Research Square

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

The Art of Choosing an Estimator: Verifiable Agents Inspired by Rudolf E. Kalman and Hassan K. Khalil

Mehdi Heydari Shahna

mehdi.heydarishahna@tuni.fi

Tampere University

Jouni Mattila

Tampere University

Research Article

Keywords:

Posted Date: November 24th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11815817/v1

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

Additional Declarations: The authors declare no competing interests.

The Art of Choosing an Estimator: Verifiable Agents Inspired by Rudolf E. Kalman and Hassan K. Khalil

Mehdi Heydari Shahna and Jouni Mattila

***Abstract*—Choosing an estimator begins with the claim an engineer needs to make about its error. Should a velocity estimate be accurate on average, or must its error stay below a limit after a deadline for every admissible disturbance? The answer can change the choice of estimator even when the plant and sensor stay the same. *Heros in Estimation* makes this judgment explicit for two classical approaches, Kalman's filtering of linear stochastic systems and the high-gain observer theory of Khalil and his coauthors. The engineer confirms or corrects a supervised language interpreter's reading of the request. A deterministic coordinator then compares the established methods against those requirements. Separately written reference implementations check the numerical calculations. In the encoder example, the Kalman filter gives the smallest steady-state mean-square speed error among linear estimators. With the same information, the sampled second-order high-gain observer gives the tighter guaranteed speed-error bound after the deadline. Its worst-case error can be calculated exactly at each sample for the stated bounded-disturbance model. In a rotary-joint simulation, the more accurate velocity estimate accompanies a slightly larger tracking error. The framework preserves a practical part of control-theoretic expertise by making the reasoning behind these choices available for engineers to examine and reuse.**

I. Introduction

SOONER or later every control engineer needs a signal that no sensor delivers. An encoder reports the angle of a robot joint, and the controller needs its velocity. A hydraulic boom carries an angle sensor and a pressure sensor, and the task needs the boom rate. Two of the most influential answers to this problem carry the names of Rudolf E. Kalman and Hassan K. Khalil. The Kalman filter [1], [2] treats what is unknown as random and returns an estimate with its error covariance; the estimate has the smallest mean-square error of all estimators when the noises are Gaussian, and of all linear estimators when only means and covariances are known. The high-gain observer treats what is unknown as bounded and returns an estimate whose error stays inside a guaranteed envelope; taken from linear feedback into nonlinear feedback in the late 1980s, it was developed to a large extent by Khalil and a long line of coauthors [3]–[5] (see the survey [4] for its origins).

Both methods are textbook material; choosing between them is not. The encoder example carries one question through this article: *with the same sensor and the same plant, how does the right estimator change when the engineer asks for the smallest mean-square error instead of a guaranteed error bound?* The answer depends on which description of the noise the facts support, on what each method's output means, and on what must be asked before a number is released. That knowledge lives mostly in experienced colleagues; this article keeps it executable.

Keeping control-design knowledge executable is an old aim: expert control [6] let an expert system carry the heuristic logic around control algorithms. Paper2Agent [7], [8] turns a paper and its code into tools for a chat agent; S2C [9] synthesizes H_∞ controllers from natural-language requirements with specialized agents; ControlAgent [10] routes design tasks to agents that tune controllers iteratively; and supervisory multi-observers [11] pick one of a bank of observers at each instant. Here, by contrast, the estimators are fixed numerical code checked against separately written references, the learned component never computes or certifies anything, and a deterministic coordinator decides once, at design time, which of two methods with *different meanings of uncertainty* the request justifies and whether the request says enough to decide.

Here an *agent* is a software component with a fixed role and scope that must justify every answer or refuse; none of the three is a language model. The *Kalman specialist* handles known linear models with stated noise covariances, the *Khalil specialist* second-order systems with one measured output and bounded uncertainty, and the *coordinator* holds a common specification, checks what each specialist can justify, and returns one of seven outcomes: a released design, a request to confirm what was read, one clarifying question, "no verified design found," a proof that no estimator can meet the request, "out of scope," or "invalid input." The article makes four contributions.

1) *Checked executable methods.* Both specialists agree with separately written references, mostly to rounding error, and Proposition 1, an elementary application of peak-to-peak analysis, gives the exact worst-case error of the observer as it is sampled and started.
2) *Coordination across assumptions.* Conditions under which a fixed design meets a requirement throughout the stated uncertainty ranges, and a comparison on identical information in which the justified method changes with the criterion.
3) *Trained interpretation behind a confirmation step.* A compact reader, tested on separately generated synthetic

This work was supported by the Finnish Foundation for Technology Promotion (Tekniikan Edistämissäätiö) through the Post Docs in Companies (PoDoCo) program.

M. Heydari Shahna and J. Mattila are with the Faculty of Engineering and Natural Sciences, Tampere University, 33720 Tampere, Finland (e-mail: mehdi.heydarishahna@tuni.fi).

requests and on plant families never used in training, whose every reading the engineer confirms before it is used.

4) *A reproducible workflow.* An exportable record for replaying the estimator and scripts for reproducing the numerical experiments (the archived GitHub release).

The names in the title acknowledge a lineage that includes R. S. Bucy and Khalil's coauthors F. Esfandiari, A. M. Dabroom, L. K. Vasiljevic, L. Praly and others; the agents implement a documented part of that work, and the names express acknowledgment, not endorsement.

II. Two Meanings of Uncertainty

The two lines of work answer the same question, "what is the state?", under different descriptions of what is not known. The difference determines which sentences about the result are true.

A. Kalman's Stochastic Reading

Kalman's 1960 paper [1] posed filtering and prediction in state-space form with a recursive solution; the continuous-time counterpart followed with Bucy [2]. Here the plant is

$$x_{k+1} = Ax_k + Bu_k + w_k, \qquad y_k = Cx_k + v_k, \tag{1}$$

with known matrices, known inputs u_k, and zero-mean white noises w_k, v_k of covariances Q and R that are uncorrelated with each other and with the initial error, whose covariance is P_0. Each measurement is processed by an update and a prediction,

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + M_k\,(y_k - C\hat{x}_{k|k-1}), \tag{2}$$

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k, \tag{3}$$

with $M_k = P_{k|k-1}C^{\mathsf{T}}(CP_{k|k-1}C^{\mathsf{T}}+R)^{-1}$ and the covariance recursion $P_{k|k} = (I - M_kC)P_{k|k-1}(I - M_kC)^{\mathsf{T}} + M_kRM_k^{\mathsf{T}}$, $P_{k+1|k} = AP_{k|k}A^{\mathsf{T}} + Q$.

What may be claimed depends on what is assumed about the distributions (see Appendix A). In every case $P_{k|k}$ is a covariance, never a bound that the error cannot exceed, and it describes the estimate that is actually computed. When the engineer asks for the constant (steady-state) gain, the exported covariance therefore starts from the stated P_0 and follows that gain; it reaches the steady-state value only as the start-up transient decays. For a scalar random walk with $Q = R = 1$ and $P_0 = 100$, the steady-state gain is $M = 0.618$; with it, the first filtered variance is $(1 - M)^2P_0 + M^2R = 14.9718$, not the steady-state variance 0.618 (equal to M because $R = 1$).

B. Khalil's Deterministic Reading

High-gain observers address a different situation: the model is nonlinear and partly unknown, and nothing statistical is asserted. The supported class is

$$\dot{x}_1 = x_2, \qquad \dot{x}_2 = f(x_1, x_2, u) + d(t), \qquad y = x_1 + v, \tag{4}$$

in which f is known only approximately, $|d(t)|$ and $|v(t)|$ are bounded, and x_2 is to be estimated. Following the motivating example in [4], the observer copies a nominal model f_0 and injects the output error with gains that grow as a small parameter ε shrinks; for the repeated-pole choice used throughout this article, and with the nominal model evaluated at the measurement rather than at the estimate (a choice of this project),

$$\dot{\hat{x}}_1 = \hat{x}_2 + \frac{2}{\varepsilon}(y - \hat{x}_1), \qquad \dot{\hat{x}}_2 = f_0(y, u) + \frac{1}{\varepsilon^2}(y - \hat{x}_1). \tag{5}$$

Three facts from that literature shape what follows. First, a small ε makes the estimate insensitive to the mismatch $f - f_0$ but amplifies measurement error: the ultimate error bound has the form $c_1\varepsilon + c_2\vartheta/\varepsilon$, where ϑ (written V from here on) bounds the measurement error [4]; error bounds for differentiating noisy signals are the subject of [12]. Second, the estimate may show a short transient whose amplitude is of order $1/\varepsilon$ (peaking), which is why feedback designs saturate the control [3]. Third, every bound holds only while the trajectory stays where the mismatch is bounded. The result is a deterministic envelope that the error cannot leave, provided the stated bounds are true.

Table I lists what each specialist implements, needs and returns; Fig. 1 places the implemented methods in their lineage.

III. From Papers to Checked Code

Between a theorem and a number lie a model class, a discretization, a gain computation and a bound evaluation, and each step can silently change what the theorem covers. A *source-to-method register* therefore records, for every executable method, the publication and statement used, the numerical adaptation, the independent check, and the tested range.

A. Independent References

The specialists reuse the numerical kernels of two earlier single-method packages supplied for this project, one for each method. Reuse alone proves nothing, so every kernel is compared with a reference implementation written separately from the mathematical definitions with different algorithms (batch conditioning instead of the Kalman recursion, no library Riccati or Lyapunov solver, no matrix-exponential shortcut). For the Kalman specialist the reference stacks all states and measurements of a record into one Gaussian vector and conditions it in 40-digit arithmetic. Over 21 cases from 7 model families the estimates agree to 2.2×10^{-14} of the prior standard deviation and the covariances to 6.5×10^{-15}; the boom record of Fig. 2(a) differs from its reference by at most 6.7×10^{-15}. For the observer, the reference superposes simulated unit responses; over 20 random cases the two worst cases agree to 8.6×10^{-14}, and over 20 further cases with the start from the first reading to 2.2×10^{-13}.

The earlier packages also provide a baseline of a different kind: their documentation lists 15 defects, 7 of which were reproduced on the supplied code, and all now have regression tests. In one, a deadline was certified as if the initial error were irrelevant, which holds only for an ultimate bound; an admissible initial error leaves the error at the deadline about 18 times the certified limit.

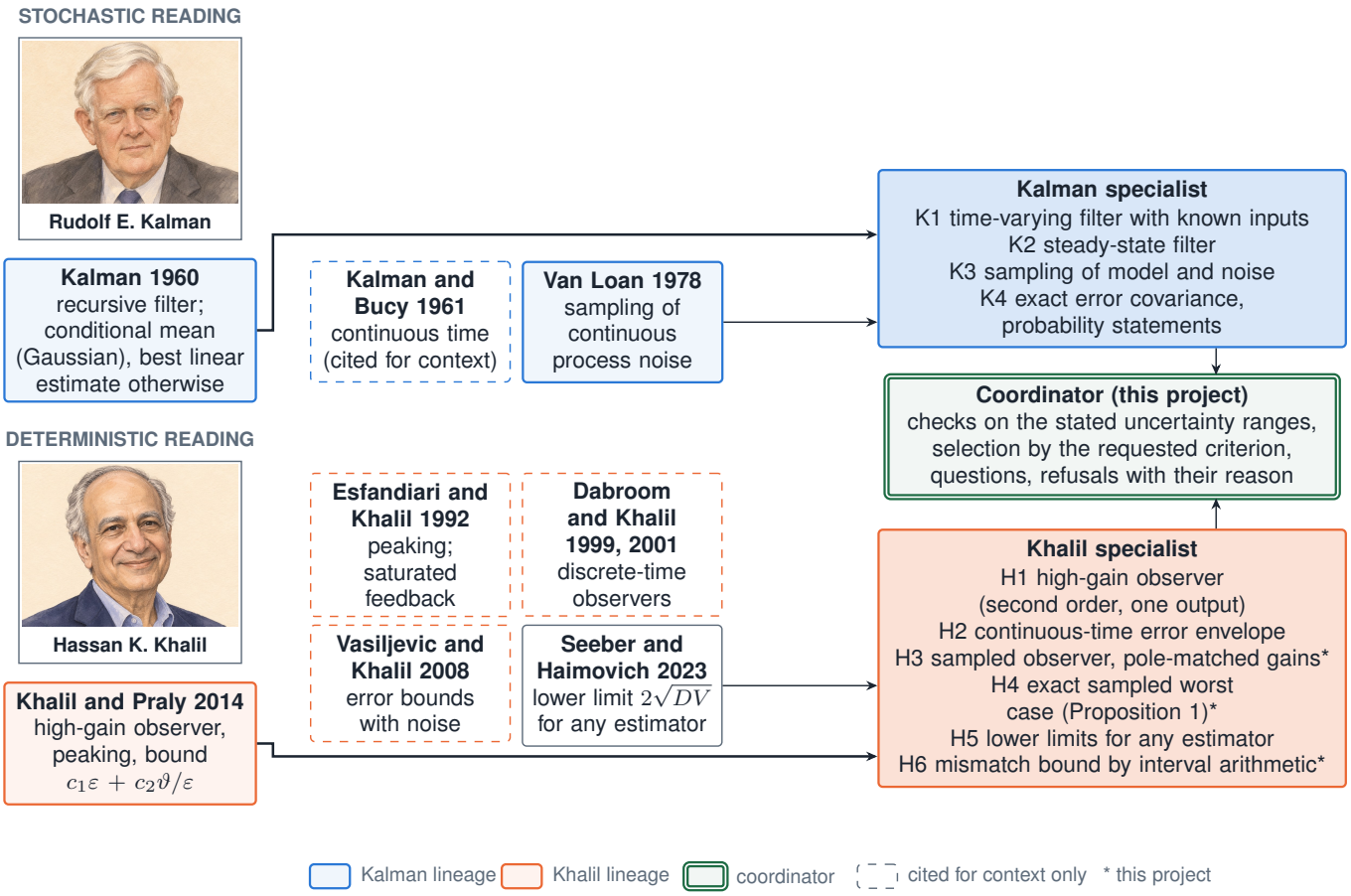


Fig. 1. Scientific lineage and scope. The portrait illustrations identify the two research lineages. A bold arrow connects each method to the software specialist that implements it. Both specialists report to the coordinator. The boxes on the right list the methods that are implemented and independently checked (K1 to K4, H1 to H6); an arrow from another source marks a statement that a specialist uses, and dashed boxes mark work cited for context only. H3, H4 (Proposition 1) and H6 are results or design choices of this project.

B. A Guarantee for the Observer That Is Actually Implemented

The bounds quoted above are for the continuous-time observer (5), but a controller runs it at a sample period T, and for the short time constants that make high-gain observers attractive, T/ε is not small. Discrete-time implementations were studied in [15], [16], where the bilinear transformation was found preferable among the discretizations compared [15]. The framework implements the observer in the update-and-predict form of (2)–(3), with $A_d = \left[\begin{smallmatrix} 1 & T \\ 0 & 1 \end{smallmatrix}\right]$, the nominal acceleration $f_0(y_k, u_k)$ held over the period, and the constant gain

$$M = \begin{bmatrix} 1-p^2 \\ (1-p)^2/T \end{bmatrix}, \qquad p = e^{-T/\varepsilon}, \tag{6}$$

which places both error poles at p, the sampled image of the repeated pole $-1/\varepsilon$ of (5). This pole-matched form is a choice of this project: it shares the structure of the Kalman filter, so that one worst-case analysis covers both, and its poles stay real and positive for every T/ε. Forward-Euler discretization, by contrast, loses stability at $T/\varepsilon = 2$ and on the joint example below already nearly doubles the guaranteed limit at $T/\varepsilon = 1$ (0.242 against $0.126\,\mathrm{rad/s}$).

At the sample instants, the error of this observer is exactly linear in the initial error, the measurement errors, and the total disturbance (model mismatch, unmodeled acceleration and the effect of holding f_0), so the worst case over the stated uncertainty set is computable exactly (see Appendix B). The specialist evaluates it in interval arithmetic with outward rounding, for the gain and period as stored in binary floating point, on the samples of the requested window, and releases a design only on that evidence; the certificate refers to the observer recursion with these stored values. The certificate holds for samples taken exactly at the instants $t_0 + kT$ of the one period it was computed for. When recorded data carry timestamps, the run-time path takes them as the actual sample instants and compares them with these instants first: each must agree to within the float64 precision of the stored times, or to within half the resolution of the time labels when the request states one, which then becomes a condition of the claim. The decision therefore does not depend on the clock origin, but the precision does: times near $10^9\,\mathrm{s}$ resolve only about $10^{-6}\,\mathrm{s}$, and a release then states this precision. Data recorded every $2\,\mathrm{ms}$ for a design certified at $1\,\mathrm{ms}$, a record with a missing sample, or a sample taken $0.5\,\mathrm{ns}$ late are refused with the reason stated; in one such record, with unit speed, the late sample

TABLE I
SUPPORTED METHODS AND WHAT EACH ONE CLAIMS.

Method	Source	Model class and evidence required	What is claimed	Scope of the claim	Reference check
Kalman filter with known inputs, time-varying or steady-state gain (K1, K2)	Kalman [1]; steady state as in [13]	Linear discrete-time model with known parameters, or continuous model with period; Q, R, P_0 and their meaning; whiteness; distribution if an all-estimator claim is wanted	Conditional mean (Gaussian) or best linear estimate; error covariance of the computed estimate from the first sample on; probability level per instant	Exactly known parameters; white, mutually uncorrelated noises; one period, checked against recorded timestamps; finite-time statements from P_0, steady-state statements for the limit	Batch conditioning in 40-digit arithmetic: estimates 2.2×10^{-14}, covariances 6.5×10^{-15}; fixed-point iteration: steady-state gain 3.4×10^{-11}
Sampling of model and process noise (K3)	Van Loan [14]	Continuous-time matrices; whether a noise value is a spectral density or a per-step covariance	Exact zero-order-hold model and noise covariance	Input held between samples	Gauss–Legendre quadrature: 1.3×10^{-15}
Error covariance of any linear gain (K4)	Standard covariance recursion	Gain sequence and second moments	Exact mean-square error, largest over the requested samples; Chebyshev level	Probability statement at each instant of the window	Kronecker-form Lyapunov solution and Monte Carlo
Sampled high-gain observer (H1, H3)	Khalil and Praly [4]; discrete-time forms in [15], [16]	Chain (4); nominal model in y, u; exact period	Estimate at the sample instants	Second order, one output, repeated pole; pole-matched gains (this project)	Separately written simulator
Exact sampled worst case with interval certificate (H2, H4)	This project (Proposition 1, with the start from the first reading); continuous-time bound of the form in [4]	Hard bounds on measurement error, total disturbance and initial error; deadline and record length	Error limit for every admissible signal, at the sample instants of the requested window	The sample instants of the requested window	Superposition of unit responses: 8.6×10^{-14}; first-reading start 2.2×10^{-13}
Lower limits for any estimator (H5)	Seeber and Haimovich [17]; first sample (this project)	Bounds on the free part of the acceleration and on the measurement error; travel, speed and input ranges; initial-error bounds	No causal estimator guarantees less than $2\sqrt{DV}$ at a requested sample from $4\sqrt{V/D}$ on, or, at $t = 0$, less than the half-width of the initial velocities that admissible trajectories can have	Worst-case requirements; trajectories admissible for the whole window, inside the stated travel and speed ranges	Explicit pair of signals with the travel for braking; separately written construction of admissible trajectories at $t = 0$
Mismatch bound (H6)	This project (design choice)	Symbolic f, f_0, parameter ranges (a one-point range states the true value), operating range with evidence	Bound D on the total disturbance; the smaller of this and a supplied bound	Valid inside the stated range; monitored at run time	Random, face and corner samples

"Evidence required" lists what the specification must contain before a design is released; "Scope of the claim" states where the released statement applies; "Reference check" names the separately written calculation and, where it is a numerical comparison, the largest relative difference observed.

alone produced a velocity error of 4.6×10^{-7} under a claimed bound of zero. A request that declares irregular sampling gets no fixed-period design, with or without timestamps. Likewise, a design whose model uses the applied input runs only with input values: the recorded ones or, without them, a fixed input that the request states, which the claim then names. Fig. 2(c) and (d) show two consequences. On the joint the sampled observer's limit after the deadline is $0.120\,\mathrm{rad/s}$, against $0.130\,\mathrm{rad/s}$ for the continuous-time observer with the same ε. And the guaranteed limit has a minimum in ε, the trade-off between mismatch and measurement error that the literature describes [4], here located for the sampled implementation.

IV. ONE SPECIFICATION, THREE QUESTIONS

The interface opens with three plain questions: *What do you want to estimate? What do you measure, and what model and uncertainty information do you have? What does successful estimation mean for you?* Equations, a model file and data, with or without timestamps, can be attached. The answers fill one *common specification* (plant and target, time, initial information, stochastic and deterministic uncertainty, objective, evidence) that both specialists read, so a comparison always refers to the same plant, measurements and sample period.

Every material field carries a status: supplied, interpreted from text, derived (with its conditions), accepted as a design choice, or unresolved. Anything interpreted from text is shown back in plain words, and nothing is released until the engineer confirms it; the confirmation is bound to a hash of the specification and is void if a material fact changes. The result is exported as an *execution record* (specification, decision, configuration, measurements, inputs, timestamps and

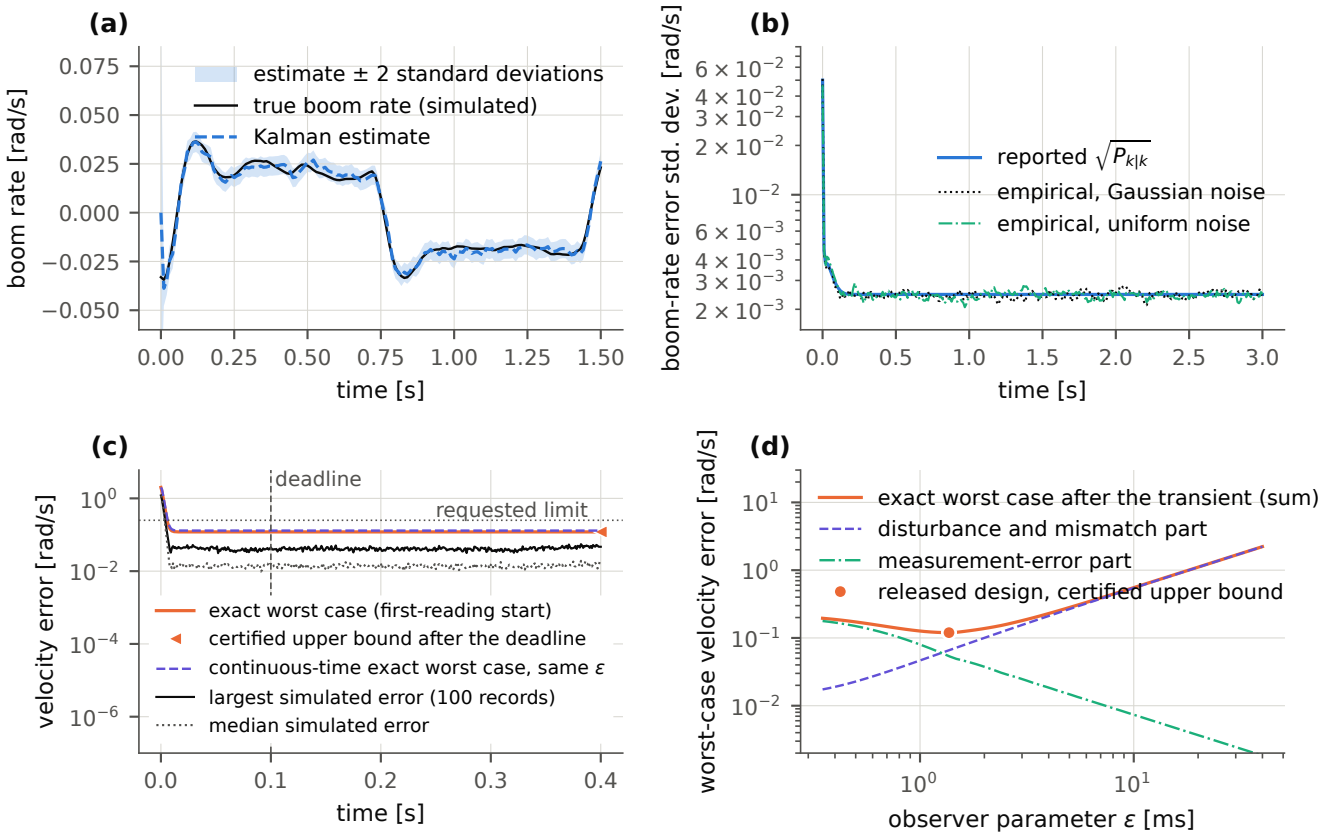


Fig. 2. Specialist fidelity. (a) Kalman filter on the simulated hydraulic boom: true and estimated boom rate for one record with the band of two standard deviations; the estimate starts from the stated prior mean. (b) Reported standard deviation of the boom-rate error and the empirical spread over 200 records, for Gaussian and for uniformly distributed noise of the same covariance. (c) High-gain observer on the simulated rotary joint, started from the first encoder reading. Exact worst case over the stated uncertainty set (orange and violet dashed curves), that is, over every initial speed error within the stated bound, every measurement error with $|v_k| \le V$ and every total disturbance with $|\delta(t)| \le D$, which may vary within a sample interval: the orange curve is the largest error of the sampled observer over this set at each sample (Proposition 1 with that start, evaluated in floating point; a signal of the set attains it), and the violet dashed curve is the same quantity for the continuous-time observer with the same ε and independent initial errors, bounded by V in position and by the initial speed uncertainty, shown for reference. The simulated joint is a narrower case (its acceleration follows the joint's dynamics), so for it the orange curve is an upper bound, not a value it reaches. Certified upper bound (marker): the orange marker at the right edge is the interval-arithmetic bound, valid at every sample after the deadline, that the released claim states; it lies one floating-point step above the largest value of the orange curve. The black and gray curves are the largest and median errors over 100 simulated records and lie below the worst case; the vertical line marks the deadline and the horizontal line the requested limit. (d) Exact worst case over the same set after the transient (curves: the value that the orange curve of (c) approaches, as a function of ε, with its disturbance and measurement-error parts) and the certified upper bound of the released design (marker).

every output, including bounds or covariances), whose replay recomputes every number, so that a certificate, not only a gain, can be reproduced.

V. The Coordinator Decides What the Evidence Supports

The deterministic coordinator (Fig. 3) proceeds in three steps.

Validity and scope. Contradictory input (a negative bound, a deadline after the end of the record, a window without a sample, a discrete model with a different stated period) is invalid, with the field named. A transient cap below the later tolerance is not a contradiction: the cap applies at every sample, the tolerance from the deadline on. Plants outside both method classes, or with unmodeled actuator or sensor dynamics, are out of scope; a supported plant with a feature no verified method covers, such as colored noise or stated irregular sampling, gets "no verified design found."

Checking a fixed design over uncertain ranges. An admissible completion assigns values to unresolved fields while respecting the stated ranges and constraints. A guarantee requires one estimator, with one fixed configuration, to meet the requested criterion at every point of the admissible set.

Every stated fact takes part (a tighter disturbance bound for an exactly known model applies together with a looser total bound), and every sample used to establish or refute a requirement lies in the requested window: a record that ends at 3.5 s with 1 s sampling is judged on the samples up to 3 s. For a fixed estimator, the error bound in Proposition 1 is nondecreasing in its component bounds. When those bounds vary independently over intervals, their joint upper endpoint gives the worst case. For coupled ranges, a vertex check suffices for an upper bound if the scalar error criterion is convex over the entire feasible polytope. In the variance-split example below, the mean-square error for a fixed gain is affine in the split.

The archived implementation accepts some requests with coupled bounds for which checking only the vertices can miss the worst case. Suppose two bounds on the same acceleration satisfy $|w| \le D$, $|w| \le d$, and $D + d = 2$ with $D, d \in [0, 2]$. The effective bound $\min(D, d)$ is zero at both endpoints and reaches one at $D = d = 1$. A release based on those endpoints alone carries no guarantee for the interior. When the coordinator identifies two admissible completions with different outcomes, it can ask for the fact that separates them; a fact that cannot change the outcome is not asked for (see Appendix C).

Comparison on the requested criterion. When both specialists qualify, the coordinator compares them on the quantity the engineer asked for, on identical information, and reports both values; it does not rank methods in general.

Refusals. If no candidate passes, the coordinator answers "No verified design found." and states the limit of each test. "No estimator can meet this requirement" needs an argument for the stated plant: the optimum of the stated linear model for a mean-square level, or one of the two lower limits for a worst-case requirement. A worst-case limit never refutes a probability level. For a speed error within $0.02\,\mathrm{rad/s}$ with probability 0.9, the limit $2\sqrt{DV} = 0.049\,\mathrm{rad/s}$ shows only that no estimator guarantees $0.02\,\mathrm{rad/s}$ for every disturbance, and Chebyshev's inequality proves a probability of only 0.84 for the best linear estimate; the answer is again "No verified design found."

Table II lists the 16 incomplete and conflicting specifications of this study, each with its outcome and an independent check; all passed.

A. A Worked Decision With the Same Facts and a Different Criterion

The specialists overlap on second-order chains with a known input gain. Return to the encoder of the introduction: a servo axis sampled at $1\,\mathrm{kHz}$ whose encoder error is at most $V = 10^{-4}\,\mathrm{rad}$ and whose unmodeled acceleration is at most $D = 6\,\mathrm{rad/s^2}$, both uniformly distributed within their bounds and independent from sample to sample. The information is then common: the Kalman specialist uses the variances, the Khalil specialist the bounds, and Proposition 1 applies to *any* constant-gain estimator of this plant, including the steady-state Kalman gain. Both estimators start from the first reading and zero velocity, with an initial speed error of at most $3\,\mathrm{rad/s}$; the requested levels below were chosen relative to computed values, to illustrate each outcome.

Fig. 4 shows the outcome on 200 common records. When the engineer asks for the smallest mean-square error of the speed, the coordinator releases the Kalman filter with the linear-estimator claim, because the noise is not Gaussian; its steady-state mean-square error is $6.35 \times 10^{-5}\,(\mathrm{rad/s})^2$, while the observer tuned for the worst case is 1.34 times larger and one tuned for mean-square error 1.04 times. When the engineer asks instead for a guaranteed error of at most $0.0906\,\mathrm{rad/s}$ from $50\,\mathrm{ms}$ onward, both qualify and the coordinator releases the observer, certified at $0.0566\,\mathrm{rad/s}$ against $0.0603\,\mathrm{rad/s}$ for the Kalman gain. A limit of $0.0392\,\mathrm{rad/s}$ lies below the lower limit $2\sqrt{DV} = 0.0490\,\mathrm{rad/s}$, and the coordinator proves that no estimator can guarantee it.

Here the reversal amounts to a factor of 1.06 in guaranteed error and 1.34 in mean-square error. Across 15 combinations of the two bounds, the ratio of the worst-case errors from $50\,\mathrm{ms}$ onward ranged from 0.97 to 10.0, in favor of the observer in 14 settings, while for the ultimate worst case the Kalman gain was better in 3. "Observer for guarantees" is a tendency, not a rule, so the coordinator computes the comparison each time.

VI. Learning to Read a Request

Only the reading of the three answers is learned. A deterministic reader extracts numbers and units, converting "half a millisecond," "1 kHz" or "5×10^{-5} rad^2" to SI units, and never guesses a number it cannot parse. The trained reader, a bidirectional gated recurrent unit (GRU) network with 113 620 parameters trained by supervised learning on 3 600 labeled requests (23 273 labeled quantities), then assigns each number a *role* from a fixed list (hard bound on the measurement error, noise standard deviation, tolerance, deadline, sample period, and so on; a mask forbids impossible assignments such as a time read as a variance) and labels the request: the kind of objective, the stated distribution, the horizon, the evidence offered for an operating range, and warning flags such as colored noise. What it learns is how the wording decides the role when the unit cannot. In one test request, the deployed reader distinguished an initial speed-error bound of $361\,\mathrm{mrad/s}$ from a requested speed-estimation tolerance of $32.5\,\mathrm{mrad/s}$. The scripted evaluation measured field corrections. On the 229 worded test requests, the trained reader needed 0.65 to 0.79 fewer field corrections per request than keyword rules across the three seeds (intervals excluding zero for 3 of 3 seeds; see Section VII-C). It also proposes an outcome, which the coordinator ignores; the proposal exists only for a "single model decides" baseline.

A. Cases, Splits, and Wording

Fig. 5 shows how the data were built. A generator draws a plant from one of eleven model families and a complete specification, then removes, blurs or contradicts information according to one of 35 categories (complete requests, harmless and essential omissions, ranges that do or do not change the decision, impossible requirements, unsupported features, invalid input). Each correct outcome is thus fixed by construction, and the categories form contrasting pairs in which one changed condition changes the justified decision. Seven families were used for training and development; four (a gantry axis, a Duffing-type stage, the hydraulic boom and a cart-pole) appear only in the reader's test set. The 12 test cases of each of the two unsupported families (bearing-only tracking, cart-pole) share one plant model, and the bearing-only specification is identical to all 106 training cases of that family.

Training requests come from three families of templates and development requests from a fourth. We generated three sets of 80 synthetic test requests and one set of 60 synthetic development requests in separate sessions of the same language model.

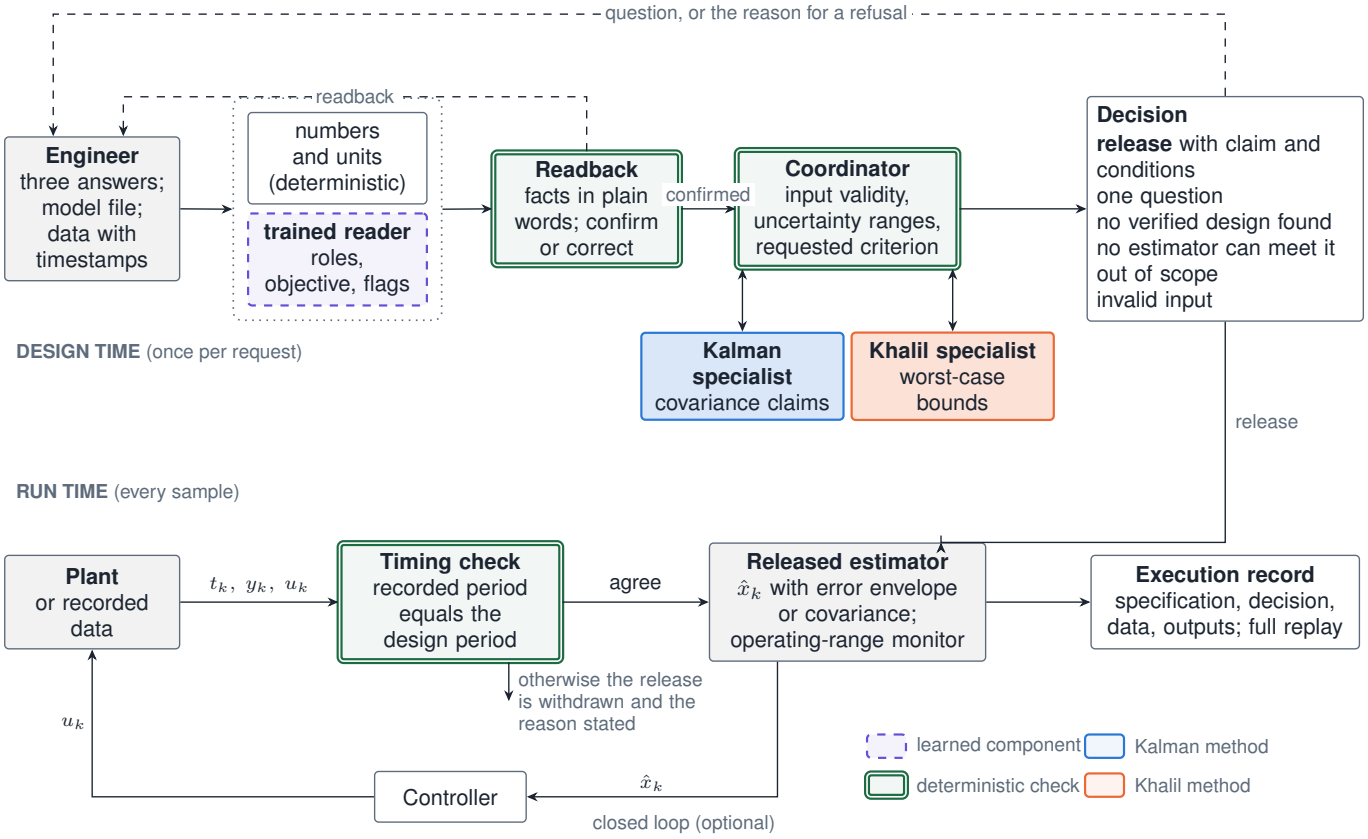


Fig. 3. The deployed framework. At design time, three agents stand between the engineer and the estimate: the coordinator (green, deterministic checks), the Kalman specialist (blue) and the Khalil specialist (orange). The only learned component is the reader of the request (violet, dashed); what it reads is used only after the engineer confirms the readback. The decision is a release with its claim and conditions, one question, or a refusal with its reason. At run time, the recorded times t_k of the measurements y_k and applied inputs u_k are compared with the period of the design before the released estimator runs; its output $\hat{x}_k$ carries the error envelope or the covariance of its claim and may feed a controller. Every run is exported and can be replayed.

Each session received its own fact sheets and style instructions. A scripted evaluator supplied confirmations and corrections during testing. Eleven of the 240 test requests belong to a category outside the three answers (a range written in reverse order) and were excluded by a rule fixed beforehand, leaving 229. Wording was kept apart as strictly as the families; the one change made to the training templates after a test request had been seen, the display of squared compound units, preceded the freezing of the protocol, and a retraining on the uncorrected templates reads 41% of the test requests without error, against 47%.

B. Choosing the Reader

Five candidates and the two models supplied with the earlier packages were compared on development data only (581 templated and 59 worded requests) by the share of requests read without any error: keyword rules reached 5% and 25%, a logistic regression on context words 40% and 36%, and a small bidirectional gated recurrent network with 113 620 parameters 51% and 51%; the same network on frozen pretrained word vectors (49% and 42%) and pretrained vectors with linear output layers only (12% and 20%) scored lower. The supplied models cannot produce the request-level labels, so none of their readings was complete.

The rule fixed beforehand selects the smallest trained model within 0.02 of the best mean score; only the recurrent network without pretrained vectors met it. It was trained with three predeclared seeds, about 34 min each on one processor core, and reads a request in about 4 ms in its deployed NumPy form. In the comparison, each candidate was trained once.

VII. An Acceptance Test Fixed in Advance

The test protocol was committed and tagged in version control, with the hash of every file that determines the test, before any reader saw a test request or any policy a test case. The software was then refined in review, each refinement with a regression test, and the test was rerun with the same cases, readers and protocol; the numbers below come from the eighth run, made with the software frozen for this article, and every run and changed outcome is kept with the software (the archived GitHub release). The limited-user variant, the selector comparison and the timing study were added after the first run.

For the 20 requests whose transient cap lies below the later tolerance, the expected outcome was decided from each case's own data without the software: the two requirements are compatible, and in all 20 the cap is below the half-width of the admissible initial velocities (for the 14 with a stated travel or speed range, of those from which explicit trajectories stay

TABLE II
INCOMPLETE AND CONFLICTING SPECIFICATIONS: WHAT IS MISSING, WHY IT MATTERS OR DOES NOT, WHAT THE COORDINATOR RETURNS, AND HOW THE OUTCOME WAS CHECKED.

Missing or conflicting fact	Consequence	Outcome	Independent check
Prior and noise variances; their sum (2) and a range for the noise variance (1 to 1.5) are known; smallest mean-square error requested	Optimal gain lies between $1/4$ and $1/2$	Question about the noise variance	Batch conditioning: gains 0.25 and 0.50 ✓
As above; a mean-square error below 0.65 is requested	None: gain $1/4$ meets it for every admissible split	Release (Kalman), no question needed	Errors 0.375 and 0.625 at the two extreme splits, affine in between ✓
Initial-error bound; an ultimate limit is requested	None: initial-error terms vanish in the limit	Release (observer), omission reported as irrelevant	Worst case after 400 steps equal for zero and large initial error ✓
Initial covariance; a steady-state (limit) statement is requested	None: the steady-state gain and its limit do not depend on it	Release (Kalman), omission reported as irrelevant	Riccati iteration from two initial covariances: same gain ✓
Distribution of the noises	Same estimate; only the linear-estimator claim holds	Release (Kalman), omission reported as irrelevant	Monte Carlo variance matches for Gaussian and uniform noise ✓
Hard bound on the measurement error	The guaranteed limit depends on it	Question about the hard noise bound	Design certified for one bound violates the limit for a larger admissible one ✓
Hard noise bound; an RMS value is supplied instead	No hard bound follows from an RMS value	Question about the hard noise bound	Noise with fixed RMS and arbitrarily large magnitude ✓
Noise bound known only as a range	Requirement is met on part of the range only	Question about the hard noise bound (with the decisive threshold)	Worst case meets the limit at the threshold, not above it ✓
Sample period stated as a range	Model, gains and certificate depend on the period	Question about the exact period	Wrong period: velocity error 2 rad/s on error-free data ✓
A deadline is added later	The initial error, irrelevant before, now matters	Question about the initial-error bound (with the decisive threshold)	Earlier design violates the deadline for an admissible initial error ✓
Evidence that the motion stays in the stated range	The mismatch bound holds only inside the range	Question about the evidence for the range	Mismatch outside the range exceeds the bound used inside ✓
Whether a noise value is a spectral density or a per-step covariance	The two readings differ by a factor of order T	Question about the meaning of the value	Quadrature: the two readings give different discrete covariances ✓
Discrete model with a different stated period	A discrete model cannot be resampled	Invalid input	Supplied matrix is not the transition for the stated period ✓
Data recorded every 2 ms, request states 1 ms	Gains and certificate hold for one period	Release withdrawn (timing conflict)	Release withdrawn for these data; the same records with 1 ms timestamps are processed ✓
None: a transient cap below the later tolerance	At $t = 0$ one reading carries no speed information	No estimator can meet it	Admissible initial speeds span ± 2 rad/s, above the cap ✓
None: a total and a tighter additive disturbance bound	The smaller bound applies	Release (observer), no question needed	Independent worst case with the additive bound meets the limit ✓

The outcome was checked independently of the coordinator's own numbers. A check mark means that the independent calculation confirmed the outcome. A root-mean-square (RMS) noise level does not imply a hard bound: a noise equal to $\pm a/\sqrt{p}$ with probability $p/2$ each and zero otherwise has RMS value a for every p, but its magnitude grows without limit as p shrinks.

inside it), so no estimator can meet it at the first sample. The frozen case files are unchanged; these outcomes are applied when the cases are loaded.

A. Design

The *structured* set contains 600 specifications from all eleven families, with correct fields and no language layer; 370 of them admit a verified design once the missing facts are supplied, and 230 do not. The *natural-language* set contains the 229 worded requests for a sample of the same cases.

The user is a program. Asked a question, it gives the hidden true value, or "unknown" if the fact is not available; in a limited variant it knows only the ranges it stated. Shown a readback, it either corrects every field that differs from the stated facts (each correction counted) or confirms without reading: the attentive mode is an ideal reviewer whose outcomes equal the oracle's by construction, the blind mode a stress test of the confirmation step. Releases and impossibility claims are judged against the hidden complete specification with the independent references, other outcomes against the outcome fixed by construction. A release is incorrect if its claim is false, answers a different question, ignores a stated fact, or rests on information the request did not contain; for nonlinear plants the judge checks that the released disturbance bound exceeds a sampled estimate of the true mismatch. An impossibility claim is false if a verified design exists for the complete facts or the judge's own construction does not confirm it.

Each baseline differs from the full framework in one decision: a specialist alone; a selector that tries the Kalman specialist first; asking for every unresolved field; never asking (defaults and range midpoints); and, for language, no reader, keyword rules, frozen word vectors with linear heads, and a "single model" whose proposal chooses the method. The coordinator on the correct structured specifications is the oracle.

B. Results

On the structured set (Table III, Fig. 6) the full framework released a correct design for all 370 requests that admit one, identified all 230 others correctly, and made no incorrect release (95% upper limit 0.64% of requests) and no false

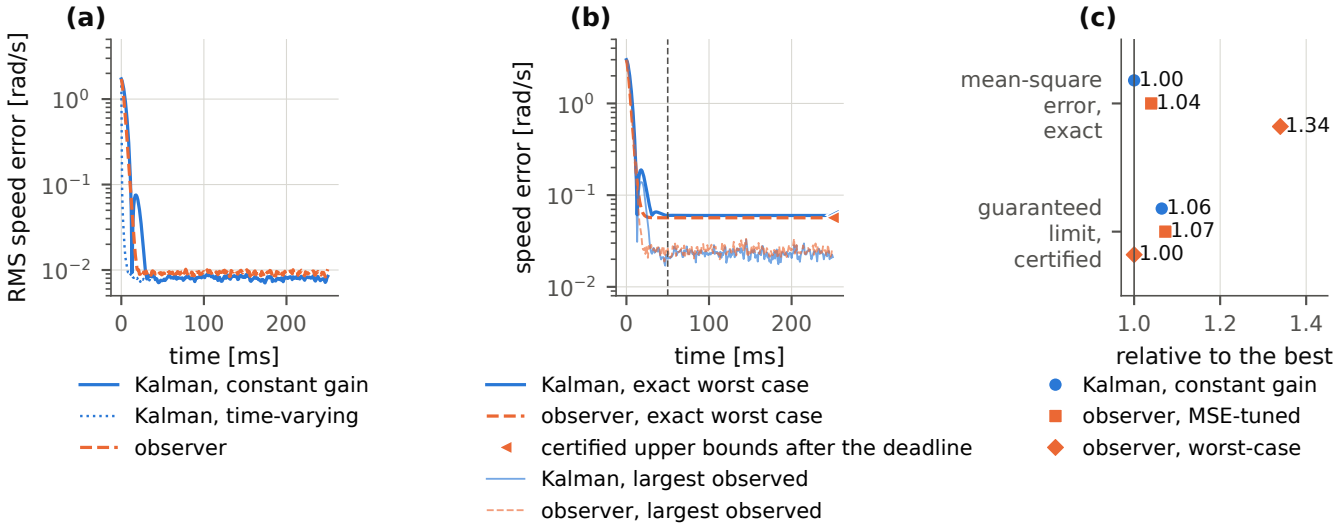


Fig. 4. One plant, one set of facts, two criteria. (a) Root-mean-square speed error over 200 common records for the Kalman filter and the observer tuned for the worst case (also in (b)): in steady state the Kalman filter is lowest, as its claim states. (b) Exact worst case over the stated uncertainty set (thick curves): for each design started from the first encoder reading, the largest error at each sample over every initial speed error within $3\,\mathrm{rad/s}$, every measurement error with $|v_k| \le V$ and every disturbance with $|\delta(t)| \le D$, which may vary within a sample interval (Proposition 1 with that start, evaluated in floating point; at $t = 0$ it equals the initial speed uncertainty). The common records hold the acceleration constant over each sample, a narrower condition, for which the thick curves are upper bounds. Certified upper bound (markers at the right edge): the interval-arithmetic bounds after the deadline that the released claims state, one floating-point step (for the Kalman filter two) above the largest value of the curve after the deadline. The thin curves are the largest errors observed on the records, and the vertical line marks the deadline. The observer has the smaller guaranteed limit, although the largest observed errors are similar. (c) Each design relative to the best one on each criterion: the exact steady-state mean-square errors for the stated noise model, and the certified upper bounds of (b). The preferred method changes with the requested criterion.

impossibility claim, in each of the 11 families and on the 266 cases of the four held-out families, and every designed category ended with its correct outcome. With a user who knew only the stated ranges, the full framework made 0 incorrect releases; 46 requests whose missing facts this user did not know ended without a design, and useful coverage was 87.6%.

None of the 600 structured requests specified a coupling constraint. In a separate double-integrator replay with the linked acceleration bounds described above, the archived implementation returned a zero speed-error bound. For the admissible trajectory $x(t) = t^2/2$ starting from rest, the speed error at $0.1\,\mathrm{s}$ was $0.05350\,\mathrm{m/s}$, exceeding the requested limit of $0.001\,\mathrm{m/s}$.

These totals equal those of the fifth to seventh runs. Between runs, rows changed where a rule was refined: in the 20 requests above, every policy proves the requirement unattainable, and in 4 probability requests the Khalil specialist alone reports “no verified design found” (the full framework releases the Kalman filter for them). The eighth run, made with the requirement window, the timestamp check, the first-sample construction and the input rule described in this article, reproduced the seventh row by row (0 structured and 0 language rows changed); every change between runs is listed row by row with the software.

Each specialist alone made no incorrect release but refused 196 and 158 requests that the full framework served. The fixed-priority selector was as correct, but in 4 cases released the Kalman filter where the observer had the better value of the requested criterion: guaranteed limits 5 to 14% larger in 3, and a provable probability level of 0.96 instead of 1.00 in one. Asking for every missing field reached the same outcomes with 1.53 questions per request; the coordinator asked 0.32, 79% fewer. Never asking released 158 incorrect designs among 338 and left 41 requests open, because a missing sample period or noise level has no default.

C. The Language Layer

The trained reader gave the correct role to 90% of the numbers in the worded requests. Its first readback was entirely correct for 49% of the requests (mean over seeds, range 47 to 50%; 38% on held-out families, 23% for keyword rules), and the scripted evaluator corrected at least one field in the other 51%, with 2.0 corrections on average when one was needed; keyword rules needed 0.65 to 0.79 more corrections per request (intervals excluding zero for 3 of 3 seeds). With attentive review every reader reached the oracle’s outcome, as the scripted reviewer guarantees. The single-model baseline made 9 incorrect releases and 3 false impossibility claims even so: what holds for the midpoint of a range need not hold for every admissible value.

Without review, the trained reader led to 14, 23, 19 incorrect releases and 6, 8, 6 false impossibility claims for the three seeds, keyword rules to 28 and 1, and frozen vectors to 47

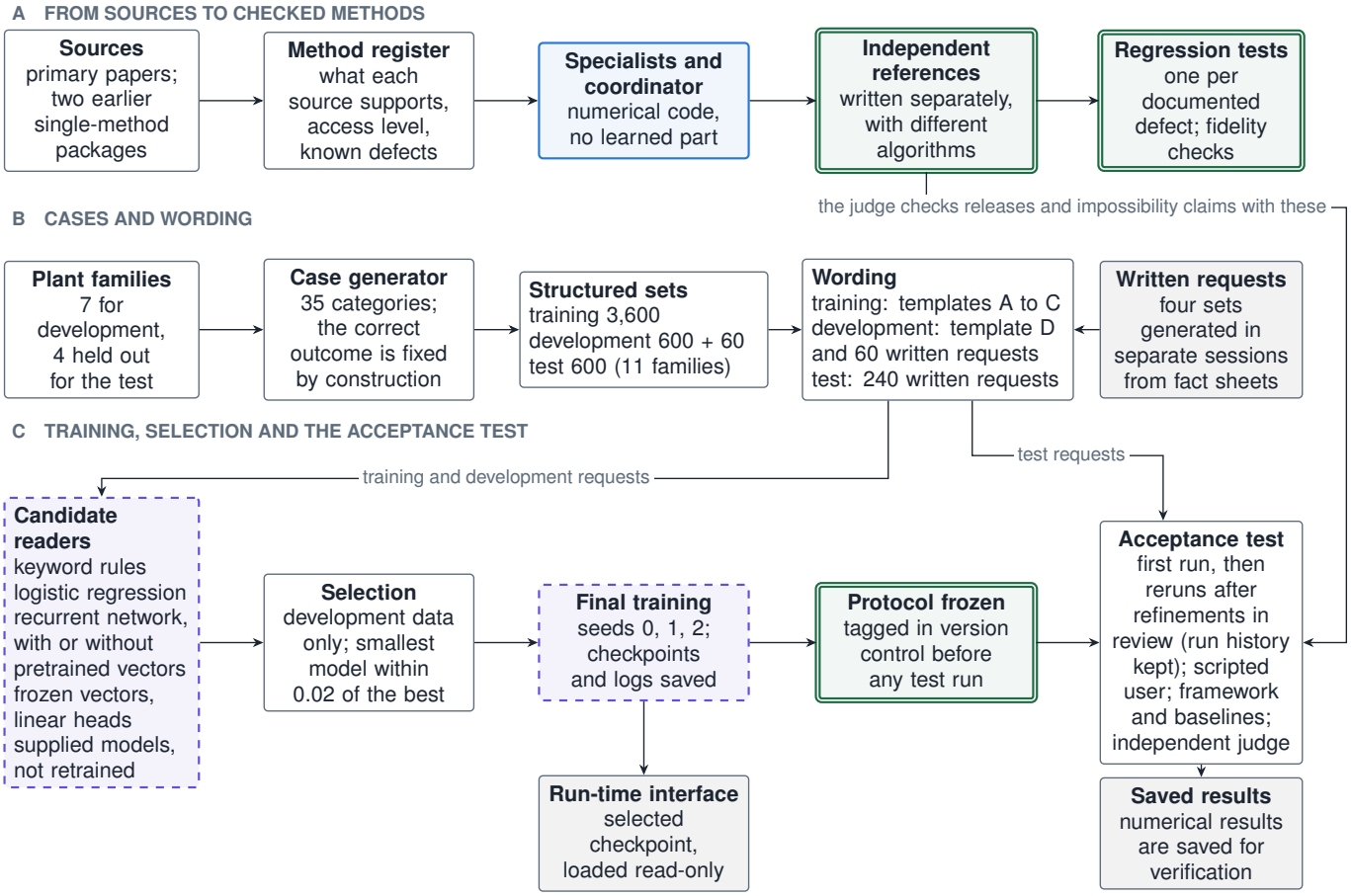


Fig. 5. From sources to the acceptance test. (A) Methods are implemented from a source register and checked against independently written references. (B) Cases are generated with their correct outcome fixed by construction; model families and wording are split before training, and the test requests are generated from fact sheets in separate sessions of the same language model. (C) Candidate readers are trained and one is selected on development data; the protocol and the checkpoints are frozen in version control before the first test run; the reported run is the eighth, after refinements made in review, and the run history is kept with the software. Learned components are dashed; the numerical methods (blue) and the checks (green) are not trained.

and 12. Paired on the same requests, the trained reader's share of incorrect releases was lower than that of keyword rules by 2.2 to 6.1 percentage points (no interval excludes zero). Its incorrect releases fell in 16 categories over the seeds, for example a probability requirement read as a mean-square one and a missed statement that the noise is colored. In 18 of the 20 false impossibility claims, the reader had turned a number into a cap at every sample (in 16 of them the tolerance that applies after the deadline), and the limit then holds for the request as read but not as meant. Each of these misreadings is visible in the readback, which is why every reading is confirmed before it is used.

A decision takes a median of 4 ms, and over 1 s for 32 structured requests with long interval certificates (at most 8.4 s); per sample, the released estimators need 82 μs (Kalman filter, four states) and 6 μs (observer) in Python. The joint request of the next section, from the three answers to the exported record, took 1.1 s on one core, and its replay recomputes 1 682 quantities in 0.5 s.

D. Fresh Requests With Independently Derived Answers

Twelve requests outside the acceptance test, V01 to V12, check the framework against answers obtained without it: 4 admit a design, 3 lack exactly one fact, and 5 must be declined. Their expected answers were derived in separate scripts, from closed forms, a short recursion or a stated construction, and committed together with the software of revision 2, the second revised version (tags validation-frozen and revision-2-frozen, both on one commit), before that software first answered them. Each request went through the public interface as a user would send it, one with a data file whose timestamps contradict the stated period. In that first run all 12 answers agreed with the expected ones. Rerun on revision 2c, the version reported here (tag revision-2c-frozen), all 12 still agree, with the same answers (Table IV), and every returned bound or limit equals the value recomputed with the same independent formulas. When the encoder noise bound is missing, the coordinator asks for it before certifying the velocity error; when a requirement applies at the first sample, it proves that no estimator can meet it; and when only Chebyshev's inequality is available, it declines to certify a probability level without claiming that the level is unattainable.

TABLE III
ACCEPTANCE TEST OF THE EIGHTH RUN, WITH THE PROTOCOL FROZEN BEFORE THE FIRST RUN.

Policy or reader	n	Incorrect releases			Useful	False	Unneeded	Open	Questions	Corrections
		of releases	Wilson CI	cluster CI	coverage (%)	impossible	refusals		(answers)	per request
Structured specifications (correct fields, no language layer)										
Full framework (also the oracle for the rows below)	600	0/370	0.0–0.6	–	100.0	0	0	0	0.32 (0.32)	–
Full framework, user knows only the stated ranges	600	0/324	0.0–0.6	–	87.6	0	46	0	0.32 (0.24)	–
Selector with fixed priority, same checks	600	0/370	0.0–0.6	–	100.0	0	0	0	0.32 (0.32)	–
Kalman specialist alone	600	0/174	0.0–0.6	–	47.0	0	196	0	0.14 (0.14)	–
Khalil specialist alone	600	0/212	0.0–0.6	–	57.3	0	158	0	0.26 (0.26)	–
Ask for every unresolved field	600	0/370	0.0–0.6	–	100.0	0	0	0	1.53 (0.37)	–
Never ask (defaults and midpoints)	600	158/338	23.0–30.0	–	48.6	0	9	41	0.00 (0.00)	–
Worded requests, readback reviewed attentively by the scripted user (consistency check)										
Oracle: correct structured specification of the same requests	229	0/145	–	–	100.0	0	0	0	0.31 (0.31)	–
Trained reader, seed 0	229	0/145	–	–	100.0	0	0	0	0.42 (0.42)	1.00
Trained reader, seed 1	229	0/145	–	–	100.0	0	0	0	0.40 (0.40)	1.13
Trained reader, seed 2	229	0/145	–	–	100.0	0	0	0	0.36 (0.36)	1.00
Frozen vectors, linear heads (no trained encoder)	229	0/145	–	–	100.0	0	0	0	0.43 (0.43)	2.65
Keyword rules	229	0/145	–	–	100.0	0	0	0	0.50 (0.50)	1.78
Reads nothing (every fact entered at the readback)	229	0/145	–	–	100.0	0	0	0	0.54 (0.54)	6.38
Single model chooses the method (trained reader, seed 0)	229	9/109	2.1–7.3	1.5–6.6	69.0	3	33	0	0.49 (0.48)	1.00
Ask for every unresolved field (trained reader, seed 0)	229	0/145	–	–	100.0	0	0	0	1.47 (0.48)	1.00
Worded requests, readback confirmed without reading (stress test)										
Trained reader, seed 0	229	14/137	3.7–10.0	2.9–9.5	84.8	6	9	0	0.68 (0.66)	–
Trained reader, seed 1	229	23/136	6.8–14.6	6.5–13.9	77.9	8	12	0	0.69 (0.67)	–
Trained reader, seed 2	229	19/135	5.4–12.6	4.9–11.8	80.0	6	14	0	0.57 (0.55)	–
Frozen vectors, linear heads (no trained encoder)	229	47/132	15.8–26.2	14.7–26.8	58.6	12	22	0	1.03 (0.98)	–
Keyword rules	229	28/131	8.6–17.1	6.6–18.1	71.0	1	21	0	1.12 (1.06)	–
Reads nothing	229	27/167	8.2–16.6	7.0–16.6	96.6	0	4	1	3.73 (5.11)	–
Single model chooses the method (trained reader, seed 0)	229	20/100	5.7–13.1	5.1–12.3	55.2	7	42	0	0.55 (0.52)	–

"Incorrect releases" are judged against the hidden complete specification with the independent references and given as a count out of the released designs. Intervals (CI) are 95% confidence intervals for the share of incorrect releases per request, in percent of n: Wilson intervals [18] over requests, and for the natural-language rows a cluster bootstrap over groups of requests with the same plant family and request set. With attentive review the scripted user corrects every field to the stated facts, so the coordinator rows of that group reproduce the oracle by construction and their intervals are omitted. Useful coverage is the share of requests that admit a verified design and ended with a correct release. "False impossible" counts impossibility claims refuted by the independent check. "Unneeded refusals" are requests that admit a verified design but were refused; "open" counts dialogues that ended with an unanswered question. Questions (with the values answered: one question can ask for several, and a question about a field the case does not have gets none) and readback corrections are counted per request. Per-family, per-set and per-category results, coverage intervals and refusals for reasons other than the designed one are in the results files.

TABLE IV
VALIDATION REQUESTS V01 TO V12, FIRST RUN ON REVISION 2 AND RERUN ON REVISION 2C (TAG REVISION-2C-FROZEN) THROUGH THE PUBLIC INTERFACE.

	Request	Expected	Returned	Mathematical reason
V01	Random-walk level, finite record; smallest mean-square error	release	release	Kalman recursion from P_0; largest $P_{k\|k}$ at $t = 0$
V02	AR(1) level; steady-state MSE ≤ 0.5	release	release	scalar Riccati equation gives 0.375
V03	Axis, $D = 4$, $V = 10^{-4}$; speed error eventually ≤ 0.06	release	release	exact worst case of one observer gain below 0.06
V04	Axis, total $D = 6$ and additive $d = 1.5$; ≤ 0.045	release	release	exact model: the additive bound applies
V05	Axis, guaranteed limit; encoder bound missing	ask V	ask V	every gain passes v_k with nonzero weight
V06	Level gauge, smallest MSE; noise variance missing	ask R	ask R	gain $P/(P + R)$ depends on R
V07	Axis, limit after 20 ms; initial error missing	ask E	ask E	initial error still weighted at the deadline
V08	Axis, speed error eventually ≤ 0.035	impossible	impossible	two-signal limit $2\sqrt{DV} = 0.04$
V09	Axis, never above 0.5; initial speed ± 2	impossible	impossible	one reading carries no speed information
V10	AR(1) level; steady-state MSE ≤ 0.3	impossible	impossible	optimum 0.375 over all estimators
V11	Axis, $\|e\| \leq 0.02$ with probability 0.9	no verified design	no verified design	Chebyshev proves 0.84; no impossibility argument
V12	Request at 1 kHz, record every 2 ms	data conflict	data conflict	certificate is for one period

The expected answers were derived without the software and committed together with the revision 2 software (commit cb8d3e3, tags validation-frozen and revision-2-frozen) before the first run, in which 12 of 12 agreed; on revision 2c (tag revision-2c-frozen), 12 of 12 agree, with the same answers. For each release the returned bound was recomputed with the same independent formulas.

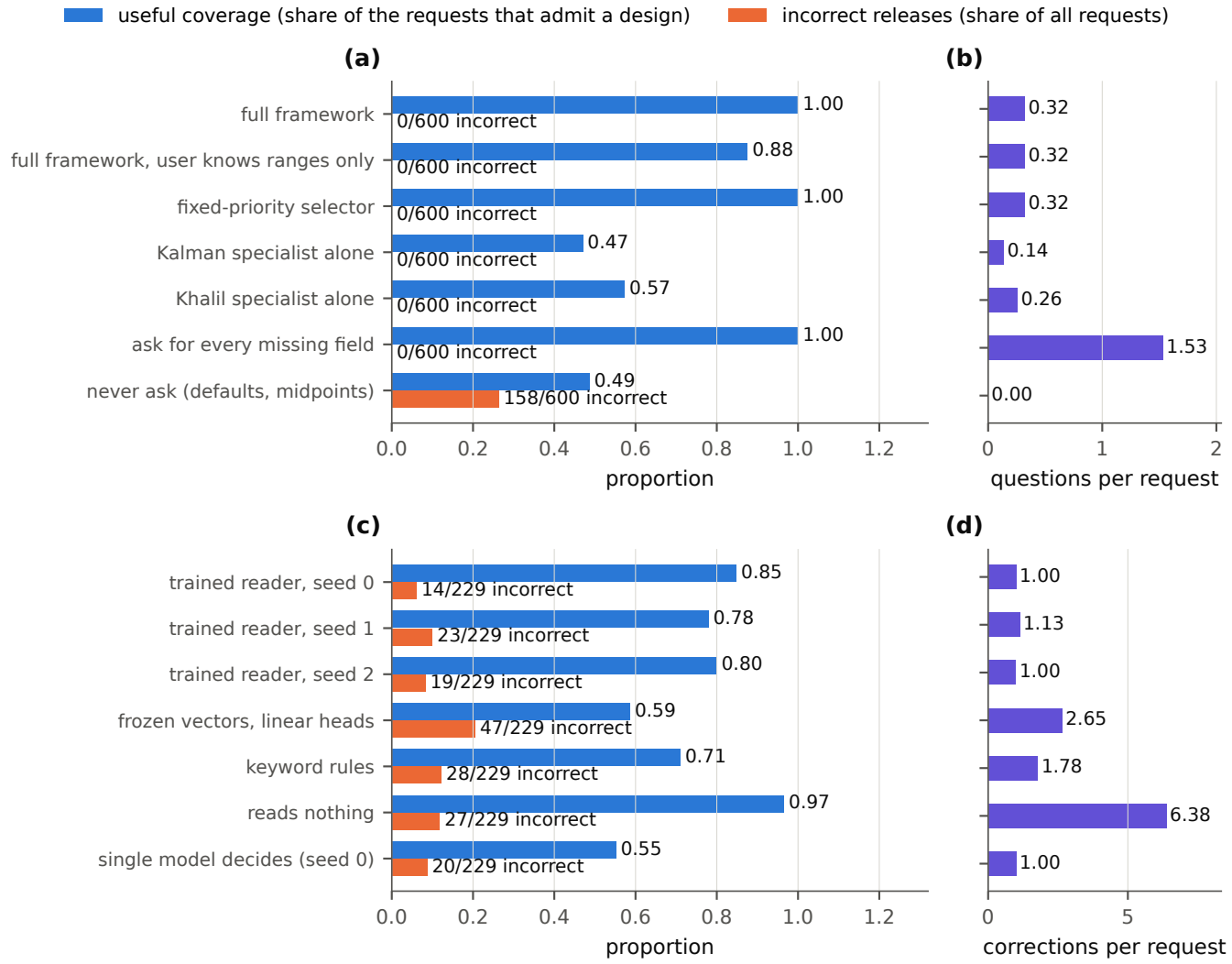


Fig. 6. Acceptance test. (a) Structured specifications: useful coverage (blue, share of the requests that admit a design) and incorrect releases (orange, share of all requests, with counts) for the full framework, the full framework with a user who knows only the stated ranges, and the baselines. (b) Questions asked per request. (c) Worded requests when the scripted user confirms the readback without reading it (stress test): every reader then leads to incorrect releases; with the attentive review of panel (d) the coordinator makes none. (d) Corrections per request needed by an attentive reviewer; the single-model baseline, which skips the comparison across methods and replaces ranges by midpoints, released 9 incorrect designs even with attentive review.

VIII. A Simulated Rotary Joint in the Loop

The last example follows a request into a feedback loop on a simulated single-link joint with gravity and friction, driven by a torque and measured by an encoder at 1 kHz; the true parameters differ from the nominal ones given to the observer. The engineer asks for the joint velocity with a guaranteed error of at most $0.25\,\mathrm{rad/s}$ from 100 ms after start-up, states an encoder error bound of $10^{-4}\,\mathrm{rad}$, a bound of $0.5\,\mathrm{rad/s^2}$ on unmodeled disturbance accelerations, and a hardware speed limit of $4\,\mathrm{rad/s}$ as the evidence for the operating range.

The request omits a bound on the initial velocity error. Because a deadline is stated, the coordinator asks for it before certifying the velocity error. With the answer ($2\,\mathrm{rad/s}$), the Khalil specialist derives the total disturbance bound $D = 27.96\,\mathrm{rad/s^2}$ by interval arithmetic over the operating and parameter ranges (27.26 of it model mismatch) and releases a sampled observer with $\varepsilon = 1.366\,\mathrm{ms}$ ($T/\varepsilon = 0.73$), started from the first encoder reading, with a certified limit of $0.120\,\mathrm{rad/s}$ after the deadline. Over 100 simulated records with different measurement-noise realizations, the largest error after the deadline was $0.069\,\mathrm{rad/s}$, a factor of 1.73 below the certified limit, with 0 violations (Fig. 2(c)). The largest speed

TABLE V
Closed loop on the simulated joint.

	Estimation error [rad/s]		Tracking error [mrad]		Torque variation
Velocity used	largest	RMS	largest	RMS	[N m]
released observer	0.061	0.021	64.7	36.5	61.6
filtered difference	0.086	0.028	60.2	34.5	28.9
true velocity	–	–	61.4	34.5	5.9

Velocity-estimation and tracking errors after the 100 ms deadline, and the total variation of the applied torque, from three closed-loop runs under matched conditions.

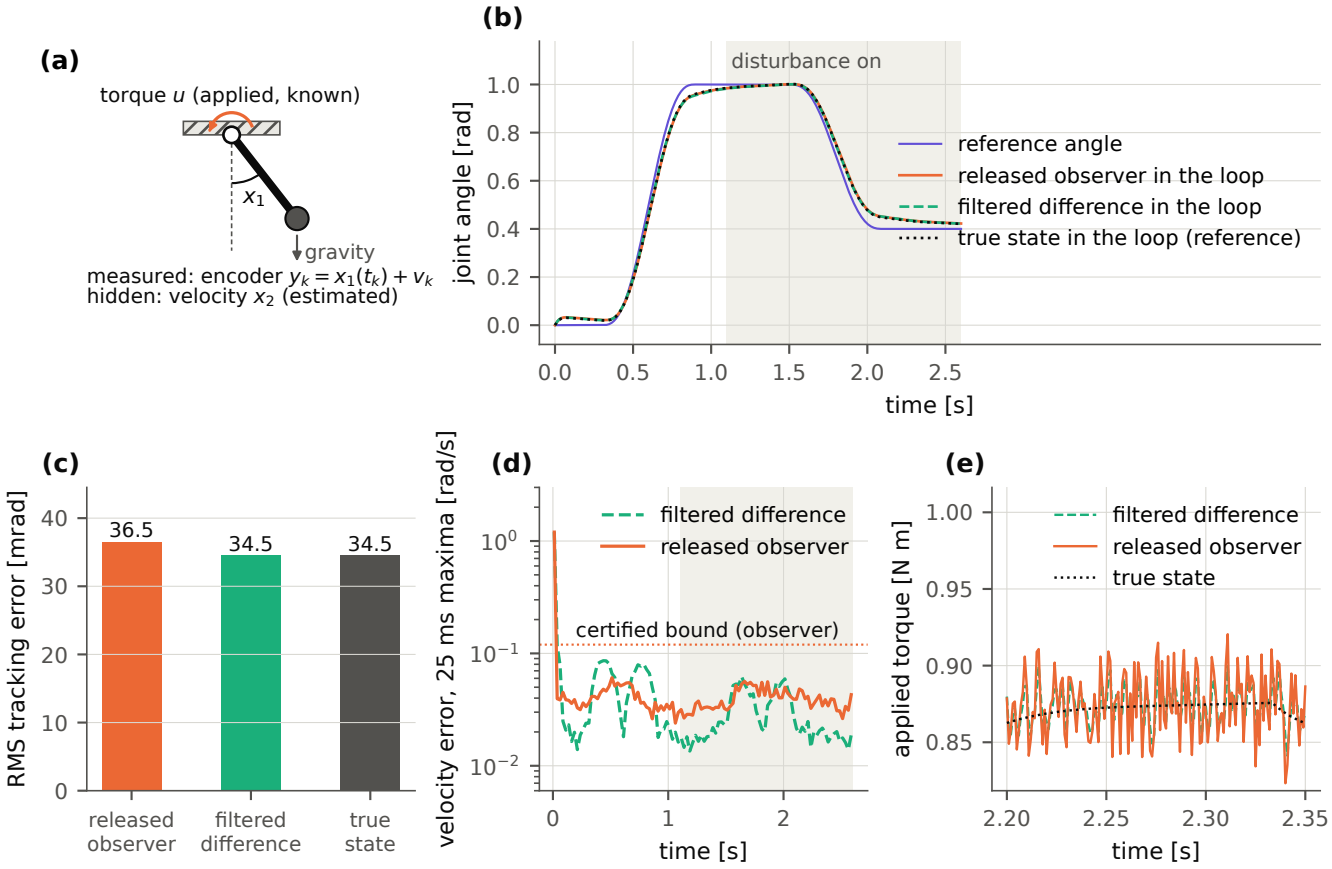


Fig. 7. Closed-loop demonstration on a simulated joint (synthetic parameters). (a) Geometry, measured and hidden quantities. (b) Reference and joint angle for the three feedback designs; the shaded interval marks an added disturbance acceleration. (c) RMS tracking error. (d) Velocity-estimation error in the loop (maxima over 25 ms windows) with the certified limit of the released observer. (e) Applied torque on a short interval: the observer-based loop is the most active. The three curves in each panel come from three separate closed-loop runs under matched conditions.

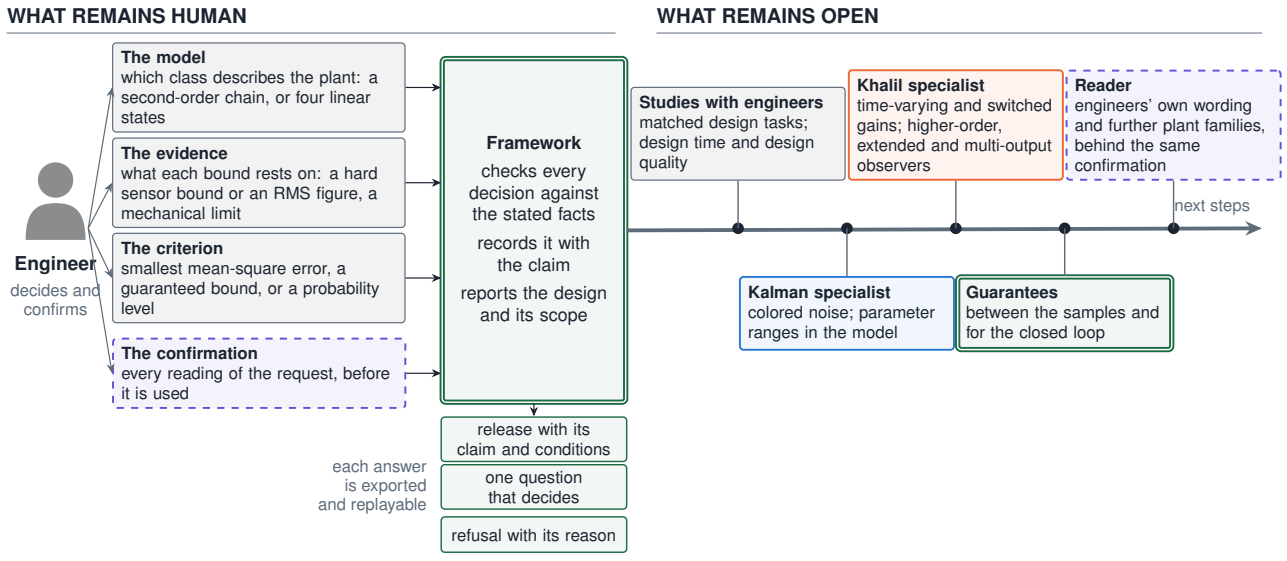


Fig. 8. What remains human, and what remains open. (Left) The engineer decides the model, the evidence behind every bound and range, and the criterion, and confirms every reading of the request; the framework checks each decision against the stated facts, records it with the claim, and reports a design with its conditions, asks a clarifying question, or explains why it cannot release a design. (Right) Next steps, each drawn in the color of the part of the framework that it extends (Figs. 5 and 3): gray for people, blue for the Kalman specialist, orange for the Khalil specialist, green for the guarantees and violet (dashed) for the reader.

was 2.91 $\mathrm{rad/s}$, inside the range on which the mismatch bound rests; a run-time monitor reports any exit.

The estimator was fixed before the closed-loop runs. Fig. 7 and Table V compare three feedback designs under identical reference, disturbance, noise and initial condition: a computed-torque tracking law (double closed-loop pole at $-12\,\mathrm{rad/s}$) using the released observer, the same law using a filtered difference of the encoder signal (backward difference and a 40 Hz first-order low-pass filter), and the same law using the true velocity. Estimation and tracking answer different questions and are reported separately.

Estimation error. After the deadline the observer's velocity estimate is the more accurate one: its largest error in the loop is $0.061\,\mathrm{rad/s}$, inside the certified limit of $0.120\,\mathrm{rad/s}$, against $0.086\,\mathrm{rad/s}$ for the filtered difference, which carries no guarantee.

Tracking error. The root-mean-square (RMS) tracking error after the deadline is $36.5\,\mathrm{mrad}$ with the observer, $34.5\,\mathrm{mrad}$ with the filtered difference and $34.5\,\mathrm{mrad}$ with the true velocity, and the largest tracking errors are 64.7, 60.2 and $61.4\,\mathrm{mrad}$. The total variation of the torque is $61.6\,\mathrm{N\,m}$ with the observer, against 28.9 and 5.9; no run reached the torque limit of $6\,\mathrm{N\,m}$.

With the same feedback law, the observer gave a lower RMS velocity-estimation error than filtered differentiation (0.021 versus $0.028\,\mathrm{rad/s}$), yet the joint tracked the reference slightly less accurately and the total torque variation more than doubled. The certificate bounds velocity-estimation error at the specified sample instants under the stated assumptions. Establishing closed-loop stability or a bound on tracking error would require a separate analysis of the observer and controller together.

IX. What Remains Human, and What Remains Open

The framework makes the judgment that estimation requires visible and checkable (Fig. 8). Three decisions stay with the engineer. *The model:* whether a joint is a second-order chain, or a boom four linear states, is a modeling decision that the specification records. *The evidence:* a hard bound on sensor error is a different commitment from an RMS figure in a data sheet, and a statement that a trajectory stays in a range rests on a reason, such as a mechanical limit, which the coordinator records with the claim. *The criterion:* the justified method follows from what is asked. The engineer also confirms every reading of the request, and the framework turns a statistical description of uncertainty into a deterministic one, or back, only when the request states the distribution that links them.

Five directions extend the framework, each with its own independent reference (Fig. 8): a controlled study with engineers on matched design tasks; colored noise and parameter ranges for the Kalman specialist; time-varying and switched gains and the higher-order, extended and multi-output observers of [19] for the Khalil specialist; guarantees between the samples and for the closed loop, for which the nonlinear separation principle [20] is a natural starting point; and a reader trained on requests worded by engineers and on further plant families.

X. Conclusion

Kalman's filter and Khalil's high-gain observer answer the same question under different descriptions of uncertainty. For the encoder of the opening example, the coordinator releases the Kalman filter when the smallest mean-square error is asked for, the observer when a guaranteed error bound is asked for, and a proof that no estimator suffices when the bound lies below what two indistinguishable signals allow. Specialists checked against separately written references, a coordinator that compares designs against the requested criterion and reports the assumptions behind its decision, and a reader whose every reading the engineer confirms keep this knowledge executable. The code, data, trained models, benchmark results, and scripts for the numerical experiments are available from the archived GitHub release.

Appendix A
What a Kalman Filter Promises

For the model (1) three statements are often blurred; the Kalman specialist issues exactly one of them, chosen from the stated assumptions.

Gaussian case. If the initial state and both noise sequences are jointly Gaussian, $\hat{x}_{k|k} = \mathbb{E}[x_k \mid y_0, \dots, y_k]$. For any other estimate z computed from the same measurements, $\mathbb{E}\|x_k - z\|^2 \geq \operatorname{tr} P_{k|k}$. This is the statement of minimum mean-square error among *all* estimators [1].

Second-order case. If only means and covariances are known, the recursion gives the orthogonal projection of x_k on the affine functions of the data. It is the best *linear* estimator; a nonlinear estimator may do better, and the specialist says so.

What P is not. $P_{k|k}$ is the error covariance when the model is correct. With Gaussian errors, a band of two standard deviations contains the error with probability 0.954 at each instant; with second moments only, Chebyshev's inequality guarantees 0.75. Neither is a bound that holds for every realization.

Simulation can check the covariance, not the optimality claims. On a simulated hydraulic boom with four states and two sensors (synthetic parameters), 200 runs each with Gaussian and with uniform noise of the same covariances gave an average normalized squared error $e^{\mathsf{T}} P^{-1} e$ of 3.94 and 3.97 (expected value 4), and two-standard-deviation bands contained the boom-rate error in 95.4% and 95.9% of the samples. The coverage under uniform noise is a property of this example, not a promise, so without a Gaussian assumption the specialist states only the linear-estimator property and the Chebyshev level.

Appendix B
An Exact Worst Case for the Sampled Observer

Consider $\dot{x}_1 = x_2$, $\dot{x}_2 = a(t)$ with measurements $y_k = Cx(kT) + v_k$, $C = [1\ 0]$, transition matrix A_d as above, and an estimator that updates and predicts with a constant gain M and a nominal acceleration a_k^0 held on $[kT, (k+1)T)$. Let

$\delta(t) = a(t) - a_k^0$ be the total disturbance, e_k^- the prediction error and e_k the error after the update. Then, exactly,

$$e_k = (I - MC)\, e_k^- - M v_k, \tag{7}$$

$$e_{k+1}^- = A_d\, e_k + \int_0^T \begin{bmatrix} T - s \\ 1 \end{bmatrix} \delta(kT + s)\, \mathrm{d}s. \tag{8}$$

Suppose $|e_{0,1}^-| \le E_1$, $|e_{0,2}^-| \le E_2$, $|v_k| \le V$ and $|\delta(t)| \le D$, and nothing else is known. Write $\Phi = A_d(I - MC)$ and $\rho_j = c^{\mathsf{T}}(I - MC)\Phi^j$, where c selects the error component of interest ($c = [0\ 1]^{\mathsf{T}}$ for the velocity).

Proposition 1: The largest value of $|c^{\mathsf{T}} e_N|$ over all admissible initial errors, measurement errors and disturbances is

$$W_N = |\rho_N| \begin{bmatrix} E_1 \\ E_2 \end{bmatrix} + V\, |c^{\mathsf{T}} M| + \sum_{j=0}^{N-1} \Big(D \int_0^T \big| \rho_j \big[\begin{smallmatrix} \tau \\ 1 \end{smallmatrix} \big] \big|\, \mathrm{d}\tau + V\, |\rho_j A_d M| \Big), \tag{9}$$

where $|\cdot|$ acts entrywise on ρ_N. The value is attained.

Proof: Unrolling the recursion expresses $c^{\mathsf{T}} e_N$ as a sum of terms, each linear in one of the independent unknowns e_0^-, $v_0, \ldots, v_N$, and the restrictions of δ to the N sampling intervals. The admissible set is a product of intervals symmetric about zero, so the supremum of the sum is the sum of the suprema, and each is attained by aligning signs: $v_k = \pm V$, and $\delta = \pm D$ with at most one sign change per interval, because the weight $\rho_j[\tau\ 1]^{\mathsf{T}}$ is affine in τ. ∎

Start from the first reading. The observer usually sets $\hat{x}_1(0) = y_0$; then $e_0 = [-v_0 \quad e_{0,2}]^{\mathsf{T}}$, so the initial position error is the first measurement error, not an independent quantity. The same argument applied to the independent unknowns $e_{0,2}, v_0, \ldots, v_N$ and δ gives, for $N \ge 1$, the exact worst case with $|\rho_N|[E_1\ E_2]^{\mathsf{T}} + V|\rho_{N-1}A_d M|$ replaced by $|\rho_{N-1}A_d|[V\ E_2]^{\mathsf{T}}$, and $W_0 = |c|^{\mathsf{T}}[V\ E_2]^{\mathsf{T}}$; it is attained by sign-aligned signals as before. Proposition 1 with $E_1 = V$ treats v_0 twice and therefore remains a valid upper bound for this start; it exceeds the exact value only during the first samples (on the servo axis below by 0.74% at $t = 0$). Figs. 2 and 4 plot these exact values, for the stated uncertainty set, as curves; a released number, shown there as a marker, is their interval-certified upper bound. A simulation whose disturbance has more structure, such as an acceleration held over each sample, is a narrower case, for which the curve is an upper bound.

The result is elementary: W_N is a weighted ℓ_1 sum whose limit is the weighted peak-to-peak gain of the sampled error system [21], with the disturbance inside each sampling interval treated exactly, a special case of the L_1 analysis of sampled-data systems [22]; worst-case estimation under unknown-but-bounded errors goes back to [23]. Each integral is that of the absolute value of an affine function, and for a stable Φ the tail beyond a finite horizon is bounded by a geometric series. Sign-aligned admissible signals reproduced the bound to within relative gaps of 3.1×10^{-13} (independent initial errors) and 3.1×10^{-13} (first-reading start).

Lower limits complement the upper one. It is shown in [17], with two admissible signals that produce the same measurements, that no causal estimator can guarantee a worst-case velocity error below $2\sqrt{DV}$ from $4\sqrt{V/D}$ after start-up on, when the acceleration is an arbitrary signal bounded by D; sampling only removes information. The coordinator claims this limit only when the pair of signals is admissible for the stated plant over the whole window: a worst-case requirement, a sample of the requested window at or after $4\sqrt{V/D}$, D the smallest stated bound on the free part of the acceleration, a rest state inside the stated travel, speed, input and initial-error bounds, and travel for the braking after the sample ($3V$ on each side of the rest state when the window continues). A second limit concerns $t = 0$: one position reading carries no velocity information, so no estimate is closer than the half-width of the initial velocities that admissible trajectories can have. A known travel range restricts them: a carriage that starts at the center of a $\pm 0.1\,\mathrm{m}$ travel and must stay inside it for $1\,\mathrm{s}$ at constant speed cannot move faster than $0.1\,\mathrm{m/s}$. With a stated travel or speed range the coordinator therefore uses only velocities from which explicit trajectories stay inside every range for the whole window, built with outward-rounded interval bounds: braking with an admissible input, every velocity on the way inside a stated speed range, and resting with the input then held, or, for a restoring force with damping, holding one input while the energy stays below its value at both ends of the travel. The construction approximates the admissible set from inside, so a claim made with it is valid, and where it does not reach the requirement no limit is claimed.

Appendix C
The Same Data, Two Different Answers

One measurement $y = x + v$ is taken of a zero-mean Gaussian quantity x with variance σ_x^2; the noise v is independent, zero-mean, Gaussian with variance σ_v^2. The conditional mean is

$$\mathbb{E}[x \mid y] = \frac{\sigma_x^2}{\sigma_x^2 + \sigma_v^2}\, y. \tag{10}$$

Suppose the request states that the measurement has variance 2 and that the noise variance lies between 1 and 1.5. The admissible completions are the pairs $(\sigma_x^2, \sigma_v^2) = (2 - r, r)$ with $1 \le r \le 1.5$; the extreme ones are $(0.5, 1.5)$ and $(1, 1)$. All produce the same distribution of y, so no amount of recorded data distinguishes them, yet the optimal estimates range from $y/4$ to $y/2$.

Whether this matters depends on what was asked (Fig. 9). An estimate Ky has mean-square error $(1 - K)^2\sigma_x^2 + K^2\sigma_v^2$. If the request is for the *smallest* mean-square error, no single K is optimal for every admissible completion ($K = 1/4$ gives 0.625 under the second, where 0.5 is attainable), and the coordinator must ask for the noise variance. If the request is only for a mean-square error *below* 0.65, the gain $K = 1/4$ meets it for every admissible completion (its error is affine in r and lies between 0.375 and 0.625) and can be released without a question. The stated range matters: if only the sum were known, the completion $(1.5, 0.5)$ would give this gain an error of 0.875; only gains between about 0.43 and 0.57 would qualify, $K = 1/2$ giving 0.5 for every split.

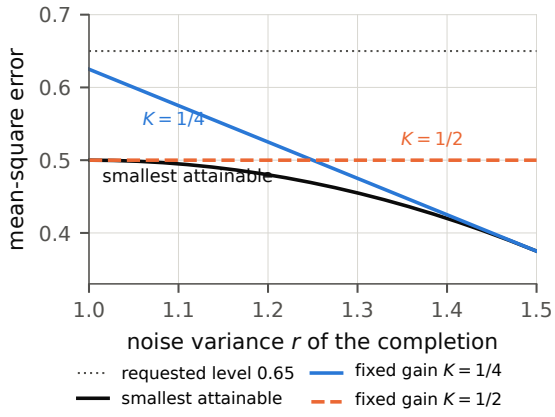


Fig. 9. The same data, two different answers. Mean-square error of a fixed gain K over the admissible completions $(\sigma_x^2, \sigma_v^2) = (2-r, r)$, $1 \le r \le 1.5$. The smallest attainable error (black) is reached by $K = 1/2$ only at $r = 1$ and by $K = 1/4$ only at $r = 1.5$, so a request for the smallest error needs the noise variance; both gains stay below the requested level of 0.65 (dotted) at every completion, so a request for an error below 0.65 is met without a question.

REFERENCES

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[2] R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME, J. Basic Eng.*, vol. 83, no. 1, pp. 95–108, Mar. 1961.

[3] F. Esfandiari and H. K. Khalil, "Output feedback stabilization of fully linearizable systems," *Int. J. Control*, vol. 56, no. 5, pp. 1007–1037, 1992.

[4] H. K. Khalil and L. Praly, "High-gain observers in nonlinear feedback control," *Int. J. Robust Nonlinear Control*, vol. 24, no. 6, pp. 993–1015, 2014.

[5] H. K. Khalil, "High-gain observers in feedback control: Application to permanent magnet synchronous motors," *IEEE Control Syst. Mag.*, vol. 37, no. 3, pp. 25–41, Jun. 2017.

[6] K. J. Åström, J. J. Anton, and K.-E. Årzén, "Expert control," *Automatica*, vol. 22, no. 3, pp. 277–286, 1986.

[7] J. Miao, J. R. Davis, Y. Zhang, J. K. Pritchard, and J. Zou, "Paper2Agent: Reimagining research papers as interactive and reliable AI agents," arXiv:2509.06917v2, Oct. 2025.

[8] J. Miao, J. R. Davis, Y. Zhang, J. K. Pritchard, and J. Zou, "Reimagining research papers as interactive and reliable AI agents," *Nature*, 2026, doi: 10.1038/s41586-026-11044-y.

[9] S. Li, J. Li, J. Xu, and D. Chen, "From natural language to certified H_∞ controllers: Integrating LLM agents with LMI-based synthesis," arXiv:2511.07894v1, Nov. 2025.

[10] X. Guo, D. Keivan, U. Syed, L. Qin, H. Zhang, G. Dullerud, P. Seiler, and B. Hu, "ControlAgent: Automating control system design via novel integration of LLM agents and domain expertise," arXiv:2410.19811v1, Oct. 2024.

[11] M. S. Chong, D. Nešić, R. Postoyan, and L. Kuhlmann, "Parameter and state estimation of nonlinear systems using a multi-observer under the supervisory framework," *IEEE Trans. Autom. Control*, vol. 60, no. 9, pp. 2336–2349, Sep. 2015.

[12] L. K. Vasiljevic and H. K. Khalil, "Error bounds in differentiation of noisy signals by high-gain observers," *Syst. Control Lett.*, vol. 57, no. 10, pp. 856–862, 2008.

[13] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1979.

[14] C. F. Van Loan, "Computing integrals involving the matrix exponential," *IEEE Trans. Autom. Control*, vol. 23, no. 3, pp. 395–404, Jun. 1978.

[15] A. M. Dabroom and H. K. Khalil, "Discrete-time implementation of high-gain observers for numerical differentiation," *Int. J. Control*, vol. 72, no. 17, pp. 1523–1537, 1999.

[16] A. M. Dabroom and H. K. Khalil, "Output feedback sampled-data control of nonlinear systems using high-gain observers," *IEEE Trans. Autom. Control*, vol. 46, no. 11, pp. 1712–1725, Nov. 2001.

[17] R. Seeber and H. Haimovich, "Optimal robust exact differentiation via linear adaptive techniques," *Automatica*, vol. 148, Art. no. 110725, 2023.

[18] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," *J. Amer. Statist. Assoc.*, vol. 22, no. 158, pp. 209–212, 1927.

[19] H. K. Khalil, *High-Gain Observers in Nonlinear Feedback Control*. Philadelphia, PA, USA: SIAM, 2017.

[20] A. N. Atassi and H. K. Khalil, "A separation principle for the stabilization of a class of nonlinear systems," *IEEE Trans. Autom. Control*, vol. 44, no. 9, pp. 1672–1687, Sep. 1999.

[21] M. A. Dahleh and J. B. Pearson, "ℓ^1-optimal feedback controllers for MIMO discrete-time systems," *IEEE Trans. Autom. Control*, vol. 32, no. 4, pp. 314–322, Apr. 1987.

[22] G. E. Dullerud and B. A. Francis, "L_1 analysis and design of sampled-data systems," *IEEE Trans. Autom. Control*, vol. 37, no. 4, pp. 436–446, Apr. 1992.

[23] F. C. Schweppe, "Recursive state estimation: Unknown but bounded errors and system inputs," *IEEE Trans. Autom. Control*, vol. 13, no. 1, pp. 22–28, Feb. 1968.

Mehdi Heydari Shahna received the D.Sc. (Tech.) degree with distinction in Automation Science and Engineering from Tampere University, Finland, in 2025. He is currently a postdoctoral researcher at Tampere University. His research interests include physical AI, robot learning, safe control, and robust control.

Jouni Mattila received a Ph.D. in automation engineering from Tampere University of Technology, Tampere, Finland, in 2000. He is currently a professor of machine automation with the Unit of Automation Technology and Mechanical Engineering at Tampere University. His research interests include machine automation, nonlinear-model-based control of robotic manipulators, and energy-efficient control of heavy-duty mobile manipulators.