Journal of Emerging Intelligence and Engineering Technologies

Research Article

Safety-Filter Interventions as Congestion Prices for Delay Accounting, Causal Attribution and Marginal-Cost Routing of Warehouse Mobile Robots

Olasubomi Akanbi[1,*], Toheeb Babatunde Ibrahim[2], Joseph Oladejo[2], Warith Akinade[2]

[1]Department of Systems Engineering, Bowling Green State University, Bowling Green, OH 43403, USA

[2]Department of Mathematics and Statistics, Bowling Green State University, Bowling Green, OH 43403, USA

* Corresponding author. Email: olasubomi.akanbi1@gmail.com (O. Akanbi), toheeb.ibrahim@aims-cameroon.org (T.B. Ibrahim), oladejojoseph0@gmail.com (J. Oladejo), warith@aims.edu.gh (W. Akinade).

Abstract

Autonomous mobile robots in warehouses are protected by control barrier function (CBF) safety filters, which minimally modify each robot's velocity command so that no collision constraint is violated. The modification, which we call the intervention, is computed at every control step and then discarded. We show that it is a congestion price with three properties that occupancy and travel-time signals lack. First, it accounts for delay exactly: along a lane, the time integral of the longitudinal intervention of all control layers, that is, planner holds plus filter corrections, equals the robot's congestion delay. This is a kinematic identity, so on a single lane the signal carries the same information as the speed deficit, and a regularised Little's-law estimator turns it into consistent edge delays while traversals are still in progress. Second, it is attributable: the Karush–Kuhn–Tucker multipliers of the filter and the hold decisions of the planner charge every second of delay to the robot whose constraint produced it, and forwarding along active headway constraints charges queue cascades to their root cause. Since this attribution is budget balanced, pricing a route by the delay its robots suffer plus the delay they impose equals the marginal social cost under pairwise conflicts and gives a conservative price under superlinear congestion. Third, it routes stably: by specialising classical day-to-day assignment stability, we derive the convergence-rate-optimal smoothing gain, its robustness margin and the gain that balances tracking lag against estimation noise. Simulations of a lane network with realistic synthetic inputs, namely order-wave demand generated by time rescaling, lognormal pick times and heterogeneous speeds, checked for collinearity, confirm the delay identity to within one integration step. The attributed imposed delay recovers 97% of the counterfactual marginal external cost at low flow, and the accrued signal improves out-of-sample delay prediction beyond occupancy and flow. Marginal-cost intervention pricing lowers the system travel time at a busy crossing by 3.5% relative to user-equilibrium pricing and recovers 83% of the gain of an oracle price. Online gain tuning did not outperform a conservatively chosen fixed gain.

Keywords: Control barrier functions; Safety filter; Warehouse robotics; Autonomous mobile robots; Congestion pricing; Marginal cost pricing; Traffic assignment

Received: 02 August 2026
Revised: 15 November 2026
Accepted: 20 November 2026
Published: 26 November 2026

Copyright: © 2026 by the authors. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0) license (https://creativecommons.org/licenses/by-nc/4.0/).

Citation: Akanbi, O.; Ibrahim, T.B.; Oladejo, J.; Akinade, W. Safety-Filter Interventions as Congestion Prices for Delay Accounting, Causal Attribution and Marginal-Cost Routing of Warehouse Mobile Robots. *J. Emerg. Intell. Eng. Technol.* 2026, 3(2), 64–97.

Nomenclature

Symbol	*Meaning*
Network and robots	
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	directed graph of the warehouse floor; vertices $\mathcal{V}$, lane segments (edges) $\mathcal{E}$
$e,\ L_e,\ \mathbf{n}_e,\ x_e^0$	edge index; edge length (m); unit direction vector of edge e; entrance point of edge e
$\mathcal{N},\ N,\ i,\ j$	set of robots; number of robots; robot indices
$x_i,\ u_i,\ \mathcal{U}_i$	position of robot i (m); commanded velocity (m/s); set of admissible velocities
$\bar{v}_i,\ m_i(t),\ w_i(t)$	cruise speed (m/s); motion indicator (0 during a planned dwell, 1 otherwise); lateral path-keeping velocity
$s_i,\ \ell(i)$	progress of robot i along its lane (m); leader of robot i on the same lane
$\tau_i,\ T_i,\ D_i$	total planned dwell time on an edge (s); traversal time (s); congestion delay (s)
Control layers and safety filter	
$K,\ k,\ \mathcal{L}_k$	number of control layers; layer index; decision rule of layer k
$u_i^{(0)},\ u_i^{(k)}$	free-flow command; output of layer k (m/s)
$\hat{u}_i,\ u_i^\star$	command entering the safety filter; filtered command (m/s)
$h_{ij},\ h_{io},\ d_s$	robot–robot barrier; static barrier; safety distance (m)
$\alpha,\ \gamma,\ \rho_{ij}$	extended class-$\mathcal{K}$ function; CBF gain (s^{-1}); responsibility share of robot i for pair (i,j)
$A_i,\ b_i,\ a_q,\ b_q$	constraint matrix and vector of the filter of robot i; row q and its right-hand side
$\mathcal{K}_i^{\mathrm{rr}}, \mathcal{K}_i^{\mathrm{st}}, \mathcal{K}_i^{\mathrm{act}}$	robot–robot, static and actuator row sets; $j(q)$ is the neighbour of robot–robot row q
$\lambda_q,\ \nu_q,\ \hat{a}_q,\ \chi_q$	KKT multiplier; normalised multiplier; unit row normal; nominal safety deficit of row q
$\zeta_i,\ w_z,\ d_{\mathrm{app}},\ k_s$	signed distance to a crossing point (m); conflict-zone half-width (m); approach distance (m); stop-line gain (s^{-1})
Interventions, delay and attribution	
$\delta_i^{(k)},\ \delta_i,\ \delta_i^{\mathrm{tot}}$	layer-k intervention; filter intervention; total intervention (m/s)
$\bar{u}_i,\ \delta_i^{\mathrm{env}},\ \delta_i^{\mathrm{peer}}$	filtered command without peer constraints; environment and peer components of δ_i (m/s)
$\iota_i^{(k)},\ \iota_i,\ \eta_i$	longitudinal layer-k intervention; total longitudinal intervention (m/s); normalised intervention
$B_i,\ B_{ij},\ \mathcal{H}_i$	blame vector of robot i; part of ι_i charged to robot j; set of robots holding a conflict zone
$o_\ell,\ \Gamma_\ell$	own-cause speed deficit of leader ℓ; total speed deficit used for forwarding (m/s)
X_j	total delay imposed by robot j on other robots (s)
$T_p,\ N_e,\ I_e,\ \Xi_e$	price-update period (s); entries to edge e in a period; accrued suffered delay; accrued imposed delay (s)
$f_e,\ d_e,\ \xi_e$	entry rate (robots/s); mean suffered delay per robot (s); mean imposed delay per robot (s)
$\hat{d}_e,\ \nu_e,\ d_e^0$	regularised edge-delay estimate (s); pseudo-entry count; prior delay (s)
$\Delta t,\ \gamma_a$	control sampling period (s); CBF gain of barrier a (s^{-1})
Routing and pricing	
$Q,\ Q_H$	routed demand; background (highway) flow (robots/s)
$r,\ r',\ \mathcal{R},\ \Delta$	route or robot-class indices; set of routes; edge–route incidence matrix
$t_e^0,\ p_e,\ C_r$	free-flow time (s); price (s); perceived cost of route r (s)
$\theta,\ \pi_r,\ y_r$	logit dispatch sensitivity (s^{-1}); dispatch probability; route flow (robots/s)
$c_e,\ \beta,\ n,\ \Phi_\beta$	price target function; smoothing gain; period index; price-update map
$\Omega_{rr\prime},\ \mathrm{TC},\ \mathrm{MEC}_r$	pairwise-encounter coefficient; total travel time per unit time; marginal external cost
$\Pi^\star,\ G,\ M$	dispatch covariance matrix; route-to-edge projected covariance; diagonal matrix of target slopes

Symbol	*Meaning*
$\mu,\ \varsigma,\ \vartheta_{\varsigma}$	largest network curvature; curvature of a price mode; $1+\theta Q\varsigma$
$\varrho,\ \beta^{\star},\ \beta_{\mathrm{opt}}$	spectral radius; stability bound on the gain; rate-optimal gain
$\sigma^2,\ \omega$	variance of price-estimation noise (s^2); drift of the equilibrium price per period (s)
$\bar{n}_e,\ \kappa_n$	mean occupancy of edge e (robots); density-price coefficient (s per robot)

1. Introduction

1.1. Background and Motivation

Fleets of automated guided vehicles (AGVs) and autonomous mobile robots (AMRs) move a large share of the material in fulfilment centres, distribution centres and factories [1–4]. Their control is organised in layers [5–7]. A planning layer assigns tasks and routes [8–10]; reservation or traffic rules resolve conflicts at intersections and in aisles [11–13]; and an execution layer keeps every robot clear of racks, people and other robots [14]. Control barrier functions (CBFs) have become a principled tool for the execution layer [15–17]. A CBF safety filter solves a small quadratic program (QP) at every control step and returns the command closest to the nominal one that keeps the system in a safe set [18–21]. Decentralised versions split each pairwise constraint between the two robots involved [22–24], and learned CBFs now serve as the local safety layer under warehouse fleet managers [25].

1.2. Research Gap

Every time a layer modifies a command, it produces a quantity that the rest of the system discards: how much the robot was held back. Congestion is instead estimated from planned paths, reservations or vehicle counts [26–30]. Counts are only a proxy, because the same number of robots can cost very different amounts of time depending on how often they stop to pick and how the safety layer resolves their conflicts. Travel times are exact but become available only after a traversal ends. Neither signal identifies *who* caused a delay, which is precisely what a system-optimal price requires [31–33]. Planning that anticipates safety-filter overrides has been studied for human–robot interaction [21, 34], and the KKT conditions of CBF-QPs have been used to analyse multi-robot deadlock [35]. To our knowledge, however, no published work uses executed safety-filter interventions as route prices, attributes them to the robots whose constraints produced them, or analyses the stability of the resulting price loop.

1.3. Objectives and Contributions

This work treats the intervention as a measurement. We characterise its mathematical structure, establish what it measures, attribute it to the robots that produce it, and analyse whether a fleet can route on it without oscillation. The contributions are as follows.

1. *Lagrangian structure.* We show that the filter intervention is a sum of constraint forces, namely unit constraint normals weighted by row-normalised Karush–Kuhn–Tucker (KKT) multipliers. The multipliers are invariant to row scaling and equal the sensitivity of the filter cost to each barrier, so the intervention can be split uniquely by constraint. An auxiliary projection without peer constraints separates the part of the correction due to other robots from the part due to walls and input limits, and a normalised intervention of one identifies a stop imposed by the control layers, which is necessary but not sufficient for deadlock (Theorem 1, Proposition 2).
2. *Layered delay accounting.* We prove that the integrated longitudinal intervention, summed over all control layers, equals the robot's congestion

delay exactly. The identity is kinematic, and on a lane it coincides with the integrated speed deficit. We derive a regularised Little's-law estimator of edge delay that needs no completed traversals and stays defined in windows without new entries, and we give a sampled-data condition under which the filter keeps the robots safe between control updates (Theorem 2, Proposition 3, Lemma 1).

3. *Causal attribution and marginal-cost prices.* We define direct and root-cause attributions of each robot's intervention to the robots that caused it and prove that both are budget balanced. Pricing a route by suffered plus imposed delay then equals the marginal social cost under pairwise encounters and lies between user-equilibrium and marginal-cost prices under superlinear congestion (Theorems 3 and 4, Proposition 4). Delays have been charged to causing agents in queue-based traffic microsimulation [36, 37]; our attribution is instead derived from the KKT structure of a continuous-space safety filter, runs online on the robots, and comes with exactness conditions. The attribution is an accounting rule that follows binding constraints, not a counterfactual statement, and Theorem 4 states when the two coincide.
4. *Gain design.* Building on the classical stability analysis of day-to-day assignment [38–40], which we specialise to intervention prices (Theorem 6), we derive the rate-optimal gain, its robustness margin, the noise it amplifies, the gain that balances tracking lag against noise (Proposition 5) and a bound on the price error under bounded estimation noise (Theorem 7). We also test an online tuning algorithm and report that it does not outperform a conservative fixed gain.
5. *Numerical evidence.* We evaluate these results on a lane network driven by realistic synthetic inputs whose collinearity we check, including a predictive test, a comparison against a counterfactual marginal external cost, and a routing comparison that includes an oracle price.

1.4. Organisation

Section 2 reviews the literature. Section 3 formulates the system model and the problem. Section 4 develops the methodology. Section 5 describes the experimental design, and Section 6 reports the results. Section 7 discusses implications and limitations, and Section 8 concludes. Appendix A lists the symbolic verifications.

2. Literature Review

2.1. Control Barrier Functions and Safety Filters

Barrier certificates were introduced for the verification of hybrid systems [41] and grew from the theory of set invariance [42]. Wieland and Allgöwer made them constructive [18], and Ames and co-workers established the CBF quadratic program [15–17], a pointwise minimum-norm controller in the tradition of [43, 44]. The theory now covers robustness and input-to-state safety [45–47], high relative degree [48, 49], adaptation [50], discrete time [51, 52], temporal-logic tasks [53], backup sets [54], regularity of the QP solution [55] and learned certificates [56, 57]. Recent surveys place CBF-QPs within the broader class of safety filters between a performance controller and the plant [19–21]. In this literature the correction is a cost to keep small; we use it as information.

2.2. Multi-Robot CBFs, Responsibility and Deadlock

Safety barrier certificates provide decentralised collision avoidance with shared responsibility [22, 23]. They underlie the Robotarium testbed [58] and have been extended to nonsmooth barriers [24], limited actuation [59], constraint-driven

coordination [60], probabilistic certificates [61], learned decentralised certificates [62, 63] and human-aware warehouse navigation [25]. Recent work optimises how responsibility is split, by auction [64], by mixed-integer programming [65] or by maximising a certified feasibility reserve [66]. These approaches change the filter; we leave it unchanged and read its output. CBF-QPs can create undesirable equilibria [67], which appear as multi-robot deadlocks [68, 69]; the KKT conditions of the filter have been used to characterise and resolve such deadlocks [35]. Shielding-aware planning anticipates prospective safety-filter overrides when it plans [34]; we instead use overrides that have already been executed to update later route costs. Other reactive methods, such as potential fields [70], the dynamic window [71], velocity obstacles [72–74], buffered Voronoi cells [75] and social forces [76], also modify a nominal command but do not solve a convex program whose multipliers can be read as prices. Unicycle robots are mapped to single integrators by a near-identity diffeomorphism [77].

2.3. Coordination of Warehouse Robot Fleets

Warehouse robot coordination spans the design and control of AGV systems [5–7], robotic mobile fulfilment [1, 78–81], decentralised traffic control on roadmaps [11, 12] and AMR planning in intralogistics [4, 82]. Multi-agent path finding (MAPF) dominates the planning side [83–87], with lifelong variants for continuous task streams [9, 10], warehouse-specific models [88] and robust execution frameworks [14]. Congestion is addressed by highways [28, 29], optimised guidance graphs [27], traffic-flow costs computed from planned paths [26] and congestion-aware coordination games [30]. Intersections are commonly managed by reservations [13], and safety requirements for driverless industrial trucks are set by ISO 3691-4 [89]. None of these works feeds execution-layer interventions back into the planner.

2.4. Traffic Assignment, Day-to-Day Dynamics and Congestion Pricing

Wardrop's principles [90], the Beckmann program [32] and Fisk's logit stochastic user equilibrium (SUE) [91, 92] define equilibria of route choice. Selfish routing loses efficiency [93], and marginal-cost tolls restore the optimum [31, 33]. Day-to-day adjustment towards equilibrium can converge, oscillate or settle elsewhere [38]; local stability conditions for general networks and a unifying theory are given in [39, 40], continuous-time analyses in [94, 95], and oscillations caused by stale information in [96]. Step-size rules for computing SUE include successive and self-regulated averages [97]. In agent-based traffic simulation, marginal congestion costs have been computed by tracing delays in link queues back to the agents that caused them [36, 37]. In communication networks, Lagrange multipliers of capacity constraints act as congestion prices [98–100]. Macroscopic traffic theory relates density and delay [101–103], a relation that density-based proxies implicitly assume.

2.5. Summary

Table 1 positions this work. Existing congestion signals either observe the state of the traffic (occupancy, planned paths) or its outcome (travel time). None of them is both exact and available while the delay accrues, and none identifies who caused the delay in continuous space.

Table 1: Properties of congestion signals for warehouse robot routing.

Signal	*Exact delay*	*Available online*	*Identifies cause*	*Stability analysis*
Occupancy or density [101, 103]	no	yes	no	classical
Planned-path traffic flow [26, 27]	no	yes	no	no
Completed travel time	yes	no	no	classical
Accrued speed deficit	yes	yes	no	classical
Queue-traced caused delay [36, 37]	queue model	offline	yes	no
Layered intervention (this work)	yes	yes	yes	yes

3. System Model and Problem Formulation

3.1. Warehouse Network and Robot Motion

We represent the warehouse floor by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, whose vertices $\mathcal{V}$ are intersections, stations and lane ends and whose edges $\mathcal{E}$ are straight lane segments. Edge $e \in \mathcal{E}$ has length L_e (m), unit direction vector $\mathbf{n}_e \in \mathbb{R}^2$ and entrance point $x_e^0 \in \mathbb{R}^2$. A fleet $\mathcal{N} = \{1, \dots, N\}$ of N robots moves on the floor. Since the onboard velocity loops of AMRs are much faster than the time scale of traffic, we model robot i as a single integrator,

$$\dot{x}_i = u_i, \qquad u_i \in \mathcal{U}_i, \tag{1}$$

where $x_i \in \mathbb{R}^2$ is its position (m), $u_i \in \mathbb{R}^2$ is its commanded velocity (m/s), and $\mathcal{U}_i \subset \mathbb{R}^2$ is the compact convex polytope of admissible velocities. For a differential-drive robot, Eq. (1) describes a point ahead of the wheel axle through a near-identity diffeomorphism [77].

3.2. Free-Flow Command and Layered Control

When robot i is alone on edge e, it moves along the lane at its cruise speed except while it performs a planned task. We therefore define its free-flow command as

$$u_i^{(0)}(t) = \bar{v}_i \, m_i(t) \, \mathbf{n}_e + w_i(t), \qquad w_i(t)^\top \mathbf{n}_e = 0, \tag{2}$$

where $\bar{v}_i > 0$ is the cruise speed of robot i (m/s), $m_i(t) \in \{0,1\}$ is its motion indicator, which equals zero exactly during planned dwells such as picking or loading, and $w_i(t) \in \mathbb{R}^2$ is a lateral path-keeping velocity orthogonal to the lane. Since $u_i^{(0)}$ is the motion the robot would execute without other robots, any difference between it and the applied velocity is caused by traffic or by constraints.

The command passes through K control layers before it reaches the drive. Layer $k \in \{1, \dots, K\}$ maps the command it receives to a new command,

$$u_i^{(k)} = \mathcal{L}_k\big(u_i^{(k-1)}, x, t\big), \tag{3}$$

where $\mathcal{L}_k$ is the decision rule of layer k and $x = (x_1, \dots, x_N)$ collects all positions; the applied velocity is $u_i = u_i^{(K)}$. In this work $K = 2$: layer 1 is a reservation planner that may hold a robot at a stop line, and layer 2 is the CBF safety filter. We write $\hat{u}_i = u_i^{(K-1)}$ for the command that enters the filter and $u_i^\star = u_i^{(K)}$ for the filtered command.

3.3. CBF Safety Filter

Robot–robot safety requires the centres of two robots to be at least the safety distance d_s (m) apart and is encoded by the barrier function

$$h_{ij}(x) = \| x_i - x_j \|^2 - d_s^2, \tag{4}$$

which is nonnegative exactly when robots i and j are safe. Static safety with respect to rack faces, walls or keep-out zone o is encoded by a continuously differentiable function $h_{io}(x_i)$ that is nonnegative in free space. The safe set is $\mathcal{C} = \{x:\ h_{ij}(x) \geq 0,\ h_{io}(x_i) \geq 0\ \forall i, j, o\}$. To keep $\mathcal{C}$ forward invariant, we require every barrier h to satisfy $\dot{h} \geq -\alpha(h)$, where $\alpha: \mathbb{R} \to \mathbb{R}$ is an extended class-$\mathcal{K}$ function [16]. Since $\dot{h}_{ij} = 2(x_i - x_j)^\top (u_i - u_j)$ depends on both robots, we split the requirement between them with responsibility shares $\rho_{ij} \geq 0$ satisfying $\rho_{ij} + \rho_{ji} = 1$ [22]. Robot i then computes

$$\begin{aligned} u_i^\star = \arg\min_{u \in \mathbb{R}^2} \quad & 1/2 \| u - \hat{u}_i \|^2 \\ \text{s.t.} \quad & 2(x_i - x_j)^\top u \geq -\rho_{ij}\, \alpha(h_{ij}), \quad j \in \mathcal{N}_i, \\ & \nabla h_{io}(x_i)^\top u \geq -\alpha(h_{io}), \quad o \in \mathcal{O}_i, \\ & \qquad\qquad\qquad\quad u \in \mathcal{U}_i, \end{aligned} \tag{5}$$

where $\mathcal{N}_i$ and $\mathcal{O}_i$ are the neighbouring robots and obstacles within the sensing range of robot i. Adding the two halves of a pair's constraint gives $\dot{h}_{ij} \geq -\alpha(h_{ij})$, so $\mathcal{C}$ is forward invariant whenever every filter is feasible [16, 22].

For the analysis, we write problem (5) compactly as $\min_u 1/2 \| u - \hat{u}_i \|^2$ subject to $A_i u \geq b_i$, where row q of the matrix A_i and of the vector b_i is a single linear constraint $a_q^\top u \geq b_q$ (the polytope $\mathcal{U}_i$ contributes rows of the same form). We partition the row indices into robot–robot rows $\mathcal{K}_i^{\mathrm{rr}}$, static rows $\mathcal{K}_i^{\mathrm{st}}$ and actuator rows $\mathcal{K}_i^{\mathrm{act}}$, and for each robot–robot row q we denote by $j(q)$ the neighbour it concerns.

Assumption 1 (Feasibility and regularity). Along closed-loop trajectories, problem (5) is feasible. Where stated, the linear independence constraint qualification (LICQ) holds at $u_i^\star$, i.e. the rows a_q of the constraints active at $u_i^\star$ are linearly independent.

Forward invariance in continuous time does not by itself cover a digital controller, which computes $u_i^\star$ at sampling instants $t_n = n\Delta t$ and holds it until the next instant. For the barriers used in this paper, a simple condition on the CBF gain closes this gap.

Lemma 1 (Sampled-data safety). *Suppose that every robot solves its filter at the instants t_n from the common state $x(t_n)$, holds the resulting command on $[t_n, t_n + \Delta t)$, and uses $\alpha(h) = \gamma_a h$ for barrier a. Suppose further that every robot–robot barrier is a squared distance, as in Eq. (4), and every static barrier is affine in x_i. If $\gamma_a \Delta t \leq 1$ for every barrier and $x(t_n) \in \mathcal{C}$, then $x(t) \in \mathcal{C}$ for all $t \in [t_n, t_n + \Delta t]$.*

Proof. Let $r = x_i - x_j$ and $\sigma \in [0, \Delta t]$. Since the commands are constant, $r(t_n + \sigma) = r(t_n) + \sigma(u_i^\star - u_j^\star)$, and therefore

$$\begin{aligned} h_{ij}(t_n + \sigma) &= h_{ij}(t_n) + 2\sigma r(t_n)^\top (u_i^\star - u_j^\star) + \sigma^2 \| u_i^\star - u_j^\star \|^2 \\ &\geq (1 - \gamma_a \sigma)\, h_{ij}(t_n) \geq 0, \end{aligned}$$

where the first inequality adds the two halves of the pair constraint in problem (5) and drops the nonnegative quadratic term. For an affine static barrier, $h_{io}(t_n + \sigma) = h_{io}(t_n) + \sigma \nabla h_{io}^\top u_i^\star \geq (1 - \gamma_a \sigma) h_{io}(t_n) \geq 0$. □

The same argument applies to the lane barriers below: the headway barrier is affine in the progress of the two robots, and the crossing barrier $\zeta_i^2 + \zeta_j^2 - d_s^2$ is a squared distance. Our simulations use $\gamma \Delta t \leq 0.1$. The lemma does not cover nonlinear static barriers, neighbour information that is stale or asynchronous, a fallback command used when a QP is infeasible, or model and perception error.

3.4. Lane Networks and Intersection Reservation

Most warehouse AGVs and AMRs follow a roadmap of lanes [4, 11, 12]. On a lane network, robot i moves along its lane and controls only its scalar speed $u_i \in [0, \bar{v}_i]$ in the direction $\mathbf{n}_e$; its position is described by its progress $s_i = \mathbf{n}_e^\top (x_i - x_e^0) \in [0, L_e]$ (m). Three kinds of constraint arise.

1. *Headway.* Let $\ell = \ell(i)$ be the robot directly ahead of robot i on the same lane, which we call its leader. Since the leader cannot observe robots behind it, the follower takes full responsibility for the gap. With the barrier $h_i^{\mathrm{hw}} = s_\ell - s_i - d_s$ and the linear choice $\alpha(h) = \gamma h$ with CBF gain $\gamma > 0$ (s^{-1}), the requirement $\dot{h}_i^{\mathrm{hw}} \geq -\gamma h_i^{\mathrm{hw}}$ gives the upper bound $u_i \leq u_\ell + \gamma h_i^{\mathrm{hw}}$.
2. *Crossings.* Let ζ_i be the signed distance of robot i to a crossing point, negative before the crossing. Since the squared centre distance of robots i and j on perpendicular lanes is $\zeta_i^2 + \zeta_j^2$, we use $h_{ij} = \zeta_i^2 + \zeta_j^2 - d_s^2$ with equal shares $\rho_{ij} = 1/2$. For $\zeta_i < 0$, the row of robot i becomes the upper bound $u_i \leq \gamma h_{ij} / (4|\zeta_i|)$.
3. *Reservation.* The conflict zone $\{|\zeta| < w_z\}$ of an intersection, with half-width w_z (m), is reserved first come, first served (FCFS) [13]. A robot that requests the zone within the approach distance d_{app} (m) of its stop line may enter only if no robot of the crossing stream holds the zone and none has waited longer. A robot that is not granted the zone is held by its planner, which caps its command at $k_s(s_{\mathrm{stop}} - s_i)$, where k_s (s^{-1}) is the stop-line gain and s_{stop} is the position of the stop line, so that the robot comes to rest at the stop line.

Since every filter row in a lane network is an upper bound $\bar{u}_{i,q}$ on u_i, problem (5) reduces to the clip

$$u_i^\star = \min\left\{\hat{u}_i,\ \max\left(0,\ \min_q \bar{u}_{i,q}\right)\right\}, \tag{6}$$

and both layers can only reduce the speed.

3.5. Interventions

Since each layer modifies the command it receives, we measure that modification directly.

Definition 1 (Interventions). The layer-k intervention of robot i is $\delta_i^{(k)} = u_i^{(k)} - u_i^{(k-1)} \in \mathbb{R}^2$; the filter intervention is $\delta_i = \delta_i^{(K)} = u_i^\star - \hat{u}_i$; and the total intervention is $\delta_i^{\mathrm{tot}} = u_i^\star - u_i^{(0)} = \sum_{k=1}^K \delta_i^{(k)}$. Because congestion delays motion along the lane, we project onto the lane direction: the longitudinal intervention of layer k is $\iota_i^{(k)} = -\mathbf{n}_e^\top \delta_i^{(k)}$ (m/s), which is positive when the layer brakes the robot, and the total longitudinal intervention is $\iota_i = \sum_{k=1}^K \iota_i^{(k)}$. When $\bar{v}_i m_i > 0$, the normalised intervention $\eta_i = \| \delta_i^{\mathrm{tot}} \| / \| u_i^{(0)} \|$ is the fraction of the intended motion that the control layers remove.

Peer and environment components. To separate the part of the filter intervention caused by other robots from the part caused by walls, keep-out zones and input limits, we also solve, at the same state, the auxiliary projection $\bar{u}_i = \mathrm{argmin}_u\ 1/2\ \|$

$u - \hat{u}_i \|^2$ subject to $u \in \mathcal{U}_i$ and the static rows of problem (5) only. We then define the environment component $\delta_i^{\text{env}} = \bar{u}_i - \hat{u}_i$ and the peer component $\delta_i^{\text{peer}} = u_i^\star - \bar{u}_i$, so that $\delta_i = \delta_i^{\text{env}} + \delta_i^{\text{peer}}$. The split is an algebraic identity at each state; the peer component is conditional on the static rows and is not the difference between two trajectories. Unlike the split of Theorem 1(iv) below, it needs no constraint qualification, but it does not identify the individual neighbour. In a lane network, the auxiliary projection clips $\hat{u}_i$ to $[0, \bar{v}_i]$, so $\delta_i^{\text{env}} = 0$ whenever $0 \le \hat{u}_i \le \bar{v}_i$ and the whole filter intervention is due to other robots.

3.6. Problem Statement

Given a fleet governed by Eqs. (1)–(6), we seek:

1. a characterisation of the filter intervention δ_i in terms of the constraints that produce it;
2. an exact online estimator of the congestion delay of each robot and of the mean delay of each edge, built from the interventions;
3. an attribution of each delay to the robots that caused it, and a route price built from this attribution that approximates the marginal social cost; and
4. conditions under which routing on these prices converges, and a rule for choosing the price-smoothing gain.

4. Methodology

4.1. the Intervention as a Sum of Constraint Forces

Let $\lambda \in \mathbb{R}_{\ge 0}^{|\mathcal{K}_i|}$ denote a vector of KKT multipliers of problem (5) at $u_i^\star$, with one entry λ_q per row. Since the multiplier of a row depends on how the row is scaled, we define for every row q the unit normal $\hat{a}_q = a_q / \| a_q \|$ and the normalised multiplier $\nu_q = \| a_q \| \lambda_q$. We also define the nominal safety deficit $\chi_q = b_q - a_q^\top \hat{u}_i$, which is positive when the incoming command would violate row q.

Theorem 1 (Lagrangian structure of the intervention). *Let problem (5) be feasible, and let $J_i^\star = 1/2 \| \delta_i \|^2$ denote its optimal value. Then:*

1. $\delta_i = \sum_q \lambda_q a_q = \sum_q \nu_q \hat{a}_q$;
2. $\| \delta_i \|^2 = \sum_q \lambda_q \chi_q$, *so that* $J_i^\star = 1/2 \sum_q \lambda_q \chi_q$;
3. *the set of normalised multipliers is invariant to a positive rescaling* $(a_q, b_q) \mapsto (\varphi a_q, \varphi b_q)$, $\varphi > 0$, *of any row, and under LICQ and strict complementarity* $\partial J_i^\star / \partial (b_q / \| a_q \|) = \nu_q$;
4. *under LICQ, λ is unique, and so is the decomposition $\delta_i = \delta_i^{\text{rr}} + \delta_i^{\text{st}} + \delta_i^{\text{act}}$ with $\delta_i^S = \sum_{q \in \mathcal{K}_i^S} \nu_q \hat{a}_q$ for $S \in \{\text{rr}, \text{st}, \text{act}\}$; in particular, the robot–robot part splits uniquely by neighbour as $\delta_i^{\text{rr}} = \sum_j \delta_i^{(j)}$, where $\delta_i^{(j)} = \sum_{q:\, j(q)=j} \nu_q \hat{a}_q$ is the part caused by robot j.*

Proof. Problem (5) is convex with affine constraints, so the KKT conditions are necessary and sufficient [104]. Its Lagrangian is $1/2 \| u - \hat{u}_i \|^2 - \lambda^\top (A_i u - b_i)$. Setting its gradient to zero at $u_i^\star$ gives $u_i^\star - \hat{u}_i = A_i^\top \lambda$, which is (i) since $\lambda_q a_q = \nu_q \hat{a}_q$. Complementary slackness states $\lambda^\top (A_i u_i^\star - b_i) = 0$. Substituting (i), we obtain $\| \delta_i \|^2 = \lambda^\top A_i (u_i^\star - \hat{u}_i) = \lambda^\top (b_i - A_i \hat{u}_i) = \sum_q \lambda_q \chi_q$, which is (ii). For (iii), rescaling row q by φ leaves the feasible set, $u_i^\star$ and δ_i unchanged; a vector λ is a multiplier of the original problem if and only if the vector with λ_q replaced by λ_q / φ is a multiplier of the rescaled problem, and since $\| a_q \|$ becomes $\varphi \| a_q \|$, the set of normalised multipliers is unchanged. The derivative follows from the

sensitivity theorem of nonlinear programming [105] applied to the normalised row $\hat{a}_q^\top u \geq b_q/\| a_q \|$, whose multiplier is ν_q. For (iv), complementary slackness sets the multipliers of inactive rows to zero, and (i) then determines the remaining multipliers uniquely when the active rows are linearly independent. □

Theorem 1 shows that the filter acts on the incoming command through constraint forces $\nu_q \hat{a}_q$: unit normals of the barriers, scaled by multipliers that measure the marginal cost of tightening each barrier. This is the structure that makes the Lagrange multipliers of link capacities congestion prices in network flow control [98–100]. Part (iv) is the basis of the causal attribution in Section 4.3. In a lane network, the binding row of the clip (6) has $\hat{a}_q = -1$ and $\nu_q = \hat{u}_i - u_i^\star$, so that the normalised multiplier equals the filter intervention.

Corollary 1 (A single active pairwise barrier). *Suppose that only the row of neighbour j is active. Let $r_{ij} = x_i - x_j$ be the relative position and $v_{ij}^{\mathrm{cl}} = -r_{ij}^\top \hat{u}_i/\| r_{ij} \|$ the nominal closing speed of robot i towards robot j. Then $\delta_i = \nu\, r_{ij}/\| r_{ij} \|$ with $\nu = \| \delta_i \| = [v_{ij}^{\mathrm{cl}} - \rho_{ij}\alpha(h_{ij})/(2 \| r_{ij} \|)]_+$.*

Proof. The active row has $a = 2r_{ij}$ and $b = -\rho_{ij}\alpha(h_{ij})$, so its safety deficit is $\chi = b - a^\top \hat{u}_i = 2 \| r_{ij} \| v_{ij}^{\mathrm{cl}} - \rho_{ij}\alpha(h_{ij})$. Since the row holds with equality at $u_i^\star = \hat{u}_i + \lambda a$, we have $\lambda = \chi/\| a \|^2$ and therefore $\nu = \lambda \| a \| = \chi/(2 \| r_{ij} \|)$. □

The intervention is thus the excess of the nominal closing speed over the closing speed that the barrier allows at the current gap.

Proposition 1 (Regularity). *Under LICQ, $u_i^\star$, λ and δ_i are locally Lipschitz functions of the data $(A_i, b_i, \hat{u}_i)$. Hence, along trajectories with locally Lipschitz data, $t \mapsto \delta_i(t)$ is locally Lipschitz and its time integrals are well defined.*

Proof. The Hessian of the objective is the identity, so the strong second-order sufficient condition holds. Together with LICQ, the KKT system is strongly regular in the sense of Robinson [106], which yields a locally Lipschitz primal–dual solution map [55, 105]. □

Proposition 2 (Stop signature). *Let $u_i^{(0)} \neq 0$. Then $u_i^\star = 0$ if and only if $\delta_i^{\mathrm{tot}} = -u_i^{(0)}$, in which case $\eta_i = 1$. At a closed-loop equilibrium in which robot i is at rest while $m_i = 1$, for instance an undesirable equilibrium [67] or a deadlock [35, 68], its longitudinal intervention equals $\bar{v}_i$, so its delay grows at one second per second. In general, $|\eta_i - 1| \leq \| u_i^\star \|/\| u_i^{(0)} \|$, and in a lane network, where every layer only reduces the speed, $0 \leq \eta_i \leq 1$.*

Proof. The equivalence follows from $\delta_i^{\mathrm{tot}} = u_i^\star - u_i^{(0)}$. At rest with $m_i = 1$, $\iota_i = -\mathbf{n}_e^\top \delta_i^{\mathrm{tot}} = \mathbf{n}_e^\top u_i^{(0)} = \bar{v}_i$ by Eq. (2). The bound is the reverse triangle inequality applied to $\delta_i^{\mathrm{tot}} = u_i^\star - u_i^{(0)}$. In a lane network with $m_i = 1$, $u_i^\star \in [0, \bar{v}_i]$ and $u_i^{(0)} = \bar{v}_i$, so $\eta_i = (\bar{v}_i - u_i^\star)/\bar{v}_i \in [0,1]$. □

A robot slowed by traffic therefore has $0 < \eta_i < 1$, and a robot stopped by its control layers has $\eta_i = 1$. A unit value identifies a stop, not a deadlock: a robot may stop to yield and later proceed. A deadlock additionally requires an unresolved goal and a persistent loss of progress, and confirming it needs a liveness analysis [35, 67]. Persistence of $\eta_i \approx 1$ is therefore a cheap, communication-free trigger for such an analysis rather than a deadlock test in itself.

4.2. Layered Delay Accounting

The delay accounting rests on the following definition of an edge traversal.

Assumption 2 (Edge traversal). Robot i occupies edge e from time t_i^{in}, when it reaches the entrance of the edge (including any time it is held there), until time t_i^{out}, the first time its progress $s_i = \mathbf{n}_e^\top (x_i - x_e^0)$ reaches L_e, with $s_i(t_i^{\text{in}}) = 0$. Its free-flow command is Eq. (2) throughout, and its planned dwells, of total duration τ_i (s), lie within $[t_i^{\text{in}}, t_i^{\text{out}}]$.

Since a robot that is never held back crosses the edge in $L_e/\bar{v}_i$ plus its dwell time, we define its congestion delay as the excess of its traversal time $T_i = t_i^{\text{out}} - t_i^{\text{in}}$ over this free-flow time,

$$D_i = T_i - \frac{L_e}{\bar{v}_i} - \tau_i. \tag{7}$$

Theorem 2 (Layered intervention–delay identity). *Under Assumption 2,*

$$D_i = \frac{1}{\bar{v}_i} \sum_{k=1}^{K} \int_{t_i^{\text{in}}}^{t_i^{\text{out}}} \iota_i^{(k)}(t)\, dt = \frac{1}{\bar{v}_i} \int_{t_i^{\text{in}}}^{t_i^{\text{out}}} \iota_i(t)\, dt. \tag{8}$$

Consequently, the delay inferred from any subset $\mathcal{S} \subset \{1, \dots, K\}$ of the layers falls short of D_i by $\bar{v}_i^{-1} \sum_{k \notin \mathcal{S}} \int \iota_i^{(k)}\, dt$. Moreover, $|D_i| \le \bar{v}_i^{-1} \int \| \delta_i^{\text{tot}} \|\, dt$, with equality if and only if δ_i^{tot} is a non-positive multiple of $\mathbf{n}_e$ almost everywhere.

Proof. Since the layer interventions telescope, $u_i^\star = u_i^{(0)} + \sum_k \delta_i^{(k)}$. Projecting Eq. (1) onto the lane and using Eq. (2) with $w_i^\top \mathbf{n}_e = 0$, the progress of robot i therefore evolves as

$$\dot{s}_i = \mathbf{n}_e^\top u_i^\star = \bar{v}_i m_i - \sum_k \iota_i^{(k)}.$$

Integrating over the traversal, and using $s_i(t_i^{\text{out}}) - s_i(t_i^{\text{in}}) = L_e$ and $\int m_i\, dt = T_i - \tau_i$, we obtain $L_e = \bar{v}_i (T_i - \tau_i) - \sum_k \int \iota_i^{(k)} dt$. Rearranging with Eq. (7) gives Eq. (8). The bound follows from $|\iota_i| \le \| \delta_i^{\text{tot}} \|$ (Cauchy–Schwarz), with equality exactly when δ_i^{tot} points against $\mathbf{n}_e$. □

The identity is local: robot i knows the output of each layer at every step, so it knows its own congestion delay without a model of the other robots or of the traffic. It also resolves a problem of measurement scope. If a planner already holds a robot at an occupied intersection, the filter receives a conflict-free command and intervenes little, so a filter-only signal under-reports the delay; in a "silent queue" of robots whose commands are already zero, the filter may report almost nothing. Theorem 2 shows that the remedy is to log the deviation of every layer from the free-flow command: their sum is exact however the work is divided between the layers.

The identity is kinematic rather than specific to CBFs. Since $\iota_i = \bar{v}_i m_i - \mathbf{n}_e^\top u_i^\star$, a robot that logs its forward speed and its planned dwell schedule obtains the same signed deficit, and on a single lane the accrued intervention and the accrued speed deficit are the same signal. Logging the control layers is still useful for two reasons. Each layer already computes its deviation, so the measurement costs nothing extra. More importantly, the filter's active rows and multipliers, which speed telemetry does not contain, are what make the attribution of Section 4.3 possible. The CBF-specific value of the signal lies in that attribution.

Corollary 2 (Sampled data). *Suppose that the layers run with zero-order hold at period Δt (s), that dwell durations are multiples of Δt, and that $\| u_i^\star \| \le u_{\max}$. Then Eq. (8) holds with the integrals replaced by the sums $\sum_n \iota_i^{(k)}(t_n) \Delta t$ over the sampling instants t_n, up to an error of at most $(1 + u_{\max}/\bar{v}_i) \Delta t$.*

Proof. We repeat the proof of Theorem 2 with sums. The terminal progress overshoots L_e by less than $u_{\max}\Delta t$, which changes the time by less than $u_{\max}\Delta t/\bar{v}_i$, and the entry time is resolved to within Δt. □

For routing we need the mean delay of an edge rather than of one robot. Consider a window $[t, t+T_p]$ of length T_p (s). We let N_e be the number of robots that enter edge e during the window, and we define the accrued suffered delay

$$I_e = \int_t^{t+T_p} \sum_{i \text{ on } e} \frac{\iota_i(s')}{\bar{v}_i}\, ds', \tag{9}$$

which every robot can accumulate by adding $\iota_i \Delta t/\bar{v}_i$ to a counter of its current edge. The ratio I_e/N_e is undefined in a window without new entries, which happens both on idle edges and on edges where robots are stuck. We therefore define the regularised estimator

$$\hat{d}_e = \frac{I_e + \nu_e d_e^0}{N_e + \nu_e}, \tag{10}$$

where $\nu_e > 0$ is a pseudo-entry count and $d_e^0 \geq 0$ (s) a prior delay. On an idle edge with $d_e^0 = 0$, the estimator returns zero; on an edge where robots accrue delay without new entries, it increases instead of dividing by zero.

Proposition 3 (Little's-law edge-delay estimator). *Suppose that the traffic on edge e is stationary and ergodic, with entry rate $f_e > 0$ (robots/s) and mean congestion delay per robot $d_e < \infty$. Then, almost surely, $\lim_{T_p\to\infty} I_e/N_e = d_e$, and $\lim_{T_p\to\infty} \hat{d}_e = d_e$ for every fixed $\nu_e > 0$ and d_e^0.*

Proof. Each robot accrues cost at rate $\iota_i/\bar{v}_i$ while it is on the edge, and by Theorem 2 its total accrued cost is D_i. The cost-accrual form of Little's law [107, 108] therefore gives $\lim_{T_p\to\infty} I_e/T_p = f_e d_e$. Since also $N_e/T_p \to f_e$, the ratio converges to d_e. Dividing the numerator and denominator of Eq. (10) by T_p shows that the fixed terms $\nu_e d_e^0$ and ν_e vanish in the limit. □

Unlike an estimator based on completed traversals, the estimator updates while the delay accrues, and returning a defined value on idle edges prevents the lock-in that occurs when an unused edge never produces a completed traversal (Section 6). Proposition 3 establishes consistency over long windows only; it does not bound the bias or variance over a single period, and the prior and pseudo-count should be reported with each price. In the experiments we used a simpler rule, the raw ratio when $N_e > 0$ and the zero-load delay otherwise; Eq. (10) is the form we recommend for deployment.

4.3. Causal Attribution and Marginal-Cost Prices

Theorem 2 quantifies the delay of robot i; Theorem 1(iv) and the hold decisions of the planner identify its sources. We formalise the attribution for lane networks, where every intervention is a braking action.

Definition 2 (Blame vectors and imposed delay). At each control step, robot i carries a blame vector $B_i \in \mathbb{R}^N_{\geq 0}$, whose entry B_{ij} (m/s) is the part of its longitudinal intervention ι_i charged to robot j. We denote by $\mathbf{1}_j$ the j-th unit vector of $\mathbb{R}^N$. The vector is built from the following rules.

1. *Planner holds.* If the reservation holds robot i, its planner intervention $\iota_i^{(1)}$ is shared equally among the robots in the set $\mathcal{H}_i$ that hold the conflict zone or, if no robot holds it, charged to the earlier-waiting robot whose priority causes the hold.

2. *Filter, crossing row.* If the binding row of the filter is a crossing row with neighbour j, the filter intervention $\iota_i^{(2)}$ is charged to robot j (ties are split equally; under LICQ the binding row is unique).
3. *Filter, headway row, direct.* If the binding row is the headway row with leader ℓ, $\iota_i^{(2)}$ is charged to ℓ.
4. *Filter, headway row, root cause.* The leader brakes robot i either because the leader itself is slower or dwelling, or because the leader is held back by others. We therefore define the own-cause deficit of the leader relative to robot i, $o_\ell = (\bar{v}_i - \bar{v}_\ell)_+ + \bar{v}_\ell(1 - m_\ell)$, where $(\cdot)_+ = \max(\cdot, 0)$, and its total deficit $\Gamma_\ell = o_\ell + \iota_\ell$, and we split $\iota_i^{(2)}$ in proportion to these components:

$$B_i \mathrel{+}= \begin{cases} \dfrac{\iota_i^{(2)}}{\Gamma_\ell}(o_\ell \mathbf{1}_\ell + B_\ell), & \Gamma_\ell > 0, \\ \iota_i^{(2)} \mathbf{1}_\ell, & \Gamma_\ell = 0. \end{cases} \tag{11}$$

A robot that waits at an edge entrance is treated as a follower of the robot directly ahead of it, with intervention $\bar{v}_i$. The direct attribution uses rules (a), (b) and (c); the root-cause attribution uses rules (a), (b) and (c′). The total delay imposed by robot j on other robots is $X_j = \sum_i \int B_{ij}(t)/\bar{v}_i \, dt$ (s).

Root-cause forwarding passes a follower's braking back along the chain of active headway constraints until it reaches a robot that is slow, dwelling, held by the planner or blocked by a crossing robot. A picking robot is therefore charged for the queue that forms behind it, and a crossing robot for the platoon its passage stops on the highway. This is the continuous-space counterpart of the backward tracing of delays in queue models [36]. We call the attribution causal in a specific sense: each charge follows a constraint that is binding at that instant. It is an accounting rule, not a counterfactual statement about how much faster robot i would have been without robot j. Theorem 4 below gives conditions under which the two coincide, and Experiment 3 measures the gap when they do not.

Theorem 3 (Conservation). *For both attributions and at every time, $B_i \geq 0$ and $\mathbf{1}^\top B_i = \iota_i$. Consequently, every second of congestion delay is charged exactly once:*

$$\sum_j X_j = \sum_i \frac{1}{\bar{v}_i} \int \iota_i \, dt = \sum_i D_i. \tag{12}$$

In particular, in every window the accrued imposed delay of edge e, Ξ_e, defined as the increase of $\sum_j X_j$ over the robots j that use edge e, satisfies $\sum_e \Xi_e = \sum_e I_e$.

Proof. Nonnegativity holds because both layers only brake and the weights in Eq. (11) are nonnegative. To show $\mathbf{1}^\top B_i = \iota_i$, we process the robots on each lane from front to back, which is the order in which the leader's vector B_ℓ becomes available. Rules (a), (b) and (c) assign exactly $\iota_i^{(1)} + \iota_i^{(2)} = \iota_i$. For rule (c′), suppose inductively that $\mathbf{1}^\top B_\ell = \iota_\ell$. Then $\mathbf{1}^\top(o_\ell \mathbf{1}_\ell + B_\ell) = o_\ell + \iota_\ell = \Gamma_\ell$, so the forwarded amount is exactly $\iota_i^{(2)}$. Dividing by $\bar{v}_i$, integrating over time and applying Theorem 2 gives Eq. (12); summing over the robots active in a window gives the edge-level statement. □

Since the attribution is budget balanced, the imposed delay can serve as an externality price. Let robots belong to classes r, which may be routes or background streams with fixed routes, with stationary flows f_r (robots/s), free-flow times t_r^0 (s) and mean suffered delays $d_r(f)$ (s), where f collects all flows. The total travel time spent per unit time is

$$\mathrm{TC}(f) = \sum_r f_r \left(t_r^0 + d_r(f)\right), \tag{13}$$

so the marginal social cost of class r' is $\partial \mathrm{TC}/\partial f_{r'} = t_{r'}^0 + d_{r'} + \mathrm{MEC}_{r'}$, where $\mathrm{MEC}_{r'} = \sum_r f_r \, \partial d_r/\partial f_{r'}$ is the marginal external cost [32, 33]. We denote by $\xi_{r'}$ the mean delay imposed per robot of class r', i.e. the long-run ratio of the delay attributed to class-r' robots to the number of such robots.

Assumption 3 (Pairwise encounters). The mean delay that a class-r robot suffers and that is attributed to class-r' robots equals $\Omega_{rr'} f_{r'}$, where the encounter coefficients $\Omega_{rr'} \geq 0$ (s²) are constants.

Theorem 4 (Suffered plus imposed delay is the marginal social cost). *Under Theorem 3 and Assumption 3, $d_r(f) = \sum_{r'} \Omega_{rr'} f_{r'}$, the mean delay imposed per class-r' robot is $\xi_{r'} = \sum_r f_r \, \Omega_{rr'}$, and*

$$t_{r'}^0 + d_{r'} + \xi_{r'} = \frac{\partial \, \mathrm{TC}}{\partial f_{r'}}, \qquad \text{i.e.} \quad \xi_{r'} = \mathrm{MEC}_{r'}, \tag{14}$$

including the externalities on classes that are not routed. If, in addition, $\Omega + \Omega^\top$ is positive semidefinite, then routing on the prices $t^0 + d + \xi$ under the logit dispatch of Section 4.4 has a unique equilibrium, and it minimises $\mathrm{TC} + \theta^{-1} \sum_r y_r \ln y_r$ over the route flows y_r, i.e. it is the system-optimal stochastic equilibrium.

Proof. By conservation, the delay of a class-r robot is the sum of its attributed parts, so $d_r = \sum_{r'} \Omega_{rr'} f_{r'}$. Delay attributed to class r' accrues on class-r robots at the rate $f_r \cdot \Omega_{rr'} f_{r'}$ (seconds of delay per second). By the renewal-reward argument used in Proposition 3, the mean delay imposed per class-r' robot on class r is this rate divided by the arrival rate $f_{r'}$, namely $f_r \Omega_{rr'}$; summing over r gives $\xi_{r'}$. Differentiating Eq. (13) gives $\partial \mathrm{TC}/\partial f_{r'} = t_{r'}^0 + \sum_r \Omega_{r'r} f_r + \sum_r f_r \, \Omega_{rr'} = t_{r'}^0 + d_{r'} + \xi_{r'}$; this identity was also verified symbolically (Appendix A). Since $t^0 + d + \xi = \nabla \mathrm{TC}$, the equilibrium statement follows from the proof of Theorem 5 with the potential TC, which is convex when $\Omega + \Omega^\top$ is positive semidefinite. □

Assumption 3 describes light traffic, in which delay arises from separate encounters. In heavier traffic, encounters overlap, queues form and delay becomes superlinear in the flow. Budget balance then has a precise consequence.

Proposition 4 (Bracketing under superlinear congestion). *Suppose that all interactions of a class are internal to it, as on an isolated aisle, and that its delay function $d(f)$ is differentiable. Then conservation gives $\xi = d(f)$, and the attribution-based price $t^0 + 2d$ satisfies $t^0 + d \leq t^0 + 2d \leq t^0 + d + f\, d'(f)$ whenever $d(f)/f$ is nondecreasing, with equality on the right if and only if d is linear. In general, the imposed delay is nonnegative, so attribution prices are never below user-equilibrium prices, and they equal marginal-cost prices exactly under Assumption 3.*

Proof. Since all imposed delay is suffered within the class, $f\xi = fd$, so $\xi = d$. The function $d(f)/f$ is nondecreasing if and only if its derivative $(fd' - d)/f^2$ is nonnegative, i.e. if and only if $fd' \geq d$. For example, for $d = a_1 f + a_3 f^3$ with $a_1, a_3 \geq 0$ we obtain $fd' - d = 2a_3 f^3$ (Appendix A). □

Budget-balanced attribution therefore cannot charge the full marginal cost under superlinear congestion, because marginal-cost tolls collect more than the total delay [33]. A second gap arises from *bunching*. A crossing robot that holds a highway platoon compresses its headways, and fast robots may later be slowed by slower leaders. The attribution charges that later delay to the slow leader rather than to the crossing robot, because it follows instantaneous binding

constraints. Section 6 measures both gaps against a counterfactual marginal external cost.

4.4. Intervention-Priced Routing and Its Equilibrium

Robots requesting transport arrive at rate Q (robots/s) and choose a route r from the set $\mathcal{R}$; the matrix $\Delta \in \{0,1\}^{|\mathcal{E}|\times|\mathcal{R}|}$ has entry $\Delta_{er} = 1$ when route r uses edge e. The fleet manager keeps a price p_e (s) for every edge and perceives the cost of route r as $C_r(p) = \sum_e \Delta_{er}\,(t_e^0 + p_e)$, where t_e^0 is the free-flow time of edge e. Since dispatchers in practice randomise among nearly equivalent routes, we model dispatch by the logit rule

$$\pi_r(p) = \frac{\exp\big(-\theta C_r(p)\big)}{\sum_{r'\in\mathcal{R}} \exp\big(-\theta C_{r'}(p)\big)}, \tag{15}$$

where π_r is the probability of dispatching a robot to route r and $\theta > 0$ (s^{-1}) is the dispatch sensitivity; $\theta \to \infty$ recovers shortest-path dispatch. The induced edge flows are $f(p) = Q\Delta\pi(p)$. Every period n of length T_p, the manager moves the prices towards a target $c_e(f_e)$ estimated from interventions, with smoothing gain $\beta \in (0,1]$:

$$p(n+1) = \Phi_\beta\big(p(n)\big) := (1-\beta)\,p(n) + \beta\,c\Big(f\big(p(n)\big)\Big). \tag{16}$$

User-equilibrium (UE) pricing uses the suffered delay, $c_e = d_e$ (Proposition 3); marginal-cost (MC) pricing uses suffered plus imposed delay, $c_e = d_e + \xi_e$ (Theorem 4). Equation (16) is the mean-field model of the update, which assumes that T_p is long compared with traversal times; we treat estimation noise in Proposition 5(c).

Assumption 4 (Targets). Each target $c_e\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is continuously differentiable and nondecreasing.

Theorem 5 (Equilibrium). *Under Assumption 4, for every $\beta \in (0,1]$ the map Φ_β has exactly one fixed point $p^\star$. The route flows $y^\star = Q\pi(p^\star)$ are the unique minimiser of*

$$\min_{y\geq 0,\ \mathbf{1}^\top y = Q} \sum_e \int_0^{(\Delta y)_e} \big(t_e^0 + c_e(w)\big)\,dw + \frac{1}{\theta}\sum_r y_r\,(\ln y_r - 1), \tag{17}$$

and $p^\star = c(\Delta y^\star)$. For UE pricing, this is Fisk's stochastic user equilibrium with respect to the true delays [91].

Proof. The objective of problem (17) is continuous on a compact set, its first term is convex because the integrands are nondecreasing, and its entropy term is strictly convex; hence a unique minimiser $y^\star$ exists. Since the derivative of the entropy term diverges as $y_r \to 0$, every $y_r^\star > 0$, and the KKT conditions with a multiplier for $\mathbf{1}^\top y = Q$ read $C_r\big(c(\Delta y^\star)\big) + \theta^{-1}\ln y_r^\star = \text{const}$, i.e. $y^\star = Q\pi\big(c(\Delta y^\star)\big)$. Setting $p^\star = c(\Delta y^\star)$, we therefore have $c\big(f(p^\star)\big) = p^\star$, so $p^\star$ is a fixed point of Φ_β for every β. Conversely, any fixed point p satisfies $p = c\big(f(p)\big)$, so $y = Q\pi(p)$ satisfies the sufficient KKT conditions and coincides with $y^\star$. □

Since intervention prices are consistent estimates of the delays that robots experience (Theorem 2 and Proposition 3), a UE-priced fleet settles where the cost it perceives is the cost it pays. A density proxy $\kappa_n \bar{n}_e$, where $\bar{n}_e$ is the mean number of robots on edge e and κ_n (s per robot) a calibration coefficient, instead leads to the equilibrium of problem (17) with c_e replaced by $\kappa_n \bar{n}_e(f_e)$, which differs from the true-delay equilibrium whenever the relation between density and delay differs between edges.

Algorithm 1. Self-tuned intervention pricing (one period)

1. For each edge e, read the accrued suffered delay I_e, the accrued imposed delay Ξ_e and the number of entries N_e; set $\hat{f}_e \leftarrow N_e/T_p$
2. Set the target $\hat{d}_e$ by Eq. (10) (UE), or add the imposed delay $(\Xi_e + \nu_e \xi_e^0)/(N_e + \nu_e)$ with prior ξ_e^0 (MC)
3. Append $\left(\hat{f}_e, \hat{d}_e\right)$ to the history of edge e; fit $c_e(f) = a_{1,e} f + a_{3,e} f^3$ with $a_{1,e}, a_{3,e} \geq 0$ by nonnegative least squares
4. Set $\hat{Q} \leftarrow \sum_e N_e / T_p$ over route-entry edges; compute $\pi(p)$, G, $\widehat{M} = \operatorname{diag}\left(c_e{}'\left(\hat{Q}(\Delta\pi)_e\right)\right)$ and $\hat{\mu} \leftarrow \lambda_{\max}\left(\widehat{M}^{1/2} G \widehat{M}^{1/2}\right)$
5. $\beta_{\text{rate}} \leftarrow 2/\left(2 + \theta \hat{Q} \hat{\mu}\right)$ ▷ *Proposition 5(a)*
6. $\widehat{\omega}^2 \leftarrow$ moving average of the squared change of the centred fitted target at smoothed flows; $\hat{\sigma}^2 \leftarrow$ moving average of the squared residual between centred measured and fitted targets; $\beta_{\text{track}} \leftarrow$ root of $\vartheta \hat{\sigma}^2 \beta^3 = \widehat{\omega}^2 (2 - \vartheta\beta)^2$ with $\vartheta = 1 + \theta \hat{Q} \hat{\mu}$ ▷ *(e)*
7. $\beta \leftarrow \min(\max(\min(\beta_{\text{rate}}, \beta_{\text{track}}), \beta_{\min}), 1)$; $p_e \leftarrow (1 - \beta) p_e + \beta \cdot \text{target}_e$

4.5. Stability and Gain Design

Let $\pi^\star = \pi(p^\star)$ be the equilibrium dispatch probabilities. Since the sensitivity of the logit rule is governed by their covariance, we define the dispatch covariance matrix, its projection onto the edges, the diagonal matrix of target slopes, and the largest and smallest network curvatures as

$$\begin{gathered} \Pi^\star = \operatorname{diag}(\pi^\star) - \pi^\star \pi^{\star\top}, \qquad G = \Delta \Pi^\star \Delta^\top, \\ M = \operatorname{diag}\left(c'_e(f_e^\star)\right), \\ \mu = \lambda_{\max}\left(M^{1/2} G M^{1/2}\right), \qquad \varsigma_{\min} = \lambda_{\min}\left(M^{1/2} G M^{1/2}\right), \end{gathered} \tag{18}$$

where $f_e^\star$ are the equilibrium edge flows and $\lambda_{\max}$, $\lambda_{\min}$ denote the largest and smallest eigenvalues. We denote by $\varrho(\cdot)$ the spectral radius of a matrix.

Theorem 6 (Local stability [38–40]. *] The Jacobian of Φ_β at $p^\star$ is $J = (1 - \beta) I - \beta\theta Q\, MG$. Its eigenvalues are $1 - \beta(1 + \theta Q \varsigma)$, where ς ranges over the spectrum of MG, which is real and contained in $[\varsigma_{\min}, \mu]$ with both ends attained. Hence $p^\star$ is locally exponentially stable if $\beta(1 + \theta Q \mu) < 2$, and unstable with a period-doubling (flip) instability if $\beta(1 + \theta Q \mu) > 2$.*

Proof. Differentiating Eq. (15) gives $\partial\pi / \partial C = -\theta \Pi$, and $\partial C / \partial p = \Delta^\top$, so the chain rule applied to Eq. (16) gives J. The matrix G is positive semidefinite, and the nonzero eigenvalues of $MG = M^{1/2}\left(M^{1/2} G\right)$ coincide with those of the positive semidefinite matrix $M^{1/2} G M^{1/2}$ [109]; hence the spectrum of MG is real and lies in $[\varsigma_{\min}, \mu]$. The largest eigenvalue of J is at most $1 - \beta < 1$, so $\varrho(J) < 1$ if and only if $1 - \beta(1 + \theta Q \mu) > -1$. The linearisation principle for maps [110] completes the proof. □

Theorem 6 is the classical local analysis of day-to-day assignment dynamics [38–40], written for the price map so that the boundary is explicit in the curvature μ. We denote the boundary by $\beta^\star = 2/(1 + \theta Q \mu)$. In a warehouse, the flip instability corresponds to robots crowding into whichever aisle was empty in the previous period. The design results that follow are new.

Proposition 5 (Gain design). *For a price mode with curvature ς, let $\vartheta_\varsigma = 1 + \theta Q \varsigma$, so that its eigenvalue is $1 - \beta\vartheta_\varsigma$.*

1. *Rate-optimal gain. The spectral radius* $\varrho(J) = \max\left(\left|1-\beta\vartheta_{\varsigma_{\min}}\right|, \left|1-\beta\vartheta_{\mu}\right|\right)$ *is minimised by* $\beta_{\text{opt}} = \frac{2}{2+\theta Q(\mu+\varsigma_{\min})}$, $\varrho_{\text{opt}} = \frac{\theta Q(\mu-\varsigma_{\min})}{2+\theta Q(\mu+\varsigma_{\min})} < 1$.
2. *Robustness margin. The gain* $\hat{\beta} = 2/(2+\theta Q\hat{\mu})$ *designed with an estimate* $\hat{\mu}$ *of* μ *is locally stabilising if and only if* $\hat{\mu} > \mu - 1/(\theta Q)$*; overestimating* μ *only slows convergence.*
3. *Noise amplification. If every price estimate carries independent zero-mean noise of variance* σ^2 *(s²), the stationary variance of the mode is* $V(\beta,\varsigma) = \frac{\beta\sigma^2}{\vartheta_{\varsigma}(2-\beta\vartheta_{\varsigma})}$*, which increases with* β *and diverges at the stability boundary.*
4. *Marginal-cost pricing. For a class whose interactions are internal, the MC target has slope* $c' = d' + \xi' = 2d'$ *(Proposition 4), so* μ *doubles and the stable range of* β *shrinks accordingly.*
5. *Lag–noise trade-off. If the equilibrium value of the mode drifts by* ω *(s) per period, as under order waves, the steady-state tracking lag is* $\omega/(\beta\vartheta_{\varsigma})$ *and the mean-square error is* $\omega^2/(\beta\vartheta_{\varsigma})^2 + V(\beta,\varsigma)$*. Its unique minimiser on* $(0, 2/\vartheta_{\varsigma})$ *solves* $\vartheta_{\varsigma}\sigma^2\beta^3 = \omega^2(2-\vartheta_{\varsigma}\beta)^2$*, and for small gains* $\beta \approx (4\omega^2/(\vartheta_{\varsigma}\sigma^2))^{1/3}$*.*

Proof. (a) Since $\varrho(J)$ is the larger of the moduli of the two extreme eigenvalues, it is minimised where they are equal and opposite, $1-\beta\vartheta_{\varsigma_{\min}} = -(1-\beta\vartheta_{\mu})$, which gives β_{opt} and ϱ_{opt}. (b) The condition $\hat{\beta}(1+\theta Q\mu) < 2$ is equivalent to $1+\theta Q\mu < 2+\theta Q\hat{\mu}$. (c) Near equilibrium, the mode coordinate z_n evolves as $z_{n+1} = (1-\beta\vartheta_{\varsigma})z_n + \beta\epsilon_n$, where ϵ_n is the estimation noise, and the stationary variance of this autoregression is $\beta^2\sigma^2/\left(1-(1-\beta\vartheta_{\varsigma})^2\right)$, which simplifies to V. (d) follows from $\xi = d$. (e) If the equilibrium value $z_n^{\star}$ increases by ω per period, the tracking error $\tilde{z}_n = z_n - z_n^{\star}$ obeys $\tilde{z}_{n+1} = (1-\beta\vartheta_{\varsigma})\tilde{z}_n - \omega$, whose fixed point is $-\omega/(\beta\vartheta_{\varsigma})$. Differentiating the mean-square error with respect to β and multiplying by the positive factor $\beta^3\vartheta_{\varsigma}^2(2-\beta\vartheta_{\varsigma})^2/2$ gives $\vartheta_{\varsigma}\sigma^2\beta^3 - \omega^2(2-\vartheta_{\varsigma}\beta)^2$, which increases in β, is negative at $\beta = 0$ and positive at $\beta = 2/\vartheta_{\varsigma}$. Parts (a)–(c) and (e) were derived and checked in SymPy (Appendix A). □

Part (a) gives the fastest deterministic convergence, part (c) shows that the same gain amplifies estimation noise, and part (e) balances the two when demand moves. Since dispatch depends only on price *contrasts* between routes, the drift and noise in part (e) are those of the π-centred prices, $p - (\pi^{\top}p)\mathbf{1}$, whose drift under demand waves is much smaller than that of the price levels. Algorithm 1 estimates μ, ω and σ online and combines parts (a) and (e).

Order waves help the online fit, because they move the flows across a wide range and therefore identify the slopes of the delay functions. Self-regulated averaging [97] adapts step sizes to compute SUE offline from the distance to an auxiliary solution; Algorithm 1 instead sets the gain online from estimated spectral, drift and noise quantities.

Theorem 7 (Global convergence in the low-gain regime). *Suppose that* $0 \le c_e' \le L_c$ *for all edges, where* L_c *(s²) is a slope bound, and that* $\kappa := \theta Q L_c \lambda_{\max}(\Delta\Delta^{\top}) < 1$*. Then, for every* $\beta \in (0,1]$*,* Φ_{β} *is a contraction in the Euclidean norm with modulus* $1-\beta(1-\kappa)$*, and* $p(n) \to p^{\star}$ *from every initial price vector. If, in addition, the implemented*

update is $p(n+1) = \Phi_\beta\big(p(n)\big) + \beta\epsilon_n$, *where the estimation error satisfies* $\| \epsilon_n \| \leq \bar{\epsilon}$ *(s), then* $\| p(n) - p^\star \| \leq (1-\beta(1-\kappa))^n \| p(0) - p^\star \| + \frac{\bar{\epsilon}}{1-\kappa}$.

Proof. For any price vectors p and p', the mean-value theorem gives $\| f(p) - f(p') \| \leq \sup \| \partial f / \partial p \| \| p - p' \|$, and $\| \partial f / \partial p \| = \theta Q \lambda_{\max}(\Delta \Pi \Delta^\top) \leq \theta Q \lambda_{\max}(\Delta\Delta^\top)$ because $\lambda_{\max}(\Pi) \leq \max_r \pi_r \leq 1$ by eigenvalue interlacing [109]. Since c acts edgewise with slopes in $[0, L_c]$, $\| c(f) - c(f') \| \leq L_c \| f - f' \|$. Hence $\| \Phi_\beta(p) - \Phi_\beta(p') \| \leq (1 - \beta + \beta\kappa) \| p - p' \|$, and the Banach fixed-point theorem gives convergence to the unique fixed point $p^\star$ of Theorem 5. With the error term, $\| p(n+1) - p^\star \| \leq q \| p(n) - p^\star \| + \beta\bar{\epsilon}$ with $q = 1 - \beta(1-\kappa)$; iterating and summing the geometric series, whose sum is at most $\beta\bar{\epsilon}/(1-q) = \bar{\epsilon}/(1-\kappa)$, gives the bound. It is informative only when a uniform bound $\bar{\epsilon}$ on the estimation error has been established independently. □

5. Experimental Design

5.1. Simulation Platforms

We use two simulators. The *lane-network simulator* implements the layered controller of Section 3 on single-lane edges: the FCFS conflict-zone reservation of rule (iii) with half-width $w_z = 1.3$ m, approach distance $d_{\text{app}} = 3$ m and stop-line gain $k_s = 1\ \text{s}^{-1}$; the one-dimensional filter (6) with safety distance $d_s = 1$ m and CBF gain $\gamma = 1\ \text{s}^{-1}$; and both attributions of Definition 2. Its control period is $\Delta t = 0.1$ s. The *planar simulator* implements the full two-dimensional filter (5) with equal responsibility shares, corridor walls and a velocity box, solved exactly by an active-set QP method, with $\Delta t = 0.05$ s; for single-lane aisles it uses the closed form of the single active headway row. The planar simulator uses homogeneous robots with cruise speed $\bar{v}_i = 1.2$ m/s, Poisson arrivals and fixed 8 s dwells. Table 3 lists which simulator each experiment uses.

5.2. Realistic Synthetic Inputs

Since no public data set records warehouse robot traffic at the resolution of individual control steps, we generate realistic synthetic inputs for the lane-network simulator. Each robot i draws its cruise speed $\bar{v}_i$ uniformly from [1.0,1.4] m/s. With the pick probability of its aisle, it stops once at a pick face drawn uniformly along the aisle, for a dwell time drawn from a lognormal distribution with mean $m_d = 8$ s and coefficient of variation $c_v = 0.5$, clipped to [2,30] s. Since a lognormal variable with log-mean $\mu_{\ln}$ and log-standard deviation $\sigma_{\ln}$ has mean $\exp(\mu_{\ln} + \sigma_{\ln}^2/2)$ and coefficient of variation $\sqrt{\exp(\sigma_{\ln}^2) - 1}$, we solve for

$$\mu_{\ln} = \ln m_d - 1/2 \ln(1 + c_v^2), \qquad \sigma_{\ln} = \sqrt{\ln(1 + c_v^2)}. \tag{19}$$

Order-picking demand arrives in waves [3], so the arrival intensity (robots/s) is modelled as a baseline with Gaussian waves,

$$\lambda(t) = \lambda_0 \left(1 + A_w \sum_{k=1}^{K_w} \exp\left(-\frac{(t - t_k)^2}{2w^2} \right) \right), \tag{20}$$

where λ_0 is the baseline intensity, A_w the relative wave amplitude, K_w the number of waves, t_k the wave centres and w the wave width (s). To generate a non-homogeneous Poisson process with this intensity, we use time rescaling [111]: if

$E_1 < E_2 < \cdots$ are the points of a unit-rate Poisson process, then $t_n = \Lambda^{-1}(E_n)$ are arrivals with intensity λ, where $\Lambda(t) = \int_0^t \lambda\,(t')\,dt'$ is the integrated intensity. Integrating Eq. (20) in SymPy [112] gives the closed form

$$\Lambda(t) = \lambda_0 t + \lambda_0 A_w w\sqrt{\pi/2}\sum_{k=1}^{K_w}\left[\operatorname{erf}\left(\frac{t-t_k}{\sqrt{2}\,w}\right) + \operatorname{erf}\left(\frac{t_k}{\sqrt{2}\,w}\right)\right], \tag{21}$$

which we invert numerically.

Correlated inputs would confound the interpretation of the experiments, so the generator is checked for collinearity. For each input z_j of a design matrix, the variance inflation factor $\mathrm{VIF}_j = 1/\left(1 - R_j^2\right)$, where R_j^2 is the coefficient of determination of the regression of z_j on the other inputs, measures how well z_j is predicted by them; the condition number of the standardised design is the ratio of its largest to its smallest singular value [113]. Table 2 reports both. With a naive encoding that sets the dwell time and pick position to zero for robots that do not pick, the largest VIF is 4.9, an artefact of the encoding. With the conditional encoding that we use, in which dwell time and position are defined only for picking robots, all VIFs equal 1.00 and the condition number is 1.02, so the inputs are not collinear.

Table 2: Collinearity diagnostics of the synthetic inputs (8 h of order-wave demand, 29,613 robots).

Input set	*Largest VIF*	*Cond. number*
Naive encoding (dwell, position = 0 if no pick)	4.93	4.38
Conditional encoding (picking robots)	1.00	1.02
Speed, pick flag, time of day	1.00	1.00

5.3. Scenarios

Table 3 summarises the six experiments.

Experiment 1 (delay identity) compares the measured delay (7) with the intervention-based delay (8) in (a) a pick aisle, (b) a crossing with FCFS reservation and (c) a planar four-way intersection.

Experiment 2 (predictive value) drives single pick aisles with three order waves for 8 h (two seeds per configuration). For every window of length 1 min or 5 min, we record the mean occupancy $\bar{n}$ (robots), the entry rate f (robots/s), the entrance queue length and the intervention accrual rate I_e/T_p, and we predict the mean delay of the robots that *arrive* in the next window. We train on two aisle configurations (40 m with pick probability 0.6; 36 m with 0.7) and test on two unseen ones (48 m with 0.4; 30 m with 0.9).

Experiment 3 (attribution versus marginal external cost) lets a highway lane H (40 m, background flow $Q_H = 0.15$ robots/s) cross a 24 m lane C at an FCFS-reserved zone. For crossing flows f from 0.02 to 0.16 robots/s we run eight coupled 2 h simulations per flow, with common random numbers for the highway stream and the robot attributes. We compare the mean delay imposed per crossing robot, ξ_C, with the counterfactual marginal external cost

$$\mathrm{MEC}(f) = \frac{\partial}{\partial f}\left(Q_H d_H(f) + f\,d_C(f)\right) - d_C(f), \tag{22}$$

where d_H and d_C are the mean delays of highway and crossing robots, estimated by finite differences between neighbouring flows. We repeat the lowest flows with homogeneous speeds to isolate the effect of speed heterogeneity.

Experiment 4 (routing at a busy crossing) lets routed demand $Q = 0.14$ robots/s choose between lane C (free time 20 s) and a 30 m detour D (free time 25 s) that crosses nothing, while the highway carries $Q_H = 0.15$ robots/s. Dispatch is logit with $\theta = 1\,\mathrm{s}^{-1}$, prices are updated every $T_p = 300$ s with $\beta = 0.2$, and each run lasts 8 h, of which the first 2 h are discarded. We compare six price signals: none; density, $\kappa_n \bar{n}_e$, with κ_n calibrated by least squares on a pilot UE run; UE intervention pricing; MC pricing with direct and with root-cause attribution; and an oracle MC price that uses the counterfactual MEC curve of Experiment 3 and is therefore not implementable online.

Experiment 5 (stability and gain design) routes demand between two parallel aisles, A (36 m, pick probability 0.7) and B (48 m, 0.4). Under stationary demand ($Q = 0.30$ robots/s, $\theta = 0.5\,\mathrm{s}^{-1}$, planar simulator) we vary the gain. Under order waves (Eq. (20) with $\lambda_0 = 0.14$ robots/s, $A_w = 1.4$, $K_w = 3$, $w = 1500$ s; lane-network simulator; $T_p = 180$ s) we compare fixed gains with Algorithm 1, using either the rate-optimal gain only (line 5) or the full noise-aware rule (lines 5–7).

Experiment 6 (deadlock signature) records η_i in a head-on encounter in a corridor and in a long run of the planar intersection.

Table 3: Experimental scenarios.

Experiment	*Scenario*	*Simulator*	*Main measures*
1 Identity	aisle; crossing; planar intersection	planar; lane; planar	$\left\lvert D_i - \bar{v}_i^{-1} \int \iota_i \, dt \right\rvert$
2 Prediction	4 aisle configurations, order waves	lane	out-of-sample R^2, MAE, VIF
3 Attribution	highway–lane crossing, 8 flows	lane	ξ_C vs. MEC
4 Routing	crossing vs. detour, 6 price signals	lane	mean travel time, share
5 Gain	two aisles; stationary and waves	planar; lane	travel time, oscillation
6 Deadlock	head-on; planar intersection	planar	η_i

5.4. Performance Measures and Statistical Procedures

We measure the mean travel time of all robots, of routed robots and of background robots; the share of routed robots using a given route; the mean and 95th-percentile congestion delay; and the oscillation of dispatch, defined as the mean absolute change of the share between consecutive periods. For prediction, we report the weighted coefficient of determination R^2 and the mean absolute error (MAE) on the unseen configurations. Since occupancy, flow and intervention rate are linked by Little's law, they are collinear; we therefore report three complementary analyses: out-of-sample accuracy, which collinearity does not bias; ridge regression [114] with the penalty chosen by leave-one-run-out validation; and a model in which the intervention rate is replaced by its residual after regression on occupancy, flow and queue, whose coefficient then measures only the information the other features lack, with heteroskedasticity-consistent (HC3) standard errors. Experiments are replicated over three to ten seeds as stated, and we report means with standard deviations or standard errors.

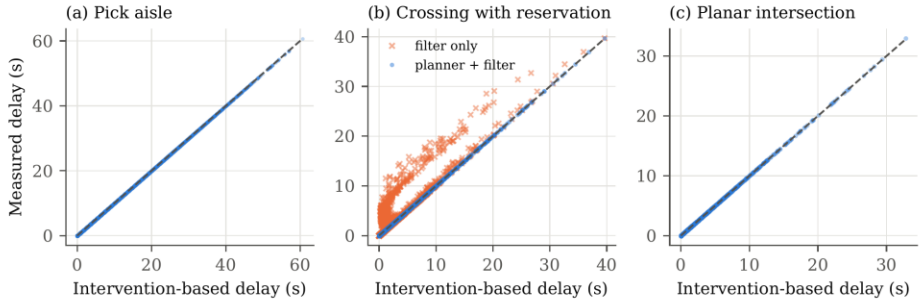


Fig. 1: Experiment 1. Measured congestion delay versus intervention-based delay. (a) Pick aisle. (b) Crossing with reservation: layered (planner + filter) and filter-only estimates. (c) Planar intersection with the full QP. The dashed line is the identity.

Table 4: Experiment 2: out-of-sample prediction of next-window delay on unseen aisle configurations. Weighted R^2 and mean absolute error (MAE, s) of ordinary least squares at horizons of 1 and 5 min, and R^2 of ridge regression at 5 min. Collinearity of the training designs: VIF of (occupancy, flow, queue, intervention) = (7.6, 3.2, 1.0, 3.9) at 1 min and (56.1, 30.8, 1.0, 9.2) at 5 min; condition numbers 5.5 and 15.9.

	1 min		5 min		
Features	R^2	***MAE***	R^2	***MAE***	R^2 ***(ridge)***
Occupancy	0.089	3.59	0.130	2.58	0.130
Occupancy + flow	0.137	3.45	0.277	2.27	0.263
Occupancy + flow + queue	0.137	3.45	0.279	2.27	0.265
Intervention rate only	0.152	3.49	0.211	2.39	0.210
All four (incl. intervention)	0.202	3.34	0.346	2.11	0.337

6. Results

6.1. Validation of the Layered Delay Identity

In the pick aisle, over 67,693 traversals, the intervention-based delay matches the measured delay to within 0.10 s (Fig. 1a). At the crossing with realistic inputs, over 151,214 traversals, the layered estimate matches to within 0.10 s, inside the bound of Corollary 2, $(1 + 1.4/1.0) \times 0.1 = 0.24$ s. The filter layer alone accounts for only 75% of the delay and the planner holds for 23%; a filter-only measurement therefore under-reports the delay of robots held at the stop line (Fig. 1b). At the planar intersection, the identity holds to within 0.097 s over 1,538 traversals, and the norm bound of Theorem 2 overestimates the delay by only 5%, because the filter acts mostly along the direction of travel (Fig. 1c).

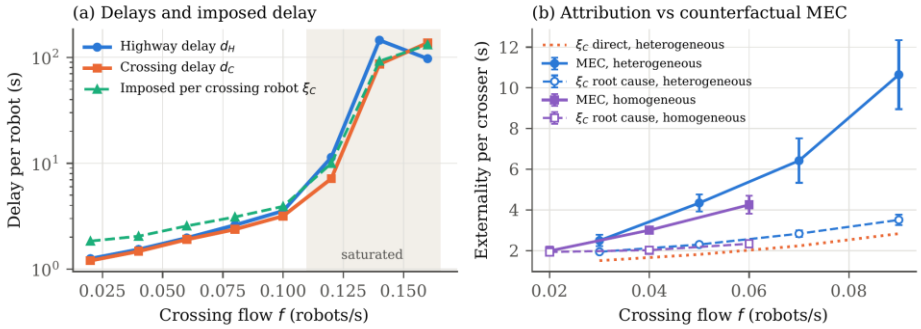


Fig. 2: Experiment 3. (a) Mean congestion delay of highway and crossing robots and attributed delay imposed per crossing robot, ξ_C, versus crossing flow; the shaded band marks flows beyond the capacity of the crossing. (b) Attributed imposed delay (root cause, direct) and counterfactual marginal external cost (finite differences, ± 1 s.e.) with heterogeneous and homogeneous speeds.

Table 5: Experiment 4: routing at a busy crossing. Mean travel time (s) of all robots, of routed robots and of highway robots, and the share of routed robots using the crossing lane C. Means $\pm$ s.d. over three seeds.

Price signal	*All robots*	*Routed*	*Highway*	*Share to C*
None[a]	36.97 ± 0.34	28.75	44.72	0.99
Density ($\kappa_n \bar{n}_e$)	31.94 ± 0.89	24.67	38.74	0.85
UE intervention	31.45 ± 0.40	24.56	37.90	0.78
MC, direct attrib.	30.47 ± 0.17	23.99	36.54	0.61
MC, root-cause attrib.	30.35 ± 0.13	23.99	36.31	0.57
MC, oracle MEC[b]	30.13 ± 0.03	24.36	35.53	0.37

[a]One of three seeds saturated the crossing (mean travel time 1232 s); the row reports the other two. [b]Uses the counterfactual MEC curve of Experiment 3; not implementable online.

6.2. Predictive Value of the Intervention Signal

The training designs are collinear (Table 4): occupancy and intervention rate correlate at 0.84 (1 min) and 0.91 (5 min), and the largest VIF is 7.6 at 1 min and 56.1 at 5 min. This gives 1828 training and 1828 test windows at 1 min, and 368 and 368 at 5 min.

Since a window holds only a few arrivals and the delay of each robot depends on the random dwell times of the robots ahead of it, next-window delay is inherently noisy, and no feature set explains more than about a third of its variance on the unseen aisles. The ranking of the feature sets is nevertheless clear. On its own, the intervention rate predicts better than occupancy (R^2 of 0.152 against 0.089 at 1 min, and 0.211 against 0.130 at 5 min). When it is added to occupancy, flow and queue length, the out-of-sample R^2 rises from 0.137 to 0.202 at 1 min and from 0.279 to 0.346 at 5 min, and the MAE falls by 3.3% and 6.8%, respectively. Ridge regression gives the same ranking. The collinearity diagnostics explain the in-sample estimates. At 1 min, the orthogonalised intervention residual has coefficient 0.89 (HC3 standard error 0.40; $p = 0.025$), so the intervention signal carries information that occupancy, flow and queue length lack. At 5 min, occupancy and flow are almost collinear (correlation 0.97), as Little's law implies over longer windows; the standard errors therefore inflate and the residual coefficient is not significant ($p = 0.405$), although the out-of-

sample gain persists. We conclude that the intervention signal adds predictive information beyond occupancy and flow and that this gain transfers to aisle configurations not seen in training, which is precisely the setting in which a density proxy fails (Experiment 5).

Since the lane simulator uses the one-dimensional filter, the intervention rate in this experiment coincides with the accrued speed deficit (Section 4.2). The result therefore shows that accrued deficit carries information that occupancy and flow lack, however it is logged; it does not by itself show an advantage of filter logs over speed telemetry.

Table 6: Experiment 5, stationary demand: closed-loop routing of CBF-filtered robots on two aisles ($Q = 0.30$ robots/s, $\theta = 0.5\,\mathrm{s}^{-1}$, $\beta^\star = 0.33$). Means and standard deviations over five seeds; the share s.d. is over price periods of one run.

Price signal	β	*Travel time (s)*	*Delay (s)*	*Share to A*	*Share s.d.*
Intervention	1.0	77.2 ± 3.1	37.6	0.48	0.49
Intervention	0.6	65.1 ± 3.0	24.3	0.32	0.42
Intervention	0.2	46.5 ± 0.5	7.7	0.58	0.17
Intervention	0.1	45.7 ± 0.2	7.0	0.59	0.08
Density ($\kappa_n \bar{n}_e$)	0.1	49.6 ± 0.4	11.7	0.70	0.06
Completed only	1.0	61.2 ± 1.1	18.2	0.02	0.00

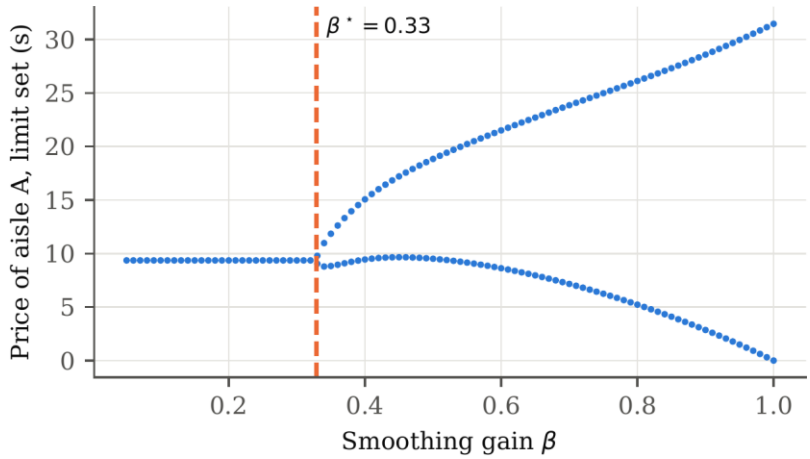


Fig. 3: Experiment 5, stationary demand. Limit set of the price of aisle A under the mean-field map (16) versus the smoothing gain β; the vertical line is the stability bound $\beta^\star = 2/(1 + \theta Q \mu)$.

6.3. Attribution versus the Marginal External Cost

Fig. 2 shows three regimes. Below about 0.1 robots/s, delays grow smoothly; between 0.12 and 0.14 robots/s the crossing saturates and delays jump by an order of magnitude. Every run kept the safety distance (minimum separation 1.00 m, no barrier violations). With homogeneous speeds at the lowest flows, the attributed imposed delay recovers 97% of the counterfactual MEC, as Theorem 4 predicts for pairwise encounters. As the flow rises, the share falls (55% at $f = 0.06$ robots/s with homogeneous speeds) because congestion becomes superlinear (Proposition 4). With heterogeneous speeds, the share is lower at every flow, from 78% at $f = 0.03$ to 33% at $f = 0.09$ robots/s, which is the

bunching effect described after Proposition 4. Root-cause forwarding raises the imposed delay over direct attribution by 27% on average, because it charges highway queue cascades to the crossing robot that started them. We conclude that attribution is a conservative, model-free externality estimate: exact in light traffic and an underestimate in heavy traffic.

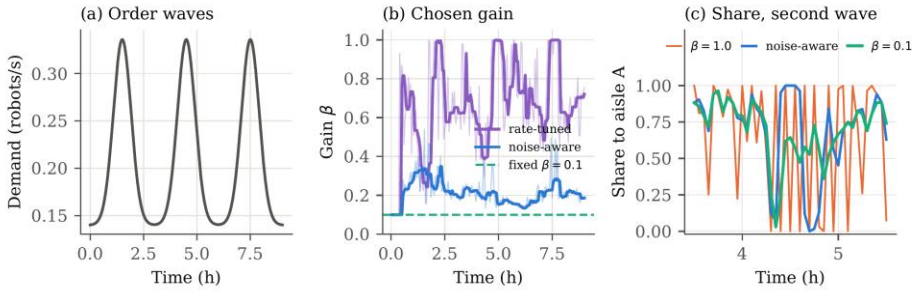


Fig. 4: Experiment 5, order waves (seed 0). (a) Demand intensity (20). (b) Gains chosen by rate-tuned and noise-aware tuning (thin: per period; thick: 5-period running median) and the best fixed gain. (c) Share of robots dispatched to aisle A in each 180 s period around the second wave.

Table 7: Experiment 5, order waves: mean travel time and congestion delay, 95th-percentile (P95) delay, and oscillation of dispatch. Means ± s.d. over three seeds. Rate-tuned: Algorithm 1 with line 5 only; noise-aware: lines 5–7.

Gain policy	*Travel (s)*	*Delay (s)*	*P95 (s)*	*Osc.*
Fixed $\beta = 0.1$	45.26 ± 0.38	7.00	26.9	0.098
Fixed $\beta = 0.2$	45.86 ± 0.93	7.45	28.0	0.125
Fixed $\beta = 0.4$	47.92 ± 0.50	9.26	34.4	0.204
Fixed $\beta = 1.0$	52.22 ± 0.33	13.29	47.9	0.519
Rate-tuned	49.31 ± 0.84	10.70	39.5	0.344
Noise-aware tuned	49.04 ± 2.08	10.17	38.2	0.148

6.4. Routing at a Busy Crossing

Without prices, nearly all routed robots take the shorter crossing lane, which pushes the crossing past capacity: in 1 of three seeds it saturated, with a mean travel time of 1232 s (row [a] of Table 5 reports the other seeds). UE intervention pricing keeps the crossing below capacity but still sends 0.78 of the routed robots across it, because each robot pays only the delay it suffers. Density pricing, with calibrated coefficient $\kappa_n = 1.36$ s per robot, is slightly worse than UE pricing: it overprices the long, uncongested detour and therefore sends even more robots across (0.85). MC pricing with root-cause attribution charges each crossing robot for the highway platoons that it stops. It lowers the crossing share to 0.57 and the mean travel time of all robots by 3.5% relative to UE pricing, by 5.0% relative to density pricing and by 18% relative to the unsaturated unpriced runs. Highway robots gain most (from 37.9 s to 36.3 s), but routed robots also gain (from 24.6 s to 24.0 s), since UE routing overloads the crossing and so slows the robots that use it. Direct attribution yields a smaller improvement (3.1%), because it does not charge the crossing robot for the highway queue cascade. The oracle price reduces the travel time by 4.2%, so attribution recovers 83% of the achievable gain without a delay model or a gradient estimate. The oracle diverts more robots

to the detour (0.37) and makes routed robots slightly slower (24.4 s) to the benefit of the highway. No run violated a barrier.

6.5. Stability and Gain Design

Stationary demand. With the delay functions fitted from data, the equilibrium sends $\pi_A^\star = 0.60$ of the robots to aisle A, the curvature is $\mu = 33.9\ \mathrm{s}^2$, the stability bound is $\beta^\star = 0.33$ and, by Proposition 5(a), $\beta_{\mathrm{opt}} = 0.28$ with contraction factor 0.72 per period, against 0.90 for $\beta = 0.1$. The mean-field map (16) converges below $\beta^\star$ and settles on a period-two orbit above it, exactly at the predicted boundary (Fig. 3). In closed loop with CBF-filtered robots (Table 6), gains above $\beta^\star$ flip-flop between the aisles and raise the mean travel time by 69% relative to $\beta = 0.1$. A gain just below $\beta^\star$ is stable but noisy, as Proposition 5(c) predicts. Density pricing is stable but converges to the wrong split and costs 8.6% more travel time, and an estimator that uses only completed traversals locks in.

Order waves. Table 7 and Fig. 4 report the order-wave runs, in which demand rises from 0.14 to 0.34 robots/s at each wave.

The smallest fixed gain, $\beta = 0.1$, gives the best result, with a mean travel time of 45.3 s and an oscillation of 0.10. The undamped update ($\beta = 1$) flip-flops between the aisles (oscillation 0.52) and costs 15% more travel time. The rate-optimal gain of Proposition 5(a) does not help. Since the fitted curvature is small over most of the run, the rate-optimal gain is large (median 0.66), and it amplifies estimation noise as Proposition 5(c) predicts, giving a travel time of 49.3 s and an oscillation of 0.34. The noise-aware rule of Proposition 5(e) lowers the median gain to 0.21 and the oscillation to 0.15, but its travel time (49.0 s) remains 8% above that of the best fixed gain and varies widely between seeds. Two causes are likely, although we have not isolated them: the cubic delay model may underestimate the curvature at the wave peaks, which would violate the robustness margin of Proposition 5(b), and the drift and noise estimates are themselves noisy over three-minute periods. The theory therefore identifies which gains work, namely small gains well inside the stability bound, whereas online tuning did not outperform a conservatively chosen fixed gain.

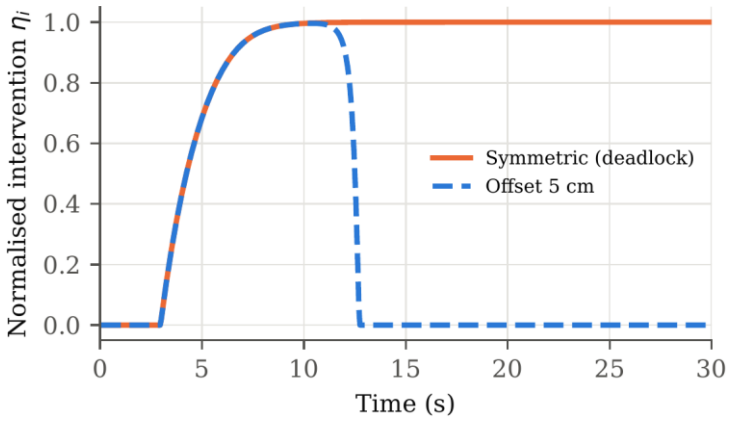


Fig. 5: Experiment 6. Normalised intervention η_i of one robot in a head-on encounter with symmetric (deadlock) and slightly perturbed initial conditions.

6.6. Stop and Deadlock Signature

When two robots meet head-on in a corridor with perfectly symmetric initial conditions, the full QP brings both to rest, which is the undesirable equilibrium described in [67], and their normalised intervention converges to $\eta_i = 1$ (Fig. 5). With a 5 cm lateral offset, the robots pass each other and η_i returns to zero. In a one-hour planar-intersection run at 0.10 robots/s per corridor, robots gridlocked after about 45 minutes. At that moment every robot within 4 m of the centre had $\eta_i = 1$ to machine precision, while robots further back, still creeping forward, had $\eta_i < 1$. Since a gridlocked edge accrues large intervention but completes few traversals, a deadlock check, triggered by persistent $\eta_i = 1$, must accompany the price.

7. Discussion

7.1. Implementation in a Warehouse Control Stack

Fig. 6 shows how the method fits into a warehouse control stack. Each robot already runs its planner logic and its filter. Per edge, it accumulates its longitudinal deviation from the free-flow command and the blame vector of that deviation, and it reports two scalars per period: the suffered delay and the imposed delay. The fleet manager converts them into edge prices (Proposition 3 and Theorem 4), smooths them with a gain chosen by Proposition 5, and uses them as edge costs in dispatching, in MAPF [10, 86] or in guidance-graph optimisation [26, 27].

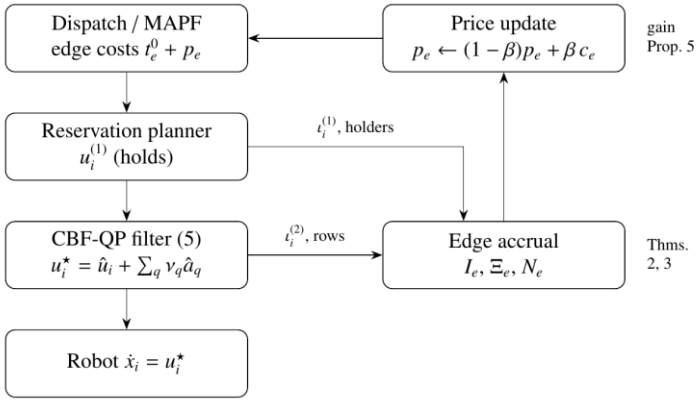


Fig. 6: Intervention pricing in a layered warehouse control stack. Planner and filter are unchanged; their deviations from the free-flow command are accumulated per edge, attributed to causing robots, and fed back as suffered-plus-imposed delay prices.

7.2. Theoretical and Practical Implications

Occupancy describes where robots are, and travel time arrives late and carries no information about cause. The layered intervention is exact as an accounting identity (Theorem 2), available while the delay accrues (Proposition 3) and attributable (Theorem 3). The first two properties are shared by any accrued speed-deficit measurement; the third requires the filter's constraint data and is the CBF-specific contribution. It makes marginal-cost routing possible without estimating the gradients of delay functions over many routes, a task that scales poorly with the number of routes and requires persistent excitation. Experiment 3 quantifies the price of this convenience: attribution misses the part

of the externality that acts through superlinear queueing and bunching, so that an online estimate of the elasticity of delay is a natural correction. Accumulated over a shift, the imposed delay maps which robots, routes and pick faces cause congestion, and the static part of Theorem 1(iv) maps geometry that forces robots off their paths. Barrier rows for people [25, 89] can be priced separately to keep robots away from zones that people use at certain times. For the smoothing gain, our results support a simple rule: choose a fixed gain well inside the stability bound, computed from an upper bound on the curvature (Theorem 7 and Proposition 5(b)) and the expected noise level.

7.3. Limitations and Future Research

The attribution theory is stated for lane networks, where interventions are braking. In the plane, individual constraint forces can have positive or negative longitudinal components; signed attribution still conserves delay, but nonnegativity is lost. Since the attribution follows instantaneous binding constraints, externalities that act through headway compression are charged to the slow robot rather than to the robot that compressed the headway. The routing analysis is mean-field and assumes that the price period is long compared with traversal times. Logit dispatch models randomised assignment; deterministic dispatch is the limit $\theta \to \infty$, for which no fixed gain guarantees stability. Online gain tuning did not outperform a conservative fixed gain in our tests. In one dimension the delay signal coincides with the speed deficit, so its advantage over speed telemetry rests on the attribution, which we have tested only in simulation. The sampled-data guarantee of Lemma 1 covers squared-distance and affine barriers with synchronous updates; nonlinear walls, asynchronous neighbour information, perception error and the fallback used when a QP is infeasible need separate analysis. The experiments use synthetic inputs in realistic ranges, single-integrator robots and an FCFS reservation. Future work should address elasticity-corrected attribution, more reliable online gain tuning, integration with a full lifelong MAPF planner, second-order dynamics with high-order barriers [48, 49], and validation on physical fleets.

8. Conclusion

The control layers of a warehouse robot already compute, at every step, how much the robot is held back and by which constraint. We showed that, summed over the layers, this intervention equals the robot's congestion delay exactly. Through the multipliers of the safety filter and the hold decisions of the planner, the delay can be attributed to the robots that caused it, with every second charged once. Suffered plus imposed delay is then the marginal social cost when conflicts are pairwise, and a conservative externality price when they are not. Routing on these prices converges to the corresponding stochastic equilibrium when the smoothing gain respects a bound inherited from classical day-to-day assignment theory, and our gain analysis explains why small fixed gains work best under noisy estimates. Simulations with realistic synthetic inputs confirmed the identity, the accuracy of the attribution in light traffic and its conservatism in heavy traffic, the predictive value of the signal beyond occupancy, and a lower system travel time at a busy crossing.

Appendix A. Symbolic Verification

We derived or verified the following results with SymPy [112]; the script is part of the supplementary code.

1. Proposition 5(a): equating the moduli of the extreme eigenvalues gives $\beta_{\mathrm{opt}} = 2/\big(2 + \theta Q(\mu + \varsigma_{\min})\big)$ and $\varrho_{\mathrm{opt}} = \theta Q(\mu - \varsigma_{\min})/\big(2 + \theta Q(\mu + \varsigma_{\min})\big)$.
2. Proposition 5(b): $2 - \hat{\beta}(1 + \theta Q\mu)$ has the sign of $1 + \theta Q(\hat{\mu} - \mu)$, so stability holds if and only if $\hat{\mu} > \mu - 1/(\theta Q)$.
3. Proposition 5(c): $\beta^2\sigma^2/(1 - r^2)$ with $r = 1 - \beta\vartheta_{\varsigma}$ simplifies to $V(\beta, \varsigma)$.
4. Proposition 5(e): the stationarity condition $\vartheta_{\varsigma}\sigma^2\beta^3 - \omega^2\big(2 - \vartheta_{\varsigma}\beta\big)^2 = 0$, the lag $-\omega/\big(\vartheta_{\varsigma}\beta\big)$ and the small-gain asymptote $(4\omega^2/\big(\vartheta_{\varsigma}\sigma^2\big))^{1/3}$.
5. Two parallel routes: the eigenvalues of $M^{1/2}\Pi^{\star}M^{1/2}$ are 0 and $\pi_1\pi_2(c_1{}' + c_2{}')$, so $\mu = \pi_1\pi_2(c_1{}' + c_2{}')$.
6. Theorem 4: for three classes with a symbolic matrix Ω, $\partial \mathrm{TC}/\,\partial f_{r'} - (t^0_{r'} + \sum_r \Omega_{r'r} f_r + \sum_r f_r \Omega_{rr'}) \equiv 0$.
7. Proposition 4: $fd' - d = 2a_3 f^3$ for $d = a_1 f + a_3 f^3$.
8. Eqs. (19) and (21): the lognormal parameters and the integrated wave intensity.

CRediT Authorship Contribution Statement

Olasubomi Akanbi: Conceptualization, Methodology, Software, Writing – original draft. **Toheeb Babatunde Ibrahim:** Formal analysis, Validation, Writing – review & editing. **Joseph Oladejo:** Visualization. **Warith Akinade:** Software.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The simulation code, the synthetic-data generator and all results are available from the corresponding author on reasonable request.

Declaration of Generative AI and AI-Assisted Technologies

During the preparation of this work the authors used Claude (Anthropic) to assist with literature search and to check the text for grammatical errors and mistakes. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

1. P. R. Wurman, R. D'Andrea, M. Mountz, Coordinating hundreds of cooperative, autonomous vehicles in warehouses, AI Magazine 29 (1) (2008) 9–19.
2. K. Azadeh, R. De Koster, D. Roy, Robotized and automated warehouse systems: Review and recent developments, Transportation Science 53 (4) (2019) 917–945.
3. N. Boysen, R. De Koster, F. Weidinger, Warehousing in the e-commerce era: A survey, European Journal of Operational Research 277 (2) (2019) 396–411.
4. G. Fragapane, R. De Koster, F. Sgarbossa, J. O. Strandhagen, Planning and control of autonomous mobile robots for intralogistics: Literature review and research agenda, European Journal of Operational Research 294 (2) (2021) 405–426.

5. T. Le-Anh, M. B. M. De Koster, A review of design and control of automated guided vehicle systems, European Journal of Operational Research 171 (1) (2006) 1–23.
6. I. F. A. Vis, Survey of research in the design and control of automated guided vehicle systems, European Journal of Operational Research 170 (3) (2006) 677–709.
7. L. Qiu, W.-J. Hsu, S.-Y. Huang, H. Wang, Scheduling and routing algorithms for AGVs: A survey, International Journal of Production Research 40 (3) (2002) 745–760.
8. B. P. Gerkey, M. J. Matarić, A formal analysis and taxonomy of task allocation in multi-robot systems, The International Journal of Robotics Research 23 (9) (2004) 939–954.
9. H. Ma, J. Li, T. K. S. Kumar, S. Koenig, Lifelong multi-agent path finding for online pickup and delivery tasks, in: Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2017, pp. 837–845.
10. J. Li, A. Tinka, S. Kiesel, J. W. Durham, T. K. S. Kumar, S. Koenig, Lifelong multi-agent path finding in large-scale warehouses, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 11272–11281.
11. I. Draganjac, D. Miklić, Z. Kovačić, G. Vasiljević, S. Bogdan, Decentralized control of multi-AGV systems in autonomous warehousing applications, IEEE Transactions on Automation Science and Engineering 13 (4) (2016) 1433–1447.
12. V. Digani, L. Sabattini, C. Secchi, C. Fantuzzi, Ensemble coordination approach in multi-AGV systems applied to industrial warehouses, IEEE Transactions on Automation Science and Engineering 12 (3) (2015) 922–934.
13. K. Dresner, P. Stone, A multiagent approach to autonomous intersection management, Journal of Artificial Intelligence Research 31 (2008) 591–656.
14. W. Hönig, S. Kiesel, A. Tinka, J. W. Durham, N. Ayanian, Persistent and robust execution of MAPF schedules in warehouses, IEEE Robotics and Automation Letters 4 (2) (2019) 1125–1131.
15. A. D. Ames, J. W. Grizzle, P. Tabuada, Control barrier function based quadratic programs with application to adaptive cruise control, in: Proceedings of the 53rd IEEE Conference on Decision and Control, 2014, pp. 6271–6278.
16. A. D. Ames, X. Xu, J. W. Grizzle, P. Tabuada, Control barrier function based quadratic programs for safety critical systems, IEEE Transactions on Automatic Control 62 (8) (2017) 3861–3876.
17. A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, P. Tabuada, Control barrier functions: Theory and applications, in: Proceedings of the 18th European Control Conference, 2019, pp. 3420–3431.
18. P. Wieland, F. Allgöwer, Constructive safety using control barrier functions, IFAC Proceedings Volumes 40 (12) (2007) 462–467.
19. K. P. Wabersich, A. J. Taylor, J. J. Choi, K. Sreenath, C. J. Tomlin, A. D. Ames, M. N. Zeilinger, Data-driven safety filters: Hamilton-Jacobi reachability, control barrier functions, and predictive methods for uncertain systems, IEEE Control Systems Magazine 43 (5) (2023) 137–177.
20. K. L. Hobbs, M. L. Mote, M. C. L. Abate, S. D. Coogan, E. M. Feron, Runtime assurance for safety-critical systems: An introduction to safety filtering approaches for complex control systems, IEEE Control Systems Magazine 43 (2) (2023) 28–65.
21. K.-C. Hsu, H. Hu, J. F. Fisac, The safety filter: A unified view of safety-critical control in autonomous systems, Annual Review of Control, Robotics, and Autonomous Systems 7 (2024).
22. L. Wang, A. D. Ames, M. Egerstedt, Safety barrier certificates for collisions-free multirobot systems, IEEE Transactions on Robotics 33 (3) (2017) 661–674.
23. U. Borrmann, L. Wang, A. D. Ames, M. Egerstedt, Control barrier certificates for safe swarm behavior, IFAC-PapersOnLine 48 (27) (2015) 68–73.

24. P. Glotfelter, J. Cortés, M. Egerstedt, Nonsmooth barrier functions with applications to multi-robot systems, IEEE Control Systems Letters 1 (2) (2017) 310–315.
25. S. Farrell, C. Li, H. Yu, R. Yoshimitsu, S. Gao, H. I. Christensen, Safe human robot navigation in warehouse scenario, arXiv preprint arXiv:2503.21141 (2025).
26. Z. Chen, D. Harabor, J. Li, P. J. Stuckey, Traffic flow optimisation for lifelong multi-agent path finding, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 38, 2024.
27. Y. Zhang, H. Jiang, V. Bhatt, S. Nikolaidis, J. Li, Guidance graph optimization for lifelong multi-agent path finding, in: Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI), 2024.
28. L. Cohen, T. Uras, S. Koenig, Feasibility study: Using highways for bounded-suboptimal multi-agent path finding, in: Proceedings of the International Symposium on Combinatorial Search (SoCS), 2015.
29. M.-F. Li, M. Sun, The study of highway for lifelong multi-agent path finding, arXiv preprint arXiv:2304.04217 (2023).
30. S. H. Q. Li, D. J. Calderone, B. Açıkmeşe, Congestion-aware path coordination game with Markov decision process dynamics, IEEE Control Systems Letters 7 (2023) 431–436.
31. A. C. Pigou, The Economics of Welfare, Macmillan, London, 1920.
32. M. Beckmann, C. B. McGuire, C. B. Winsten, Studies in the Economics of Transportation, Yale University Press, New Haven, CT, 1956.
33. W. S. Vickrey, Congestion theory and transport investment, The American Economic Review 59 (2) (1969) 251–260.
34. H. Hu, K. Nakamura, J. F. Fisac, SHARP: Shielding-aware robust planning for safe and efficient human-robot interaction, IEEE Robotics and Automation Letters 7 (2) (2022) 5591–5598.
35. J. Grover, C. Liu, K. Sycara, The before, during, and after of multi-robot deadlock, The International Journal of Robotics Research 42 (6) (2023) 317–336.
36. I. Kaddoura, B. Kickhöfer, Optimal road pricing: Towards an agent-based marginal social cost approach, VSP Working Paper 14-01, Technische Universität Berlin, Transport Systems Planning and Transport Telematics (2014).
37. I. Kaddoura, Marginal congestion cost pricing in a multi-agent simulation: Investigation of the greater Berlin area, Journal of Transport Economics and Policy 49 (4) (2015) 560–578.
38. J. L. Horowitz, The stability of stochastic equilibrium in a two-link transportation network, Transportation Research Part B: Methodological 18 (1) (1984) 13–28.
39. D. Watling, Stability of the stochastic equilibrium assignment problem: A dynamical systems approach, Transportation Research Part B: Methodological 33 (4) (1999) 281–312.
40. G. E. Cantarella, E. Cascetta, Dynamic processes and equilibrium in transportation networks: Towards a unifying theory, Transportation Science 29 (4) (1995) 305–329.
41. S. Prajna, A. Jadbabaie, Safety verification of hybrid systems using barrier certificates, in: Hybrid Systems: Computation and Control (HSCC 2004), Vol. 2993 of Lecture Notes in Computer Science, Springer, 2004, pp. 477–492.
42. F. Blanchini, Set invariance in control, Automatica 35 (11) (1999) 1747–1767.
43. E. D. Sontag, A 'universal' construction of Artstein's theorem on nonlinear stabilization, Systems & Control Letters 13 (2) (1989) 117–123.
44. R. A. Freeman, P. V. Kokotović, Robust Nonlinear Control Design: State-Space and Lyapunov Techniques, Birkhäuser, Boston, 1996.
45. X. Xu, P. Tabuada, J. W. Grizzle, A. D. Ames, Robustness of control barrier functions for safety critical control, IFAC-PapersOnLine 48 (27) (2015) 54–61.

46. M. Jankovic, Robust control barrier functions for constrained stabilization of nonlinear systems, Automatica 96 (2018) 359–367.
47. S. Kolathaya, A. D. Ames, Input-to-state safety with control barrier functions, IEEE Control Systems Letters 3 (1) (2019) 108–113.
48. Q. Nguyen, K. Sreenath, Exponential control barrier functions for enforcing high relative-degree safety-critical constraints, in: Proceedings of the American Control Conference, 2016, pp. 322–328.
49. W. Xiao, C. Belta, High-order control barrier functions, IEEE Transactions on Automatic Control 67 (7) (2022) 3655–3662.
50. A. J. Taylor, A. D. Ames, Adaptive safety with control barrier functions, in: Proceedings of the American Control Conference, 2020, pp. 1399–1405.
51. A. Agrawal, K. Sreenath, Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation, in: Robotics: Science and Systems XIII, 2017.
52. J. Zeng, B. Zhang, K. Sreenath, Safety-critical model predictive control with discrete-time control barrier function, in: Proceedings of the American Control Conference, 2021, pp. 3882–3889.
53. L. Lindemann, D. V. Dimarogonas, Control barrier functions for signal temporal logic tasks, IEEE Control Systems Letters 3 (1) (2019) 96–101.
54. Y. Chen, M. Jankovic, M. Santillo, A. D. Ames, Backup control barrier functions: Formulation and comparative study, in: Proceedings of the 60th IEEE Conference on Decision and Control, 2021.
55. B. Morris, M. J. Powell, A. D. Ames, Sufficient conditions for the Lipschitz continuity of QP-based multi-objective control of humanoid robots, in: Proceedings of the 52nd IEEE Conference on Decision and Control, 2013.
56. R. Cheng, G. Orosz, R. M. Murray, J. W. Burdick, End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 3387–3395.
57. C. Dawson, S. Gao, C. Fan, Safe control with learned certificates: A survey of neural Lyapunov, barrier, and contraction methods for robotics and control, IEEE Transactions on Robotics 39 (3) (2023) 1749–1767.
58. D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, M. Egerstedt, The Robotarium: A remotely accessible swarm robotics research testbed, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2017, pp. 1699–1706.
59. Y. Chen, A. Singletary, A. D. Ames, Guaranteed obstacle avoidance for multi-robot operations with limited actuation: A control barrier function approach, IEEE Control Systems Letters 5 (1) (2021) 127–132.
60. G. Notomista, M. Egerstedt, Constraint-driven coordinated control of multi-robot systems, in: Proceedings of the American Control Conference, 2019, pp. 1990–1996.
61. W. Luo, W. Sun, A. Kapoor, Multi-robot collision avoidance under uncertainty with probabilistic safety barrier certificates, in: Advances in Neural Information Processing Systems 33 (NeurIPS 2020), 2020.
62. Z. Qin, K. Zhang, Y. Chen, J. Chen, C. Fan, Learning safe multi-agent control with decentralized neural barrier certificates, in: International Conference on Learning Representations (ICLR), 2021.
63. V. Zinage, A. Jha, R. Chandra, E. Bakolas, Decentralized safe and scalable multi-agent control under limited actuation, arXiv preprint arXiv:2409.09573 (2024).
64. J. Autenrieb, M. Spiller, Auction-based responsibility allocation for scalable decentralized safety filters in cooperative multi-agent collision avoidance, arXiv preprint arXiv:2510.21546 (2025).
65. J. Autenrieb, M. Spiller, H.-S. Shin, N. Cho, Combinatorial safety-critical coordination of multi-agent systems via mixed-integer responsibility allocation and control barrier functions, arXiv preprint arXiv:2603.05762 (2026).

66. C. K. Sah, J. Keshavan, Exact feasibility certification and optimal responsibility allocation for multi-robot CBF safety filters, arXiv preprint arXiv:2609.14935 (2026).
67. M. F. Reis, A. P. Aguiar, P. Tabuada, Control barrier function-based quadratic programs introduce undesirable asymptotically stable equilibria, IEEE Control Systems Letters 5 (2) (2021) 731–736.
68. J. S. Grover, C. Liu, K. Sycara, Deadlock analysis and resolution for multi-robot systems, in: Algorithmic Foundations of Robotics XIV (WAFR 2020), Springer Proceedings in Advanced Robotics, Springer, 2021.
69. M. Müller, Multi-agent reinforcement learning for deadlock handling among autonomous mobile robots, Doctoral dissertation, arXiv preprint arXiv:2511.07071 (2025).
70. O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, The International Journal of Robotics Research 5 (1) (1986) 90–98.
71. D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, IEEE Robotics & Automation Magazine 4 (1) (1997) 23–33.
72. P. Fiorini, Z. Shiller, Motion planning in dynamic environments using velocity obstacles, The International Journal of Robotics Research 17 (7) (1998) 760–772.
73. J. van den Berg, M. Lin, D. Manocha, Reciprocal velocity obstacles for real-time multi-agent navigation, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2008, pp. 1928–1935.
74. J. van den Berg, S. J. Guy, M. Lin, D. Manocha, Reciprocal n-body collision avoidance, in: Robotics Research, Vol. 70 of Springer Tracts in Advanced Robotics, Springer, 2011, pp. 3–19.
75. D. Zhou, Z. Wang, S. Bandyopadhyay, M. Schwager, Fast, on-line collision avoidance for dynamic vehicles using buffered Voronoi cells, IEEE Robotics and Automation Letters 2 (2) (2017) 1047–1054.
76. D. Helbing, P. Molnár, Social force model for pedestrian dynamics, Physical Review E 51 (5) (1995) 4282–4286.
77. R. Olfati-Saber, Near-identity diffeomorphisms and exponential ϵ-tracking and ϵ-stabilization of first-order nonholonomic SE(2) vehicles, in: Proceedings of the American Control Conference, 2002.
78. T. Lamballais, D. Roy, M. B. M. De Koster, Estimating performance in a robotic mobile fulfillment system, European Journal of Operational Research 256 (3) (2017) 976–990.
79. D. Roy, S. Nigam, R. De Koster, I. Adan, J. Resing, Robot-storage zone assignment strategies in mobile fulfillment systems, Transportation Research Part E: Logistics and Transportation Review 122 (2019) 119–142.
80. M. Merschformann, T. Lamballais, M. B. M. De Koster, L. Suhl, Decision rules for robotic mobile fulfillment systems, Operations Research Perspectives 6 (2019) 100128.
81. M. Merschformann, L. Xie, D. Erdmann, Multi-agent path finding with kinematic constraints for robotic mobile fulfillment systems, arXiv preprint arXiv:1706.09347 (2017).
82. R. Keith, H. M. La, Review of autonomous mobile robots for the warehouse environment, arXiv preprint arXiv:2406.08333 (2024).
83. R. Stern, N. R. Sturtevant, A. Felner, S. Koenig, H. Ma, T. T. Walker, J. Li, D. Atzmon, L. Cohen, T. K. S. Kumar, R. Barták, E. Boyarski, Multi-agent pathfinding: Definitions, variants, and benchmarks, in: Proceedings of the International Symposium on Combinatorial Search (SoCS), 2019, pp. 151–158.
84. G. Sharon, R. Stern, A. Felner, N. R. Sturtevant, Conflict-based search for optimal multi-agent pathfinding, Artificial Intelligence 219 (2015) 40–66.
85. H. Ma, D. Harabor, P. J. Stuckey, J. Li, S. Koenig, Searching with consistent prioritization for multi-agent path finding, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019.

86. K. Okumura, M. Machida, X. Défago, Y. Tamura, Priority inheritance with backtracking for iterative multi-agent path finding, Artificial Intelligence 310 (2022) 103752.
87. J. Li, Z. Chen, D. Harabor, P. J. Stuckey, S. Koenig, MAPF-LNS2: Fast repairing for multi-agent path finding via large neighborhood search, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022.
88. S. Varambally, J. Li, S. Koenig, Which MAPF model works best for automated warehousing?, in: Proceedings of the International Symposium on Combinatorial Search (SoCS), 2022.
89. International Organization for Standardization, ISO 3691-4:2023 industrial trucks — safety requirements and verification — part 4: Driverless industrial trucks and their systems, ISO, Geneva (2023).
90. J. G. Wardrop, Some theoretical aspects of road traffic research, Proceedings of the Institution of Civil Engineers 1 (3) (1952) 325–362.
91. C. Fisk, Some developments in equilibrium traffic assignment, Transportation Research Part B: Methodological 14 (3) (1980) 243–255.
92. Y. Sheffi, Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods, Prentice-Hall, Englewood Cliffs, NJ, 1985.
93. T. Roughgarden, É. Tardos, How bad is selfish routing?, Journal of the ACM 49 (2) (2002) 236–259.
94. M. J. Smith, The stability of a dynamic model of traffic assignment—an application of a method of Lyapunov, Transportation Science 18 (3) (1984) 245–252.
95. W. H. Sandholm, Population Games and Evolutionary Dynamics, MIT Press, Cambridge, MA, 2010.
96. S. Fischer, B. Vöcking, Adaptive routing with stale information, in: Proceedings of the 24th ACM Symposium on Principles of Distributed Computing (PODC), 2005.
97. H. X. Liu, X. He, B. He, Method of successive weighted averages (MSWA) and self-regulated averaging schemes for solving stochastic user equilibrium problem, Networks and Spatial Economics 9 (4) (2009) 485–503.
98. F. P. Kelly, A. K. Maulloo, D. K. H. Tan, Rate control for communication networks: Shadow prices, proportional fairness and stability, Journal of the Operational Research Society 49 (3) (1998) 237–252.
99. S. H. Low, D. E. Lapsley, Optimization flow control—I: Basic algorithm and convergence, IEEE/ACM Transactions on Networking 7 (6) (1999) 861–874.
100. R. Srikant, The Mathematics of Internet Congestion Control, Birkhäuser, Boston, 2004.
101. B. D. Greenshields, A study of traffic capacity, Highway Research Board Proceedings 14 (1935) 448–477.
102. M. J. Lighthill, G. B. Whitham, On kinematic waves II. A theory of traffic flow on long crowded roads, Proceedings of the Royal Society of London. Series A 229 (1178) (1955) 317–345.
103. C. F. Daganzo, The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory, Transportation Research Part B: Methodological 28 (4) (1994) 269–287.
104. S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, 2004.
105. A. V. Fiacco, Introduction to Sensitivity and Stability Analysis in Nonlinear Programming, Academic Press, New York, 1983.
106. S. M. Robinson, Strongly regular generalized equations, Mathematics of Operations Research 5 (1) (1980) 43–62.
107. J. D. C. Little, A proof for the queuing formula: $L = \lambda W$, Operations Research 9 (3) (1961) 383–387.

108. D. P. Heyman, S. Stidham, The relation between customer and time averages in queues, Operations Research 28 (4) (1980) 983–994.
109. R. A. Horn, C. R. Johnson, Matrix Analysis, 2nd Edition, Cambridge University Press, Cambridge, 2013.
110. H. K. Khalil, Nonlinear Systems, 3rd Edition, Prentice Hall, Upper Saddle River, NJ, 2002.
111. E. N. Brown, R. Barbieri, V. Ventura, R. E. Kass, L. M. Frank, The time-rescaling theorem and its application to neural spike train data analysis, Neural Computation 14 (2) (2002) 325–346.
112. A. Meurer, C. P. Smith, M. Paprocki, et al., SymPy: Symbolic computing in Python, PeerJ Computer Science 3 (2017) e103.
113. D. A. Belsley, E. Kuh, R. E. Welsch, Regression Diagnostics: Identifying Influential Data and Sources of Collinearity, Wiley, New York, 1980.
114. A. E. Hoerl, R. W. Kennard, Ridge regression: Biased estimation for nonorthogonal problems, Technometrics 12 (1) (1970) 55–67.