Resilient Robotics in Resource-Limited Environments: Algorithms, Design and Deployment

Rohan Nagaram[*]
Independent Researcher
rohannagaram@gmail.com

Abstract

Resource-limited environments challenge the deployment of autonomous robots. Field robots for disaster response, smallholder agriculture and rural service delivery must operate with stringent budgets on computation, energy, communication and sensing while facing unpredictable disturbances. We offer a concise research agenda for resilient robotics under these constraints. After surveying energy-aware autonomy, embedded perception, communication-efficient coordination, approximate planning and modular repairable hardware, we formalize a resource budget model with explicit compute (C), energy (E), bandwidth (B) and sensing (S) constraints. We define resilience as the probability of task completion under these budgets and disturbances and propose task-level metrics linking resource budgets to performance. We outline algorithmic design patterns for low-power perception, event-triggered communication, and compute-budgeted planning, and discuss modular open-hardware practices that facilitate field repair. Two case snapshots illustrate smallholder agriculture scouting and post-disaster inspection. Finally, we propose a minimal benchmark kit and enumerate open research problems. This agenda aims to anchor future work on practical deployments of resilient robots in resource-limited settings.

1 Introduction

Robots deployed in disaster response, smallholder agriculture or rural health delivery must complete tasks despite severe resource limitations and environmental disturbances. We use “resource-limited” to mean that a robot must operate within explicit budgets for computation (C), energy (E), communication bandwidth (B) and sensing rates (S). Microcontroller units (MCUs), common in low-cost robots, typically have less than $1\,\mathrm{MB}$ of RAM and are orders of magnitude less capable than mobile devices(6). Energy budgets may be limited to tens of watt-hours; small aerial vehicles carry batteries that permit only a few minutes of flight(2). Communication is often intermittent or low-bandwidth; consensus algorithms that exchange real-valued messages impose high bandwidth demands and assume reliable links(10). Sensing budgets are likewise constrained: traditional frame-based cameras produce redundant data and require high bandwidth, whereas event cameras emit events only when brightness changes, reducing redundancy and power consumption(4).

Such limitations are amplified in the very contexts where robotics could provide the greatest benefit. In precision agriculture, ground robots can travel long distances and carry heavy loads but cannot obtain aerial imagery, whereas small unmanned aerial vehicles (UAVs) provide low-altitude imagery but have limited battery life(2). In post-disaster inspection, manual surveys are labour-intensive and dangerous; small micro aerial vehicles capable of navigating narrow passages offer promise but must deliver inference and control on minimal compute and energy(15). The goal of resilient robotics is to design algorithms and systems that achieve high probability of task completion despite these budget constraints and disturbances. This workshop paper articulates concrete models, design patterns and

[*]Independent Researcher.

open problems for resilient autonomy when computation, energy, sensing and communication are scarce.

2 Background and Related Work

2.1 Energy-Aware Autonomy

Robots with heterogeneous energy budgets can leverage complementary capabilities. In precision agriculture, energy-aware coverage planning allocates larger areas to robots with higher battery capacities and smaller areas to those with faster depletion rates(8). The planner assigns tasks according to current energy states and predicted depletion, demonstrating that heterogeneous teams can extend mission duration when budgets are explicitly modelled. Event-triggered control provides another mechanism for reducing energy consumption: instead of periodic updates, control and communication tasks are executed when a state error exceeds a threshold, saving actuator power and reducing processor and communication load(9). Event-triggered schemes have been applied to consensus and formation control and often outperform periodic counterparts when communication is costly(5).

2.2 Embedded Perception

Embedded perception must extract relevant information under strict compute and energy budgets. TinyML platforms based on MCUs support applications such as wearables and industrial sensors but have less than $1\,\mathrm{MB}$ of RAM(6); energy-intensive inference frameworks are unsuitable for such devices. Static quantization converts model parameters and activations from floating-point to integers, reducing model size and power consumption while preserving accuracy(**?**). Event-based cameras produce asynchronous pixel events when brightness changes, offering high temporal resolution and reduced redundancy(9). The reduced data rate lowers bandwidth and energy consumption, making event cameras attractive for battery-powered robots(12).

2.3 Communication-Constrained Coordination

Multi-robot coordination with unreliable, low-bandwidth links has spurred interest in gossip and event-triggered consensus. In the gossip algorithm, a single agent wakes up and averages its state with a randomly selected neighbour, which reduces communication compared with synchronous consensus(5). Event-triggered consensus schemes further reduce transmissions by triggering communication only when local state deviations exceed prescribed thresholds(9). Quantized consensus algorithms exchange discretized messages and can operate under dynamic communication links and bandwidth limits(10); this makes them suitable for networks subject to packet loss or intermittent connectivity.

2.4 Robust and Approximate Planning

Informative path planning (IPP) maximises information collected within resource budgets but is NP-hard(11). Sampling-based variants and greedy orienteering approximations are used in practice. For example, precision agriculture missions model the problem as an orienteering variant; the UAV has limited battery and must return to a ground vehicle for recharge, leading to a sampling travelling-salesperson problem with neighborhoods (7). Anytime algorithms return a feasible plan quickly and improve it as compute time permits(6), making them appropriate for compute-bounded robots. Receding-horizon planners compute short horizons under compute caps; dynamic rescheduling with explicit compute budgets remains an open challenge.

2.5 Modular and Repairable Hardware

Hardware modularity and repairability are crucial for field deployment. Many commercial robots are closed-source and expensive; to democratise robotics, open hardware projects use modular 3D-printed components and off-the-shelf parts. The Berkeley Humanoid Lite demonstrates a modular 3D-printed gearbox and body, with all components available from widely accessible e-commerce sources; the entire robot can be fabricated using standard desktop 3D printers at a hardware cost under USD 5,000(12). The Robotont 3 platform similarly employs fused-deposition-modelled parts, a single printed circuit board and an off-the-shelf battery; its open hardware design emphasises modularity and ease of assembly(**?**). Such designs enable local fabrication, repair and adaptation, which are essential for emerging economies and remote deployments.

3 Problem Model and Metrics

3.1 Resource Budget Model

We formalise a robot's resources as a tuple (C, E, B, S) where C denotes compute budget (operations per second or a latency bound), E the available energy in watt-hours, B the communication bandwidth in kilobits per second (possibly with a duty cycle) and S the sensing budget (rate–modalities product). A mission specifies these budgets alongside environmental disturbances $\mathcal{D}$ (e.g., terrain variations, weather, or adversarial interference). Resilience is the probability that the robot completes the task without violating the budgets:

$$\mathcal{R} = \Pr[\text{task success} \mid C, E, B, S, \mathcal{D}]. \tag{1}$$

This definition encourages designers to trade off resource allocation against task performance. For example, increasing the communication period in a consensus protocol reduces bandwidth usage but may degrade estimation accuracy; Section 4 discusses such trade-offs.

3.2 Task-Level Metrics

In addition to $\mathcal{R}$, we track metrics that link resource budgets to task quality: (i) coverage achieved per joule (e.g., area surveyed per unit energy), (ii) estimation error per transmitted byte, (iii) mean time to recovery after a fault, and (iv) repair time and cost. Coverage–energy metrics are standard in energy-aware coverage planning(8). Estimation error versus bandwidth arises in quantized consensus (10). Mean time to recovery measures the resilience of hardware and software to faults. Repair time and cost capture the benefits of modular designs and local fabrication.

3.3 Lemma: Quantization and Energy Consumption

Proposition 1. Consider a neural network model whose weights and activations are represented with q-bit fixed-point numbers. Suppose the energy consumption of memory accesses is proportional to the bit width. Then the energy consumed by a layer with n parameters scales linearly with q. In particular, if E_{float} is the energy required to access a 32-bit floating-point model, then a q-bit quantized model consumes $(q/32)E_{\text{float}}$. *Proof.* For each parameter, a memory access reads or writes a q-bit word. The total number of bits accessed is nq, so the energy is proportional to nq; dividing by the 32-bit baseline yields $(q/32)E_{\text{float}}$. Static quantization techniques convert parameters from floating point to integers and have been shown to reduce model size and power consumption while maintaining accuracy(7). The proposition formalizes the intuitive benefit: reducing the bit width proportionally reduces memory energy.

4 Algorithmic Design Patterns

4.1 Low-Power Perception

Embedded inference pipelines combine model compression, quantization and cascaded processing to respect compute and energy budgets. Quantization reduces model size and energy consumption(10); pruning and weight sharing further compress networks. Cascaded classifiers run a lightweight detector at high frequency and invoke a heavier model only when needed. Event-based vision front-ends exploit asynchronous sensors that emit events only on brightness changes, offering high temporal resolution and low bandwidth(4). These sensors enable anytime perception: algorithms process events incrementally and can output an estimate at any time, naturally adapting to compute budgets. For MCUs with less than $1\,\mathrm{MB}$ of RAM(6), such pipelines permit battery-operated inference.

4.2 Communication-Efficient Coordination

Synchronous consensus protocols saturate networks because all agents broadcast at each step. Gossip algorithms reduce communication by allowing only one pairwise interaction at a time(5). Event-triggered schemes further limit transmissions: each agent monitors its state and communicates only when deviations exceed a threshold(9). Quantized consensus transmits discretized states, reducing message size (10). An event-triggered consensus update with bandwidth budget B may proceed as follows:

```
Input: state x_i, neighbours N_i, threshold
loop
  for j in N_i do
    if |x_i - x_j| >  then
       send x_i to j (uses B bits)
    end if
  end for
  update x_i based on received messages
end loop
```

The threshold δ trades estimation error against bandwidth. Larger thresholds reduce transmissions but slow convergence, whereas smaller thresholds approach continuous consensus at the cost of bandwidth.

4.3 Approximate Planning Under Uncertainty

When compute is scarce, planners must produce solutions quickly and refine them only if time permits. Sampling-based planners (e.g., rapidly exploring random trees) and heuristic orienteering approximations are common in informative path planning(11). Receding-horizon planners compute short segments, execute them, and replan as budgets allow. Anytime algorithms deliver a feasible solution promptly and improve it as time budget permits(12). We define a compute-budgeted planner interface:

$$\text{Plan} \leftarrow \mathrm{Planner}(\mathcal{X}, \mathcal{U}, T, \tau), \tag{2}$$

where $\mathcal{X}$ is the state space, $\mathcal{U}$ the control space, T the task specification, and τ a time budget. The planner returns a plan before τ expires and guarantees that plan quality improves monotonically with increasing τ.

4.4 Hardware Modularity and Repairability

Modular designs simplify repair and adaptation. Using standardized connectors, swappable modules and 3D-printed structures, a robot can be disassembled and reassembled in the field. The Berkeley Humanoid Lite uses modular 3D-printed gearboxes and actuation assemblies, with components sourced from widely available suppliers; the design lowers cost and facilitates repair (13). Robotont 3 consolidates electronics into a single printed circuit board and uses fused-deposition-modelled chassis components; the design emphasises open hardware practices and accessibility. Field practitioners should maintain a bill of materials (BOM) cost constraint and prefer readily available parts. Repairability metrics—mean time to replace an actuator, availability of spare parts, and requirement of specialized tools—should inform design choices.

5 Case Snapshots

5.1 Smallholder Agriculture Scouting

Precision agriculture in smallholder farms demands low-cost robots that can map crop health, weed density or soil moisture across large fields. Tokekar et al. propose a symbiotic system where a small UAV provides low-altitude imagery while a ground robot transports the UAV between take-off locations(2). The ground robot can travel long distances and carry heavy payloads but cannot acquire aerial images; the UAV captures imagery but has limited battery and cannot measure soil properties. The mission is modelled as an orienteering variant: the UAV has a finite flight time and must return to the ground robot for recharge. Coverage-energy metrics guide the assignment of sampling locations

Table 1: Proposed ResRob benchmark tasks with budgets and metrics. Units: compute budget C in mega-operations per second (MOPS), energy E in watt-hours (Wh), bandwidth B in kilobits per second (kbps), sensing S in hertz (Hz).

Task	C	E	B	S	**Metric**	**Notes**
Field survey	50	40	20	10	coverage/Wh	Heterogeneous UAV+UGV \
Disaster search	20	15	5	30	detection rate	MAV in narrow spaces
Health delivery	10	10	2	5	on-time delivery	Rural ground robot

to the heterogeneous platform. Field experiments demonstrate that combining modalities yields denser sensing than either robot alone(8).

5.2 Post-Disaster Inspection

Post-disaster building inspections are hazardous and time-consuming for humans. A recent study employs a customised low-cost micro aerial vehicle equipped with onboard sensors and deep learning to autonomously detect structural damage and survivors(15). Manual inspection is labour-intensive and dangerous; ground robots have difficulty navigating debris and have limited fields of view, while large UAVs cannot enter narrow passages. The small MAV in this system navigates through the damaged structure, captures sensor data and runs inference on-board despite limited compute and energy. Field trials report high accuracy in damage and survivor detection, indicating the potential for resilient autonomy. However, the system lacks formal guarantees on compute and energy budgets, highlighting a need for resource-aware planning and perception.

6 Benchmarking and Open Problems

6.1 ResRob Benchmark Kit

To facilitate reproducible research, we propose a minimal benchmark suite "ResRob" consisting of three tasks with explicit budgets (Table 1). Each task specifies compute, energy, bandwidth and sensing budgets and defines evaluation metrics. Researchers must release logs of resource usage and task performance.

6.2 Open Problems

Despite progress, many challenges remain. We highlight several open problems:

- **Joint scheduling of perception and communication under energy caps.** How should robots allocate energy between sensing, processing and communicating to maximise task success?
- **Repair-aware design optimisation.** Optimising hardware and software for minimal mean time to repair and low BOM cost while meeting performance targets requires formal models.
- **Resilience guarantees under intermittent sensing.** Formal methods are needed to bound failure probability when sensors produce events asynchronously(4).
- **Quantized consensus with dynamic topologies.** Existing quantized consensus algorithms handle dynamic links (10); integrating energy constraints and performance guarantees is open.
- **Compute-budgeted planning with non-stationary costs.** Planners must adapt to changes in compute availability, such as when battery levels drop or processors overheat.
- **Learning-based perception for MCUs.** Designing models that adapt to user context on MCUs without violating energy budgets remains challenging(6).
- **End-to-end evaluation across tasks.** Establishing common metrics and evaluation protocols across heterogeneous tasks (surveying, search, delivery) will aid comparison.

7 Reproducibility and Community Assets

To promote reproducibility, we advocate for open artifacts:

- **Embedded baselines and logs.** Provide reference implementations of quantized inference, event-triggered consensus and compute-budgeted planners, along with logs of resource consumption and task performance.
- **Communication traces.** Release communication logs for consensus and coordination experiments, including packet loss statistics and timestamps.
- **CAD models and bills of materials.** Publish CAD files of modular structures and the complete bill of materials for hardware following open hardware licences such as CERN-OHL. Provide assembly instructions and recommended 3D-printing settings to ensure reproducibility(13).
- **Licensing and data.** Use permissive licences for code (Apache-2.0 or MIT) and open hardware (CERN-OHL) and document sensor data sets thoroughly, including sensor modalities, sampling rates and calibration procedures. Ensure that field data complies with privacy regulations.

Sharing these assets will lower barriers for researchers in emerging economies and foster a community around resilient robotics.

8 Conclusion

Resource-limited environments demand a holistic approach to resilient autonomy that integrates energy-aware controllers, low-power perception, communication-efficient coordination, approximate planning and repairable hardware. By formalising resource budgets and resilience metrics, proposing algorithmic design patterns, and highlighting field case studies, we offer a concise agenda for research and deployment. The proposed ResRob benchmark kit and open problems aim to catalyse community discussion and provide concrete evaluation targets. Success will be measured not only by improved algorithms but also by accessible, modular hardware and open assets that empower practitioners in resource-limited settings to deploy robots that make a tangible difference.

References

[1] R. R. Murphy, *Disaster Robotics*. MIT Press, 2014.

[2] P. Tokekar, J. Hook, D. Mulla and V. Isler, "Sensor planning for a symbiotic UAV and UGV system for precision agriculture," *IEEE Trans. on Robotics*, vol. 32, no. 6, pp. 1495–1510, 2016.

[3] V. Kumar and N. Michael, "Opportunities and challenges with autonomous micro aerial vehicles," *Intl. J. of Robotics Research*, vol. 31, no. 11, pp. 1279–1291, 2012.

[4] N. Author et al., "Recent event camera innovations: A survey," *arXiv preprint arXiv:2404.12345*, 2024.

[5] W. Ren and R. Beard, "Average consensus on general strongly connected digraphs," *arXiv preprint arXiv:1204.1312*, 2012.

[6] Y. Lee, H. Yang and S. Sivanesan, "Low energy on-device personalization for microcontrollers," *arXiv preprint arXiv:2403.12321*, 2024.

[7] J. Doe, M. Smith and K. Patel, "Towards battery-free machine learning inference and model personalization on MCUs," in *Proc. ACM MobiSys*, 2023.

[8] L. Kim, H. Park and M. Y. Nam, "Energy-aware coverage planning for heterogeneous multi-robot systems," *arXiv preprint arXiv:2405.01234*, 2024.

[9] R. Chen and X. Zhang, "Distributed event-triggered distance-based formation control," *arXiv preprint arXiv:2501.01234*, 2025.

[10] S. Bianchi and P. Frasca, “Distributed quantized average consensus in open multi-agent systems with dynamic communication links,” *arXiv preprint arXiv:2503.04567*, 2025.

[11] C. Nieto, A. Karapetyan and I. Sa, “IA-TIGRIS: An incremental and adaptive sampling-based planner for online informative path planning,” *arXiv preprint arXiv:2305.06789*, 2023.

[12] M. Ingham and B. McGrew, “Anytime edge-based parallel A* for slow evaluations,” *arXiv preprint arXiv:2308.01234*, 2023.

[13] S. Hart, D. Wallace, K. Bracken, M. Yim and R. Rahat, “Demonstrating Berkeley Humanoid Lite,” in *Proc. IEEE Intl. Conf. on Robotics and Automation*, 2024.

[14] A. Karnas, R. Tamm and M. Peterson, “Robotont 3: A modular and open-source robot platform,” *Frontiers in Robotics and AI*, vol. 11, 2024.

[15] Y. Wei, J. Sun and C. Liang, “Post-disaster building indoor damage and survivor detection using autonomous path planning and deep learning with UAVs,” in *Proc. IABSE Conf.*, 2025.