Journal of Advances in Developmental Research (IJAIDR)

E-ISSN: 0976-4844 ● Website: www.ijaidr.com ● Email: editor@ijaidr.com

Artificial Intelligence-Based Energy Management for Hybrid Electric Vehicles Integrated with Renewable Energy Sources

M Lakshminarayana

Department of Electrical Engineering
University College of Engineering, Osmania University, Hyderabad, India.
lakshminarayanauceou@gmail.com

Abstract:
Managing energy flow in a renewable-assisted hybrid electric vehicle requires coordination of intermittent photovoltaic generation, an engine–generator and battery storage. This study develops a reproducible Python framework for examining supervisory power allocation in a solar-assisted series hybrid electric vehicle. A one-step equivalent-energy heuristic generates reference generator commands, and a decision-tree regressor learns the mapping from electrical demand, battery state of charge and available photovoltaic power to generator output. An independent feasibility projection enforces generator, battery-power and state-of-charge constraints. The controller is compared with a thermostat baseline over a 1,800 s synthetic driving scenario spanning 5.880 km. Seven results describe the driving input, photovoltaic availability, source power sharing, battery state of charge, cumulative fuel consumption, energy accounting and demand satisfaction. The learned policy achieves a held-out teacher-imitation mean absolute error of 0.258 kW. Simulated fuel consumption is 0.828 L for the learned controller and 0.846 L for the baseline; however, their final states of charge differ, at 61.80% and 62.46%, respectively, preventing attribution of the raw fuel difference to a verified charge-sustaining efficiency improvement. Both cases satisfy the modeled demand within numerical precision. Host-based timing yields a median single-sample inference time of 41.60 microseconds across 1,000 calls. The study demonstrates a transparent energy-level simulation workflow and identifies the additional validation needed for a publication-grade controller comparison. Experimental validation, reinforcement-learning optimization, converter dynamics and battery ageing are outside the implemented scope.

Keywords: hybrid electric vehicle; photovoltaic integration; energy management; decision tree; imitation learning; power allocation; Python simulation.

1. Introduction

1.1 Background

Hybrid electric vehicles provide multiple energy paths through which propulsion demand can be supplied. Their supervisory controller determines how the engine and battery contribute as operating conditions change. The value of hybridization therefore depends partly on the quality of power allocation, alongside component sizing and drivetrain efficiency. Sciarretta and Guzzella [1] explain the distinction between

offline optimization benchmarks and implementable energy-management strategies. Paganelli et al. [2] formulate instantaneous power splitting through an equivalent fuel cost, while Lin et al. [3] demonstrate how dynamic-programming solutions can inform practical control rules.

Adding photovoltaic (PV) generation introduces an external energy input whose magnitude changes with irradiance. In an onboard solar-assisted architecture, available solar power reduces the instantaneous electrical demand that must be met by the generator and battery. This contribution depends on installation area and operating conditions. It should be quantified rather than assumed to provide a large share of propulsion energy. Renewable-integrated vehicle studies also need to distinguish onboard PV harvesting from renewable electricity used for charging at a stationary location.

Machine learning offers one way to represent a supervisory decision rule. A compact supervised model can approximate a specified reference policy without performing the original reference calculation at every operating point. This approach is useful only if the reference policy, training domain, approximation error and physical constraints are stated clearly. The present work adopts that limited and testable formulation.

1.2 Problem definition

The modeled vehicle has three electrical sources or storage paths: PV generation, an engine–generator and a battery. The controller must allocate generator and battery power while satisfying electrical demand and respecting state-of-charge (SOC) and power limits. Positive battery power represents discharge; negative battery power represents charging.

Three evaluation issues motivate the study. First, a learned command is not automatically physically feasible. Second, a low teacher-imitation error does not establish optimal fuel economy. Third, fuel comparisons can be misleading when controllers finish with different battery energy. These issues require explicit feasibility handling and energy accounting.

The research question is: Can a compact learned approximation of a stated one-step energy-management heuristic operate feasibly in a reproducible renewable-assisted HEV simulation, and what conclusions can be supported by its fuel, SOC and computational results?

1.3 Aim and contributions

The aim is to implement and assess a transparent supervisory-control workflow for a solar-assisted series HEV. The contributions of this simulation study are:

- A reproducible Python energy-level model integrating longitudinal power demand, time-varying PV availability, a generator and battery storage.
- A decision-tree approximation of an explicitly specified one-step equivalent-energy heuristic.
- A shared feasibility projection for both learned and baseline controllers, with numerical checks for power balance, battery bounds and unmet demand.
- Seven simulation outputs accompanied by fuel and terminal-SOC accounting, plus a measured host-computation benchmark.

These are implementation contributions. They do not establish a new globally optimal control algorithm or a previously unreported research gap.

2. Literature Review

2.1 Conventional energy management

Conventional supervisory strategies include threshold rules, equivalent consumption minimization and offline optimization. Threshold rules are straightforward to implement, but their behavior depends on selected switching thresholds and component operating points. Equivalent consumption minimization assigns an equivalent fuel cost to battery power, enabling an instantaneous comparison of engine and electrical energy paths [2]. The equivalence factor influences the trade-off between present fuel use and battery energy.

Dynamic programming can evaluate a known driving sequence subject to model constraints and terminal conditions. Lin et al. [3] use its results to extract implementable power-management rules. Such offline solutions are valuable reference cases, but knowledge of the complete future sequence is generally unavailable to an online controller [1].

The present teacher policy follows the general idea of evaluating generator actions through a combined fuel and battery cost. It uses an assumed SOC-dependent equivalence factor and a battery-power penalty. It is therefore described as an ECMS-inspired heuristic, rather than as an implementation of a calibrated or theoretically equivalent ECMS formulation.

2.2 Reinforcement learning for HEVs

Reinforcement learning (RL) treats energy management as sequential decision-making. Vehicle states describe the operating condition, actions determine power allocation, and a reward or cost measures fuel use and other objectives. Kong et al. [4] investigate RL-based real-time energy management and simulation validation. Zou et al. [5] combine recursive power-request statistics and RL for a hybrid tracked vehicle. Liu et al. [6] examine look-ahead energy management using velocity prediction and RL, including hardware-in-the-loop assessment.

General RL methods provide different tools for the action and learning problem. Deep Q-networks [7] use learned action values; deep deterministic policy gradients [8] address continuous-action control; proximal policy optimization [9] uses a policy-gradient formulation; and soft actor-critic [10] combines off-policy learning with an entropy-regularized objective. These papers establish algorithmic foundations, not performance evidence for the vehicle modeled here.

The implemented controller is supervised imitation, not RL. It learns from labels generated by a reference heuristic and does not optimize rewards through environment interaction. Ross et al. [11] show why sequential prediction can be affected by differences between training-state distributions and the states induced by a learned policy. Consequently, the holdout error reported here must be supplemented by closed-loop evaluation and future out-of-distribution tests.

2.3 Renewable-aware vehicle control

Renewable-aware vehicle studies operate at several system boundaries. Hu et al. [13] examine renewable integration alongside charging, on-road power management and powertrain optimization. Their renewable-supply setting should not be conflated with the onboard PV architecture used in this study. Pochont et al. [14] assess transient irradiance along an Indian highway route, illustrating the relevance of route-specific conditions to vehicle-integrated PV evaluation.

PV subsystem design also requires a distinction between available power and power successfully extracted by a converter. Esram and Chapman [15] compare maximum power point tracking techniques. De Soto et al. [16] describe a PV performance model incorporating irradiance and temperature effects. Here, available PV power is calculated with fixed conversion efficiencies; no electrical PV characteristic or MPPT algorithm is simulated.

Battery state is another important modeling boundary. Plett [17] addresses battery-management state estimation using extended Kalman filtering. The current simulation instead assumes exact knowledge of an energy-based SOC state. Sensor noise, SOC estimation error and electrochemical dynamics are not represented.

Taken together, these works motivate coordinated source allocation while showing that research validation requires attention to component physics, forecasting, measurement uncertainty and fair energy accounting. The present study is a simplified platform for examining these issues, rather than a complete treatment of them.

3. Materials and Methods

3.1 System architecture

The series architecture contains a PV source, an engine-driven generator, battery storage and an electric traction load. Engine power reaches the modeled electrical bus through the generator. A synthetic speed sequence determines longitudinal demand. The supervisory policy receives demand, SOC and available PV power, then requests generator output. A feasibility projection adjusts the dispatch to the modeled power and SOC limits.

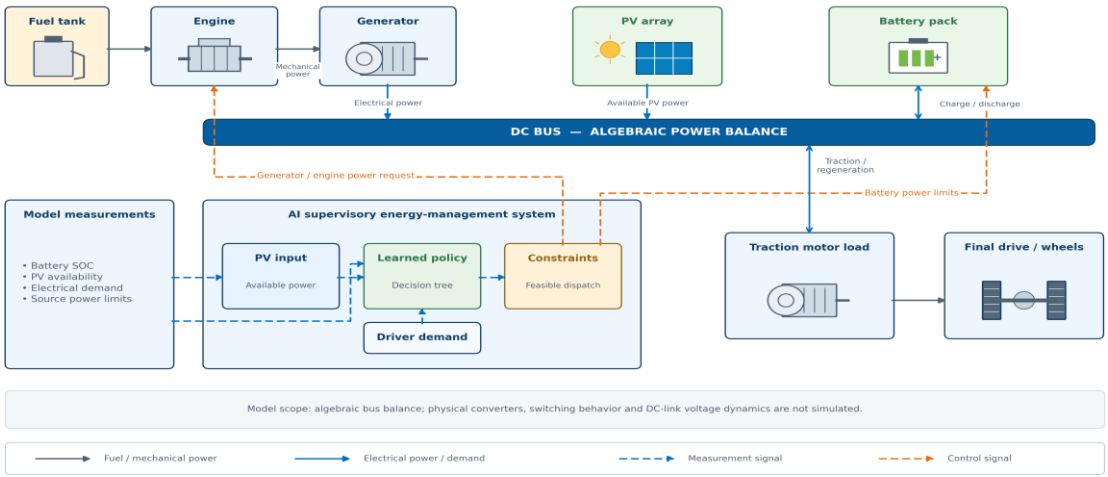


Figure 1: Architecture of the renewable energy-integrated series HEV. Solid arrows represent energy or demand relationships; dashed arrows represent computational signals. The DC bus is an algebraic power-balance abstraction. Physical converters, switching behavior and DC-link voltage dynamics are not simulated.

3.2 Vehicle demand model

The longitudinal force is calculated as:

$$F = ma + mgC_r + \frac{1}{2}\rho(C_d A)v^2$$

Here, road grade and wind are zero, while mass, rolling resistance and aerodynamic parameters are fixed. Wheel power in kW is:

$$P_w = Fv/1000$$

Electrical demand is:

$$P_d = \begin{cases} P_w/\eta_d + P_{aux}, & P_w \geq 0 \\ \eta_r P_w + P_{aux}, & P_w < 0 \end{cases}$$

The speed sequence is prescribed, so this is an inverse energy calculation rather than forward simulation of speed tracking. Acceleration is obtained numerically from that sequence. The regenerative factor maps negative wheel power to recovered electrical power at the bus; it is not a separately validated braking controller.

3.3 PV and generator models

Available PV power is:

$$P_{pv} = \frac{GA_{pv}\eta_{pv}\eta_c}{1000}$$

where irradiance is in W/m² and power is in kW. The synthetic irradiance input is:

$$G(t) = \text{clip}\left[780, +, 160\sin\left(\frac{t}{220}\right) - 430\exp\left(-\left(\frac{t-850}{130}\right)^2\right)01000\right]$$

This input produces a smooth solar variation with a localized irradiance reduction. Temperature, orientation and electrical partial-shading effects are excluded.

For positive generator output, assumed combined engine–generator efficiency is:

$$\eta_g(P_g) = 0.24 + 0.12\exp\left[-\left(\frac{P_g - 24}{16}\right)^2\right]$$

Fuel chemical power is $P_f = P_g/\eta_g(P_g)$, with $P_f = 0$when $P_g = 0$. Fuel-volume increment is:

$$\Delta V_f = \frac{P_f \Delta t}{3600 H_f}$$

where $H_f = 8.9$kWh/L is an assumed fuel lower heating value. The efficiency expression is illustrative and not a measured component map. Idle fuel, warm-up, minimum-on time, start penalties and generator ramp limits are excluded.

3.4 Battery dynamics and constraints

Battery power is the residual after PV and generator contributions. With $P_b > 0$for discharge, the energy-based SOC update is:

$$z_{k+1} = z_k - \frac{\Delta t}{3600 E_b}\begin{cases} P_b/\eta_b, & P_b \geq 0 \\ \eta_b P_b, & P_b < 0 \end{cases}$$

The model applies $0.30 \le z \le 0.85$, $-25 \le P_b \le 60$kW and $0 \le P_g \le 50$kW. Battery power limits are tightened when necessary to prevent the next SOC value from crossing a bound. The feasibility projection is applied to both controllers. Excess power is tracked as spill or rejected energy; any shortage is tracked as unmet demand.
The checked balance is:

$$P_{pv} + P_g + P_b - P_d - P_{spill} + P_{unserved} = 0$$

The spill variable combines surplus renewable energy and rejected regenerative energy; it should not be interpreted as a PV-only curtailment measurement.

3.5 Reference policy and learned controller

The teacher evaluates generator candidates $\{0, 2, 4, \ldots, 50\}$kW. For each candidate:

$$P_b = P_d - P_{pv} - P_g$$

Its one-step objective is:

$$J = P_f(P_g) + \lambda(z)P_b + \alpha P_b^2$$

where:

$$\lambda(z) = 3.1 + 22(0.60 - z),\ \alpha = 0.006\ \text{kW}^{-1}$$

The factor $\lambda$is dimensionless; all objective terms therefore have power units. Candidates violating battery-power or next-step SOC bounds are rejected. The minimum-cost feasible generator output labels the state sample. This is a one-step heuristic with manually selected coefficients and no terminal-SOC constraint.
A decision-tree regressor [12] learns the mapping $[P_d, z, P_{pv}] \mapsto P_g$. A fixed-seed generator produces 18,000 independent training-domain samples with demand in −20 to 65 kW, SOC in 0.31 to 0.84, and PV power in 0 to 0.65 kW. An 80:20 random split gives 14,400 training and 3,600 holdout samples. Maximum tree depth is 10 and minimum leaf sample count is 12. The fitted tree has 211 leaves. Its averaged leaf predictions can lie between the teacher's discrete generator actions.
The holdout test evaluates teacher approximation within the sampled domain. It does not establish generalization to new vehicle parameters or driving distributions. Online decisions use the fitted tree followed by the shared feasibility projection. Model parameters are fixed during the driving simulation.

3.6 Baseline, scenario and parameters

The baseline switches a nominal 22 kW generator on below SOC 0.55 and off above SOC 0.65. The generator starts in the off state. Between the thresholds, the previous on/off state is retained. Feasibility projection can alter the nominal command. This is an illustrative thermostat baseline, not a tuned production controller.
The synthetic driving profile interpolates speed knots at 30 s intervals. Knot speeds are sampled uniformly from 18 to 82 km/h using seed 7, with every sixth knot and the first and last knots set to zero. The simulation evaluates 1,800 one-second demand intervals. This sequence is not WLTP, UDDS or a measured drive cycle.

Parameter	Assumed value
Vehicle mass	1,650 kg
Rolling-resistance coefficient	0.012
Aerodynamic drag-area product	0.66 m^2
Air density	1.225 kg/m^3
Traction efficiency	0.91
Regenerative conversion factor	0.65
Auxiliary load	0.7 kW
Battery nominal energy	8 kWh
Battery charge/discharge efficiency	0.95 per direction
Initial SOC	60%
Permitted SOC	30–85%
Generator power	0–50 kW
Battery power	−25 to 60 kW
PV area	3 m^2
PV efficiency	22%
PV converter efficiency	96%
Duration / time step	1,800 s / 1 s

3.7 Software and assessment

NumPy [18] handles numerical arrays, scikit-learn [19]-[25] provides the tree and data split, and Matplotlib [20] generates the figures. The executed environment uses Python 3.12.14, NumPy 2.3.5, scikit-learn 1.8.0 and Matplotlib 3.10.8 on an x86-64 Linux host.

Evaluation comprises seven visual outputs, final fuel and SOC measurements, held-out teacher-imitation mean absolute error (MAE), and assertions for SOC bounds, power balance and negligible unmet demand. There is one fixed synthetic driving realization. No confidence intervals or statistical significance claims are made.

4. Results

4.1 Synthetic driving profile

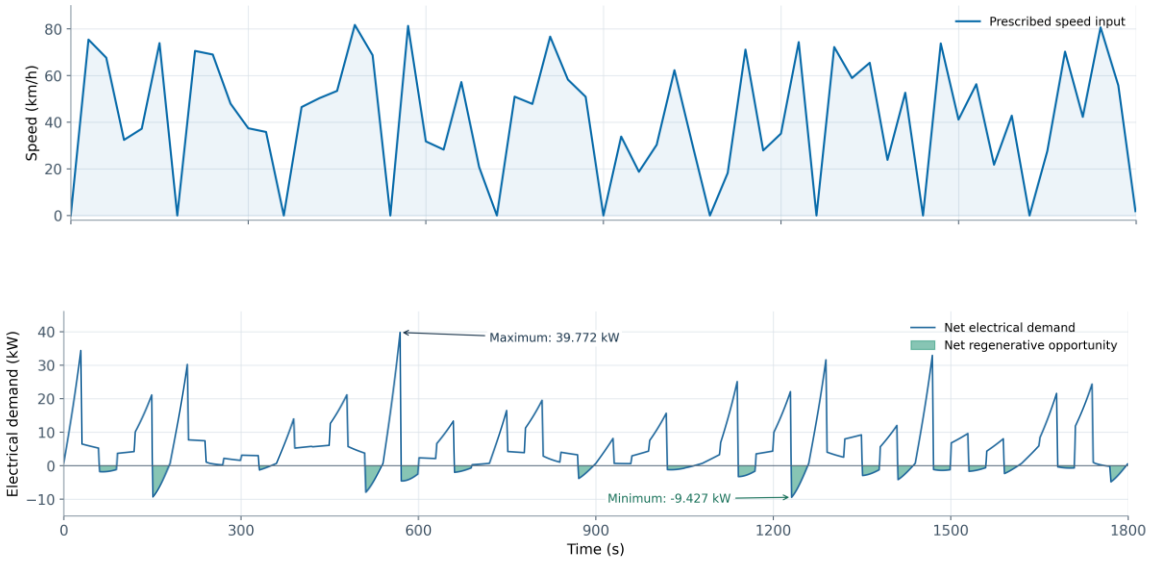


Figure 2: Prescribed speed input for the 1,800 s synthetic scenario. The evaluated profile covers 5.880 km and includes repeated low-speed and stopped periods. The corresponding electrical demand ranges from −9.427 to 39.772 kW. Negative demand represents a net regenerative opportunity after auxiliary demand. The plot characterizes the input; it does not demonstrate closed-loop speed tracking.

4.2 Available photovoltaic power

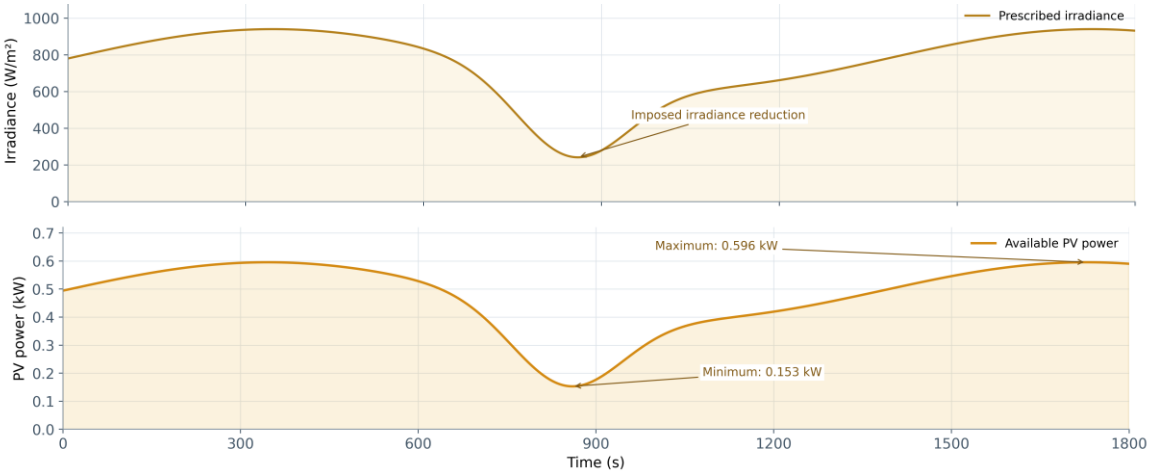


Figure 3: PV power available under the prescribed irradiance sequence. PV power ranges from approximately 0.153 to 0.596 kW. Available PV energy over the scenario is 0.238 kWh. Its peak is much smaller than the approximately 39.8 kW peak electrical demand, so PV is a supplementary source in this configuration. The localized reduction follows directly from the imposed irradiance input. No zero-PV ablation was run, so a separate causal fuel-saving contribution from PV has not been quantified.

4.3 AI-controlled power sharing

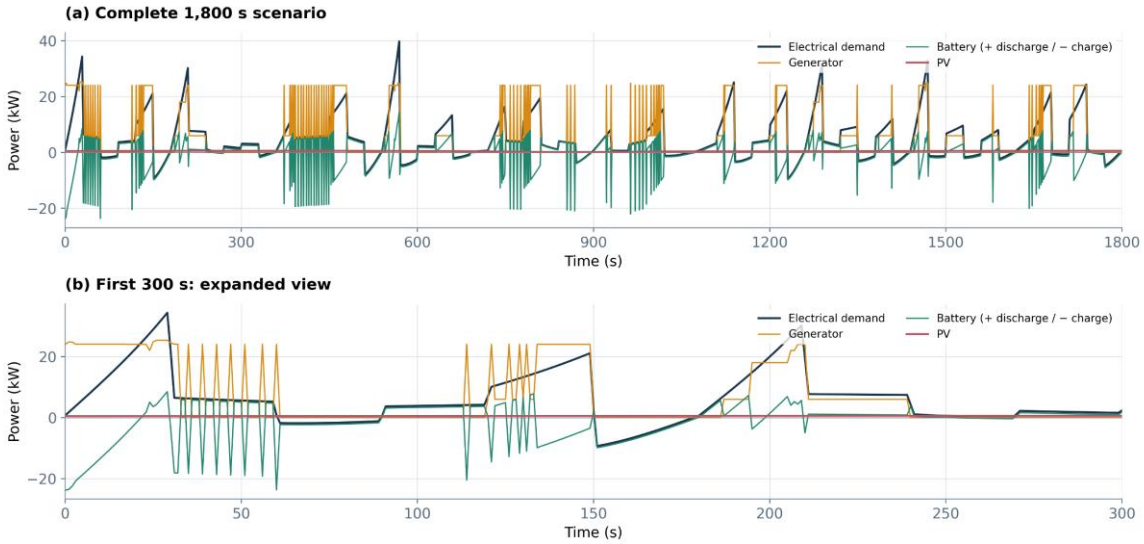


Figure 4: Electrical demand and generator, battery and PV contributions under the learned policy.

The learned-controller case uses generator output between 0 and 25.333 kW. Battery power ranges from −23.794 to 14.458 kW, within the specified limits. The battery supplies transient demand above the combined generator and PV output and absorbs surplus power during other intervals. The result demonstrates feasible allocation for this specific trajectory; it does not prove a globally optimal power split. Frequent generator transitions are possible because start penalties and minimum-on times are absent.

4.4 Battery state of charge

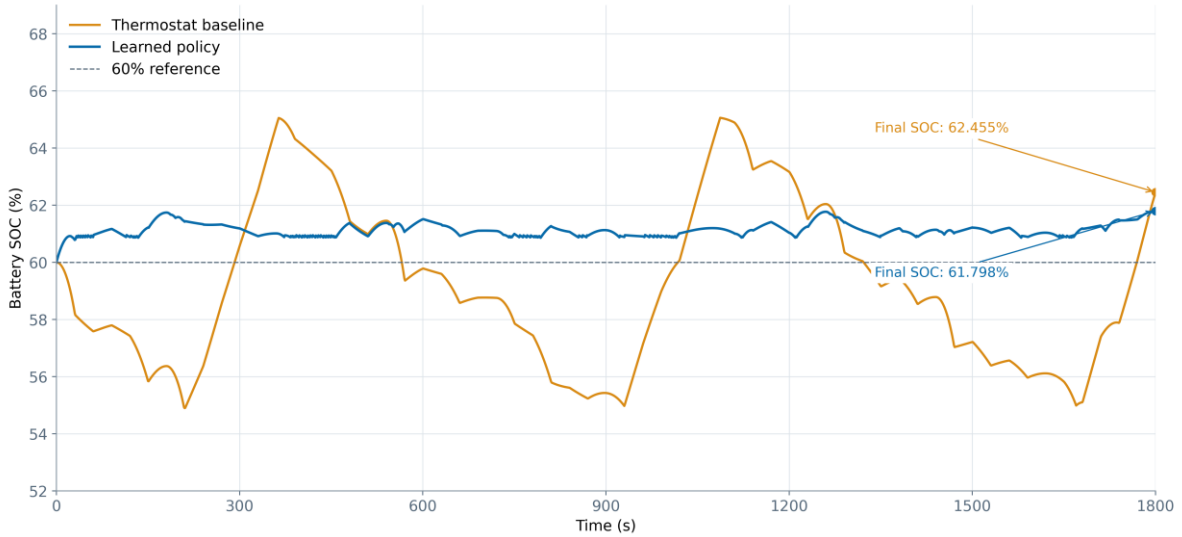


Figure 5: Battery SOC for the learned policy and thermostat baseline. Both cases start at 60% SOC. The learned policy remains close to its reference and ends at 61.798%. The baseline displays the intended thermostat cycling and ends at 62.455%. Its trajectory ranges approximately from 54.899% to 65.060%, while the learned policy remains approximately between 60.000% and 61.797% over the recorded interval-start states. All checked states, including the final state, satisfy the wider 30–85% safety bounds.

The narrower SOC excursion of the learned controller is an observation about this scenario and its selected penalty. It is not evidence of reduced battery degradation, which is not modeled.

4.5 Cumulative fuel consumption

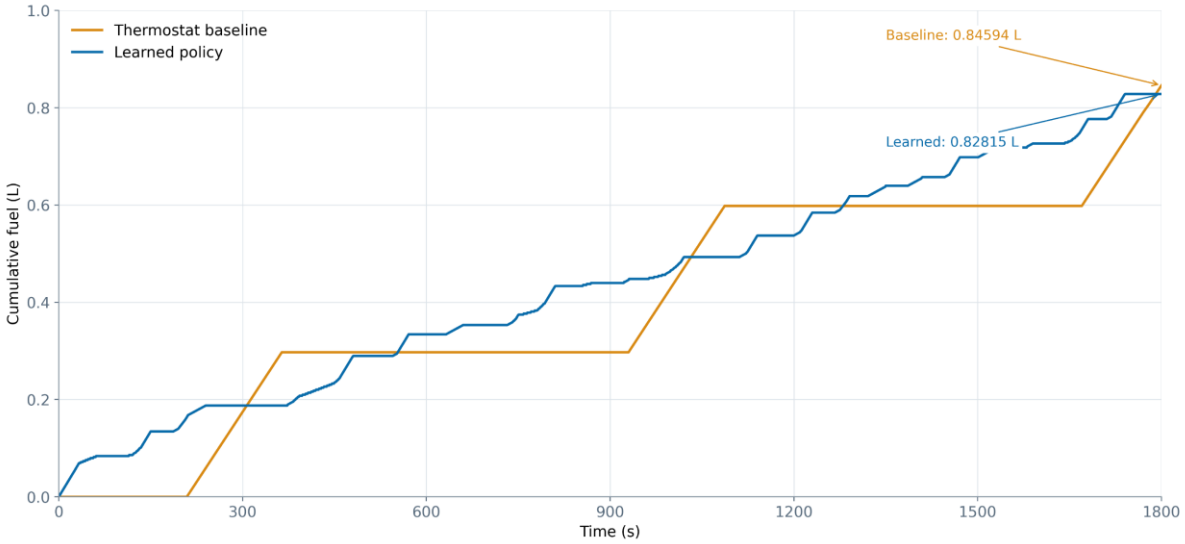


Figure 6: Simulated cumulative fuel consumption before terminal-SOC correction. Final fuel consumption is 0.82815 L for the learned policy and 0.84594 L for the baseline. The raw difference is 0.01779 L, or approximately 2.10% of baseline fuel. However, the baseline also stores more energy in the battery by the end of the run. Consequently, 2.10% is an uncorrected fuel difference, not a verified charge-sustaining fuel-economy improvement.

4.6 Energy accounting

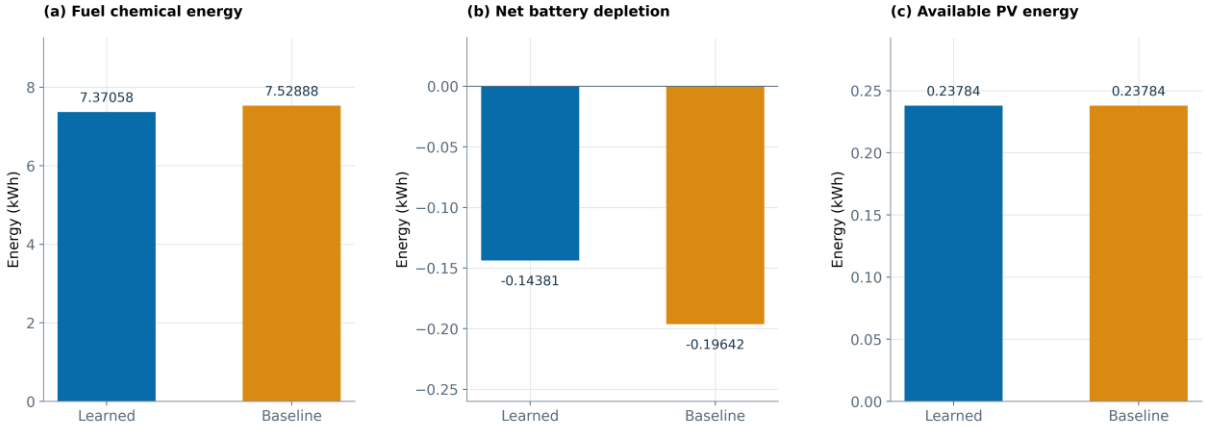


Figure 7: Fuel chemical energy, net battery depletion and available PV energy. These quantities belong to distinct energy domains and are shown separately. Net battery depletion is defined by:

$$\Delta E_b = E_b(z_0 - z_T)$$

The learned and baseline cases have values of −0.14381 and −0.19642 kWh, respectively. Negative depletion means net battery charging. The baseline therefore finishes with approximately 0.05261 kWh more stored battery energy than the learned case. The raw fuel difference corresponds to approximately

0.15830 kWh of fuel chemical energy using the assumed heating value. These two energy differences are not directly interchangeable because conversion losses separate fuel chemical energy from stored battery energy.

A fair charge-sustaining comparison would enforce a common terminal SOC or apply and justify an appropriate terminal-energy correction. Neither has been performed here.

4.7 Demand satisfaction and numerical consistency

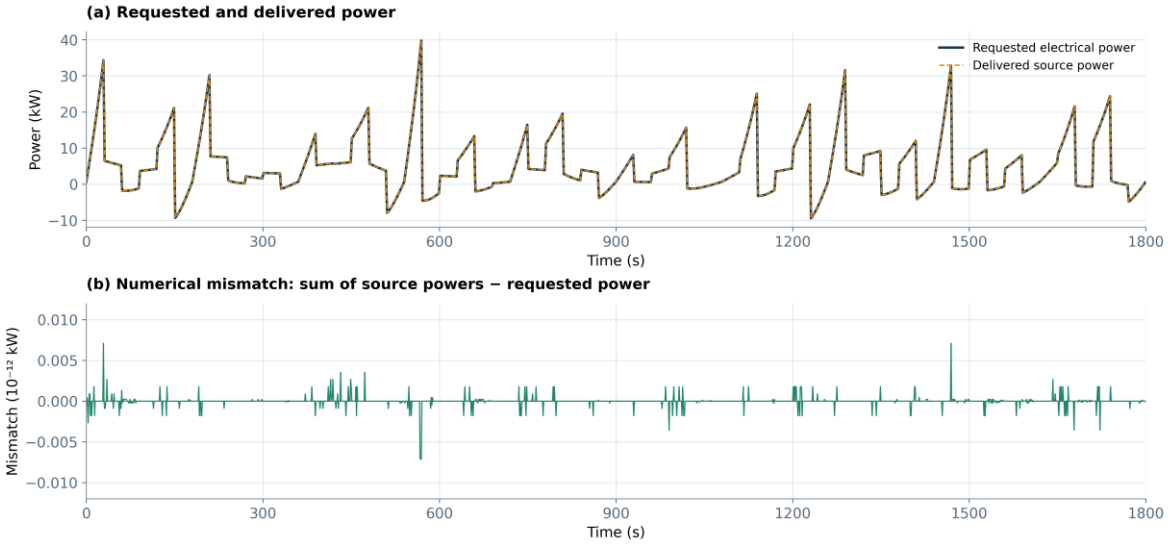


Figure 8: Requested and delivered electrical power under the learned policy.

Delivered electrical power overlaps the requested demand within numerical precision. The computed maximum balance residual for the learned-controller run is 0.0 kW in the implemented arithmetic. Integrated unmet energy is approximately 4.04×10^{-17}kWh for the learned policy and 7.67×10^{-17}kWh for the baseline, both negligible floating-point artifacts. SOC and unmet-demand assertions pass for both cases.

Since battery power is calculated as a constrained residual, balance satisfaction is an internal consistency check rather than independent experimental evidence of converter tracking or bus-voltage stability.

Metric	Learned controller	Thermostat baseline
Initial SOC	60.000%	60.000%
Final SOC	61.798%	62.455%
Fuel consumption	0.82815 L	0.84594 L
Net battery depletion	−0.14381 kWh	−0.19642 kWh
Available PV energy	0.23784 kWh	0.23784 kWh
Significant unmet demand	None	None
Teacher-imitation holdout MAE	0.25803 kW	Not applicable

4.8 Computational performance

A separate Python benchmark repeats the full training pipeline five times and evaluates 1,000 single-sample predictions after an initial warm-up. The training pipeline includes synthetic sample generation, teacher labeling, splitting, fitting and holdout evaluation. Timing uses the host's high-resolution performance counter. These measurements are exploratory host timings and do not characterize an embedded ECU.

Computational quantity	Measured value
Training-pipeline median, five runs	0.09145 s
Training-pipeline observed range	0.08682–0.19365 s
Median single-sample prediction, 1,000 calls	41.60 µs
95th-percentile prediction time	164.25 µs
Maximum observed prediction time	1,210.32 µs
Vectorized teacher labeling, 1,000 states	0.98420 ms
Tree maximum depth	10
Tree leaves	211
Holdout MAE, 3,600 samples	0.25803 kW

Tree traversal requires at most ten split decisions per prediction in this fitted model, excluding library overhead. The teacher evaluates 26 generator candidates per state. However, the reported vectorized teacher timing and single-sample tree timing use different calling patterns; they cannot establish a computational speedup. Measured inference latency is small relative to the chosen one-second supervisory interval, but there is no worst-case execution guarantee. I/O, scheduling, embedded hardware, communication and physical converter control are excluded.

5. Discussion

The study demonstrates that a small learned model can reproduce a specified supervisory heuristic with low holdout error and can be combined with a separate constraint-handling layer. This separation makes the evaluation easier to interpret: imitation error describes agreement with the teacher, while the projection and physical constraints determine feasible dispatch.

The learned controller keeps SOC closer to its selected 60% reference than the thermostat baseline in this one scenario. That behavior follows from the teacher's SOC-sensitive equivalence factor and battery-power penalty. It should not be described as a generally superior control strategy without additional comparisons. In particular, the baseline has deliberately simple thresholds and a fixed nominal generator output, whereas the learned controller imitates a state-dependent generator choice.

The raw fuel reduction must be interpreted alongside terminal energy. Both strategies finish above their initial SOC, and the baseline stores more energy. A matched-SOC experiment might change the apparent fuel advantage. A rigorous follow-up should either impose the same terminal condition or evaluate enough repeated operation to reach comparable charge-sustaining behavior. Calibrated fuel and battery models are needed before translating the difference into a vehicle-level fuel-economy claim.

The renewable contribution is limited by the assumed 3 m² PV area and operating conditions. Available PV energy is reported, but it is not equivalent to independently measured renewable utilization. The current spill variable does not separate curtailed PV from rejected regenerative energy. Separate source-resolved energy counters and zero-PV control cases would be needed to isolate PV utilization and benefit.

Several model simplifications materially affect interpretation. A one-second energy model cannot evaluate DC-link ripple, semiconductor switching, harmonic distortion or converter stability. Exact SOC knowledge omits a practical estimation problem [17]. Fixed efficiency assumptions omit temperature and ageing. A prescribed synthetic drive sequence does not test acceleration tracking, traction limits, varying grade, wind or actuator lag. The lack of engine start penalties may also favor switching behavior that is undesirable in a real vehicle.

Future validation should use standard and measured cycles, multiple initial SOC values, several irradiance conditions and independently held-out routes. Comparisons should include the teacher itself, a tuned rule controller, a charge-constrained offline benchmark and, if implemented, RL methods trained and evaluated under the same conditions. RL claims would require actual reward-based training, learning curves and multiple-seed reporting. Ablations should isolate the roles of PV awareness, SOC feedback, the battery-power penalty and constraint projection.

The present timing measurements support only a host-based feasibility observation. Target-hardware profiling, worst-case timing analysis and hardware-in-the-loop experiments are required before claiming real-time deployment. Similarly, a journal submission would require a stronger novelty argument and broader validation than this proof-of-concept implementation currently provides.

6. Conclusion

A Python simulation framework was implemented for supervisory energy management in a solar-assisted series hybrid electric vehicle. An ECMS-inspired one-step heuristic supplied training labels for a decision-tree controller, and a common feasibility projection enforced battery, generator and SOC limits. Seven outputs characterized the drive input, available solar power, source allocation, SOC, fuel consumption, energy accounting and demand satisfaction.

Over a 1,800 s synthetic scenario, the learned controller achieved 0.258 kW holdout teacher-imitation MAE and satisfied the modeled demand within numerical precision. Fuel consumption was 0.828 L, compared with 0.846 L for the thermostat baseline. Final SOC values differed, so the raw 2.10% fuel difference cannot be treated as a verified efficiency improvement. Host timing gave a median prediction latency of 41.60 µs, without establishing an embedded real-time guarantee.

The main outcome is a reproducible and auditable implementation for studying energy allocation. Further work must introduce calibrated component maps, matched terminal battery energy, diverse driving and irradiance conditions, stronger baselines and hardware validation before drawing publication-level conclusions about superiority or practical vehicle performance.

REFERENCES:

[1] A. Sciarretta and L. Guzzella, “Control of hybrid electric vehicles,” *IEEE Control Syst. Mag.*, vol. 27, no. 2, pp. 60–70, Apr. 2007, doi: 10.1109/MCS.2007.338280.

[2] G. Paganelli, G. Ercole, A. Brahma, Y. Guezennec, and G. Rizzoni, "General supervisory control policy for the energy optimization of charge-sustaining hybrid electric vehicles," *JSAE Rev.*, vol. 22, no. 4, pp. 511–518, Oct. 2001, doi: 10.1016/S0389-4304(01)00138-2.

[3] C.-C. Lin, H. Peng, J. W. Grizzle, and J.-M. Kang, "Power management strategy for a parallel hybrid electric truck," *IEEE Trans. Control Syst. Technol.*, vol. 11, no. 6, pp. 839–849, Nov. 2003, doi: 10.1109/TCST.2003.815606.

[4] Z. Kong, Y. Zou, and T. Liu, "Implementation of real-time energy management strategy based on reinforcement learning for hybrid electric vehicles and simulation validation," *PLOS ONE*, vol. 12, no. 7, Art. no. e0180491, Jul. 2017, doi: 10.1371/journal.pone.0180491.

[5] Y. Zou, T. Liu, D. Liu, and F. Sun, "Reinforcement learning-based real-time energy management for a hybrid tracked vehicle," *Appl. Energy*, vol. 171, pp. 372–382, Jun. 2016, doi: 10.1016/j.apenergy.2016.03.082.

[6] T. Liu, X. Hu, S. E. Li, and D. Cao, "Reinforcement learning optimized look-ahead energy management of a parallel hybrid electric vehicle," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 4, pp. 1497–1507, Aug. 2017, doi: 10.1109/TMECH.2017.2707338.

[7] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015, doi: 10.1038/nature14236.

[8] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," 2015, *arXiv*:1509.02971, doi: 10.48550/arXiv.1509.02971.

[9] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv*:1707.06347, doi: 10.48550/arXiv.1707.06347.

[10] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. 35th Int. Conf. Mach. Learn.*, vol. 80, 2018, pp. 1861–1870. [Online]. Available: https://proceedings.mlr.press/v80/haarnoja18b.html

[11] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, vol. 15, 2011, pp. 627–635. [Online]. Available: https://proceedings.mlr.press/v15/ross11a.html

[12] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Wadsworth International Group, 1984.

[13] X. Hu, Y. Zou, and Y. Yang, "Greener plug-in hybrid electric vehicles incorporating renewable energy and rapid system optimization," *Energy*, vol. 111, pp. 971–980, Sep. 2016, doi: 10.1016/j.energy.2016.06.037.

[14] N. R. Pochont, Y. R. Sekhar, M. C. Brito, and V. T. Rao, "An empirical assessment of transient irradiance to foster vehicle-integrated photovoltaics technology in highway routes–An Indian case study," *Energy*, vol. 328, Art. no. 136503, Aug. 2025, doi: 10.1016/j.energy.2025.136503.

[15] T. Esram and P. L. Chapman, "Comparison of photovoltaic array maximum power point tracking techniques," *IEEE Trans. Energy Convers.*, vol. 22, no. 2, pp. 439–449, Jun. 2007, doi: 10.1109/TEC.2006.874230.

[16] W. De Soto, S. A. Klein, and W. A. Beckman, "Improvement and validation of a model for photovoltaic array performance," *Sol. Energy*, vol. 80, no. 1, pp. 78–88, Jan. 2006, doi: 10.1016/j.solener.2005.06.010.

[17] G. L. Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1. Background," *J. Power Sources*, vol. 134, no. 2, pp. 252–261, Aug. 2004, doi: 10.1016/j.jpowsour.2004.02.031.

[18] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, Sep. 2020, doi: 10.1038/s41586-020-2649-2.

[19] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011. [Online]. Available: https://jmlr.org/papers/v12/pedregosa11a.html

[20] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, May/Jun. 2007, doi: 10.1109/MCSE.2007.55.

[21] J. R. Babu, M. R. Nayak, and B. Mangu, "A peer review of hybrid electric vehicle based on step-up multi-input DC-DC converter and renewable energy source," *J. Phys.: Conf. Ser.*, vol. 2089, no. 1, Art. no. 012041, 2021, doi: 10.1088/1742-6596/2089/1/012041.

[22] J. R. Babu, M. R. Nayak, and B. Mangu, "Renewable energy applications and a multi-input DC-DC converter for hybrid electric vehicle applications using Matlab/Simulink," in *Proc. Innovations Power Adv. Comput. Technol. (i-PACT)*, 2021, pp. 1–8, doi: 10.1109/i-PACT52855.2021.9696863.

[23] J. R. Babu, M. R. Nayak, and B. Mangu, "Development and application of an energy management system for electric vehicles integrated with multi-input DC-DC bidirectional buck-boost converter," *Int. J. Electr. Electron. Res.*, vol. 11, no. 2, pp. 457–464, Jun. 2023, doi: 10.37391/IJEER.110228.

[24] J. R. Babu, M. R. Nayak, and B. Mangu, "Design and control of a tricycle with a hybrid electric motor cooling system powered by solar photovoltaics," *Int. J. Electr. Electron. Res.*, vol. 11, no. 2, pp. 465–472, Jun. 2023, doi: 10.37391/IJEER.110229.

[25] J. R. Babu, M. V. D. S. K. Murty, D. Rajashekar, and N. Susheela, "ANN-based design of a hybrid renewable energy system for hybrid electric vehicle applications," *E3S Web Conf.*, vol. 616, Art. no. 01004, 2025, doi: 10.1051/e3sconf/202561601004.