



[Home](#) [Licensing](#) [Patents](#) [Articles](#)

## Robotic Capability Assessment Before Commitment

by [Nick Clark](#) | Published March 27, 2026 | [PDF](#)

A warehouse robot accepts a picking task, navigates to the location, discovers the item weighs more than its gripper can handle, and must abort. A delivery drone accepts a route, launches, discovers headwinds exceed its range capability, and must divert. In both cases, the system committed to a task it could not complete because it lacked a structural mechanism for assessing capability before commitment. Capability awareness solves this through capability envelopes that evaluate the joint condition of physical capability, temporal feasibility, and uncertainty in a single pre-commitment check.

---

### The commitment problem in robotics

Robotic task assignment today is a matching problem: match available robots to pending tasks based on proximity, type, and availability. The assignment assumes the robot can complete the task. But capability is not binary. A robot's ability to complete a task depends on its current physical state, the

environmental conditions it will encounter, the time available, and the uncertainty in all of these factors.

Current systems check individual constraints: is the robot's payload capacity sufficient? Is the destination reachable? Is the battery charged? Each check passes individually, but the joint condition may fail. A robot with sufficient payload capacity and sufficient battery may lack sufficient battery for the payload at the required distance. The individual checks pass. The task fails.

## Why task-matching is not capability assessment

Task-matching systems assign based on static capability specifications: maximum payload, maximum range, supported task types. These specifications describe the robot's ideal capability, not its current capability. A robot with a worn gripper has reduced payload capability. A robot with a degrading battery has reduced range. A robot operating in high humidity has reduced sensor accuracy. None of these current-state factors appear in the static specification.

Temporal feasibility is rarely evaluated structurally. A task that is feasible now may not be feasible in ten minutes if the robot's battery is declining. A task that requires thirty minutes of operation is only feasible if the robot's capability remains adequate for the full duration, not just the start.

## How capability awareness addresses this

Capability awareness treats capability as a first-class state variable that the robot maintains and evaluates continuously. The capability envelope defines the boundary of what the robot can accomplish given its current physical state, environmental conditions, and uncertainty bounds. Before accepting a task, the robot evaluates the task requirements against its current capability envelope, not its static specification.

Temporal executability forecasting projects whether the capability envelope will remain adequate for the duration of the task. A robot with declining battery capacity evaluates not just current range but projected range at every stage of the task. If the projection indicates capability will drop below task requirements before completion, the task is declined before commitment.

The joint condition of capability, time, and uncertainty is evaluated as a single pre-commitment check. The robot does not check payload, then range, then time independently. It evaluates whether it can carry this payload over this distance within this time given its current uncertainty about environmental conditions. This joint evaluation catches failures that sequential individual checks miss.

Envelope negotiation enables multi-robot task decomposition. A task that exceeds one robot's capability envelope may be decomposable into sub-tasks that each fall within a different robot's envelope. The robots negotiate task decomposition through their capability declarations, dividing work based on structural capability rather than arbitrary assignment.

## What implementation looks like

A warehouse deploying capability awareness equips each robot with a real-time capability envelope that reflects its current physical state. The task assignment system queries the robot's envelope before assignment rather than consulting a static specification database. Robots that cannot demonstrate current capability for the task are not assigned.

For drone delivery operators, capability awareness prevents launch commitments that will result in diversions. Each drone evaluates its capability envelope against the complete delivery route, including projected wind conditions, payload weight, and return energy requirements. Only routes within the current capability envelope are accepted.

For construction robotics, capability awareness enables autonomous equipment to assess whether site conditions match their operational requirements before beginning work, preventing costly mobilization failures and reducing safety incidents from equipment operating outside its capability bounds.

[Capability Awareness All 21 steps →](#)

Know what you can do before you try.

Primary Technical Disclosure

[◦ Capability-, Time-, and Uncertainty-Aware Execution in Autonomous Computational Networks](#)

Secondary Technical

[◦ Capability as First-Class Computational State](#)[◦ Capability Envelope for Substrates](#)[◦ Temporal Executability Forecasting](#)[◦ Uncertainty as First-Class Propagated Variable](#)[◦ Capability Envelope Negotiation](#)[◦ Capability Genealogy Tracking](#)[◦ Biological Capability Extension](#)[◦ Network-Level Capability Pressure](#)[◦ Capability-Permission Distinction](#)[◦ Capability-Native Computation](#)[◦ Execution Synthesis and Non-Synthesis](#)[◦ Agent Behavior Under Constraints](#)[◦ Predictive Network Planning](#)[◦ Multi-Agent Contention Resolution](#)[◦ Capability Robustness Mechanisms](#)[◦ Capability-Modulated Discovery](#)[◦ Capability as Confidence Input](#)[◦ Embodied Capability Envelopes](#)[◦ Substrate Resource Negotiation](#)

Applications (General)

[● Robotic Capability Assessment Before Commitment](#)[◦ Edge Computing Resource Governance Through Capability Envelopes](#)[◦ Capability Awareness for Surgical Robotics](#)[◦ Capability Awareness for Agricultural Robotics](#)[◦ Capability Awareness for Mining Operations](#)[◦ Capability Awareness for Offshore Energy Operations](#)[◦ Capability Awareness for Warehouse Logistics Robotics](#)[◦ Capability Awareness for Construction Robotics](#)

Applications (Specific)

[◦ Tesla FSD Does Not Know What It Cannot Do](#)[◦ John Deere's Autonomous Tractors Cannot Assess Their Own Limits](#)[◦ KUKA Robots Execute Without Knowing Their Envelope](#)[◦ FANUC Robots Have No Adaptive Capability Envelope](#)[◦ Universal Robots Cobots Execute Without Knowing Their Limits](#)[◦ ABB Robots Perform Without Self-Assessing Capability](#)[◦ Yaskawa Motoman Robots Move Without Tracking Capability Drift](#)[◦ Doosan Cobots Collaborate Without Capability Self-Knowledge](#)[◦ Agility Robotics' Digit Walks Without Knowing What It Can Carry](#)[◦ Figure's Humanoid Learns Tasks Without Knowing Its Envelope](#)

[Capability Awareness overview →](#)

AQ

deterministic

autonomy

Legal

Subject to one or more pending U.S. and international patent applications, see [Patents](#) for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see [Licensing](#). Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See [Legal](#) for terms and conditions.

Adaptive Query™ is a trademark of Nicholas Clark. U.S. federal registration is pending. federal registration. AQ™, AQ Inside™, Adaptive Index™, Adaptive Network™, Semantic Agent™, @AQ™, AQID™, and Adaptive Coin™ are used as trademarks in connection with the Adaptive Query platform and brand. Other names may be trademarks of their respective owners.

Platform operated by Adaptive Query LLC, which provides patent and trademark licensing services. Copyright © 2025-2026 Nicholas Clark. All rights reserved.

Last updated: 2026-03-03



- [Inventive Steps](#)
- [Licensing](#)
- [Patents](#)
- [Articles](#)
- [Legal](#)
- [Opportunities](#)
- [Sitemap](#)



- 
- [nick@qu3ry.net](mailto:nick@qu3ry.net)
- 72 28 14 36 01



[Invented by Nick Clark](#) | Founding Investors: Devin Wilkie