

How to Add a Safety Governor That Pauses an Autonomous Robot's Actions

You have a robot that can move, grip, dose, or steer, and you need a layer that can permit, throttle, or pause a committed action the instant confidence drops, then default to a safe idle when it cannot justify moving at all. This guide describes an architectural approach for that layer: a confidence-governed actuation stage that gates every physical action through an admissibility check and records the decision. It is an architecture disclosed in U.S. Provisional Application No. 64/049,409, not a shipping library. The home inventive step is the Confidence-Governed Actuation inventive step.

What You Are Building

You are building a governor: a stage that sits between your robot's planner and its actuators, so that no physical action reaches a motor, valve, brake, or manipulator without first being evaluated. When the evaluation says conditions are good, the action executes normally. When confidence is marginal, the action executes in a reduced form. When confidence is insufficient, the action does not execute at all and the robot holds a safe idle. Every one of those outcomes is written to a durable record.

This is the problem a searcher means by "how do I add a safety governor that pauses an autonomous robot's actions." The naive version of that request is an emergency stop. The version you actually want is a graduated decision that can pause, but can also

permit at reduced magnitude, defer, or ask a human, and that leaves an auditable trail of why. That is what the disclosed architecture provides.

Who needs this: teams shipping mobile robots, manipulator arms, autonomous vehicles, dosing or dispensing devices, or any actuated effector where "just run the plan" is not safe when the world stops matching the plan's assumptions.

Why the Obvious Approaches Fall Short

The common approaches are real and useful, but each leaves a structural gap.

An emergency stop or hardware interlock is binary: the actuator either runs or is cut. That is correct for a guard door or a light curtain, but it has no middle. A robot that can only choose between full speed and dead stop cannot slow down, cannot finish a reversible sub-step, and cannot degrade gracefully when confidence is merely low rather than zero.

A confidence threshold bolted onto the planner also tends to be binary: above the line, act; below it, do not. It rarely accounts for the consequence of the specific action, whether that action is reversible, or whether the sensor evidence behind the decision was itself trustworthy. And when it fires, it usually leaves nothing behind to reconstruct the decision.

Functional-safety interlocks and e-stops, accurately described, provide the cut but no budget or expiration semantics on overrides, and limited audit of what tripped and why. They are typically domain-specific to one machine class.

The structural gap common to all three: the decision to act is treated as a fixed algorithm over raw sensor readings, terminating at the actuator command, with no graduated response, no reversibility awareness, and no recorded provenance. The disclosed architecture closes that gap by making every actuation a governed decision with a spectrum of outcomes and a lineage record.

The Architecture

The disclosed approach, in the terms of the filing, is a confidence-governed execution primitive: every physical actuation of the environment by an autonomous, semi-autonomous, or operator-assisted unit is evaluated through a composite admissibility evaluator before execution, and the result selects among a graduated set of actuation modes rather than a binary permit-or-deny.

The core pieces:

An actuation chain, not a call. The filing structures execution as an ordered chain: consume the governed observations relevant to the action context; generate a proposed actuation (which specifies an actuator, a command, and parameters); evaluate that proposed actuation jointly with the surrounding evidence; select a mode; record the selected mode and the evaluation inputs to lineage before the actuator moves; command the actuator at the selected mode; observe the effect; verify the effect against what was expected; broadcast the executed action; and record the outcome. The action is a governed step, and every step is written down.

A composite admissibility evaluator gates the action. Rather than thresholding a single confidence number, the proposed actuation is evaluated jointly against consumed observations and further cognitive inputs described elsewhere in the filing (a dispositional field, forecasting observations, and a capability envelope). The point for your build is that admissibility is a composite of consequence, evidence quality, and capability, not a lone scalar.

A graduated actuation mode selector is where "pause" lives. The filing enumerates a plurality of modes and states the mapping from admissibility to mode is continuous and bounded, not binary. The enumerated modes include, among others: a disabled mode where the actuator does not execute and the non-execution is recorded; a simulated (dry-run) mode; an advisory mode that records what would have been done without doing it; a consultative mode that asks a human or higher authority and waits;

a partial mode at fractional magnitude, reduced rate, or reduced scope; a constrained mode subject to added limits; a stage-gated mode that executes in stages with re-evaluation between stages; a deferred mode; and a full mode at nominal execution. As admissibility rises the selector moves toward more autonomous modes; as it falls, toward less autonomous ones. Your "pause" is the disabled mode; your "slow down" is partial or constrained; your "ask first" is consultative.

A safe idle is the default when readiness is insufficient. The filing states that a harm-minimization path exceeding governance-defined risk thresholds proceeds in the disabled mode with the disabled decision recorded and the agent transitioned to a confidence-degraded operating mode. In the broader confidence-governor discussion, loss of governance-credentialed evidence reduces execution readiness, and the governor restores readiness only as corroborating observations accumulate. The design intent is: when the governor cannot justify motion, it does not move, and it says so in the record.

Reversibility shapes the decision. A reversibility-aware commitment-point evaluator classifies each proposed action (reversible, partially reversible, irreversible, time-bounded, and so on), identifies the commitment point beyond which the action can no longer be undone, elevates thresholds for irreversible actions, and prefers reversible paths and late commitment points among admissible candidates. Practically, this lets a staged action be interrupted before its irreversible stage when new evidence lowers admissibility.

Overrides are budgeted, not free. An emergency-preemption mechanism lets an authority-credentialed observation override ordinary thresholds and compel elevated execution, but subject to a preemption budget and an expiration. The filing distinguishes this from a bare interlock precisely because the override is credentialed, rate-limited, time-bounded, and recorded.

Verification closes the loop. After execution, a post-actuation verification mechanism compares observed effects against an expected-effect specification, classifies any discrepancy, and feeds the result back: repeated clean verifications can lower thresholds, repeated discrepant ones raise them or reduce the actuator's published capability.

Lineage is not optional. The filing records every actuation evaluation, mode selection, preemption event, commitment-point transit, harm-minimization selection, and verification outcome in a lineage field. That record is what makes each decision reconstructible after the fact.

How to Approach the Build

Implement this yourself, in this order.

1. **Interpose a governor between planner and drivers.** No actuator call should originate anywhere but the governor. Give every proposed action an explicit shape: `{ actuator, command, parameters }`. This one refactor is what makes the rest possible.
2. **Model the modes as an ordered enum, not a boolean.** Start with a faithful subset: `disabled`, `partial`, `constrained`, `consultative`, `full`. The selector's job is to map an admissibility result onto one of these.
3. **Build the admissibility check as a composite.** Even a first version should combine more than one factor: the consequence class of the actuator, the quality or freshness of the evidence behind the action, and current capability. Return a determination, not just a pass or fail.
4. **Set per-actuator thresholds by consequence.** A high-consequence effector (a brake in traffic, a dosing pump, a life-safety actuator) should require elevated admissibility across multiple factors to reach full mode; a low-consequence effector

(an indicator light) needs only baseline. The filing describes threshold sets defined per actuator class.

5. **Make the safe idle the default outcome.** When admissibility is below the lowest execution threshold, select `disabled`, hold the safe state, mark the unit confidence-degraded, and record the non-execution. Insufficient readiness must resolve to not moving, by construction, not by a caught exception.
6. **Add reversibility as a modifier.** Tag each actuator or command with a reversibility class. For irreversible actions, raise the threshold; for staged actions, define the commitment point and allow interruption before it.

An illustrative interface sketch (not shipping code; faithful to the chain above):

```
# ILLUSTRATIVE ONLY. You implement this.
def govern(proposed, evidence):
    a = admissibility(proposed, evidence)          # composite, not a scalar
    a = apply_reversibility(a, proposed)          # raise bar for irreversi
    mode = select_mode(a, thresholds[proposed.actuator])
    record_lineage(proposed, evidence, a, mode)    # BEFORE the actuator mov
    if mode == DISABLED:
        hold_safe_idle(); return
    effect = drive(proposed, mode)                 # partial/constrained/ful
    verify(effect, expected(proposed), mode)       # feeds back into thresho
```

7. **Record before you move.** Write the selected mode and the evaluation inputs to lineage prior to commanding the actuator, so a pause or an execution is reconstructible even if the actuator step fails.
8. **Close the loop with verification.** Predict the expected effect, observe the actual effect, and let sustained discrepancy raise thresholds or shrink the actuator's capability envelope.

9. **Add budgeted preemption last.** Only after the ordinary path is solid, add credentialed overrides with a rate budget and an expiration, and record every preemption.

What This Does Not Give You

This is an architecture, not a drop-in library. There is no package to install and nothing here "just works." You implement the evaluator, the mode selector, the thresholds, the lineage store, and the actuator drivers yourself, against your own hardware and your own safety case.

It is not benchmarked or productized here, and no performance or safety numbers are claimed; the filing describes the mechanisms, not measured outcomes. The admissibility evaluator, dispositional field, forecasting, and capability envelope are described in the filing at an architectural level, so quality depends entirely on how you realize them. The governor gates decisions; it does not replace the hardware interlocks, functional-safety measures, or certification your domain requires, and it does not by itself make an unsafe planner or unsafe sensors safe. If your effector is genuinely binary and single-purpose, a plain interlock may be the right tool and this layer may be more than you need.

Disclosure Scope

The confidence-governed actuation approach described here, including the graduated actuation modes, the composite admissibility gate, the safe-idle default under insufficient readiness, reversibility-aware commitment points, budgeted emergency preemption, post-actuation verification, and the lineage record of each decision, is disclosed in U.S. Provisional Application No. 64/049,409. This guide is educational: it explains an architecture you can build yourself. It is not a warranty, a safety certification, or an offer of software, and nothing here guarantees any particular safety outcome in your deployment.

Confidence-Governed Actuation (</govern> [All 40 steps →](#) </inventive-steps>)

ned-actuation

Eleven graduated modes. Preemption budget. Reversibility-aware commitment. Harm-minimization.

Provisional application

PRIMARY TECHNICAL DISCLOSURE

- [Confidence-Governed Actuation: Graduated Modes for Physical Systems](/articles/confidence-governed-actuation-graduated-modes-for-physical-systems) (</articles/confidence-governed-actuation-graduated-modes-for-physical-systems>)

SECONDARY TECHNICAL

- [The Graduated Actuation Mode Selector](/articles/governed-actuation/graduated-mode-set) (</articles/governed-actuation/graduated-mode-set>)
- [Preemption Budget for Rate-Limited Override](/articles/governed-actuation/preemption-budget) (</articles/governed-actuation/preemption-budget>)
- [Reversibility Classifier](/articles/governed-actuation/reversibility-classifier) (</articles/governed-actuation/reversibility-classifier>)
- [Stage-Gated Commitment for Irreversible Actions](/articles/governed-actuation/staged-commitment) (</articles/governed-actuation/staged-commitment>)
- [Harm-Minimization Deviation](/articles/governed-actuation/harm-minimization-deviation) (</articles/governed-actuation/harm-minimization-deviation>)
- [Policy-Configurable Harm Ordering](/articles/governed-actuation/policy-configurable-harm-ordering) (</articles/governed-actuation/policy-configurable-harm-ordering>)
- [Post-Actuation Verification](/articles/governed-actuation/post-actuation-verification) (</articles/governed-actuation/post-actuation-verification>)
- [Discrepancy Classification Taxonomy](/articles/governed-actuation/discrepancy-classification) (</articles/governed-actuation/discrepancy-classification>)
- [Actuation State as Mesh-Broadcast Observation](/articles/governed-actuation/actuation-state-broadcast) (</articles/governed-actuation/actuation-state-broadcast>)
- [Cross-Unit Coordination Through Actuation-State Broadcast](/articles/governed-actuation/cross-unit-coordination) (</articles/governed-actuation/cross-unit-coordination>)
- [Lineage-Recorded Provenance for Every Actuation](/articles/governed-actuation/lineage-recorded-provenance) (</articles/governed-actuation/lineage-recorded-provenance>)

APPLICATIONS · GENERAL

- [Autonomous Aviation Execution Under Governed Actuation](/articles/governed-actuation/autonomous-aviation-execution) (</articles/governed-actuation/autonomous-aviation-execution>)
- [Compliant Autonomous Defense Engagement: Governing the Kill Chain With Admissibility-Gated Actuation](/articles/governed-actuation/autonomous-defense-execution) (</articles/governed-actuation/autonomous-defense-execution>)

- [Autonomous Surgical Execution Under Governed Actuation \(/articles/governed-actuation/autonomous-surgical-execution\)](/articles/governed-actuation/autonomous-surgical-execution).
- [FDA PCCP for Surgical Robots: An Architecture for Change-Control Eligibility \(/articles/governed-actuation/fda-pccp-surgical-pathway\)](/articles/governed-actuation/fda-pccp-surgical-pathway).
- [Auditable Harm-Minimization for Autonomous Vehicles: Declared Harm Ordering, Not Emergent Heuristics \(/articles/governed-actuation/harm-minimization-autonomous-driving\)](/articles/governed-actuation/harm-minimization-autonomous-driving).
- [Certifying Autonomous Aircraft Under ICAO: Structural Evidence for Phase, Authority, and Sense-and-Avoid \(/articles/governed-actuation/icao-autonomous-aviation\)](/articles/governed-actuation/icao-autonomous-aviation).
- [Cobot Safety Compliance Under ISO 10218 and ISO/TS 15066: Architecting Auditable Collaborative-Robot Mode Decomposition \(/articles/governed-actuation/industrial-collab-robotics-iso\)](/articles/governed-actuation/industrial-collab-robotics-iso)
- [Preemption Budget for Multi-Vehicle Fleet Operations \(/articles/governed-actuation/preemption-budget-fleet-operations\)](/articles/governed-actuation/preemption-budget-fleet-operations).
- [14 CFR Part 107 Commercial Drone Operations \(/articles/governed-actuation/cfr-part-107-drones\)](/articles/governed-actuation/cfr-part-107-drones).
- [Enforcing 14 CFR Part 135 Operational Control in Software for Commuter, Charter, and eVTOL Air Carriers \(/articles/governed-actuation/cfr-part-135-commercial-air\)](/articles/governed-actuation/cfr-part-135-commercial-air).
- [Certifying Adaptive Airborne Software Under DO-178C with Governed Actuation \(/articles/governed-actuation/do-178c-airborne-software\)](/articles/governed-actuation/do-178c-airborne-software)
- [FDA 510\(k\) and De Novo Pathway for Autonomous Medical Devices \(/articles/governed-actuation/fda-510k-de-novo\)](/articles/governed-actuation/fda-510k-de-novo).
- [Meeting IEC 61508 Safe-State and Graduated-Response Requirements with Governed Actuation \(/articles/governed-actuation/iec-61508-functional-safety\)](/articles/governed-actuation/iec-61508-functional-safety)
- [ISO 14971 Risk Management for Autonomous Medical Devices \(/articles/governed-actuation/iso-14971-medical-risk\)](/articles/governed-actuation/iso-14971-medical-risk).
- [ISO/SAE 21434 Compliance: Binding Automotive Cybersecurity to Runtime Control \(/articles/governed-actuation/iso-21434-automotive-cyber\)](/articles/governed-actuation/iso-21434-automotive-cyber).
- [Meeting ISO 21448 SOTIF for Autonomous Driving with Governed Actuation \(/articles/governed-actuation/iso-21448-sotif\)](/articles/governed-actuation/iso-21448-sotif)
- [ISO 26262 Functional Safety for Autonomous Driving \(/articles/governed-actuation/iso-26262-functional-safety\)](/articles/governed-actuation/iso-26262-functional-safety).
- [NHTSA ADS-DV Requirements and AV STEP \(/articles/governed-actuation/nhtsa-ads-dv-requirements\)](/articles/governed-actuation/nhtsa-ads-dv-requirements)

APPLICATIONS · SPECIFIC

- [Aurora Driver vs Governed Actuation for Reversibility-Aware Execution \(/articles/governed-actuation/aurora-self-driving\)](/articles/governed-actuation/aurora-self-driving).

- [Governed Surgical Autonomy Beyond Intuitive da Vinci \(/articles/governed-actuation/intuitive-davinci\)](/articles/governed-actuation/intuitive-davinci).
- [Waymo Driver vs Governed Actuation: The Externally Verifiable Commitment Gate \(/articles/governed-actuation/waymo-driver\)](/articles/governed-actuation/waymo-driver).
- [CMR Versius and Governed Actuation for Partial Surgical Autonomy \(/articles/governed-actuation/cmr-versius\)](/articles/governed-actuation/cmr-versius).
- [Cruise Robotaxi vs Governed Actuation: A Stage-Gated, Reversibility-Aware Substrate \(/articles/governed-actuation/cruise-self-driving\)](/articles/governed-actuation/cruise-self-driving)
- [Kodiak Robotics vs Governed Actuation for Autonomous Trucking \(/articles/governed-actuation/kodiak-self-driving-trucking\)](/articles/governed-actuation/kodiak-self-driving-trucking).
- [Mobileye SuperVision vs Governed Actuation: The Tier-Transition Gap \(/articles/governed-actuation/mobileye-supervision\)](/articles/governed-actuation/mobileye-supervision)
- [Tesla FSD Supervised vs Governed Actuation: The Stage-Architecture Gap \(/articles/governed-actuation/tesla-fsd-supervised\)](/articles/governed-actuation/tesla-fsd-supervised).
- [Zoox Alternative: Governed Actuation Beyond the Purpose-Built Robotaxi \(/articles/governed-actuation/zoox-amazon\)](/articles/governed-actuation/zoox-amazon).
- [Governed Actuation Beyond Archer Aviation Midnight eVTOL \(/articles/governed-actuation/archer-maker\)](/articles/governed-actuation/archer-maker)
- [Asensus Senhance vs Governed Surgical Actuation \(/articles/governed-actuation/asensus-senhance\)](/articles/governed-actuation/asensus-senhance).
- [BMW Personal Pilot vs Governed Actuation \(/articles/governed-actuation/bmw-personal-pilot\)](/articles/governed-actuation/bmw-personal-pilot).
- [Boston Dynamics Atlas vs Governed Humanoid Actuation \(/articles/governed-actuation/boston-dynamics-atlas\)](/articles/governed-actuation/boston-dynamics-atlas).
- [Governed Actuation Beyond Boston Dynamics Spot \(/articles/governed-actuation/boston-dynamics-spot\)](/articles/governed-actuation/boston-dynamics-spot).
- [Boston Dynamics Stretch Alternative: Governed Actuation for Warehouse Robotics \(/articles/governed-actuation/boston-dynamics-stretch\)](/articles/governed-actuation/boston-dynamics-stretch).
- [EHang EH216-S vs Governed Actuation: Audit-Time Evidence for Autonomous eVTOL \(/articles/governed-actuation/ehang-evtol\)](/articles/governed-actuation/ehang-evtol)
- [Eve Air Mobility eVTOL Alternative: Governed Actuation Beyond the Flight Control Law \(/articles/governed-actuation/eve-air-mobility\)](/articles/governed-actuation/eve-air-mobility).
- [Ford BlueCruise vs Governed Actuation: Beyond Per-Feature Gating \(/articles/governed-actuation/ford-bluecruise\)](/articles/governed-actuation/ford-bluecruise).
- [GM Super Cruise vs Governed Actuation: Stage-Gated L2+/L3 Commitment \(/articles/governed-actuation/gm-super-cruise\)](/articles/governed-actuation/gm-super-cruise).
- [Honda SENSING 360 alternative: governed actuation beyond mode-switch ADAS \(/articles/governed-actuation/honda-sensing-360\)](/articles/governed-actuation/honda-sensing-360).

- [Hyundai HDA and Highway Driving Pilot vs Governed Actuation \(/articles/governed-actuation/hyundai-hda\)](/articles/governed-actuation/hyundai-hda).
- [Ottava vs Governed Surgical Autonomy: Actuation-Authority Decomposition Beyond a Single-Snapshot Review \(/articles/governed-actuation/jnj-ottava\)](/articles/governed-actuation/jnj-ottava).
- [Joby Aviation and Governed eVTOL Actuation: A Beyond-Flight-Control Alternative \(/articles/governed-actuation/joby-aviation\)](/articles/governed-actuation/joby-aviation).
- [Lilium Jet alternative: governed actuation beyond fly-by-wire \(/articles/governed-actuation/lilium-jet\)](/articles/governed-actuation/lilium-jet).
- [Medtronic Hugo Alternative: Governed Actuation for Bounded Surgical Autonomy \(/articles/governed-actuation/medtronic-hugo\)](/articles/governed-actuation/medtronic-hugo).
- [Mercedes Drive Pilot vs Governed Actuation: the Certified L3 Hand-Back \(/articles/governed-actuation/mercedes-drive-pilot\)](/articles/governed-actuation/mercedes-drive-pilot).
- [Rivian Driver+ Alternative: Governed Actuation Beyond L2 Handoff \(/articles/governed-actuation/rivian-driver-plus\)](/articles/governed-actuation/rivian-driver-plus).
- [Smith Nephew CORI vs Governed Surgical Actuation \(/articles/governed-actuation/smith-nephew-cori\)](/articles/governed-actuation/smith-nephew-cori).
- [Stellantis STLA AutoDrive vs Governed Actuation: A Multi-Brand, Multi-Region Level 3 Alternative \(/articles/governed-actuation/stellantis-stla-autodrive\)](/articles/governed-actuation/stellantis-stla-autodrive).
- [Stryker Mako vs Governed Surgical Actuation: A Reversibility-Aware Alternative \(/articles/governed-actuation/stryker-mako\)](/articles/governed-actuation/stryker-mako).
- [Toyota Teammate Alternative: Governed Actuation Beyond Binary Driver Handoff \(/articles/governed-actuation/toyota-teammate\)](/articles/governed-actuation/toyota-teammate).
- [Vertical Aerospace VX4 vs governed actuation \(/articles/governed-actuation/vertical-aerospace\)](/articles/governed-actuation/vertical-aerospace).
- [Volocopter VoloCity Air Taxi vs Governed Actuation \(/articles/governed-actuation/volocopter-air-taxi\)](/articles/governed-actuation/volocopter-air-taxi).
- [Wing Delivery Drones vs Governed Actuation: A Governed Alternative for BVLOS Flight \(/articles/governed-actuation/wing-delivery-drones\)](/articles/governed-actuation/wing-delivery-drones).
- [Wisk Aero vs Governed Actuation: A Certification Substrate for Autonomous eVTOL \(/articles/governed-actuation/wisk-aero\)](/articles/governed-actuation/wisk-aero).
- [Zimmer Biomet ROSA vs Governed Surgical Actuation \(/articles/governed-actuation/zimmer-rosa\)](/articles/governed-actuation/zimmer-rosa).
- [Zipline Drone Delivery vs Governed Actuation for Autonomous Aircraft \(/articles/governed-actuation/zipline-drone-delivery\)](/articles/governed-actuation/zipline-drone-delivery).
- [Does AGCO Fendt IDEAL and PTx OutRun Have Governed Actuation for Autonomous Farming? \(/articles/governed-actuation/agco-fendt-ideal\)](/articles/governed-actuation/agco-fendt-ideal).
- [Bear Robotics Servi vs Governed Actuation: A Serving Robot That Can Defer, Constrain, and Roll Back \(/articles/governed-actuation/bear-robotics-servi\)](/articles/governed-actuation/bear-robotics-servi).

- [Built Robotics Alternative: Governed Actuation for Autonomous Construction Equipment \(/articles/governed-actuation/built-robotics-construction\)](#)
- [Governed Actuation Beyond Cat MineStar Autonomous Mining \(/articles/governed-actuation/caterpillar-minestar\)](#)
- [CNH Industrial Autonomous Agriculture vs Governed Actuation \(/articles/governed-actuation/cnh-industrial-autonomous\)](#)
- [Governed Actuation Beyond Embark and Locomotion: An Autonomous-Trucking Alternative \(/articles/governed-actuation/embark-locomotion-legacy\)](#)
- [Epiroc Mobilaris vs Governed Actuation for Mining Fleets \(/articles/governed-actuation/epiroc-mobilaris\)](#)
- [Gatik Alternative: Governed Actuation for Middle-Mile Trucking \(/articles/governed-actuation/gatik-trucking\)](#)
- [John Deere See and Spray alternative: governed actuation beyond selective spraying \(/articles/governed-actuation/john-deere-see-spray\)](#)
- [Komatsu FrontRunner Alternative: Governed Actuation for Autonomous Haulage \(/articles/governed-actuation/komatsu-frontrunner\)](#)
- [Plus autonomous trucking vs governed actuation: a Class 8 alternative frame \(/articles/governed-actuation/plus-ai-trucking\)](#)
- [Sandvik AutoMine vs Governed Actuation: A Commitment-Layer Alternative for Underground Autonomy \(/articles/governed-actuation/sandvik-automine\)](#)
- [Governed Autonomous Trucking Beyond Torc Robotics \(/articles/governed-actuation/torc-robotics-trucking\)](#)
- [TuSimple Autonomous Trucking vs Governed Actuation \(/articles/governed-actuation/tusimple-trucking\)](#)
- [Waabi Foundation Model Trucking vs Governed Actuation \(/articles/governed-actuation/waabi-trucking\)](#)
- [ABB Robotics vs Governed Actuation \(/articles/governed-actuation/abb-robotics\)](#)
- [FANUC Robotics vs Governed Actuation \(/articles/governed-actuation/fanuc-robotics\)](#)
- [Kiwibot Delivery Robots vs Governed Actuation: A Sidewalk-Robot Alternative \(/articles/governed-actuation/kiwibot-delivery\)](#)
- [KUKA Robotics vs Governed Actuation: The Commitment Gate Above the Controller \(/articles/governed-actuation/kuka-robotics\)](#)
- [Lyft Autonomous vs Governed Actuation: A Marketplace Dispatch Gate Beyond Vendor Availability \(/articles/governed-actuation/lyft-autonomous-magna\)](#)
- [Governed Actuation vs Nuro Autonomous Delivery \(/articles/governed-actuation/nuro-delivery\)](#)
- [Starship Technologies Alternative: Governed Sidewalk-Delivery Actuation \(/articles/governed-actuation/starship-technologies\)](#)

- [Governed Dispatch Beyond Uber Freight Autonomous Trucking \(/articles/governed-actuation/uber-freight-autonomous\)](/articles/governed-actuation/uber-freight-autonomous).
- [Universal Robots Cobots vs Governed Actuation: Admissibility-Gated Cobot Execution \(/articles/governed-actuation/universal-robots-cobots\)](/articles/governed-actuation/universal-robots-cobots).
- [Yaskawa Motoman Alternative: Governed Actuation Beyond Certified Motion \(/articles/governed-actuation/yaskawa-robotics\)](/articles/governed-actuation/yaskawa-robotics).

[Confidence-Governed Actuation overview → \(/governed-actuation\)](/governed-actuation)