_{ref}	Cruise speed (m/s)
SAC + energy reward (still-air training)	38.4%	9.32	11.1% [5.2, 16.2]	10.52
SAC, no energy term (still-air training)	44.0%	10.04	15.7% [10.4, 25.1]	9.74
SAC + energy constraint (still-air training)	34.5%	9.95	12.9% [9.9, 21.5]	10.80
TD3 + energy reward (still-air training)	30.0%	9.54	13.3% [10.3, 17.8]	11.97
SAC trained in wind	41.1%	9.49	11.5% [9.9, 13.5]	11.21
SAC trained in wind, wind observed	**48.9%**	9.58	14.0% [9.0, 19.4]	10.49

Every method loses efficiency in wind: gaps rise by three to four points and cruise speeds fall. Training in wind recovers success relative to the still-air policies, and adding an observation of the wind recovers it further, to the highest rate of any configuration tested (48.9%), at the cost of a slightly larger gap. The wind-observing policy is the only one whose cruise speed does not fall as much as the others', which is the behaviour the theory of Sect. 6.1 predicts; with four seeds the difference is suggestive rather than established.

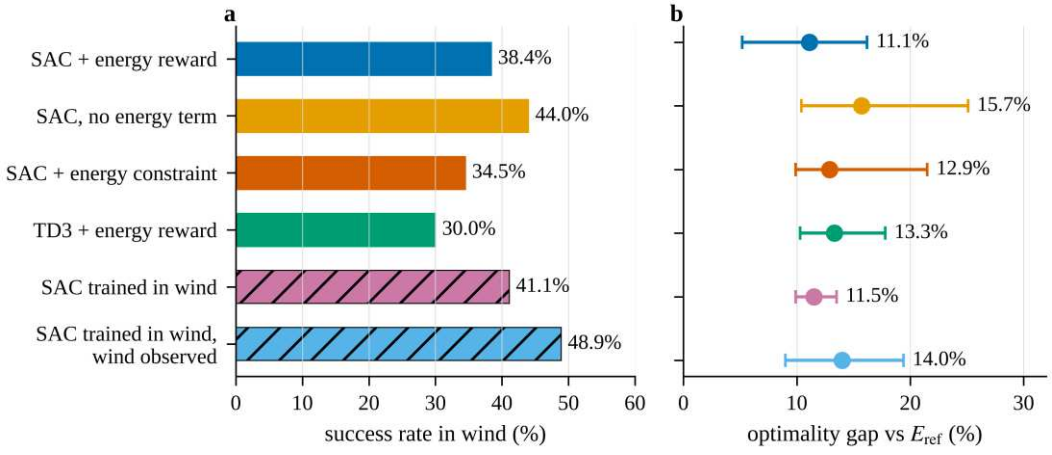


Fig. 7 Policies in a 6 m/s steady wind with first-order gusts (49 scenarios). **a** Success rate; policies trained in wind are hatched. **b** Optimality gap against E_{ref} (interquartile mean with 95% interval)

6.7 Budget as a constraint or as a penalty

The budget was set at 1.25 times the rigorous lower bound. That turns out to be tighter than the achievable reference itself, which uses 1.31 times the bound on average, so the constraint is infeasible on most scenarios by construction – violation rates are 95–100% for every method. The result is still informative: the Lagrangian agent reaches a mean energy ratio of 1.174 of its budget against 1.184 for the unconstrained reward and 1.123 for the energy-reward variant, and a gap of 10.7% against 14.8%. It therefore sits between the two reward variants rather than beating both, which is a weaker claim than one seed suggested. A budget factor at or above the reference's own 1.31 would make the constraint attainable and is the setting we would use next; we report the tighter threshold as flown.

6.8 Sensitivity to the energy model

The gap is a ratio of two energies computed with the same model, so a uniform scaling of power cancels. What does not cancel is a change in the *shape* of the power curve, because that moves the optimal speed and therefore the reference. Each uncertain coefficient was perturbed, the achievable reference re-planned and re-flown, and two energy-aware SAC policies re-evaluated on the perturbed vehicle over 60 test scenarios (Table 6, Fig. 8).

Table 6 Sensitivity of the gap, the energy-optimal speed v^* and the bracket width E_{ref}/E_{lb} to the energy-model coefficients (60 test scenarios)

Perturbation	v^* (m/s)	Gap	E_{ref} / E_{lb}
nominal	14.85	7.0%	1.294
profile drag coefficient +25%	15.14	7.2%	1.294
profile drag coefficient −25%	14.53	8.5%	1.291
induced-power factor +15%	15.16	7.6%	1.313
flat-plate area +25%	13.85	7.8%	1.250
flat-plate area −25%	16.21	9.4%	1.344
chain efficiency −6%	14.83	7.5%	1.294
avionics power doubled	15.14	7.5%	1.293
vehicle mass +10%	15.54	8.2%	1.306

Across perturbations of ±25% in the two calibrated aerodynamic coefficients, +15% in the induced-power factor, and the two quantities fixed a priori, the gap stays between 7.0% and 9.4% and the width of the bracket between 1.25 and 1.34. The two quantities that were not calibrated matter least: halving the assumed chain efficiency margin or doubling the avionics power moves the gap by half a point. The largest single effect is a 25% reduction in flat-plate area, which pushes the energy-optimal speed from 14.9 to 16.2 m/s and so widens the distance between the policies' cruise speed and the optimum. No perturbation changes the ordering of the methods or the conclusion that the speed term is the one the energy signal moves.

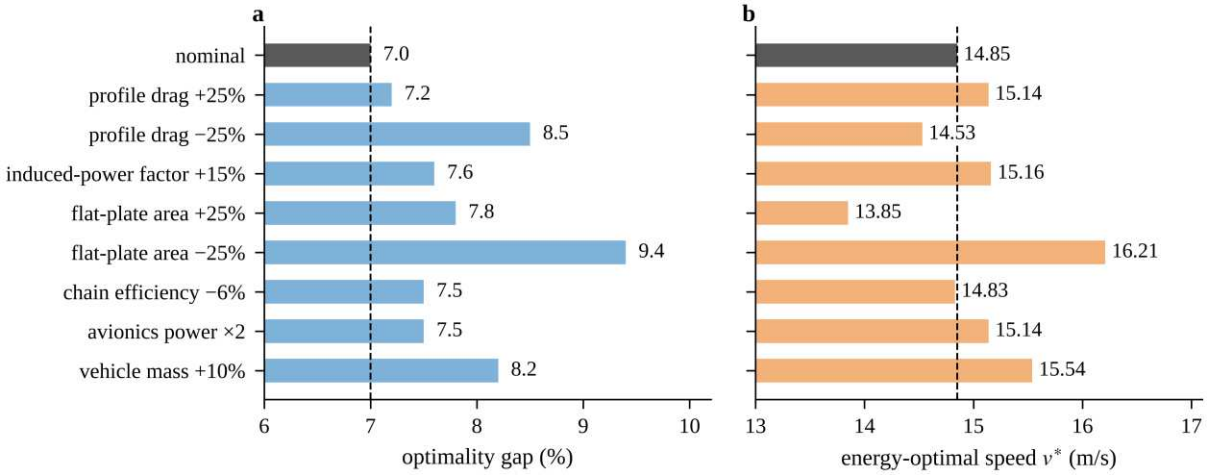


Fig. 8 Sensitivity to the energy-model coefficients (60 test scenarios). **a** Optimality gap of the energy-aware SAC policies. **b** Energy-optimal speed v^*. Dashed lines mark the nominal model

6.9 Caveats on the experimental scale

These results come from four seeds per configuration at 90 000 environment steps, a reduced budget. Three consequences follow.

Success rates are well below convergence and vary strongly between seeds – up to ±23 percentage points – so the matched comparison subset is small (23 of 200 scenarios in still air, 49 in wind) and the gaps are measured on the easier scenarios. With four seeds, Welch tests on per-seed means detect none of the differences after Holm correction even where the effect sizes are large ($|g| = 0.9$–1.3); the experiment is under-powered, not null. And PPO's failure should be read as a statement about this sample budget rather than about the algorithm.

One seed is not enough for this task, and we say so because our own first pass showed it. Trained on a single seed the same experiment gave a 2.8% gap against 26.8%, an apparent order-of-magnitude separation; four seeds put the same comparison at 7.3% against 14.8%, and the constrained variant that looked best on one seed is mid-field on four. The ordering of the methods survived; the magnitudes did not. Any study reporting energy gaps from one or two seeds should be read with that in mind.

What the reduced scale does not threaten is the methodology. The bounds, the matched subset and the decomposition are independent of the training budget, and the decomposition's qualitative message – that speed selection is comparable to or larger than routing, and that it is the term the energy signal moves – is consistent across every method and every seed.

7 Discussion

7.1 Why the speed term is the one that matters

The reference planner is privileged: it holds the obstacle map, while the policies see only range readings. Part of the routing term therefore measures partial observability rather than a defect of the learning algorithm, and a fair reading of the routing numbers is *this is what a map would be worth*.

The speed term admits no such excuse. The energy-optimal cruise speed is a property of the vehicle and the wind, not of the obstacle field; it is inferable from the policy's own velocity and from the energy signal it already receives, and a map would not reveal it. When a policy cruises away from the minimum of the energy-per-metre curve it is leaving energy on the table for reasons internal to the learning problem – the shaping that makes it value time, the entropy that keeps its actions stochastic, or simply a flat reward gradient in that direction.

That is why the decomposition is worth computing rather than reporting a single gap. The two terms have different remedies: routing improves with better perception, memory or a planner in the loop; speed improves with reward shaping, a constraint, or an explicit speed command.

7.2 Reading the numbers at mission scale

The missions here are short urban legs – about 164 m of path in 14 s for roughly 8 kJ, which is 1.7% of the 130 Wh pack. Two consequences follow.

A short leg is dominated by what is not cruise. The vehicle accelerates from rest, climbs over or steers around buildings, and arrives still moving, so the rigorous lower bound sits about 30% below what any trajectory can achieve. On a longer leg the cruise fraction grows, the bracket narrows, and the speed term becomes a larger share of the total. The gaps reported here therefore apply to dense short-range delivery, and the speed term is a conservative estimate of what it would be over kilometres.

The practical currency is flights per charge and usable range. At the reference energy a full pack supports about 59 such legs; a policy that spends 30% more supports about 45. For a fixed energy reserve the usable one-way range shrinks in the same proportion. Neither number depends on the absolute calibration of the model, because both are ratios.

7.3 Reward penalty or explicit constraint

A reward penalty sets an implicit exchange rate between energy and every other objective, and the designer does not control where the policy lands on that trade-off: the weight must be tuned, and the result moves with every other reward term. A constraint states the requirement directly – finish within a stated multiple of the theoretical floor – and lets the dual variable find the exchange rate. Because our budget is derived from the rigorous lower bound it is known before the flight and transfers across scenarios of different length without retuning, which a fixed reward weight does not.

7.4 Recommendations

1. **Report energy against a bound, not against another learned policy.** A percentage improvement over an unknown baseline is not interpretable. Both bounds used here are cheap: the lower bound is a one-dimensional search over the vehicle's own power curve, and the achievable reference is a planner plus a dynamic programme.

2. **Calibrate the power model against measured flight data and state the out-of-sample error.** A gap is a ratio of energies and inherits the model's bias; a power model that is constant or linear in speed cannot represent the trade-off that decides energy-optimal flight.
3. **Separate routing from speed selection when diagnosing a policy.** They have different causes and different remedies.
4. **Prefer an explicit budget constraint when the requirement is "arrive within a given energy".**
5. **Give the policy the wind, or an estimate of it.** The energy-optimal ground speed moves by several metres per second over a realistic wind range, and a policy that cannot observe the wind cannot track it.
6. **Match the evaluation set across methods.** Restricting the comparison to scenarios every method solves removes a selection effect that otherwise flatters whichever policy fails on the hard cases.

7.5 Limitations

- The energy model is calibrated on one airframe class and one published data set, and its out-of-sample error is a few per cent. Conclusions are stated as ratios, which cancels much but not all of that bias.
- The raw per-flight records of the reference data set were not reachable from the environment in which this study was run, so calibration and validation use the published per-regime regression and its stated uncertainties rather than individual flights.
- The achievable reference is not provably optimal, so the reported gap is a rigorous lower bound on the true gap, not an estimate of it. Tightening the reference – with a full non-linear trajectory optimisation, for instance – would increase every reported gap.
- The reference planner is privileged. Part of the routing term is the value of a map, not a failure of learning.
- Scenarios are cylindrical buildings in a bounded volume with a steady wind field and first-order gusts; real urban airspace has richer geometry, shear, moving obstacles and regulatory constraints.
- Hyperparameters were fixed once and shared across algorithms rather than tuned per algorithm, and the sample budget is matched in environment steps, which disadvantages on-policy methods. PPO did not reach a usable policy even at five times the budget, on three of four seeds.
- Four seeds per configuration is too few to certify the differences reported: effect sizes are large but no pairwise comparison survives Holm correction. Eight to ten seeds would be needed to settle the ordering.

- No flight experiment is reported. The validation this work would need is a power-against-airspeed sweep on the airframe of interest, followed by a logged mission flown by both the reference planner and the learned policy on the same route.

8 Conclusions

A percentage energy saving over another learned policy says nothing about how much energy is left on the table. This paper measures the quantity that does: how far a learned quadrotor navigation policy is from energy-optimal, bracketed between a rigorous lower bound and a trajectory that is actually flown in the same simulator.

The energy model that makes the measurement meaningful is calibrated on the cruise regime of 188 instrumented delivery flights at two payloads and predicts, out of sample, cruise power at a third payload to 2.3% and the power of two flight regimes it never saw to 6.9% and 1.9%, for a pooled held-out error of 3.5%. Against that model the achievable reference uses 1.31 times the rigorous lower bound, so the optimum is bracketed within about 30%.

Measured against that reference over four seeds per configuration, soft actor-critic with an energy term in the reward spends 7.3% more energy than a flyable trajectory, while the same algorithm without the energy term spends 14.8% more – at least that much, since the reference is achievable rather than optimal. The effect is large (Hedges' $g = -0.94$) but four seeds cannot certify it after multiplicity correction, and we report that rather than overstate it.

Decomposing the excess shows that roughly half of it is speed selection rather than routing, and that speed is the term the energy signal actually moves: it falls from 11.1% to 4.9% when the energy term is added, while routing changes much less. Every policy cruises on the left branch of the energy-per-metre curve, between 10.3 and 12.8 m/s, where the vehicle's energy-optimal speed is 14.9 m/s. Routing is the part a map would fix; speed is not, which makes it the diagnostic worth reporting.

Energy awareness is not free: the energy reward costs about nine points of success rate at this training budget. Imposing the same requirement as a constrained-MDP constraint, with the budget derived from the lower bound, landed between the two reward variants rather than beating both; the budget we used, 1.25 times the lower bound, is tighter than the reference's own 1.31 and was therefore infeasible on most scenarios. In wind, every policy loses efficiency; training in wind recovers success, and observing the wind recovers more, to the highest success rate of any configuration tested.

These numbers come from four seeds at a reduced training budget and establish an ordering, not a benchmark. We also report what one seed would have concluded – a 2.8% gap against 26.8%, an apparent order-of-magnitude separation that four seeds reduce to a factor of two – because a literature that commonly reports one or two seeds should see what that costs. The methodology does not depend

on the budget: the bounds are cheap, the decomposition needs only a dynamic programme along the flown path, and both are reusable by any study that reports the energy of a learned flight policy.

Declarations

Funding No funding was received for conducting this study.

Competing interests The authors have no competing interests to declare.

Ethics approval and consent to participate Not applicable.

Consent for publication Not applicable.

Author contributions Iqrar Ali: conceptualization, methodology, software, writing – original draft. Sayeed Omar: formal analysis, investigation, visualization. Ma Guoli: supervision, writing – review and editing. All authors read and approved the final manuscript.

Relation to other work Two companion manuscripts by the authors [22, 23] study structural energy harvesting on the same airframe family and conclude that endurance is gained by energy-aware flight rather than by harvesting. The present paper contains no results from them; it takes up that conclusion as its starting point and shares no model, code or data with them beyond the general modelling approach.

Data and code availability The energy model, the six-degree-of-freedom simulator, the planner and bounds, the learning algorithms, the unit-test suite, the per-run raw results and all figure scripts are provided as supplementary material. The only dependencies are NumPy, SciPy and Matplotlib.

A note on the calibration data The raw per-flight records of the reference data set [4] were not reachable from the environment in which this study was run. Calibration and validation therefore use the published per-regime regression of [1], which was fitted to those flights, together with its stated coefficient uncertainties, rather than individual flight records. All validation statements in Sect. 6.1 are made against that regression and its 1σ band.

Use of AI tools An AI assistant (Claude, Anthropic) was used under the authors' direction to help implement and verify the simulation code, to run the experiments, and to draft and revise the manuscript. The authors reviewed all code, results and text and take full responsibility for the content of this article.

Appendix A Parameters and hyperparameters

Table A1 Vehicle, environment and model parameters

Symbol	Quantity	Value	Source
m	mass before payload	3.68 kg	instrumented platform [4]
n	rotors	4	–
R	rotor radius	0.1651 m	DJI 1345 propeller
c, N_b	blade chord, blades	0.024 m, 2	geometry
s	solidity N_b c/(πR)	0.0925	derived
C_T	thrust coefficient	0.0123	geometry
δ	effective profile drag coefficient	0.0480	calibrated
κ	induced-power factor	1.000	calibrated
S_{fp}	equivalent flat-plate area	0.0741 m²	calibrated
K_μ	advance-ratio coefficient	4.65	[6]
η	motor + ESC efficiency	0.80	fixed a priori
P_{av}	avionics power	30 W	fixed a priori
ℓ	rotor arm	0.325 m	geometry
I	inertia (xx, yy, zz)	0.064, 0.064, 0.118 kg m²	estimate
τ_m	rotor time constant	45 ms	typical
–	per-rotor thrust limit	42.6 N	from Ω_{max}
–	tilt limit	32°	flight-controller limit
v_{max}	speed limit	17 m/s	manufacturer, GPS mode
–	battery	130 Wh, 19.8–25.2 V, 42 mΩ	DJI TB48D
ρ	air density	1.225 kg/m³	ISA sea level
–	hover power	496 W	model

Calibrated values are marked; everything else is geometric, from the manufacturer, or fixed a priori.

Environment: volume 180 × 120 × 58 m; 6–12 cylinders of radius 5–10 m and height 15–40 m; goal tolerance 5 m; control interval 100 ms with ten 10 ms dynamics steps; episode limit 300 steps; range sensor 40 m.

Table A2 Learning hyperparameters, identical across seeds and configurations

	SAC	TD3	PPO
hidden layers	2 × 64, ReLU	2 × 64, ReLU	2 × 64, ReLU (separate π, V)
optimiser, learning rate	Adam, 3×10^{-4}	Adam, 3×10^{-4}	Adam, 3×10^{-4}
discount γ	0.99	0.99	0.99
batch / minibatch	256	256	256 (rollout 128 × 32 envs, 10 epochs)
replay buffer	3×10^5	3×10^5	–
warm-up (uniform random)	5 000 steps	5 000 steps	–
target network	Polyak τ = 0.005	Polyak τ = 0.005	–
algorithm-specific	auto-α (init 0.2), target entropy −3; tanh-squashed Gaussian, log σ ∈ [−5, 2]	policy noise 0.2, clip 0.5, delay 2, exploration noise 0.1	clip 0.2, GAE λ = 0.95, value coefficient 0.5, state-independent log σ
gradient-norm clipping	none	none	0.5
gradient steps per env step	1	1	–
parallel environments	32	32	32

Constrained variant: cost budget 1.0, that is $\Sigma E_t \le 1.25 E_{lb}$; separate undiscounted cost critic with the same architecture; multiplier on a softplus parameterisation, dual learning rate 3×10^{-3}, capped at 50; cost estimate smoothed with an exponential moving average (0.98) over Q_c at initial states.

Reward weights: progress 1.0 per metre, step cost 0.05, goal +20, crash −20, $\lambda_E = 0.1$ where used.

Appendix B Verification

Table B1 Unit-test suite (35 of 35 pass)

#	Test	Result
1	induced velocity in hover equals $\sqrt{(T/2\rho A)}$	exact
2	induced velocity solves the momentum quartic (200 random states)	max rel. 1.1×10^{-15}
3	climb costs more than hover	492 > 443 W
4	shallow descent costs less than hover	370 < 443 W
5	energy per metre has an interior minimum	holds
6	optimal airspeed rises and ground speed falls in headwind	holds
7	battery terminal power equals the load	10^{-6} W
8	chemical drain equals load plus ohmic loss	10^{-6} W
9	open-circuit voltage is monotone in state of charge	holds
10	hover is an equilibrium of the 6-DOF model	3.6×10^{-15} m
11	6-DOF hover power equals the steady model	10^{-6} W
12	rotation matrix stays orthonormal	2.2×10^{-16}
13	thrust allocation round-trips	10^{-10}
14	integrated energy matches the power trace	10^{-9} J
15	observation has the declared dimension	30
16	scenarios are reproducible from the seed	holds
17	ray to a cylinder 20 m ahead reads 15 m	exact
18	ray away from the cylinder reads maximum range	exact
19	sensor reports the roof height of the building it hits	exact
20	no horizontal hit above the roof	exact
21	downward ray reads the height above ground	exact
22–23	collision inside / no collision above a building	holds
24	lower bound equals min energy per metre × distance	10^{-6} J
25	lower bound is below any achievable straight-line energy	holds
26	DP optimum respects the rigorous lower bound	holds
27–28	dense-network parameter and input gradients	6.8×10^{-10}, 1.5×10^{-9}
29	SAC critic gradients	6.6×10^{-10}
30	SAC actor gradients (tanh + entropy, reparameterised)	6.6×10^{-9}
31	SAC temperature gradient	10^{-6}
32	TD3 actor gradients	1.5×10^{-8}
33–34	PPO policy and log-σ gradients	9.6×10^{-7}, 1.6×10^{-9}
35	GAE matches its explicit definition	10^{-10}

Gradient checks use central differences with $h = 10^{-6}$ on double-precision copies of the networks; training itself runs in single precision.

Table B2 Benchmark against published reference scores

Algorithm	Budget	This implementation	Stable-Baselines3 (published)
SAC	20 000 steps	**−123.8 ± 73.9**	−176.3 ± 101.6
TD3	20 000 steps	**−126.7 ± 75.3**	−196.0 ± 119.0
PPO	100 000 steps	**−136.7 ± 82.6**	−230.4 ± 142.5

Because the algorithms are written from their original formulations rather than taken from a library, they were run on Pendulum-v1 – whose dynamics are reproduced exactly as in the Gymnasium source – under the budgets and learning rate used for the published Stable-Baselines3 trained agents, and the deterministic policy was evaluated over 20 episodes. Each entry is a single seed, as is each published reference value.

All three match or exceed the published scores, so the implementations are not deficient relative to a widely used reference library. The PPO row is the least comparable: the published PPO agent uses generalised state-dependent exploration, which is not implemented here. This benchmark says nothing about the sample efficiency of these implementations on the delivery task of Sect. 4, where PPO fails;

it establishes only that the algorithms are correctly implemented on a task where the reference scores are known.

References

[1] Rodrigues TA, Patrikar J, Oliveira NL, Matthews HS, Scherer S, Samaras C (2022) Drone flight data reveal energy and greenhouse gas emissions savings for very small package delivery. *Patterns* **3**:100569. https://doi.org/10.1016/j.patter.2022.100569

[2] Zeng Y, Xu J, Zhang R (2019) Energy minimization for wireless communication with rotary-wing UAV. *IEEE Trans Wireless Commun* **18**(4):2329–2345. https://doi.org/10.1109/TWC.2019.2902559

[3] Gong H, Huang B, Jia B, Dai H (2022) Modelling power consumptions for multi-rotor UAVs. arXiv:2209.04128. https://arxiv.org/abs/2209.04128

[4] Rodrigues TA, Patrikar J, Choudhry A, Feldgoise J, Arcot V, Gahlaut A, Lau S, Moon B, Wagner B, Matthews HS, Scherer S, Samaras C (2021) In-flight positional and energy use data set of a DJI Matrice 100 quadcopter for small package delivery. *Sci Data* **8**:155. https://doi.org/10.1038/s41597-021-00930-x

[5] Stolaroff JK, Samaras C, O'Neill ER, Lubers A, Mitchell AS, Ceperley D (2018) Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery. *Nat Commun* **9**:409. https://doi.org/10.1038/s41467-017-02411-5

[6] Leishman JG (2006) *Principles of Helicopter Aerodynamics*, 2nd edn. Cambridge University Press, Cambridge

[7] Johnson W (2013) *Rotorcraft Aeromechanics*. Cambridge University Press, Cambridge

[8] Mellinger D, Kumar V (2011) Minimum snap trajectory generation and control for quadrotors. In: Proc IEEE ICRA, pp 2520–2525

[9] Richter C, Bry A, Roy N (2016) Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In: Robotics Research, STAR **114**, pp 649–666

[10] Bertsekas DP (2017) *Dynamic Programming and Optimal Control*, 4th edn. Athena Scientific, Belmont MA

[11] Mnih V, Kavukcuoglu K, Silver D, et al (2015) Human-level control through deep reinforcement learning. *Nature* **518**:529–533

[12] Schulman J, Wolski F, Dhariwal P, Radford A, Klimov O (2017) Proximal policy optimization algorithms. arXiv:1707.06347

[13] Schulman J, Moritz P, Levine S, Jordan M, Abbeel P (2016) High-dimensional continuous control using generalized advantage estimation. In: Proc ICLR. arXiv:1506.02438

[14] Haarnoja T, Zhou A, Abbeel P, Levine S (2018) Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. In: Proc ICML, PMLR **80**:1861–1870

[15] Haarnoja T, Zhou A, Hartikainen K, et al (2018) Soft actor-critic algorithms and applications. arXiv:1812.05905

[16] Kalidas AP, Joshua CJ, Md AQ, Basheer S, Mohan S, Sakri S (2023) Deep reinforcement learning for vision-based navigation of UAVs in avoiding stationary and mobile obstacles. *Drones* **7**:245

[17] Lei B, Hu W, Ren Z, Ji S (2025) DRL-based UAV autonomous navigation and obstacle avoidance with LiDAR and depth camera fusion. *Aerospace* **12**:848

[18] Yin Y, Wang Z, Zheng L, Su Q, Guo Y (2024) Autonomous UAV navigation with adaptive control based on deep reinforcement learning. *Electronics* **13**:2432

[19] Zhang Y, Yan C, Xiao J, Feroskhan M (2025) NPE-DRL: enhancing perception constrained obstacle avoidance with nonexpert policy guided reinforcement learning. *IEEE Trans Artif Intell* **6**:184–198

[20] Kaufmann E, Bauersfeld L, Loquercio A, Müller M, Koltun V, Scaramuzza D (2023) Champion-level drone racing using deep reinforcement learning. *Nature* **620**:982–987

[21] Bahi A, Ourici A (2026) Self-sustaining drone operations through deep reinforcement learning and piezoelectric energy harvesting. *Int J Intell Robot Appl*

[22] Omar S, Ma G (under review) Four arm-mounted piezoelectric harvesters on a quadrotor: a verified energy account and its consequences for energy-aware deep reinforcement learning navigation

[23] Omar S, Ma G (under review) Can an arm-mounted piezoelectric harvester pay for its own mass on a multirotor? A fully coupled finite-element design study

[24] Henderson P, Islam R, Bachman P, Pineau J, Precup D, Meger D (2018) Deep reinforcement learning that matters. In: Proc AAAI, pp 3207–3214

[25] Agarwal R, Schwarzer M, Castro PS, Courville A, Bellemare MG (2021) Deep reinforcement learning at the edge of the statistical precipice. In: NeurIPS **34**, pp 29304–29320

[26] Altman E (1999) *Constrained Markov Decision Processes*. Chapman and Hall/CRC, Boca Raton

[27] Achiam J, Held D, Tamar A, Abbeel P (2017) Constrained policy optimization. In: Proc ICML, PMLR **70**:22–31

[28] Ray A, Achiam J, Amodei D (2019) Benchmarking safe exploration in deep reinforcement learning. OpenAI technical report

[29] Stooke A, Achiam J, Abbeel P (2020) Responsive safety in reinforcement learning by PID Lagrangian methods. In: Proc ICML, PMLR **119**:9133–9143

[30] Lee T, Leok M, McClamroch NH (2010) Geometric tracking control of a quadrotor UAV on SE(3). In: Proc 49th IEEE Conference on Decision and Control, pp 5420–5425

[31] Hart PE, Nilsson NJ, Raphael B (1968) A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans Syst Sci Cybern* **4**(2):100–107

[32] Fujimoto S, van Hoof H, Meger D (2018) Addressing function approximation error in actor-critic methods. In: Proc ICML, PMLR **80**:1587–1596

[33] Raffin A, Hill A, Gleave A, Kanervisto A, Ernestus M, Dormann N (2021) Stable-Baselines3: reliable reinforcement learning implementations. *J Mach Learn Res* **22**(268):1–8

[34] Sutton RS, Barto AG (2018) *Reinforcement Learning: An Introduction*, 2nd edn. MIT Press, Cambridge MA