COOPERATIVE JAMMING SCHEMES FOR SECURE COMMUNICATIONS

Alexander J Smith

Alcuin College

University of York

May 2026

3rd Year Project Final Report for degree of BEng

in Electronic Engineering

Abstract

Cooperative jamming can improve physical-layer security (PLS), but only if the legitimate receiver (Bob) suppresses the friendly interference better than the eavesdropper (Eve) does. Much of the conventional literature assumes this cancellation is perfect, yet real systems always leak some residual, and two things go wrong when that isn't modelled. First, residual leakage at Bob shifts the optimal jammer covariance away from the idealised solution. Second, the leakage cap a nominal design enforces is only as reliable as the channel estimates behind it, and small errors trigger violations. This report addresses both in MATLAB/CVX.

I first reproduce three baselines: Bengtsson-Ottersten semidefinite-programming (SDP) beamforming, the Cumanan 2017 geometric-programming successive convex approximation (GP-SCA) for jammer-power allocation, and a multiple-input multiple-output (MIMO) secrecy-rate SCA. I then propose three new designs and evaluate them on matched Monte Carlo channels: a leakage-aware jammer SDP with a fixed source, its S-procedure robust extension, and a two-direction closed-form optimiser for the fixed-source proxy. Joint-SCA and the public UPCT channel-frequency-response (CFR) proxies sit alongside these as stress tests, not as a stand-in for real radio-frequency (RF) deployment.

Residual Bob leakage changes which jammer is preferred. At 10 dB, nominal leakage-aware jamming roughly doubles the no-jammer secrecy and matches Eve-directed jamming when $\beta = 0$, but at $\beta = 1$ it holds 3.48 bit/s/Hz while Eve-directed jamming drops to 2.66. With norm-bounded channel uncertainty, nominal designs violate the Bob-leakage replay constraint 49.2% of the time. The robust SDP removes these violations entirely, sacrificing a fraction of the average rate for constraint safety. The fixed-source SDP is rank-one, and the closed-form two-direction implementation matches the SDP rate while cutting design time by about 7400×. Evidence here is Monte Carlo plus measured-propagation proxies, not deployed RF hardware.

Acknowledgements

I would like to thank Professor Kanapathippillai Cumanan for supervision, technical guidance, and provision of reading material, as well as Professor Paul Mitchell for his support and advice throughout my degree. Their feedback was instrumental in refining this project from a broad survey to a focused study. I am also grateful to the University and Department of Electronic Engineering for providing the academic environment and technical foundation that supported this work. Finally, I extend my thanks to my family and friends for their patience, encouragement, and understanding during the most demanding phases of the project.

Contents

Chapter 1

Introduction

Wireless security normally relies on cryptographic encryption. An older idea is that the channel itself can do some protection work. Wyner's wiretap-channel result is the starting point: whenever the legitimate receiver (Bob) has a better channel than the eavesdropper (Eve), a positive confidential rate is reachable before any key material is added [1]. Csiszár and Körner generalised this to broadcast channels with confidential messages [2], and the Gaussian wiretap formulation tied the same reasoning to signal-to-noise ratio (SNR) [3].

A perfect model hides important cases. Eve may sit closer to the transmitter (Alice), carry more antennas, or occupy a favourable propagation path. Cooperative jamming responds by transmitting interference that hurts Eve more than Bob. On paper this can work, but trouble starts when the perfect-cancellation assumption is relaxed. A friendly jammer also damages the legitimate link, so a scheme that looks strong with Bob's residual set to zero can lose most of its advantage once modest leakage is added.

After supervisor feedback, the original scope narrowed to three aims: reproduce the convex-optimisation methods behind secure beamforming, including the Cumanan 2017 GP-SCA cooperative-jamming method and planned source/jammer SCA extensions [4]; implement cooperative-jamming baselines under fair, matched simulations; and evaluate nominal, robust and low-complexity leakage-aware designs, with residual Bob interference controlled by β, CSI uncertainty handled by sampled replay and S-procedure linear-matrix-inequality (LMI) constraints, and the schemes benchmarked against a fixed-source perfect-cancellation upper bound.

Component	**Role in this dissertation**	**Status**	**Limitation**
Bengtsson-Ottersten SDP	CVX, signal-to-interference-plus-noise ratio (SINR) and rank validation baseline	Reproduced	Not a secrecy design
Cumanan GP-SCA and MIMO SCA	Literature benchmarks for secrecy optimisation	Reprod./adapt.	Local or scalar-power scope
Leakage-aware jammer SDP	Main project contribution: cap Bob leakage while jamming Eve	Original extension	Fixed source and proxy objective
Robust leakage-aware SDP	Main CSI-uncertainty extension via S-procedure LMIs	Original extension	Norm-bounded model only
Closed-form fixed-source proxy optimiser	Low-complexity payoff from the rank-one result	Derived here	Fixed-source, single-Eve scope
Joint-SCA, branch-and-bound (BnB), runtime and repeatability checks	Supporting comparison and audit evidence	Supporting	Local or scoped certificates
UPCT measured-channel proxy	Realistic-channel stress test for effective links	Supporting	Proxy, not deployed RF hardware

Table 1.1: Contribution map: reproduced baselines validate the pipeline; leakage-aware SDP, robust extension and fixed-source proxy optimiser carry the main contribution.

Chapter 2

Literature Review

2.1 Physical-layer secrecy foundations

For the Gaussian wiretap channel, the secrecy rate is the positive difference between Bob's and Eve's information rates. The expression is convenient, as it treats channel quality as a design variable, but importantly, it provides no recipe for shaping a friendly jammer once Bob is left with residual interference. The information-theoretic background here follows Bloch and Barros [5], and the scalar validation experiment is checked against the Gaussian wiretap expression of Leung-Yan-Cheong and Hellman [3]. Moving to multiple antennas turns the design variable into a covariance, with spatial power free to strengthen Bob or to place artificial noise where Bob is meant to be protected.

The MIMO capacity papers act as ideal-CSI references rather than cooperative-jammer recipes. Khisti and Wornell showed, in the multiple-input single-output multiple-eavesdropper (MISOME) case, that beamforming is capacity-achieving when channels are known [6]; their multiple-input multiple-output multiple-eavesdropper (MIMOME) analysis also relies on fixed, fully known matrices [7]. Oggier and Hassibi closed the arbitrary-antenna MIMO wiretap capacity using a Sato upper bound and a Riccati characterisation [8]. The covariance model used here borrows that machinery but drops the most convenient assumption – a harmless, perfectly known jammer channel.

Artificial noise (AN) contributes the clearest starting point. Goel and Negi place jamming-style noise in receiver null directions [9]; Lin et al. then show that quantised feedback makes this null-space benefit leak away quickly [10]. The β model used later follows the same critical lesson: a design that nulls interference algebraically can still leave Bob with residual jammer power after estimation and receiver processing.

2.2 Cooperative jamming and convex optimisation

Cooperative jamming is essentially artificial noise that has been pushed out of Alice and into a helper node. That adds spatial separation, but the relay and friendly-jammer literature shows the cost: helpers have to be trusted, one has to be picked, more CSI is needed, and much of the conventional literature assumes that Bob can clean their interference off the signal afterwards. Dong et al. study cooperating relays [11], Huang and Swindlehurst extend the model to MIMO relay jamming [12], and Zheng et al. consider relay selection [13]. Surveys by Mukherjee et al. and Jameel et al. show the same assumptions resurfacing across the field [14, 15]. The cancellation assumption is therefore treated here as a key under-examined limitation, motivating the leakage-constrained Bob model used in this report.

Secure-transmission problems become non-convex once Bob and Eve share a rate expression. Convex optimisation still carries much of the practical load: convex reformulations give global solutions, support strong duality, and run on reliable algorithms under standard regularity conditions [16]. Luo and Yu show why SOCP and SDP fit communications and signal processing: beamforming, power allocation, filter design and detection reduce to quadratic objectives, norm bounds, covariance matrices, or SINR constraints, all of which are cone programs or admit tight relaxations [17].

Bengtsson and Ottersten's downlink beamforming SDP relaxation [18] predates the secrecy literature, but its lifted positive-semidefinite (PSD) covariances, SINR constraints and rank interpretation all turn up in later cooperative-jamming work. I used it as my first CVX sanity check: if this simpler lifted SDP and rank recovery failed, later secrecy models would not be trustworthy.

For the secrecy work itself, Li and Ma recast multiple-input single-output (MISO) secrecy-rate problems through SDP, including spherical-CSI robust variants [19]. Cumanan et al. extend the same machinery to MIMO secrecy with a multiple-antenna eavesdropper, using Taylor approximations for the non-convex log-det [20]. Chu et al. add a cooperative jammer through alternating covariance optimisation [21]. I found no directly comparable treatment for the case I needed: robust source-side SDPs do not produce a practical jammer covariance, scalar GP-SCA cannot exploit a full jammer covariance, and alternating SCA stays local. The 2017 GP-SCA jammer-power paper from Cumanan et al. [4] is therefore the methodological anchor; its single-condensation step rests on the arithmetic-geometric-mean (AGM) principle used in

GP power control [22].

2.3 Recent context and project gap

Recent work shows that cooperative-jamming assumptions are still in flux. Jang et al. use distributed cooperative beamforming [23]. Yu et al. drop the perfect-Eve-CSI assumption [24]. Mao et al. share the residual-CJ and imperfect-CSI concern, but they handle it by optimising Bob's diversity-combining receiver with a single-antenna jammer and a SIMO Bob [25]. Li et al. embed the jamming problem in cognitive-radio non-orthogonal-multiple-access (CR-NOMA) interference constraints [26]. These are the natural comparators because they differ on CSI, cancellation model, architecture and solver burden. None is a drop-in answer for a fixed-source, multi-antenna jammer covariance with an explicit Bob-leakage cap and a low-complexity surrogate.

RIS/IRS and UAV work sits outside the scope. Reconfigurable surfaces turn h_b, h_e, g_b, g_e into functions of phase, placement and cascaded channels [27, 28, 29]. UAV papers add mobility, altitude, trajectory and energy as design variables [30, 31]. The measured-propagation data used later is treated only as a channel proxy.

The gap targeted here is narrow by design: combine residual Bob leakage with bounded jammer-channel error and a practical covariance choice, then study one low-complexity approximation in the same simulation harness. Candidate covariances are compared on shared channels, β is swept across its full range, and leakage-limit, runtime and seed-repeatability checks are reported alongside, so the extension is not judged on one favourable sweep.

Source	Model	CSI	Method	Limitation	Role here
Wyner; Csiszár-Körner	Wiretap/broadcast secrecy	Channel model	Capacity theory	Not a practical jammer design	Defines secrecy-rate objective.
Goel-Negi; Lin et al.	AN with multiple antennas	Legitimate-channel CSI; leakage under quantisation	Null-space AN and feedback analysis	Ideal nulling is sensitive to CSI error	Motivates leakage at Bob.
Bengtsson-Ottersten	Downlink beamforming	Known downlink channels	SDP relaxation	Not a secrecy method	Validates CVX/PSD/rank workflow.
Li-Ma; Cumanan et al.	MISO/MIMO secrecy and cooperative jamming	Perfect CSI plus robust variants	SDP, GP-SCA and SCA	Full robust/joint designs are complex	Supplies reproduction and theory base.
Chu et al.	MIMO secrecy with cooperative jammer	Perfect CSI, alternating covariance design	Taylor/SDP-style alternating optimisation	Not jointly convex	Frames the ambitious extension not claimed here.
Yu; Mao; Li; Jang	Recent CJ/AN systems	Imperfect/no CSI; residual interference; architecture-specific constraints	Outage, STC and numerical optimisation	Validation is mostly simulated and assumptions vary	Used as critical comparators for β, CSI and implementation cost.
RIS/IRS and UAV work	Surface-assisted or aerial secure links	Geometry, trajectory and energy become design variables	Alternating optimisation, trajectory/surface design	Outside fixed-node channel model	Defines explicit limitations and further work.

Table 2.1: Critical comparison of the main literature groups and their role in this project.

Chapter 3

System Model and Theory

3.1 Signal model and secrecy rate

The cooperative-jamming setup used throughout the report consists of four nodes: the source Alice, the legitimate receiver Bob, an eavesdropper Eve and a friendly jammer. The source-to-Bob and source-to-Eve channels are written h_b and h_e, while the jammer-to-Bob and jammer-to-Eve channels are g_b and g_e.

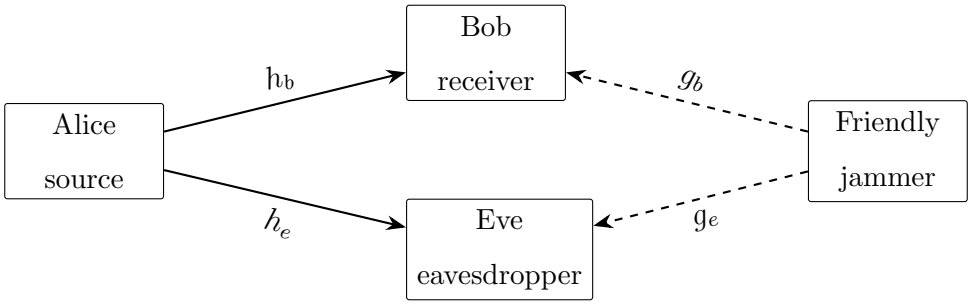

Figure 3.1: System model used in the cooperative-jamming experiments. The jammer should reduce Eve's SINR while limiting residual interference at Bob.

Let s be a scalar source symbol, $x_s = ws$ the source transmit vector, x_j the jammer signal, P_s and Q_j the source power and jammer covariance, and n_b, n_e the receiver noises. The received signals are then

$$y_b = h_b x_s + g_b x_j + n_b, \tag{3.1}$$

$$y_e = h_e x_s + g_e x_j + n_e. \tag{3.2}$$

When Bob cancels most, but not all, of the known jamming signal, the residual jammer power can be written as $\beta g_b Q_j g_b^H$, where $0 \leq \beta \leq 1$. This leads to

$$\mathrm{SINR}_b = \frac{|h_b w|^2}{\sigma^2 + \beta g_b Q_j g_b^H}, \tag{3.3}$$

$$\mathrm{SINR}_e = \frac{|h_e w|^2}{\sigma^2 + g_e Q_j g_e^H}. \tag{3.4}$$

For the cooperative-jamming sweeps, the source beamformer is fixed at maximum-ratio transmission (MRT) toward Bob,

$$w = \sqrt{P_s}\frac{h_b^H}{\|h_b\|}, \tag{3.5}$$

so $\|w\|^2 = P_s$. This convention keeps the source fixed while the jammer covariance varies. Reading the SINR expressions with the desired source term as signal and the noise plus residual jammer as effective noise, the jamming secrecy rate is

$$R_s = \left[\log_2(1 + \mathrm{SINR}_b) - \log_2(1 + \mathrm{SINR}_e)\right]^+ . \tag{3.6}$$

This is the Gaussian wiretap rate specialised to the SINR model. With Gaussian signalling, Bob and Eve see mutual-information terms $\log_2(1 + \mathrm{SINR}_b)$ and $\log_2(1 + \mathrm{SINR}_e)$, so the achievable secrecy rate is the positive part of their difference.

The parameter β does a lot of heavy lifting as an aggregate stand-in for residual cancellation error. Instead of simulating phase noise, CFO and timing, PA distortion, I/Q imbalance, quantisation, channel-estimation and thermal terms individually, the model absorbs their combined power effect into $\beta g_b Q_j g_b^H$ [10, 32, 33]. A sweep over β is therefore a sensitivity test for imperfect cancellation, not a claim that the receiver-calibrated stress check or the UPCT proxy validates any deployed receiver.

The two endpoints of the sweep are intuitive. At $\beta = 0$ Bob removes the jammer perfectly: the ideal upper-bound case. At $\beta = 1$ Bob has no cancellation, and sees the jammer the way Eve does. Holding β as a scalar isolates how much residual Bob leakage the secrecy design can absorb before the preferred jammer flips.

For MIMO covariance validation, the implemented secrecy-rate function is

$$R_s(Q) = \left[\log_2 \det\left(I + \frac{H_b Q H_b^H}{\sigma^2}\right) - \log_2 \det\left(I + \frac{H_e Q H_e^H}{\sigma^2}\right)\right]^+ , \tag{3.7}$$

with $Q \succeq 0$ the source covariance. The log-determinant is computed by a small helper that replaces Q with its Hermitian part $\frac{1}{2}(Q + Q^H)$ before taking the determinant, guarding against round-off skew on poorly conditioned channels. The helper was added after the first robust sweep returned a handful of complex-valued logs.

3.2 Simulation parameters

Every figure's headline metric is the mean secrecy rate over Monte Carlo channel realisations, shown with a 95% descriptive band of $\pm 1.96 s/\sqrt{n}$. The bands describe the simulation ensemble, not asymptotic statistics. Solver diagnostics are separate: a high secrecy rate is discarded whenever the SDP failed, exceeded its power budget, or returned a non-PSD covariance.

Neither β nor the leakage-limit values are pinned to any specific device; they are chosen as design stresses. Three pieces of context motivate the chosen range. The successive-interference-cancellation (SIC) receiver literature treats cancellation quality as a first-order assumption [33]. Residual-hardware models then map leftover distortion to error-vector magnitude (EVM) [32]. And the January 2025 European Telecommunications Standards Institute (ETSI) New-Radio (NR) base-station requirements give EVM limits ranging from a few percent up to tens of percent in amplitude, depending on modulation [34]. The ETSI release is cited for scale context only; it is neither a one-to-one β/EVM calibration nor the latest release. This is why the β sweep runs from 0 to 1 with a conservative default of $\beta = 0.1$. Choosing $L = 0.25\sigma^2$ caps residual jammer interference at $-6\,\mathrm{dB}$ below the noise floor, so the cap (when active) inflates Bob's effective noise denominator by at most a quarter.

Parameter	**Value**	**Purpose**
Source antennas, jammer antennas	$N_s = 3$, $N_j = 3$	Beamformer and jammer covariance dimensions
Default source and jammer powers	$P_s = 10$, $P_j = 5$	Default values; varied only in the SNR and jammer-power sweeps
Noise variance	$\sigma^2 = 1$	SINR and leakage-cap reference
Default residual leakage	$\beta = 0.1$	Partial jammer cancellation at Bob
Path-loss exponent	2.4	Indoor / local-area channel loss
Leakage-limit factor	0.25	Default $L = 0.25\sigma^2$ when $\beta > 0$
Monte Carlo: nominal/robust sweeps	100 realisations, 20 uncertainty samples per channel	β, SNR, Eve-position and leakage-limit sweeps
Monte Carlo: proxy optimiser	500 realisations	Two-direction closed-form benchmark
Monte Carlo: channel stress	150 realisations	Alternate-channel and geometry stress
Monte Carlo: source-optimised	50 realisations	Source/jammer joint design
Monte Carlo: Joint-SCA full-cov	100 channels, with a 100-channel single-Eve BnB audit	Full-covariance Joint-SCA benchmark
Monte Carlo: runtime trials	50 per antenna size, up to $N_j = 32$	Scaling and design-time characterisation
UPCT proxy scenarios	12 four-node measured-propagation scenarios	Measured-CFR cross-check
SNR and jammer-power grids	-5 to 15 dB in 5 dB steps; -10 to 15 dB in 5 dB steps	Source and jammer power sensitivity
β grid	0, 0.02, 0.05, 0.1, 0.2, 0.4, 0.7, 1.0	Residual-interference sensitivity
Eve x-positions	linspace$(0.35, 1.4, 7)$	Geometry sensitivity
CSI perturbation grid	0, 0.03, 0.06, 0.1, 0.15, 0.25, 0.35	Sampled and robust jammer-channel mismatch
Alternate channel checks	Rayleigh, Rician $K = 6$, correlated Rayleigh $\rho_c = 0.6$; multiple Eve positions and path-loss exponents	Simulated stress test alongside the measured-CFR proxy; not a deployed-RF claim
Frame-level proxy-optimiser demo	500 frames, 1 ms frame budget, 50 ms coherence-time assumption	Update-timing replay (Gauss-Markov); simulation only
Software	MATLAB R2026a, CVX 2.2; SDPT3 4.0, SeDuMi 1.02 and MOSEK 11.1.11, with the active solver logged per run	Reproducibility and solver disclosure

Table 3.1: Main simulation parameters. Defaults are stored in the shared MATLAB parameter helper.

Chapter 4

Optimisation Methods

4.1 Bengtsson-Ottersten SDP

Reproduced validation baseline. The Bengtsson-Ottersten reproduction follows the covariance form of the spatial-scattering example in Section IV of the original paper. Lifting one transmit covariance $W_k \succeq 0$ per user produces the SDP

$$\begin{aligned} \min_{\{W_k \succeq 0\}} \quad & \sum_k \mathrm{Tr}(W_k) \\ \text{s.t.} \quad & \mathrm{Tr}(R_i W_i) \geq \gamma_i \left(\sum_{k \neq i} \mathrm{Tr}(R_{i,k} W_k) + \sigma_i^2 \right), \quad \forall i. \end{aligned} \tag{4.1}$$

When W_k returns as rank-one, it factors into a beamforming vector. Higher-rank outputs are kept as covariance designs rather than collapsed onto one fixed beam. Each run logs solver status, objective, runtime, achieved SINR, PSD checks and rank. The reproduction checks modelling and solver behaviour, not secrecy.

4.2 Cumanan et al.-inspired MIMO SCA

Adapted comparison benchmark. The MIMO secrecy-rate problem is non-convex because Eve's concave log-det term enters with a minus sign, producing a difference-of-concave structure that no convex solver will accept directly. Writing

$$f_e(Q) = \log_2 \det(\sigma^2 I + H_e Q H_e^H),$$

the gradient at iteration n, which seeds the affine upper bound used by SCA, is

$$\nabla f_e(Q^{(n)}) = \frac{1}{\ln 2} H_e^H \left(\sigma^2 I + H_e Q^{(n)} H_e^H \right)^{-1} H_e.$$

Linearising at $Q^{(n)}$ gives

$$f_e(Q) \leq f_e(Q^{(n)}) + \mathrm{Re}\left\{ \mathrm{Tr}\left(\nabla f_e(Q^{(n)})(Q - Q^{(n)}) \right) \right\},$$

and substituting this affine majorant leaves a convex SCA subproblem: maximise a concave log-det minus an affine term, subject to $Q \succeq 0$ and $\mathrm{Tr}(Q) \leq P_{\max}$. Standard SCA and block-successive-upper-bound-minimisation (BSUM) convergence results require three conditions: local tightness, gradient consistency, and replay against the true objective [35]. The validation run shows the evaluated secrecy rate increasing monotonically, but I do not claim global optimality from that alone.

4.3 Cumanan 2017 GP-SCA reproduction

Reproduced literature benchmark. Cumanan et al.'s 2017 cooperative-jamming model is single-antenna at every node: one source, one Bob, multiple Eves and multiple cooperative jammers [4]. With the source power fixed at P_s, jammer powers stacked as $p = [P_1, \ldots, P_N]^T$, and effective noise terms

$$u_b(p) = \sum_i P_i |g_b^{(i)}|^2 + \sigma_b^2, \qquad u_{e,m}(p) = \sum_i P_i |g_{e,m}^{(i)}|^2 + \sigma_{e,m}^2,$$

the positive-secrecy branch reduces to minimising the worst Eve ratio across m,

$$\Gamma_m(p) = \frac{1 + \gamma_{e,m}(p)}{1 + \gamma_b(p)} = \frac{\Phi_m(p)}{\Psi_m(p)} \leq \tau. \tag{4.2}$$

Both Φ_m and Ψ_m are posynomials, so their ratio is not a direct GP constraint. At iterate $\tilde{p}$, the denominator gets swapped for the AGM monomial lower bound

$$\widehat{\Psi}_m(p; \tilde{p}) = \prod_k \left(\frac{w_{m,k}(p)}{\alpha_{m,k}} \right)^{\alpha_{m,k}}, \qquad \alpha_{m,k} = \frac{w_{m,k}(\tilde{p})}{\Psi_m(\tilde{p})},$$

where the $w_{m,k}$ are the monomial terms of Ψ_m. The GP-SCA subproblem then reads

$$\begin{aligned} \min_{p,\tau} \quad & \tau \\ \text{s.t.} \quad & \Phi_m(p) / \widehat{\Psi}_m(p; \tilde{p}) \leq \tau, \quad m = 1, \ldots, M, \\ & \epsilon_{\mathrm{GP}} \leq P_i \leq \bar{P}_i, \quad i = 1, \ldots, N. \end{aligned} \tag{4.3}$$

Table 4.1: Pseudo-code for the reproduced Cumanan 2017 GP-SCA jammer-power loop.

1	Start from feasible jammer powers $p^{(0)}$ and set iteration $r = 0$.
2	Build the AGM monomial lower bound $\widehat{\Psi}_m(p; p^{(r)})$ for each Eve.
3	Solve the convex GP subproblem for $p^{(r+1)}$ and $\tau^{(r+1)}$ under the jammer power bounds.
4	Stop when τ changes by less than tolerance; otherwise increment r and repeat.
5	Replay the final powers in the exact secrecy-rate expression.

A dedicated Cumanan-2017 reproduction routine wraps the loop; the log-domain GP update lives in the shared multi-jammer GP-SCA routine. On the deterministic two-Eve, three-jammer instance, τ falls from 0.1357 to 0.1298 over four solved subproblems, and the final power vector replays to 2.95 bit/s/Hz. The reproduction targets the algorithm, not a digitised figure replay.

4.4 Joint-SCA multi-antenna source extension

Adapted local benchmark. Optimising source and jammer covariances together is non-convex because they help Bob and hurt Eve through coupled, opposing terms. SCA breaks the problem into convex local subproblems and replays the exact secrecy rate after every step, giving an auditable local-improvement path rather than a global certificate.

The first Joint-SCA variant lifts the model to a multi-antenna source while keeping Bob and the Eves single-antenna, with the source covariance constrained as $Q_s \succeq 0$, $\mathrm{Tr}(Q_s) \le P_s$. Define

$$f_1(Q_s, p) = \log_2\left(u_b(p) + h_b Q_s h_b^H\right), \qquad f_2(p) = \log_2 u_b(p),$$
$$g_{1,m}(Q_s, p) = \log_2\left(u_{e,m}(p) + h_{e,m} Q_s h_{e,m}^H\right), \quad g_{2,m}(p) = \log_2 u_{e,m}(p).$$

The unclipped secrecy-rate objective is

$$R_s(Q_s, p) = f_1(Q_s, p) - f_2(p) - \max_m \{g_{1,m}(Q_s, p) - g_{2,m}(p)\}. \tag{4.4}$$

This diagonal-jammer variant optimises the full source covariance Q_s but keeps the jammer as a scalar power vector p. The default warm-start uses an MRT source covariance and half of each jammer cap, and each iteration logs the unclipped worst-Eve objective alongside the positive-part secrecy rate.

Each iteration upper-bounds the two concave functions that appear with a minus in front, evaluated at the current expansion point $(\tilde{Q}_s, \tilde{p})$:

$$\widetilde{f}_2(p;\tilde{p}) = \log_2 \tilde{u}_b + \frac{1}{\ln 2\, \tilde{u}_b} \sum_i |g_b^{(i)}|^2 (P_i - \tilde{P}_i),$$

$$\widetilde{g}_{1,m}(Q_s, p; \tilde{Q}_s, \tilde{p}) = \log_2 \tilde{v}_m + \frac{h_{e,m}(Q_s - \tilde{Q}_s)h_{e,m}^H + \sum_i |g_{e,m}^{(i)}|^2 (P_i - \tilde{P}_i)}{\ln 2\, \tilde{v}_m},$$

where $\tilde{u}_b = u_b(\tilde{p})$ and $\tilde{v}_m = u_{e,m}(\tilde{p}) + h_{e,m}\tilde{Q}_s h_{e,m}^H$. Introducing an epigraph variable t for the worst-Eve term, each Joint-SCA subproblem becomes

$$\begin{aligned} \max_{Q_s,p,t} \quad & f_1(Q_s,p) - \widetilde{f}_2(p;\tilde{p}) - t \\ \text{s.t.} \quad & \widetilde{g}_{1,m}(Q_s,p;\tilde{Q}_s,\tilde{p}) - g_{2,m}(p) \le t, \quad m = 1,\dots,M, \\ & Q_s \succeq 0, \quad \mathrm{Tr}(Q_s) \le P_s, \quad 0 \le P_i \le \bar{P}_i. \end{aligned} \tag{4.5}$$

This is convex: f_1 and $g_{2,m}$ are logarithms of positive affine functions, and the Taylor terms are affine. After each step the solver replays the exact secrecy rate, exposing any surrogate-only improvement.

The full-covariance variant goes one step further, promoting p to a matrix $Q_j \succeq 0$ with $\mathrm{Tr}(Q_j) \le P_j$ and adding an optional leakage cap $\beta g_b Q_j g_b^H \le L$. Its exact replay objective is

$$R_s(Q_s,Q_j) = \left[\log_2\left(1 + \frac{h_b Q_s h_b^H}{\sigma^2 + \beta g_b Q_j g_b^H}\right) - \max_m \log_2\left(1 + \frac{h_{e,m} Q_s h_{e,m}^H}{\sigma^2 + g_{e,m} Q_j g_{e,m}^H}\right)\right]^+. \tag{4.6}$$

Each subproblem linearises $-\log(\sigma^2 + \beta g_b Q_j g_b^H)$ to lower-bound Bob's rate, and linearises the concave total-signal log to upper-bound each Eve rate while leaving the convex noise-plus-jammer term untouched. The routine records Q_s, Q_j, exact-rate histories, CVX statuses, PSD and rank metrics, power use and residual Bob leakage.

For the single-Eve case, a BnB audit was added so that SCA itself never has to stand in as proof. The audit branches over $y = h_e Q_s h_e^H$ and $u = \beta g_b Q_j g_b^H$; inside each box, two independent SDPs upper-bound Bob source power and Eve jammer power; the covariances they return are witnesses rather than a feasible joint design. Monotonicity of the scalar secrecy expression yields a deterministic box upper bound, while exact replay of SCA and random-start covariances supplies a feasible lower bound. The certificate is numerical, solver-status-limited and scoped to one Eve; Appendix B states the proposition.

4.5 Leakage-aware jammer SDP

Main optimisation contribution. The leakage-aware extension keeps the source beamformer fixed and designs Q_j on its own:

$$\begin{aligned} \max_{Q_j,t} \quad & t \\ \text{s.t.} \quad & g_e Q_j g_e^H \geq t, \\ & \beta g_b Q_j g_b^H \leq L, \\ & \mathrm{Tr}(Q_j) \leq P_j, \\ & Q_j \succeq 0, \quad t \geq 0. \end{aligned} \tag{4.7}$$

The default implementation uses

$$L = \begin{cases} \infty, & \beta = 0, \\ 0.25\sigma^2, & \beta > 0. \end{cases}$$

With $\beta > 0$ this caps Bob's residual jammer interference at a quarter of thermal-noise power; at $\beta = 0$ the cap is switched off, since the model in that case already assumes perfect cancellation.

The problem is an SDP because $g_e Q_j g_e^H = \mathrm{Tr}(g_e^H g_e Q_j)$ and $g_b Q_j g_b^H = \mathrm{Tr}(g_b^H g_b Q_j)$ are affine trace inner products in Q_j. Direct secrecy-rate maximisation over Q_j is avoided because residual Bob leakage and Eve interference create a non-convex logarithmic trade-off.

Fixing w and letting $S_b = |h_b w|^2$, $S_e = |h_e w|^2$, $I_b = \beta g_b Q_j g_b^H$, $I_e = g_e Q_j g_e^H$, any feasible Q_j with $I_b \leq L$ and $I_e \geq t$ certifies

$$R_s(Q_j) \geq \left[\log_2\left(1 + \frac{S_b}{\sigma^2 + L}\right) - \log_2\left(1 + \frac{S_e}{\sigma^2 + t}\right) \right]^+ . \tag{4.8}$$

Three cases matter. If $\beta = 0$, Q_j leaves Bob's term alone, so all useful jammer power can be aimed at Eve. If $\beta > 0$ but the cap has slack, the proxy is still a good guide with Bob leakage replayed. With an active cap, the SDP is conservative: it maximises Eve interference inside a leakage budget and can leave rate on the table relative to a non-convex design willing to leak more into Bob. I therefore replay every covariance in the exact secrecy-rate expression.

Proposition 1 (fixed-source leakage-aware jammer). For the fixed-MRT-source, single-Eve nominal problem above, with $Q_j \succeq 0$, $\mathrm{Tr}(Q_j) \leq P_j$ and $\beta g_b Q_j g_b^H \leq L$, an optimal rank-one covariance $Q_j^\star = x^\star x^{\star H}$ exists. The optimum lies in the span of the Bob-null and Bob-parallel

jammer directions. For $\beta > 0$, the leakage cap is inactive exactly when

$$\beta P_j \frac{|g_b g_e^H|^2}{\|g_e\|^2} \le L. \tag{4.9}$$

Then $Q_j^\star = P_j g_e^H g_e / \|g_e\|^2$; otherwise the cap-active beam is the $x^\star$ given below.

Proof sketch. Huang and Palomar's rank-reduction bound gives rank ≤ 1 for the two scalar quadratic constraints [36]. To expose the beam, let

$$u_b = \frac{g_b^H}{\|g_b\|}, \quad e_\perp = (I - u_b u_b^H) g_e^H, \quad r = \|e_\perp\|, \quad c = |g_e u_b|.$$

If $r > 0$, set $u_0 = e_\perp / r$. Power outside $\mathrm{span}\{u_0, u_b\}$ neither helps Eve nor relaxes Bob leakage, so it can be discarded. For any remaining covariance, define Bob-parallel power $z = u_b^H Q_j u_b$ and Bob-null power $y = u_0^H Q_j u_0$. Cauchy-Schwarz on $Q_j \succeq 0$ gives

$$g_e Q_j g_e^H \le (r\sqrt{y} + c\sqrt{z})^2,$$

with equality when $Q_j = x x^H$. A rank-one beam is therefore optimal.

Let $z_L = L/(\beta \|g_b\|^2)$ for $\beta > 0$. The unconstrained Eve-directed beam uses Bob-parallel power

$$z_E = P_j \frac{c^2}{r^2 + c^2}.$$

The displayed cap test is $z_E \le z_L$: Eve-directed jamming is feasible only when g_e is sufficiently orthogonal to g_b after scaling by $L/(\beta P_j)$. In the cap-active case the optimum spends the full cap budget on Bob-parallel power, placing the remainder on the Bob-null direction, so that $z^\star = z_L$, $y^\star = P_j - z_L$, and

$$x^\star = \sqrt{P_j - z_L}\, u_0 + \sqrt{z_L}\, e^{-j \arg(g_e u_b)} u_b, \qquad Q_j^\star = x^\star x^{\star H}. \tag{4.10}$$

The collinear edge case $r = 0$ collapses to the Bob-parallel beam with power $\min(P_j, z_L)$. The proposition is scoped only to the nominal fixed-source single-Eve proxy; it does not cover robust SDP, Joint-SCA, multi-Eve or hardware cases.

4.6 Low-complexity proxy optimiser and fixed-source upper bound

Low-complexity payoff. The SDP is useful offline but unattractive when a system must track channel coherence and solve repeatedly. The low-complexity routine is the closed-form beam derived above: a one-parameter angle search between the Bob-null direction u_0 and the Eve direction the earlier sweep used, with no online cone solve. When the Bob-null projection is numerically zero, the routine falls back to the Bob-parallel edge case.

Cone-solver work differs sharply. The nominal SDP carries an $N_j \times N_j$ Hermitian PSD covariance with $O(N_j^2)$ decisions and dense cone algebra. The robust S-procedure adds two $(N_j + 1)$ LMIs and two scalar multipliers. The closed-form routine needs a projection, a cap test and a rank-one outer product. The SDP stays as the offline benchmark; the closed form is the practical candidate.

A fixed source perfect cancellation upper bound retains the same MRT source beamformer and jammer budget, sets $\beta = 0$, and dumps all jammer power along g_e^H. With w fixed and $\beta = 0$, Bob's rate does not depend on Q_j and Eve's rate is monotonic in $g_e Q_j g_e^H$. Under $Q_j \succeq 0$ and $\mathrm{Tr}(Q_j) \le P_j$,

$$g_e Q_j g_e^H \le P_j \|g_e\|^2,$$

with equality at rank-one Eve-directed jamming. The resulting curve bounds the fixed-source designs from above; I do not claim it as a formal capacity theorem.

Every jamming baseline uses the same channel realisations and source beamformer: no jammer, isotropic, null-space, Eve-directed, leakage-aware SDP, and the low-complexity closed form. The perfect-cancellation curve sits above them as a non-deployable reference.

4.7 S-procedure robust leakage-aware jammer SDP

Main uncertainty extension. The robust SDP generalises the nominal leakage-aware design under norm-bounded jammer-channel uncertainty, requiring the Bob-leakage and Eve-jamming constraints to hold for every channel inside the error ball. Let the estimates be $\hat{g}_b$ and $\hat{g}_e$, with

$$g_b = \hat{g}_b + \Delta_b, \qquad g_e = \hat{g}_e + \Delta_e, \qquad \|\Delta_b\|_2 \le \rho_b, \quad \|\Delta_e\|_2 \le \rho_e.$$

The robust leakage-aware jammer then solves

$$\begin{aligned} \max_{Q_j,t} \quad & t \\ \text{s.t.} \quad & \min_{\|\Delta_e\|\le\rho_e} (\hat{g}_e+\Delta_e)Q_j(\hat{g}_e+\Delta_e)^H \ge t, \\ & \max_{\|\Delta_b\|\le\rho_b} \beta(\hat{g}_b+\Delta_b)Q_j(\hat{g}_b+\Delta_b)^H \le L, \\ & \mathrm{Tr}(Q_j)\le P_j, \qquad Q_j\succeq 0, \qquad t\ge 0. \end{aligned} \tag{4.11}$$

With column vectors $q_b=\hat{g}_b^H$ and $q_e=\hat{g}_e^H$, the S-procedure converts the Eve and Bob-leakage worst-case constraints into two affine LMIs:

$$\begin{aligned} &\begin{bmatrix} Q_j+\lambda_e I & Q_j q_e \\ q_e^H Q_j & q_e^H Q_j q_e - t - \lambda_e\rho_e^2 \end{bmatrix} \succeq 0, \qquad \lambda_e\ge 0, \\ &\begin{bmatrix} \lambda_b I-\beta Q_j & -\beta Q_j q_b \\ -\beta q_b^H Q_j & L-\beta q_b^H Q_j q_b - \lambda_b\rho_b^2 \end{bmatrix} \succeq 0, \qquad \lambda_b\ge 0. \end{aligned} \tag{4.12}$$

Both LMIs are affine in $Q_j, t, \lambda_b, \lambda_e$, so the robust scheme stays inside the SDP class.

Chapter 5

MATLAB Implementation and Verification

5.1 Code structure

The MATLAB code is built around shared model functions rather than one-off plotting scripts – the first version, with channel generation inlined into every figure script, became impossible to debug. Channel generation, secrecy-rate evaluation, optimisation and plotting now live in separate modules, with every experiment script calling the same helpers. The numerical utilities cover PSD checks, dB conversion and a stable log-det evaluator. Channel helpers handle Rayleigh, geometry-based, reduced measured-channel and four-node proxy cases. Secrecy-rate functions are split by setting (single-input single-output (SISO), MISO, MIMO, diagonal-jammer, full-covariance). The optimisation routines cover SDP, GP-SCA, both Joint-SCA variants, the S-procedure robust SDP and the closed-form jammer. The rest is seeded experiment drivers and class-based MATLAB tests.

Every plotted claim has a raw MAT file, JSON metadata and processed CSV summary behind it. Appendix B records the file trail.

5.2 Environment, tests and solver logging

The environment log captures MATLAB R2026a, CVX 2.2 [37], a Hermitian PSD verification solve, and solver versions SDPT3 4.0, SeDuMi 1.02 and MOSEK 11.1.11. Unit testing uses MATLAB's `matlab.unittest` framework. Tests cover numerical helpers, secrecy-rate cases, channel generation, measured-proxy mapping, jammer construction, summary code, the portable CVX setup, the experiment API, file-output helpers and solver interfaces. The lightweight CI entry point handles environment setup, Code Analyzer, tests with coverage and a fast secrecy-rate check; long Monte Carlo sweeps stay in their own experiment drivers.

CVX solver statuses are never silently accepted. Status, objective, runtime, PSD, rank and leakage values are written into raw bundles and logs. MOSEK replaced SDPT3 partway through, after the first SDPT3 robust grid was too slow. Table 5.1 summarises solver provenance.

Experiment block	**Solver producing report-facing numbers**	**Report role**
Bengtsson-Ottersten and Cumanan reproductions	Retained SDPT3/CVX reproduction logs	Validation figures and algorithm checks.
N=100 nominal cooperative-jamming sweeps	MOSEK 11.1.11	Headline nominal values in Table 6.4.
N=100 robust uncertainty, β, SNR, Eve-position and leakage-limit sweeps	MOSEK 11.1.11	Robust headline comparisons and replay diagnostics.
Proxy-optimiser, upper-bound and source-optimised benchmarks	Closed form for the fixed-source proxy optimiser and upper bound; MOSEK 11.1.11 for comparison SDPs	Table 6.3 and source-optimised benchmark.
Receiver-calibrated residual check	MOSEK 11.1.11 for nominal and robust LA SDPs	Small N=50 EVM/SIC-scale residual-leakage check.
Measured-proxy and Joint-SCA validation	MOSEK 11.1.11 plus public UPCT measured CFR data	Reduced and four-node measured-proxy checks; diagonal and full-covariance Joint-SCA figures.

Table 5.1: Solver provenance for the report-facing experiment blocks.

Claim	**Evidence**	**Limitation**
The numerical pipeline is reproducible.	MATLAB R2026a/CVX 2.2 environment log, passing unit-test suite, raw MAT/JSON bundles and processed CSVs for every figure.	Reproducibility is numerical and platform-dependent at solver tolerance.
Leakage-aware SDP solves are reliable in the tested grids.	3330 nominal leakage-aware statuses, 640 diagnostic robust statuses and 6400 robust N=100 statuses had zero failures.	Feasibility checks do not prove the model is the only valid physical model.
Robust design gives constraint safety.	Robust replay produced zero Bob-leakage violations in the specified uncertainty balls; nominal replay violations reached 49.2%.	The safety claim is only for the chosen norm-bounded uncertainty model.
The proxy optimiser is relevant to deployment.	Rank-one closed form plus 500-channel benchmark: 99%+ SDP-rate recovery in every channel and about 7400× lower mean design time.	Exact for the fixed-source leakage-aware proxy, not for robust or joint global secrecy optimisation.
Channel realism risk is reduced, but not removed.	Rician/correlated/geometric stress tests, UPCT reduced and four-node measured-data proxies, source-optimised benchmark and N=50 receiver-calibrated residual-leakage check.	No dedicated source/jammer hardware campaign with independently measured h_b, h_e, g_b, g_e was available.
Full-covariance Joint-SCA improves the joint benchmark.	At N=100: full-covariance Joint-SCA mean 4.221 bit/s/Hz; all 100 single-Eve channels have audited BnB certificates.	The method is local SCA/DC; the certificate bounds these single-Eve instances only.

Table 5.2: Claim, evidence and limitation summary. Detailed solver diagnostics are moved to Appendix B.

5.3 Reproducibility handling

Each experiment writes MAT, JSON and CSV evidence side by side. The MAT files preserve MATLAB-native structures, the JSON files hold lightweight metadata, and the CSVs store processed summaries that feed report figures. The practical pay-off is that any plotted value can be traced back to the solver output that produced it.

Hidden randomness is kept out by setting deterministic seeds inside each experiment. Re-running the package should regenerate the same CSV summaries and figures, up to small solver or platform-level numerical differences. This matters because several conclusions rest on Monte Carlo rankings, not a single example channel.

Chapter 6

Results and Discussion

6.1 Validation experiments

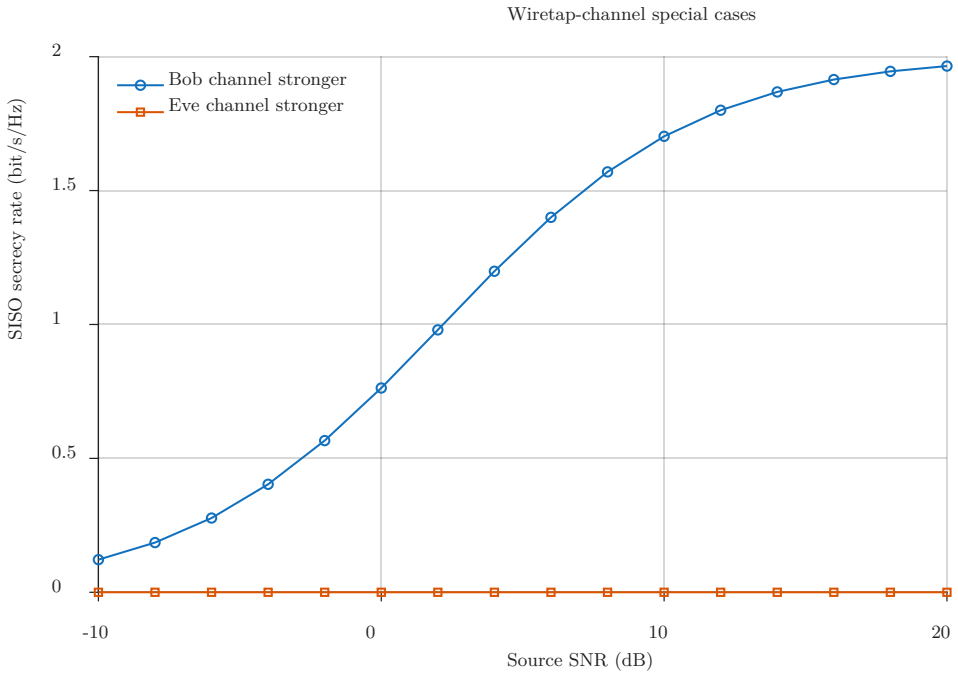

Figure 6.1: Scalar secrecy-rate validation: the favourable case is positive, while the stronger-Eve case clips to zero.

Before the jammer designs could be trusted, the secrecy-rate definition, Hermitian SDP workflow and SCA machinery were checked. Bengtsson-Ottersten is reproduced from the paper's example: the regenerated SDP hits the 10 dB SINR target at every separation, total power falls from 39.59 dB at $\Delta = 5°$ to 6.25 dB at $\Delta = 25°$, and the digitised curve matches within 0.01 dB. Cumanan 2017 GP-SCA drives τ from 0.1357 to 0.1298. The MIMO SCA validation lifts secrecy rate from 3.90 to 4.83 bit/s/Hz, with the rank-2 covariance reported as a covariance solution, not a rank-one beamformer.

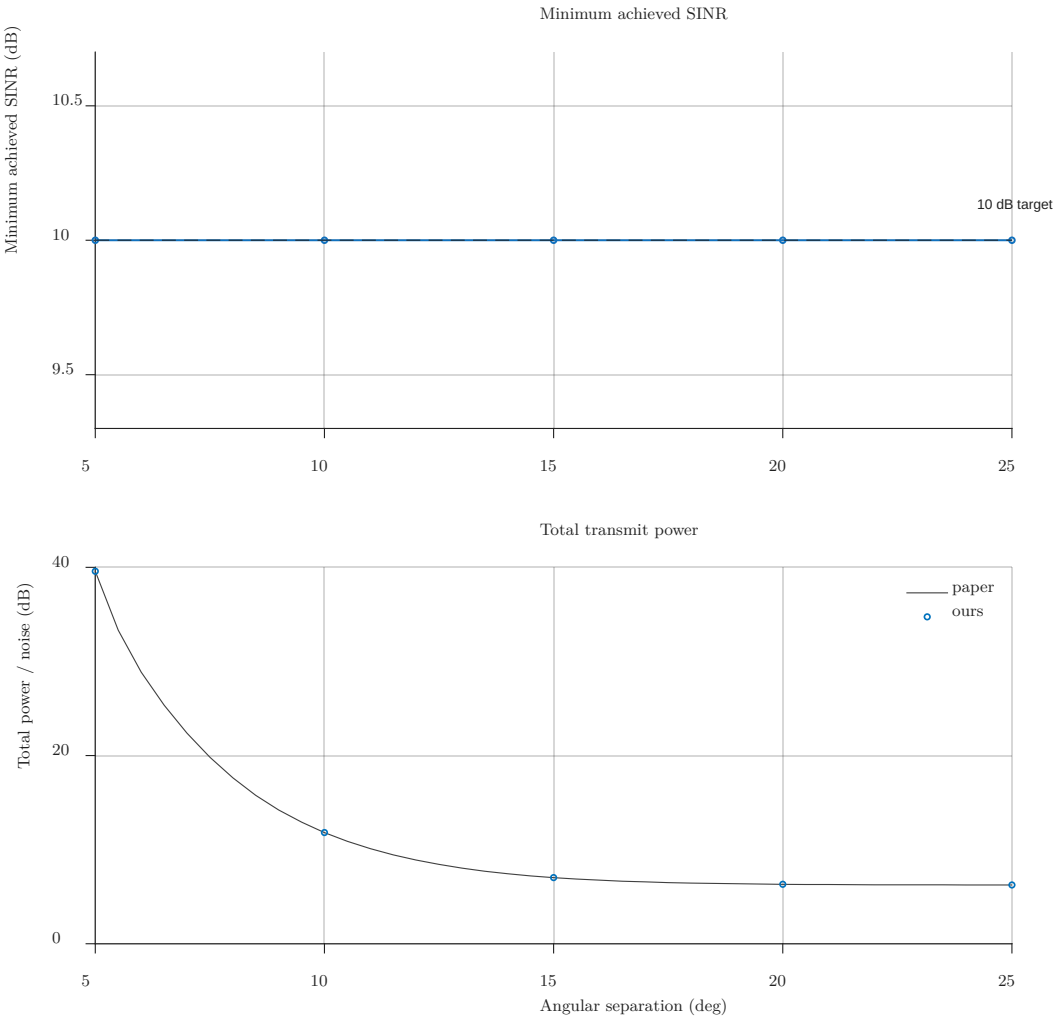

Figure 6.2: Bengtsson-Ottersten SDP reproduction: recomputed CVX points match the digitised Figure 2 “1. Optimal” curve within the paper setup.

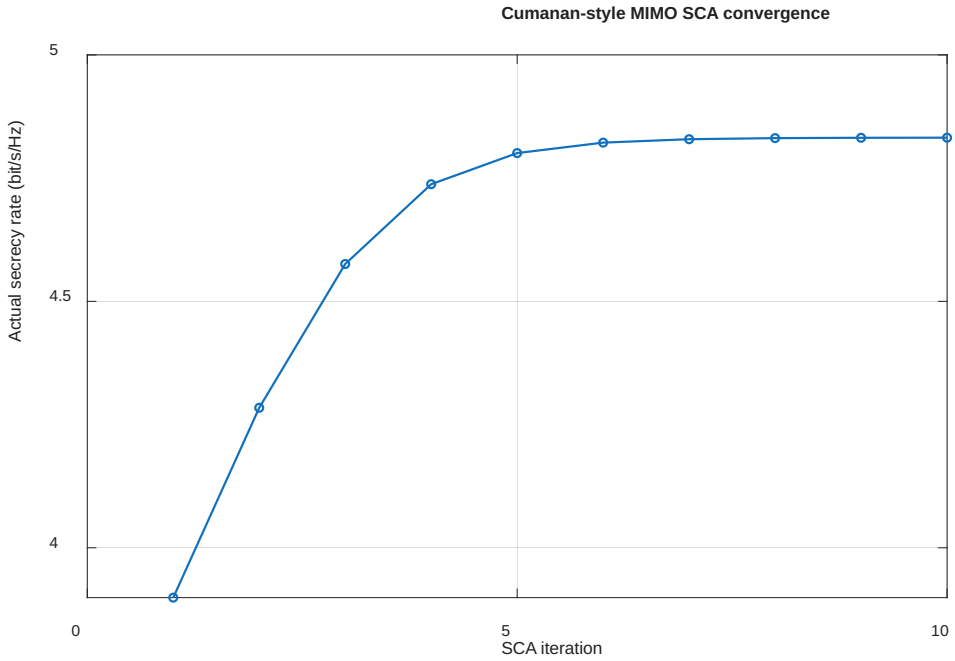

Figure 6.3: Cumanan et al.-inspired SCA validation: evaluated secrecy rate rises from 3.90 to 4.83 bit/s/Hz.

6.2 Cooperative-jamming baselines

At 10 dB the no-jammer secrecy is 1.81 bit/s/Hz; Eve-directed and leakage-aware jamming reach 3.72 and 3.76 bit/s/Hz. The jammer-power sweep in Appendix B shows the recurring trade-off: when the jammer can avoid Bob cleanly, null-space jamming is difficult to beat, and leakage-aware design is not a replacement for structural Bob protection.

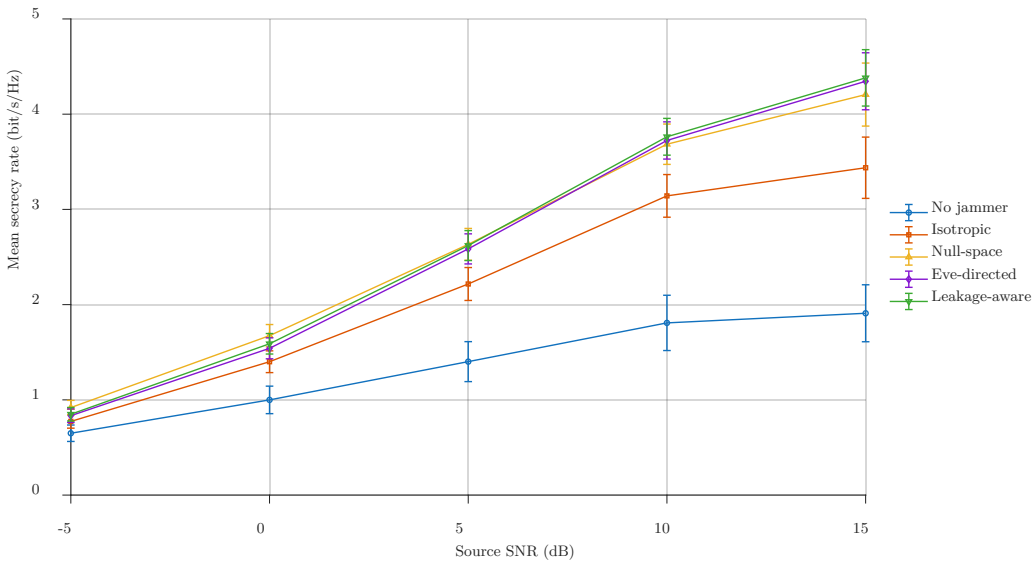

Figure 6.4: Source-SNR sweep over 100 channel realisations. Error bars are 95 percent descriptive intervals; cooperative jamming separates clearly from no-jammer transmission.

6.3 Core result: leakage-aware extension

At $\beta = 0$ the leakage-aware and Eve-directed designs are numerically indistinguishable. By $\beta = 0.2$ leakage-aware reaches 3.68 bit/s/Hz against 3.52 for Eve-directed; at $\beta = 1$ it still holds 3.48 while Eve-directed drops to 2.66. The leakage-aware design interpolates between Eve-directed and Bob-protecting behaviour; it is not a claim to dominate null-space jamming in the full-leakage limit. Appendix B's Eve-position heatmap shows the same geometry.

Regime	**Recommended design**	**Reason**
$\beta \approx 0$, accurate CSI	Eve-directed or leakage-aware	Bob leakage is negligible.
Moderate β	Leakage-aware SDP or closed form	Balances Eve damage and Bob protection.
High β	Null-space or leakage-aware	Bob protection dominates.
Uncertain g_b, g_e	Robust leakage-aware SDP	Removes leakage-cap replay violations.
Fast updates	Closed-form/two-direction design	Matches SDP rate with far lower design time.

Table 6.1: Design rules from leakage and robustness sweeps.

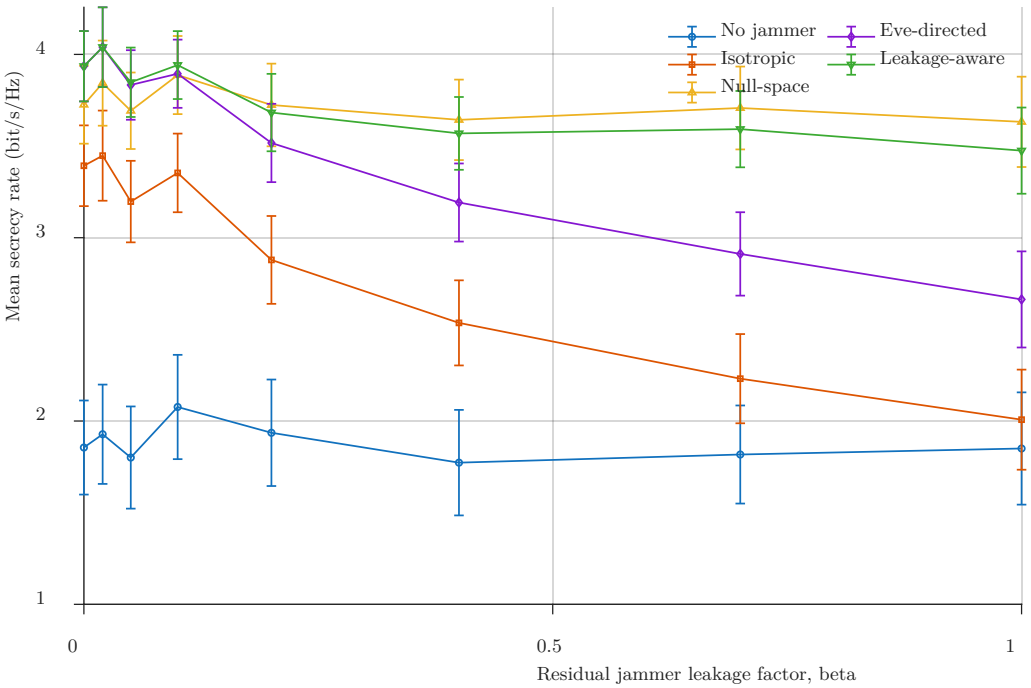

Figure 6.5: Residual-leakage sweep: leakage-aware design increasingly outperforms Eve-directed jamming as nonzero Bob leakage changes the preferred covariance.

6.4 Core result: robust leakage-aware evaluation

The robust SDP is for leakage safety, not higher rates. The N=100 robust scripts replay 20 perturbations per channel across uncertainty, β, SNR, Eve-position and leakage-limit axes.

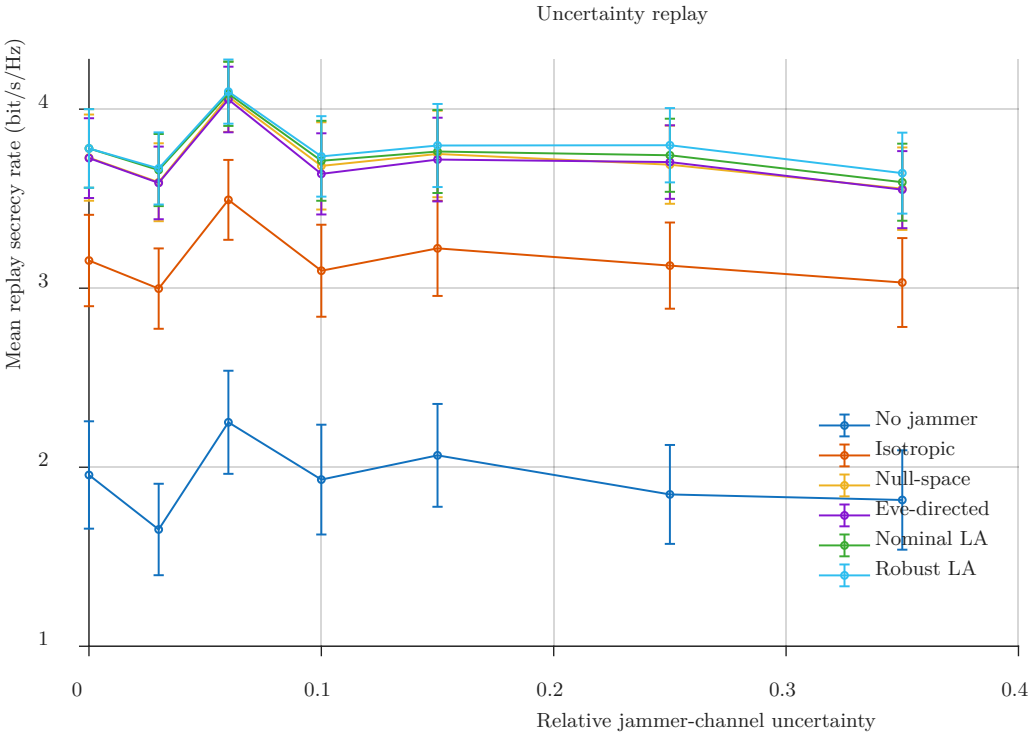

Figure 6.6: Robust uncertainty replay: the S-procedure design removes nominal Bob-leakage violations inside the tested uncertainty balls.

Scenario	**Null-space**	**Nom. LA**	**Robust LA**	**Robust–nominal gain**	**Interpretation**
$\rho = 0.15$	3.75 ± 0.24	3.76 ± 0.23	3.80 ± 0.23	0.034 [0.024, 0.043]	Removes 19.1% nominal leakage-violation rate in replay.
$\rho = 0.25$	3.69 ± 0.22	3.74 ± 0.20	3.80 ± 0.21	0.056 [0.042, 0.070]	Removes 24.5% violation rate; slight rate gain.
$\beta = 0.2, \rho = 0.10$	3.70 ± 0.22	3.69 ± 0.20	3.74 ± 0.21	0.053 [0.043, 0.063]	Close to null-space while enforcing leakage safety.
$\beta = 1, \rho = 0.10$	3.81 ± 0.22	3.68 ± 0.21	3.81 ± 0.22	0.135 [0.119, 0.150]	Improves nominal LA; null-space stays competitive at full leakage.

Table 6.2: Robust headline comparisons: the main payoff is leakage safety, with paired descriptive intervals of 95% for robust minus nominal LA.

Inside the specified uncertainty balls, the robust SDP prevents Bob-leakage replay violations on every robust row; the nominal design violates the cap in 0.8–49.2% of matched scenarios. The robust formulation buys constraint safety, not average-rate gain. At $\rho = 0.25$ it removes a 24.5% violation rate at no rate penalty. The N=100 sweeps return positive robust-minus-nominal paired gains across all SNR points, all Eve positions, and most leakage-limit settings.

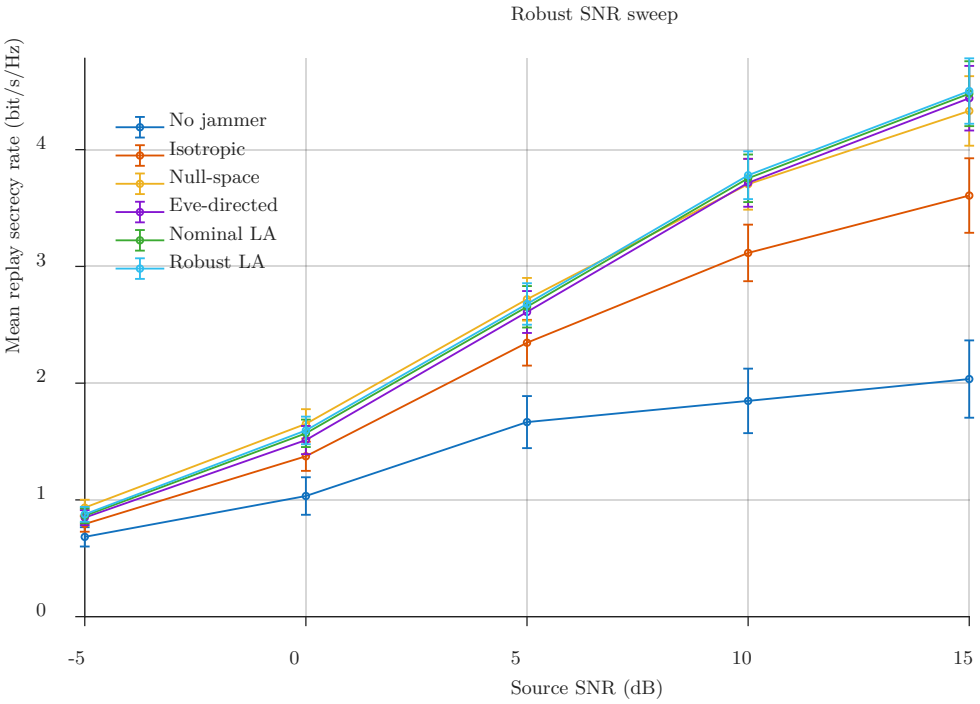

Figure 6.7: Robust SNR sweep over 100 channel realisations per SNR. Robust leakage-aware jamming removes nominal replay violations and gives positive paired gains at every SNR point.

6.5 Core result: proxy optimiser, upper bound and source benchmark

The proxy-optimiser benchmark asks whether the rank-one leakage-aware structure can replace online CVX solves. In this model, yes: the closed-form optimiser and nominal SDP both reach 3.746 bit/s/Hz, the paired SDP-minus-optimiser gap is -1.4×10^{-6} bit/s/Hz, every channel is within 99% of the SDP rate, and mean design time drops from 0.135 s to 1.83×10^{-5} s, about 7400×. Mao et al.'s single-antenna jammer plus SIMO receiver model has no faithful mapping onto this fixed-source multi-antenna covariance design [25].

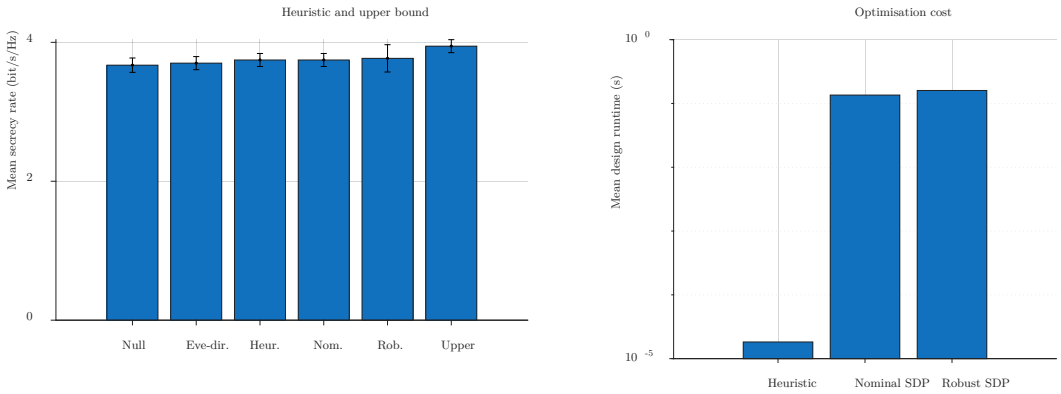

Figure 6.8: Closed-form two-direction proxy optimiser equals the leakage-aware SDP over 500 shared channels, while the $\beta = 0$ upper bound shows the fixed-source gap.

Method	**N**	**Mean rate**	**SDP-method gap**	**Gap to upper**	**Leak. viol.**	**Runtime**
Heuristic	500	3.746 ± 0.094	$-1.4\times10^{-6} \pm 0.7\times10^{-6}$	0.199±0.012	0%	1.83×10^{-5} s
Nominal LA SDP	500	3.746 ± 0.094	0	0.199±0.012	0%	0.135 s
Robust LA SDP	100	3.770 ± 0.197	-0.019 ± 0.007	0.171±0.025	0%	0.160 s
Fixed-source upper	500	3.945 ± 0.094	–	0	reference only	closed form

Table 6.3: Heuristic, SDP and fixed-source upper-bound summary. Units are bit/s/Hz except runtimes; negative SDP–method gaps are tiny paired numerical differences.

A frame-level timing check adds a deployment-style view without claiming hardware real time. Across 500 simulated one-millisecond Gauss-Markov frames, the warmed MATLAB update stayed within budget: median 0.073 ms, 95th percentile 0.089 ms and worst case 0.335 ms. Exact replay averaged 4.10 bit/s/Hz, and peak residual leakage saturated at $0.25\sigma^2$ as expected.

Both the nominal SDP and proxy optimiser land about 0.20 bit/s/Hz below the fixed-source upper bound. A separate 50-channel benchmark fixes the nominal leakage-aware jammer, then replaces MRT with a MISO source-covariance SDP (Appendix B, fig. 20). That lifts the mean from 4.013 to 4.338 bit/s/Hz, putting the cost of fixed MRT at 0.325 ± 0.072 bit/s/Hz.

The diagonal Joint-SCA result stays mainly as a convergence comparator. On the single-channel run, secrecy rate climbs from 3.92 to 4.26 bit/s/Hz across eight subproblems; the diagonal

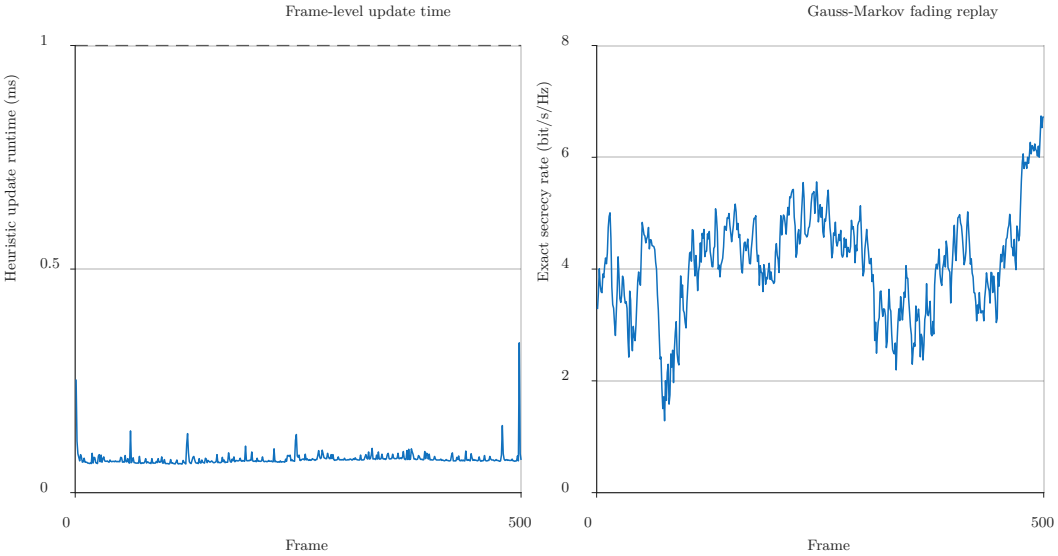

Figure 6.9: Frame-level proxy-optimiser demonstration: warmed MATLAB updates stay inside the simulated 1 ms frame budget, without claiming RF deployment evidence.

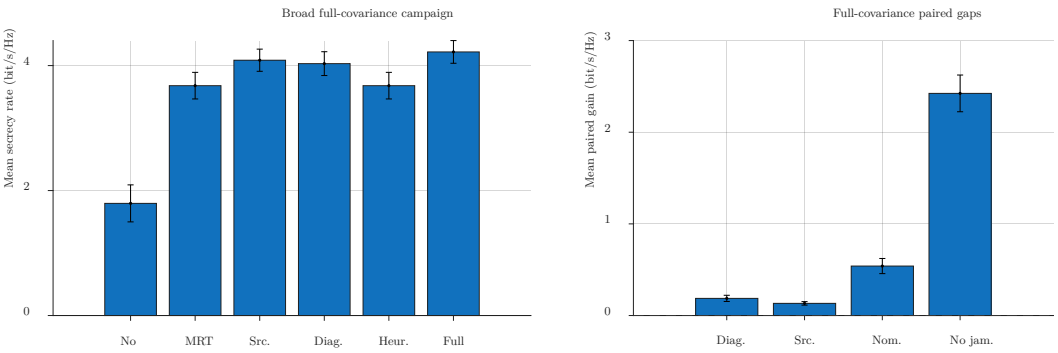

Figure 6.10: Full-covariance Joint-SCA gives positive local gains over diagonal and fixed-jammer benchmarks on 100 seeded channels; intervals are descriptive.

jammer restriction is the binding limit. On the same channel, full-covariance Joint-SCA reaches 5.00 bit/s/Hz against 4.70 diagonal, 4.83 source-optimised fixed nominal, and 4.07 fixed MRT with nominal LA.

Over the 100-channel campaign, full-covariance Joint-SCA averages 4.221 bit/s/Hz against 4.032 diagonal, 4.086 source-optimised fixed-jammer and 3.679 fixed MRT with nominal LA. The paired full-minus-diagonal gain is 0.187 bit/s/Hz with bootstrap interval [0.155, 0.224]; the full-minus-source-optimised gain is 0.133 bit/s/Hz, [0.114, 0.152]. Every history solved cleanly and monotonically, and the certificate run used three SCA/random-start incumbents per channel.

A BnB audit then checked all 100 single-Eve channels from the same seed stream, with row-level provenance in the processed CSV/JSON. All 100 certified cleanly and met the 2% relative criterion; 14 also met the 0.05 bit/s/Hz absolute target. The median relative gap was 1.952%, the worst certified relative gap 1.999%, and the worst certified absolute gap 0.117 bit/s/Hz. This does not make Joint-SCA a global algorithm; it bounds the reported single-Eve Monte Carlo solutions against deterministic global upper bounds.

6.6 Supporting evidence: channel realism

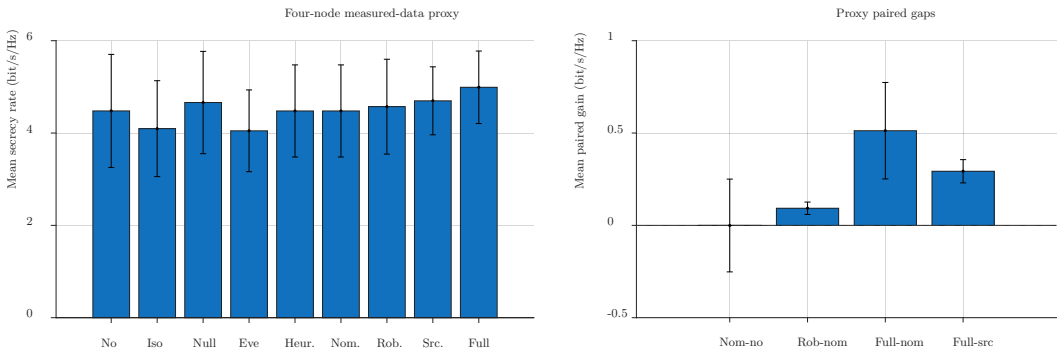

Figure 6.11: UPCT four-node proxy stress test: measured propagation is reused through separate transmitter files and disjoint receiver subarrays, not a deployed RF campaign.

No supervisor-provided four-link measurement campaign was available. A public Tampere/IEEE DataPort cooperative-jamming dataset stores baseband known-interference recordings for cancellation benchmarking, not reusable h_b, h_e, g_b, g_e matrices [38]. The public UPCT dataset is used instead: each corridor MAT file holds a single-transmitter vector-network-analyser (VNA) channel frequency response observed by a 5×5 receiver uniform rectangular array (URA)

over 4096 frequency samples [39]. It is the closest measured-propagation evidence available without hardware. Simulation-only stress tests, the reduced UPCT proxy and receiver-calibrated residual-leakage check sit in Appendix B.

The four-node mapping is auditable: Bob and Eve come from different transmitter-position files, source and jammer use disjoint URA subarrays, and the central frequency window is logged per scenario with a shared scale factor for the four effective links. The construction still leans on reciprocity and array partitioning, so it is not a hardware campaign.

The stress tests agree qualitatively. The leakage-aware SDP and proxy optimiser stay close, while the fixed-source upper bound remains above practical designs. Under Rician fading, nominal LA averages 2.957 bit/s/Hz and the upper bound 3.506; under correlated Rayleigh the figures are 3.278 and 3.510. In the four-node proxy, full covariance Joint-SCA averages 4.991 bit/s/Hz against 4.698 for source-optimised fixed LA, 4.571 for robust LA, 4.479 for nominal LA and 4.479 for no jammer. The older nominal-versus-no-jammer measured comparison remains non-significant, but the full-covariance proxy gain over nominal LA is positive in all 12 scenarios.

A receiver-calibrated residual-leakage stress check runs separately from the measured proxy. EVM/SIC-scale residuals make the cap relevant. With $L = 0.25\sigma^2$ and Rician $K = 6$, $\beta = 0.02$ leaves the cap inactive; at $\beta = 0.10$ Eve-directed jamming violates it in 96% of channels and the LA designs avoid violations. This is a numerical receiver-side stress test, not measured propagation, RF synchronisation or real-time cancellation evidence.

6.7 Supporting audit evidence

The leakage limit, runtime, sampled-CSI, repeatability and solver-status checks back up the three core results above; they are not separate headline claims. A tight leakage cap starves Eve-directed jamming, while a loose cap hurts Bob. Runtime sweeps show robust LMIs getting heavier from $N_j \geq 8$ onward, and the sampled-CSI sweep motivates the robust replay shown above.

Table 6.4 gathers the main numerical decisions with 95% descriptive intervals.

Scenario	**No jammer**	**Null-space**	**Eve-dir.**	**Leakage-aware**	**Interpretation**
10 dB source SNR	1.81 ± 0.29	3.68 ± 0.21	3.72 ± 0.20	3.76 ± 0.19	Cooperative jamming gives a large improvement over no jammer.
$\beta = 0.2$	1.94 ± 0.29	3.72 ± 0.23	3.52 ± 0.21	3.68 ± 0.21	Residual leakage separates LA from an Eve-only objective.
$\beta = 1.0$	1.85 ± 0.31	3.63 ± 0.25	2.66 ± 0.26	3.48 ± 0.24	Full residual leakage penalises Eve-directed jamming; null-space remains strongest.
$\epsilon = 0.15$ CSI perturbation	–	3.52 ± 0.27	–	3.61 ± 0.23	Nominal sampled sensitivity; robust replay is reported separately.

Table 6.4: Headline numerical summary from processed CSV evidence. Units are bit/s/Hz; intervals are descriptive. The CSI row uses only the two nominal designs from that experiment.

Because the schemes share channel realisations, paired differences carry more information than independent error bars. A 1000-resample bootstrap shows clear leakage-aware gains over the no-jammer baseline at 10 dB, and over Eve-directed once residual leakage matters. The limits stay visible: at $\beta = 1$ null-space is stronger here, and the older measured-proxy nominal-versus-no-jammer comparison is non-significant. The full-covariance rows are positive throughout; the broad Monte Carlo campaign beats diagonal Joint-SCA and the source-optimised fixed-jammer benchmark, and the four-node proxy beats nominal LA.

To check that this ranking is not just a single-seed artefact, I rerun the main β cases over three independent ten-trial seed groups.

β	**Seed groups**	**LA mean range**	**LA $\geq$ Eve**	**Top schemes by group**	**Interpretation**
0	3×10	4.33–4.59	3/3	Leakage-aware; Leakage-aware; Leakage-aware	Eve-directed and leakage-aware are numerically tied under ideal cancellation.
0.2	3×10	3.34–4.23	3/3	Leakage-aware; null-space; null-space	Leakage-aware consistently beats Eve-directed, while null-space can still top the group.
1.0	3×10	3.26–4.00	3/3	Null-space; null-space; null-space	Full leakage makes Bob-protecting designs dominate Eve-only jamming.

Table 6.5: Seed-repeatability summary. LA denotes leakage-aware.

6.7.1 Threats to validity

Seven modelling choices set the limits of what these results can fairly say. The channel evidence comes from Rayleigh, Rician, correlated and UPCT proxy channels, not a deployed four-link measurement campaign. The parameter β is an aggregate residual-cancellation term, not a calibrated hardware-impairment model. The 95% intervals and bootstrap ranges are descriptive summaries from finite Monte Carlo samples. The SDP conclusions rely on CVX tolerances, status checks and PSD screening. The UPCT construction uses separate transmitter files, disjoint URA subarrays and a shared scale factor, so it is measured-propagation proxy evidence. The BnB audit certifies only the reported single-Eve Joint-SCA channels, not multi-Eve global optimality. Finally, no claim is made about RF hardware, synchronisation or real-time cancellation validation.

Chapter 7

Project Management, Risk, Sustainability and Ethics

7.1 Actual versus planned work

In practice the project ran iteratively rather than to a fixed schedule. The loop was: pick the next claim, run the smallest script that could test it, save the evidence, and decide whether the result belonged in the main report or an appendix. Relay variants were pushed back, while covariance designs, robustness analysis and reproducibility checks were verified more thoroughly than planned.

The schedule I drew up at the start assumed I would already know which directions were worth chasing before I ran anything. I did not, and that is the main reason the work moved to a more iterative approach.

All six initial-report objectives closed, though several changed shape. The literature review (Obj. 1) narrowed from a broad relay/jamming sweep to PLS foundations, convex optimisation, leakage modelling and recent CSI-aware cooperative jamming. The modelling work (Obj. 2) covered wiretap, MISO/MIMO secrecy and residual-leakage models; DF/AF relay equations were dropped during review. The MATLAB implementation (Obj. 3) moved from many friendly/relay schemes to a controlled set: no-jammer, null-space, Eve-directed, leakage-aware SDP, robust S-procedure SDP and fixed-source proxy optimiser. The comparison work (Obj. 4) followed through SNR, β, Eve-position, leakage-limit, CSI and channel-stress sweeps. The novel modification (Obj. 5) became the leakage-aware jammer family. The evaluation work (Obj. 6) forms Chapter 6: matched Monte Carlo runs, paired comparisons, runtime checks and solver diagnostics.

Work area	**Original plan**	**Actual controlled outcome**
Literature	Broad cooperative-jamming survey	Narrowed to PLS foundations, convex optimisation, residual leakage and recent CSI-aware work.
Convex optimisation	Learn CVX and apply it	Reproduced Bengtsson-Ottersten SDP with SINR/rank checks.
PLS optimisation	Implement supplied secrecy methods	Implemented Cumanan 2017 GP-SCA reproduction and Cumanan et al.-inspired MIMO SCA validation with solver-status history.
Jamming schemes	Many friendly/relay-assisted schemes	Five controlled baselines under shared channels and source beamformer.
Novel modification	Unspecified modification	Nominal, robust and proxy-optimiser leakage-aware jammer designs with beta, leakage limit, CSI, runtime, upper bound gap and repeatability checks.
Robustness	Imperfect CSI and channel assumptions	Sampled CSI, robust S-procedure N=100 sweeps, alternate channels, full covariance Joint-SCA and reduced/four-node UPCT proxies.

Table 7.1: Actual-versus-planned execution after scope control.

7.2 Risk register

Three risks dominated the project: technical validity, reproducibility, and scope control.

Risk	**Potential effect**	**Mitigation and outcome**
CVX or solver setup failure	No reproducible optimisation evidence	Environment script, CVX 2.2 log and Hermitian PSD verification.
Non-convex secrecy formulation	Invalid global-optimum claims	Use SDP only where convex; describe SCA as iterative approximation.
Silent failed solves	False performance claims	Save status, objective, runtime, PSD, rank and leakage checks.
Monte Carlo runtime	Too few samples for broad claims	Split robust axes and rerun SNR, uncertainty, β, Eve-position and leakage-limit sweeps at N=100 with 20 replays.
Overclaiming robustness	Misleading novelty claim	Treat nominal sampled-CSI designs and robust S-procedure designs as distinct approaches, not interchangeable ones. Any claim of constraint safety should be scoped to the specific uncertainty sets and replay diagnostics under test.
Overclaiming channel realism	Simulation mistaken for deployed RF evidence	Search supplied/public data; add UPCT proxies and stress tests while stating that no measured h_b, h_e, g_b, g_e hardware campaign was available.
Scope creep	Shallow report and unverified code	Drop DF/AF relay variants and prioritise verified covariance designs.

Table 7.2: Main project risks and mitigations.

7.3 Debugging and failure mode

The largest technical obstacle was robust-grid runtime, as combining all sweep axes at 100 realisations made SDPT3 impractical; the first long call produced no usable artefacts. A two-realisation trial took several minutes; one MOSEK replay solve took 0.2–0.3 s.

The mitigation was to split the robust run. One script carries uncertainty and β; three more take SNR, Eve position and leakage-limit factors. If one axis fails later, the others still produce raw, processed and figure evidence at N=100.

7.4 Statement of Ethics

The project was entirely simulation based, no human participants were involved, no personal data was collected, and no physical experiments were carried out. The work used MATLAB, published channel models, local random channels and offline public UPCT CFR matrices. The UPCT data is radio propagation measurement; it identifies no person, device or private communication.

Under the University of York's guidance for simulation-only engineering research, this does not require formal ethics approval. There are no live transmissions, field trials, real attacks, user data, confidential operational data or public interactions. Cooperative jamming appears only as an offline PLS benchmark.

7.5 Sustainability

The main sustainability concern is compute energy. Because the project is software-only, design, build and decommissioning impacts are limited to compute energy and preserved code/data governance. SDP and SCA methods give design insight but perform poorly against the closed-form jammer. In the 50-trial benchmark to $N_j = 32$, nominal LA needed 0.121–0.137 s under MOSEK; the robust S-procedure climbed from 0.137 s to 0.448 s. The robust $N_j \geq 8$ tail fit gives exponent 0.75 ($R^2 = 0.86$).

At a 65 W CPU-power assumption (the TDP of my own processor, which all simulations were run on), one nominal LA solve consumes 7.85–8.89 J and one robust solve 8.89–29.13 J, which is negligible offline. A coherence-interval SDP loop would multiply cost and latency by the number of frames in the window.

A real system also has to budget jammer transmit energy. Reading the model powers as normalised watts, the 10 dB row gives no-jammer efficiency $1.81/10 = 0.181$ bit/s/Hz/W and leakage-aware efficiency $3.76/(10 + 5) = 0.251$ bit/s/Hz/W. The incremental secrecy gain is $1.95/5 = 0.391$ bit/s/Hz per jammer watt; this falls to 0.325 at $\beta = 1$ as more jammer energy is spent protecting Bob. The biggest computational gain is the proxy optimiser. It holds the nominal SDP rate while cutting mean design time from 0.135 s to 1.83×10^{-5} s – about 7400×.

Leakage-aware design also discourages jammer energy that would damage Bob: do not transmit

interference you cannot justify. Full-covariance Joint-SCA averaged 0.574 s per local solve, comparable to diagonal Joint-SCA and far above the proxy optimiser. I treat CVX runs as design guidance for lower-complexity approximations, not deployment timings. There is no physical disposal stage, but the code should stay tied to lawful simulation context.

Chapter 8

Conclusion and Further Work

The dissertation reports a MATLAB/CVX reproduction-extension study of cooperative jamming for secure communications. Three reproductions establish the fundamentals in the form of convex beamforming, GP-SCA and MIMO SCA. The extension is the leakage-aware jammer family: a nominal SDP, its robust S-procedure counterpart, and a low-complexity two-direction optimiser for the fixed-source proxy, evaluated on shared channels.

The headline finding is narrow but specific. The residual Bob leakage changes which jammer is preferred. Eve-directed jamming is attractive under ideal cancellation, but non-zero residual leakage forces a trade-off between Eve interference and Bob protection. Null-space jamming stays competitive once Bob protection dominates, and thus the result is a design rule, not a universal winner.

The robust formulation buys leakage safety, not average-rate gain. Replay cap violations vanish inside the stated uncertainty sets. The nominal leakage-aware SDP is rank-one optimal, and the cap-active beam has a closed-form Bob-null/Bob-parallel split. That gives a route away from online CVX solves for the fixed-source proxy, while remaining a heuristic relative to the larger joint non-convex secrecy problem.

A true four-link measurement campaign was outside the scope of this project, and was therefore replaced with the public UPCT measured-CFR dataset, mapped into reduced and four-node proxies with effective h_b, h_e, g_b, g_e links. This provided me with workable data, but the evidence comes through a proxy and not deployed RF.

Joint source/jammer optimisation moved the benchmark, but only locally. The full-covariance Joint-SCA campaign solved all 100 broad test channels with positive paired gains over diagonal Joint-SCA and the source-optimised fixed-jammer design. I still treat it as SCA/DC, not a global solver; the stronger point is that all 100 single-Eve Monte Carlo channels are now bounded against deterministic BnB global upper bounds within the stated relative/absolute certificate policy.

Design takeaways. Fixed-source leakage-aware jamming has a closed-form optimal direction in $(g_b, g_e, \beta, L, P_j)$. Robust design should be sold as leakage safety. Full covariance Joint-SCA improves the local benchmark, and the 100-channel single-Eve BnB audit reduces single-Eve optimality risk without extending to multi-Eve optimality.

Further work, in priority order, would replace the UPCT proxy with a four-radio software-defined-radio (SDR) channel campaign, recast the audit as a primary single-Eve BnB solver with SCA/random-start incumbents rather than a secondary check, and turn the fixed-source proxy optimiser into an adaptive rule stress-tested under correlated channels, uncertainty sets and frame-level evolution. These steps target the remaining gaps in realism, joint optimality and deployable complexity.

References

[1] A. D. Wyner, "The wire-tap channel," *Bell System Technical Journal*, vol. 54, pp. 1355–1387, Oct. 1975. DOI: https://doi.org/10.1002/j.1538-7305.1975.tb02040.x.

[2] I. Csiszár and J. Körner, "Broadcast channels with confidential messages," *IEEE Transactions on Information Theory*, vol. 24, pp. 339–348, May 1978. DOI: https://doi.org/10.1109/TIT.1978.1055892.

[3] S. K. Leung-Yan-Cheong and M. E. Hellman, "The Gaussian wire-tap channel," *IEEE Transactions on Information Theory*, vol. 24, pp. 451–456, July 1978. DOI: https://doi.org/10.1109/TIT.1978.1055917.

[4] K. Cumanan, G. C. Alexandropoulos, Z. Ding, and G. K. Karagiannidis, "Secure communications with cooperative jamming: Optimal power allocation and secrecy outage analysis," *IEEE Transactions on Vehicular Technology*, vol. 66, pp. 7495–7505, Aug. 2017. DOI: https://doi.org/10.1109/TVT.2017.2657629.

[5] M. Bloch and J. Barros, *Physical-Layer Security: From Information Theory to Security Engineering*. Cambridge, UK: Cambridge University Press, 2011. DOI: https://doi.org/10.1017/CBO9780511977985.

[6] A. Khisti and G. W. Wornell, "Secure transmission with multiple antennas I: The MISOME wiretap channel," *IEEE Transactions on Information Theory*, vol. 56, pp. 3088–3104, July 2010. DOI: https://doi.org/10.1109/TIT.2010.2048445.

[7] A. Khisti and G. W. Wornell, "Secure transmission with multiple antennas—part II: The MIMOME wiretap channel," *IEEE Transactions on Information Theory*, vol. 56, pp. 5515–5532, Nov. 2010. DOI: https://doi.org/10.1109/TIT.2010.2068852.

[8] F. Oggier and B. Hassibi, "The secrecy capacity of the MIMO wiretap channel," *IEEE Transactions on Information Theory*, vol. 57, pp. 4961–4972, Aug. 2011. DOI: https://doi.org/10.1109/TIT.2011.2158487.

[9] S. Goel and R. Negi, "Guaranteeing secrecy using artificial noise," *IEEE Transactions on Wireless Communications*, vol. 7, pp. 2180–2189, June 2008. DOI: https://doi.org/10.1109/TWC.2008.060848.

[10] S.-C. Lin, T.-H. Chang, Y.-L. Liang, Y.-W. P. Hong, and C.-Y. Chi, "On the impact of quantized channel feedback in guaranteeing secrecy with artificial noise: The noise leakage problem," *IEEE Transactions on Wireless Communications*, vol. 10, pp. 901–915, Mar. 2011. DOI: https://doi.org/10.1109/TWC.2011.010411.100374.

[11] L. Dong, Z. Han, A. P. Petropulu, and H. V. Poor, "Improving wireless physical layer security via cooperating relays," *IEEE Transactions on Signal Processing*, vol. 58, pp. 1875–1888, Mar. 2010. DOI: https://doi.org/10.1109/TSP.2009.2038412.

[12] J. Huang and A. L. Swindlehurst, "Cooperative jamming for secure communications in MIMO relay networks," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4871–4884, Oct. 2011. DOI: https://doi.org/10.1109/TSP.2011.2161295.

[13] G. Zheng, L.-C. Choo, and K.-K. Wong, "Optimal cooperative jamming to enhance physical layer security using relays," *IEEE Transactions on Signal Processing*, vol. 59, pp. 1317–1322, Mar. 2011. DOI: https://doi.org/10.1109/TSP.2010.2092774.

[14] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst, "Principles of physical layer security in multiuser wireless networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1550–1573, 2014. DOI: https://doi.org/10.1109/SURV.2014.012314.00178.

[15] F. Jameel, S. Wyne, G. Kaddoum, and T. Q. Duong, "A comprehensive survey on cooperative relaying and jamming strategies for physical layer security," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2734–2771, 2019. DOI: https://doi.org/10.1109/COMST.2018.2865607.

[16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004. DOI: https://doi.org/10.1017/CBO9780511804441.

[17] Z.-Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE Journal on Selected Areas in Communications*, vol. 24, pp. 1426–1438, Aug. 2006. DOI: https://doi.org/10.1109/JSAC.2006.879347.

[18] M. Bengtsson and B. Ottersten, "Optimal downlink beamforming using semidefinite optimization," in *Proceedings of the 37th Annual Allerton Conference on Communication, Control, and Computing*, pp. 987–996, 1999.

[19] Q. Li and W.-K. Ma, "Optimal and robust transmit designs for MISO channel secrecy by semidefinite programming," *IEEE Transactions on Signal Processing*, vol. 59, pp. 3799–3812, Aug. 2011. DOI: https://doi.org/10.1109/TSP.2011.2146775.

[20] K. Cumanan, Z. Ding, B. Sharif, G. Y. Tian, and K. K. Leung, "Secrecy rate optimizations for a MIMO secrecy channel with a multiple-antenna eavesdropper," *IEEE Transactions on Vehicular Technology*, vol. 63, pp. 1678–1690, May 2014. DOI: https://doi.org/10.1109/TVT.2013.2285244.

[21] Z. Chu, K. Cumanan, Z. Ding, M. Johnston, and S. Y. Le Goff, "Secrecy rate optimizations for a MIMO secrecy channel with a cooperative jammer," *IEEE Transactions on Vehicular Technology*, vol. 64, pp. 1833–1847, May 2015. DOI: https://doi.org/10.1109/TVT.2014.2336092.

[22] M. Chiang, C. W. Tan, D. P. Palomar, D. O'Neill, and D. Julian, "Power control by geometric programming," *IEEE Transactions on Wireless Communications*, vol. 6, pp. 2640–2651, July 2007. DOI: https://doi.org/10.1109/TWC.2007.05960.

[23] G. Jang, D. Kim, I.-H. Lee, and H. Jung, "Cooperative beamforming with artificial noise injection for physical-layer security," *IEEE Access*, vol. 11, pp. 22553–22573, 2023. DOI: https://doi.org/10.1109/ACCESS.2023.3252503.

[24] K. Yu, J. Yu, Z. Feng, and M. Guo, "Cooperative jamming aided securing wireless communications without CSI of eavesdroppers," *Computer Networks*, vol. 234, 2023. Art. no. 109906. DOI: https://doi.org/10.1016/j.comnet.2023.109906.

[25] Y. Mao, W. Li, W. Wang, H. Zhao, and S. Shao, "Analysis of optimal diversity combining with imperfect channel state information for cooperative jamming-aided Internet of Things security," *Electronics*, vol. 13, no. 20, 2024. Art. no. 4026. DOI: https://doi.org/10.3390/electronics13204026.

[26] M. Li, P. Xue, H. Yuan, and Y. Han, "Physical layer security for CR-NOMA network with cooperative jamming," *Tsinghua Science and Technology*, vol. 30, pp. 708–720, Apr. 2025. DOI: https://doi.org/10.26599/TST.2023.9010128.

[27] J. Youn, W. Son, and B. C. Jung, "Physical-layer security improvement with reconfigurable intelligent surfaces for 6G wireless communication systems," *Sensors*, vol. 21, no. 4, 2021. Art. no. 1439. DOI: https://doi.org/10.3390/s21041439.

[28] J. Zhang, H. Du, Q. Sun, B. Ai, and D. W. K. Ng, "Physical layer security enhancement with reconfigurable intelligent surface-aided networks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3480–3495, 2021. DOI: https://doi.org/10.1109/TIFS.2021.3083409.

[29] S. Li, B. Duo, M. Di Renzo, M. Tao, and X. Yuan, "Robust secure UAV communications with the aid of reconfigurable intelligent surfaces," *IEEE Transactions on Wireless Communications*, vol. 20, pp. 6402–6417, Oct. 2021. DOI: https://doi.org/10.1109/TWC.2021.3073746.

[30] W. Wang, X. Li, M. Zhang, K. Cumanan, D. W. K. Ng, G. Zhang, J. Tang, and O. A. Dobre, "Energy-constrained UAV-assisted secure communications with position optimization and cooperative jamming," *IEEE Transactions on Communications*, vol. 68, pp. 4476–4489, July 2020. DOI: https://doi.org/10.1109/TCOMM.2020.2989462.

[31] W. Wang, X. Li, R. Wang, K. Cumanan, W. Feng, Z. Ding, and O. A. Dobre, "Robust 3D-trajectory and time switching optimization for dual-UAV-enabled secure communications," *IEEE Journal on Selected Areas in Communications*, vol. 39, pp. 3334–3347, Nov. 2021. DOI: https://doi.org/10.1109/JSAC.2021.3088628.

[32] E. Björnson, P. Zetterberg, M. Bengtsson, and B. Ottersten, "Capacity limits and multiplexing gains of MIMO channels with transceiver impairments," *IEEE Communications Letters*, vol. 17, pp. 91–94, Jan. 2013. DOI: https://doi.org/10.1109/LCOMM.2012.112012.122003.

[33] Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan, and V. K. Bhargava, "A survey on non-orthogonal multiple access for 5G networks: Research challenges and future trends," *IEEE Journal on Selected Areas in Communications*, vol. 35, pp. 2181–2195, Oct. 2017. DOI: https://doi.org/10.1109/JSAC.2017.2725519.

[34] ETSI, "5G; NR; base station (BS) radio transmission and reception (3GPP TS 38.104 version 18.8.0 release 18)," Tech. Rep. ETSI TS 138 104 V18.8.0, European Telecommunications Standards Institute, Jan. 2025. Available: https://www.etsi.org/deliver/etsi_ts/138100_138199/138104/18.08.00_60/ts_138104v180800p.pdf.

[35] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM Journal on Optimization*, vol. 23, no. 2, pp. 1126–1153, 2013. DOI: https://doi.org/10.1137/120891009.

[36] Y. Huang and D. P. Palomar, "Rank-constrained separable semidefinite programming with applications to optimal beamforming," *IEEE Transactions on Signal Processing*, vol. 58, pp. 664–678, Feb. 2010. DOI: https://doi.org/10.1109/TSP.2009.2031732.

[37] M. Grant and S. Boyd, "CVX: MATLAB software for disciplined convex programming, version 2.2." https://cvxr.com/cvx, 2020. January 2020, Build 1148.

[38] K. Pärlin, T. Riihonen, M. Turunen, V. Le Nir, and M. Adrat, "Securing the physical layer of IEEE 802.15.4 through cooperative jamming." Dataset, IEEE DataPort, 2023. Baseband known-interference measurement dataset. Dataset DOI: https://doi.org/10.21227/9mty-pf96.

[39] F. Correa Quinchía, M. T. Martínez Inglés, J. M. Molina García-Pardo, and J. Pascual García, "Measurement dataset and MATLAB codes for the analysis of the wireless channel at the FR3 band at indoor and indoor corridor scenarios." Dataset, Universidad Politécnica de Cartagena, Repositorio Digital UPCT, 2026. Dataset DOI: https://doi.org/10.82432/10317/21348.

AI/Assistance Declaration

I acknowledge that I received assistance from Grammarly for proofreading, spelling, punctuation and grammar in this assessment, in line with the Policy on Acceptable Assistance with Assessments (including Generative AI, Proofreading and Translation). This assistance was limited to language-level checking and was not used to generate technical content, change results, weaken limitations, or artificially strengthen the original analysis, claims or contribution.

Word Count Statement

Using `texcount`, the counted report sections total 6725 words. The total across all included sections is 8325 words.

Chapter A

Reproducibility Notes

The code and data that produced the evidence sit outside the formal report submission, and this appendix is meant to give a reader enough of the procedure to audit the work without prescribing a fixed set of filesystem instructions.

All of the MATLAB evidence in the report was produced from a single, consistent development checkout. The sequence I followed was, in order: initialise the project environment, set the experiment seeds, run the full experiment driver, run the MATLAB unit-test suite, and finally run the lightweight CI check. The same sequence also rebuilds the processed numerical summaries and the report figures from the saved solver outputs, so re-running it on a fresh checkout reproduces both the underlying evidence and the artefacts the evidence was distilled into.

The retained evidence record falls into four groups:

- MATLAB-native raw result bundles, each paired with lightweight JSON metadata for detailed inspection;
- processed CSV-style numerical summaries, which are the actual inputs to the report tables and plots;
- environment, solver-status, Code Analyzer, coverage and unit-test logs;
- the figure files and LaTeX source used to build the report PDF itself.

Rebuilding the report PDF needs only the LaTeX source, the bibliography and the figure set, together with a standard `latexmk` build. Re-running the numerical campaign is more involved: it needs MATLAB R2026a, CVX 2.2 and the same solver stack recorded in the logs. Small floating-point differences across platforms and solver versions are to be expected, and do not invalidate the results.

Chapter B

Code Structure and Evidence Record

The retained development evidence record contains:

- core numerical helpers: PSD checks, log-det and dB conversion;
- channel-model helpers: Rayleigh, geometry-based, CSI perturbation and measured-CFR/four-node proxy mapping;
- secrecy-rate models: SISO, MISO, MIMO, diagonal-jammer and full-covariance jamming;
- optimisation routines: SDP, GP-SCA, diagonal/full-covariance Joint-SCA, S-procedure and jammer covariance designs;
- evaluation utilities: simulation settings, confidence intervals and result saving;
- experiment drivers: seeded scripts for each figure and result bundle;
- test suite: MATLAB unit tests and integration checks.

Logs and matrices

The development evidence record keeps the solver-status summary, detailed literature matrix, environment log and unit-test log. They are not needed for reading the report; I used them to check that each figure, table and claim traces back to saved numerical output.

Detailed Solver Diagnostics

The full diagnostic table is retained outside the main narrative as a solver-status summary plus a processed diagnostic table. The main audit totals are: 3330 nominal leakage-aware statuses; 1000 runtime-scaling SDP solves; 640 diagnostic robust statuses and 6400 robust N=100 statuses; 600 heuristic/upper-bound statuses; 100 source-optimised statuses; 1500 channel-stress statuses; 100 receiver-calibrated nominal/robust statuses; 84 reduced measured-proxy evaluations; 100 full-covariance Joint-SCA channels with 100 BnB certificates; 108 four-node measured-proxy method evaluations and 500 frame-level heuristic updates.

Retrospective Work Breakdown and Schedule

Figure B.1 records how the broad project plan turned into auditable evidence streams.

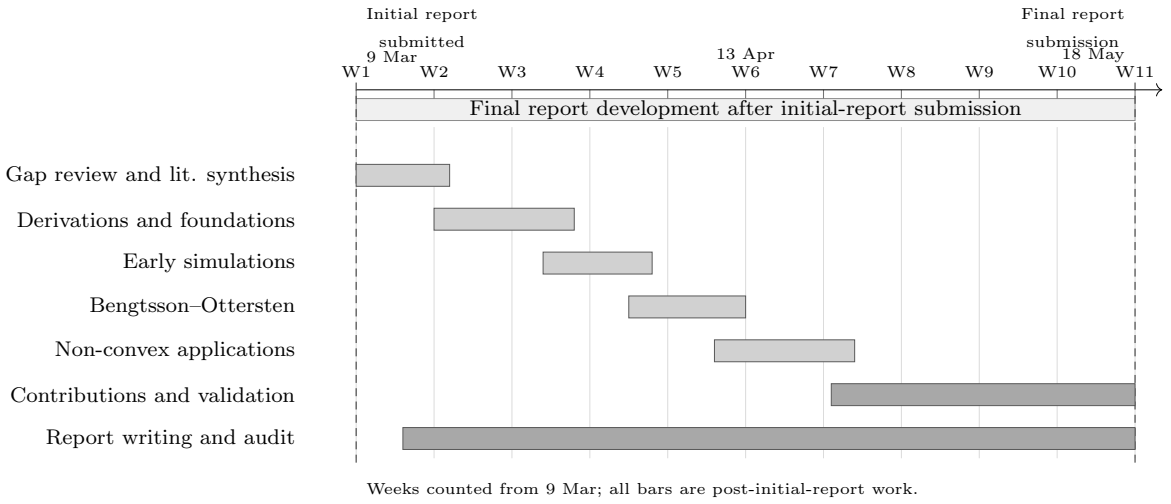

Figure B.1: Retrospective week-based WBS/Gantt view for the final-report phase: work starts after the initial-report submission on 9 March and runs through final submission on 18 May.

Auxiliary Sensitivity Figures

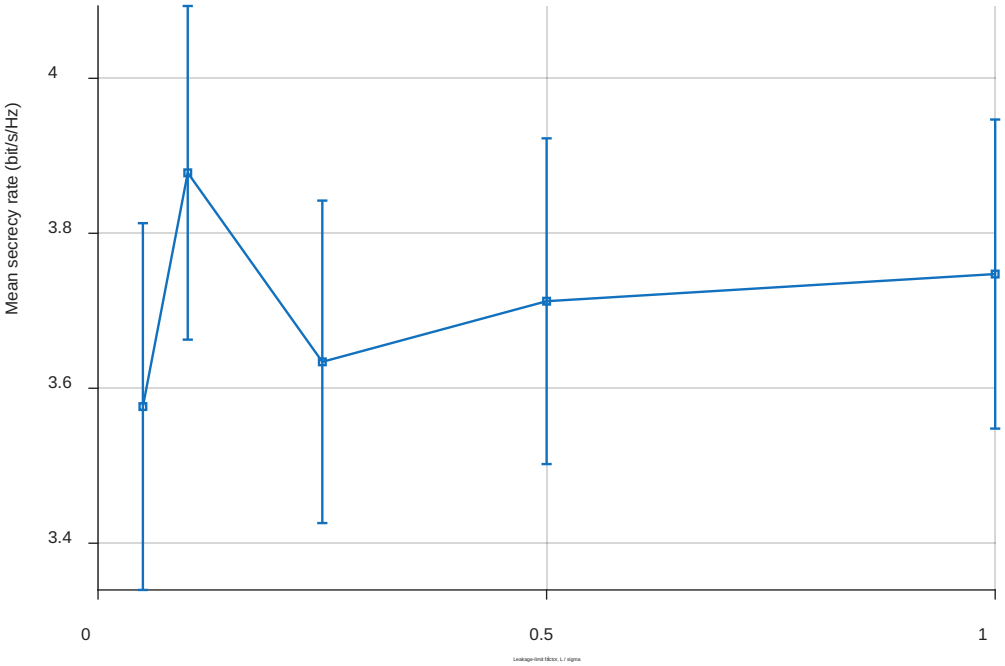

Figure B.2: Leakage-limit sensitivity for the leakage-aware SDP over 100 channel realisations. L/σ^2 is varied while $P_s = 10$, $P_j = 5$, $\beta = 0.1$ and $N_s = N_j = 3$ are fixed.

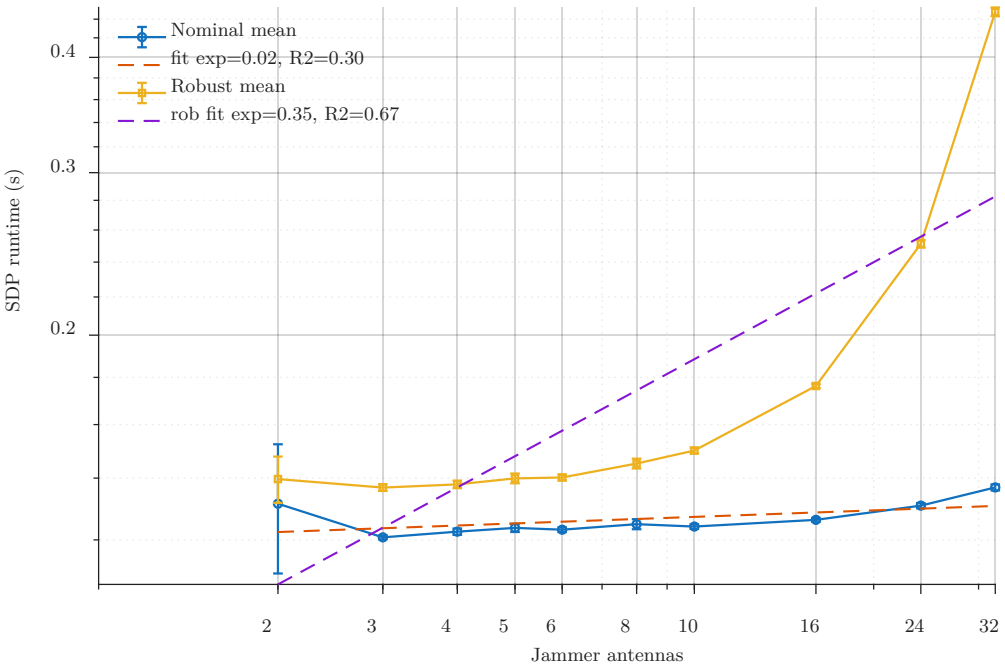

Figure B.3: Nominal and robust leakage-aware SDP runtime scaling from N_j=2 to N_j=32 (ten grid points) with 50 random channel trials per size and robust relative uncertainty radius 0.10. Solid points show means with 95 percent intervals; dashed lines are log-log diagnostic fits.

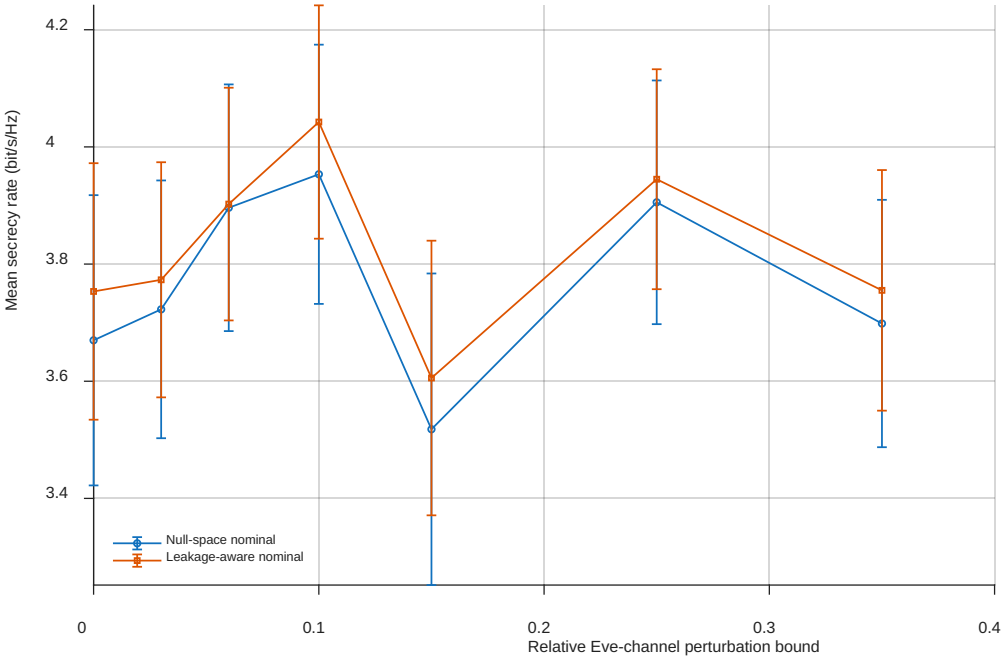

Figure B.4: Sampled Eve-channel CSI sensitivity over 100 channel realisations with $P_s = 10$, $P_j = 5$, $\beta = 0.1$ and nominal null-space/leakage-aware designs.

Auxiliary Results Figures

The figures below support specific paragraphs in Chapter 6, and have been moved here to keep the main results chapter readable.

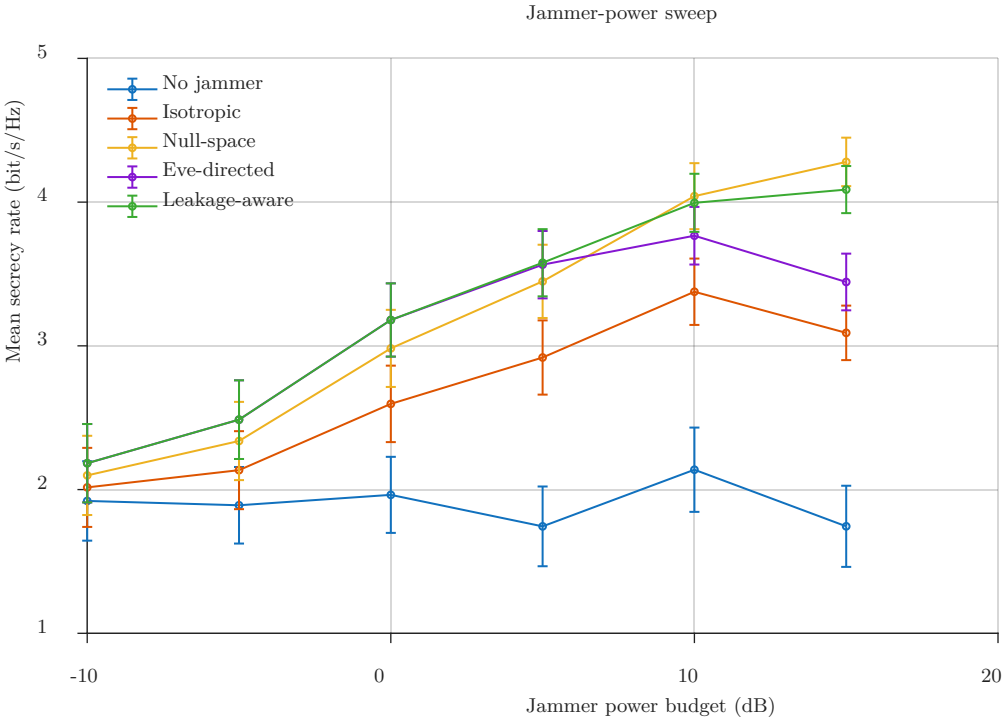

Figure B.5: Jammer-power sweep over 100 channel realisations with $P_s = 10$, $N_s = N_j = 3$, $\sigma^2 = 1$ and $\beta = 0.1$. Spatially controlled jamming is more useful than adding undirected jammer power.

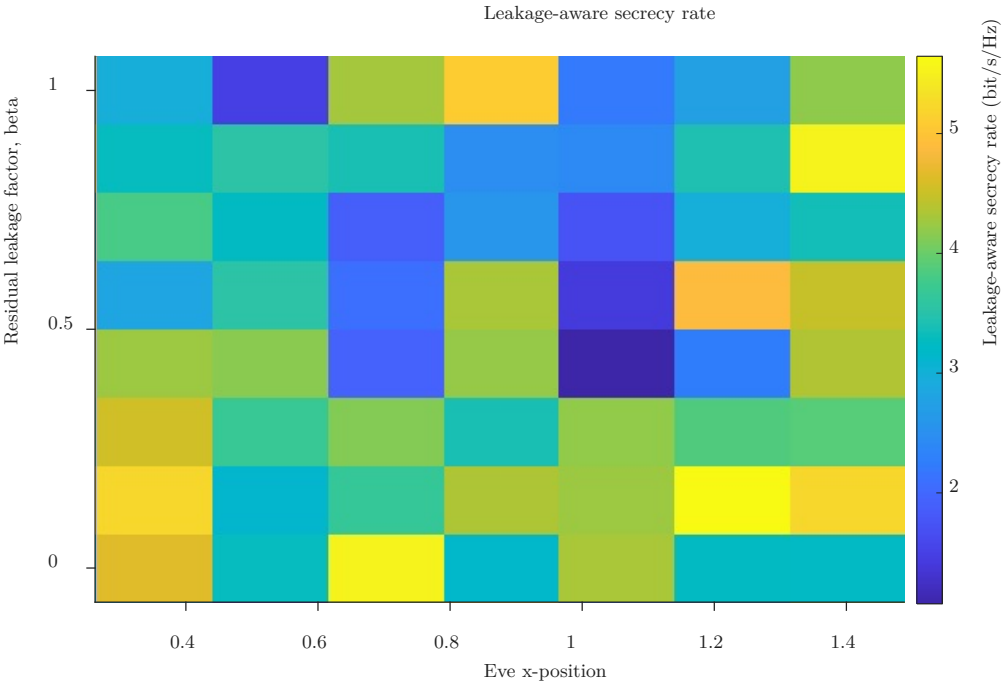

Figure B.6: Leakage-aware secrecy-rate heatmap over β and Eve x-position with one seeded channel per grid point, $P_s = 10$, $P_j = 5$, $N_s = N_j = 3$ and path-loss exponent 2.4. Illustrative seeded geometry, not a general spatial law.

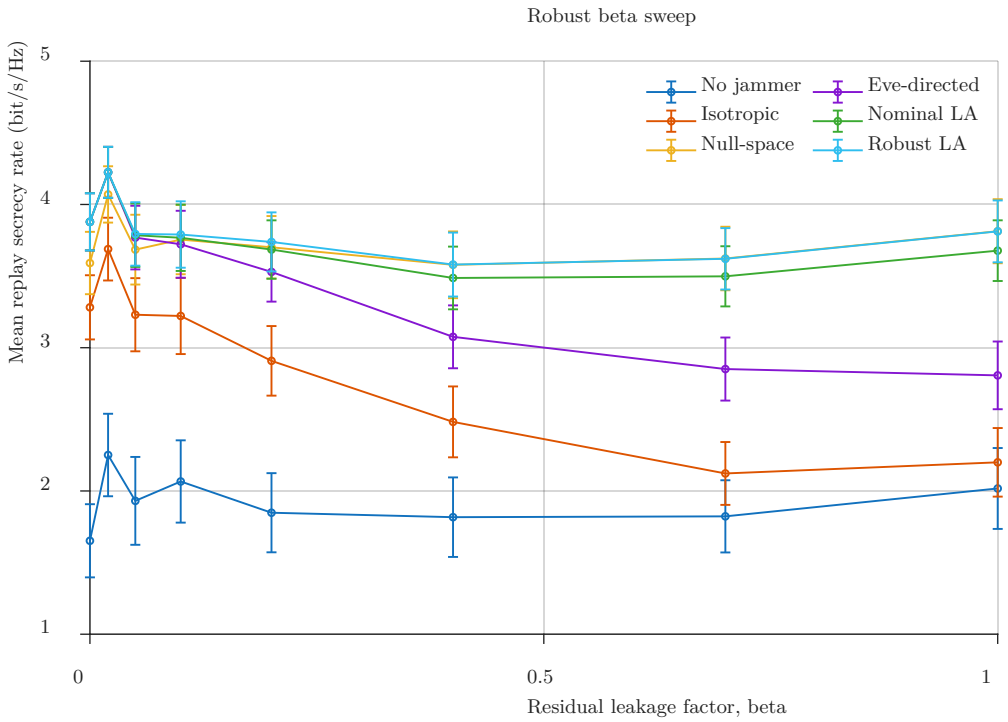

Figure B.7: Robust beta sweep over 100 channel realisations with 20 perturbation replays per realisation, relative uncertainty radius 0.10. Robust design mainly enforces leakage safety as β grows.

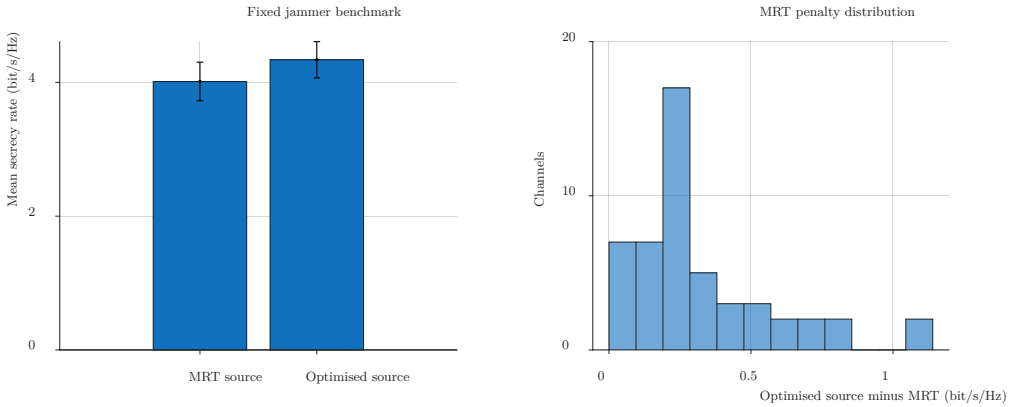

Figure B.8: Source-optimised benchmark over 50 shared channels. Only the source covariance changes from fixed MRT to MISO SDP; the jammer covariance is the nominal leakage-aware SDP in both cases.

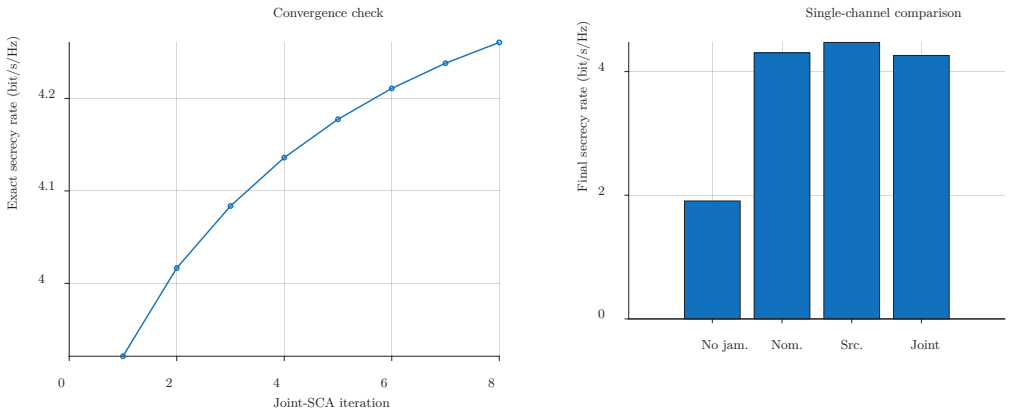

Figure B.9: Focused diagonal Joint-SCA validation. Exact secrecy rate increases monotonically over eight SCA iterations in one reproducible channel; single-channel and diagonal-jammer only.

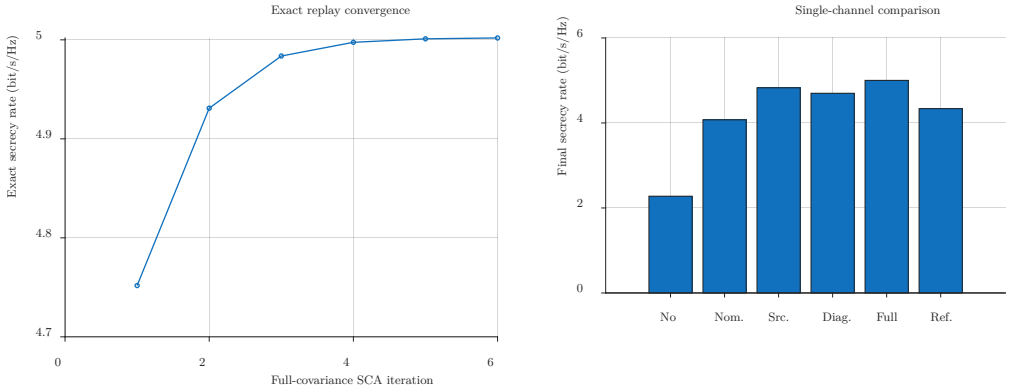

Figure B.10: Focused full-covariance Joint-SCA validation. Single-channel local solver jointly optimising full Q_s, Q_j, with exact-rate replay at every iteration.

The final Exp29 export is an audited consolidation of the predetermined sample indices 1–100. For each row, the samples CSV records the provenance the audit actually relies on: certificate run id, generation time, settings hash, solver and version text, source-CSV timestamp, node-trace availability, and the hard tolerances that applied. The certificate policy itself is fixed at `absGapTolerance=0.05` bit/s/Hz, `relGapTolerance=0.02`, and `continueUntilRelativeCriterion=true`, so every final row carries a certificate criterion of `relative` or `both`.

The accompanying JSON and log show 99 rows reused from the traced batch runs together with a targeted sample-94 retry at higher node and runtime caps, all under a single shared options hash. No row ended as timeout, node-limit, solver-limited or inaccurate, and node traces are present for every sample. The certificate gaps are retained numerically: 14 rows met the 0.05 bit/s/Hz absolute target, the median absolute gap was 0.072815 bit/s/Hz, and the worst certified absolute gap was 0.117491 bit/s/Hz. Upper-bound SDP status traceability is preserved

in the node-trace CSV through its source and jammer status fields, while the final sample rows themselves carry a clean solver status and the full solver/version text.

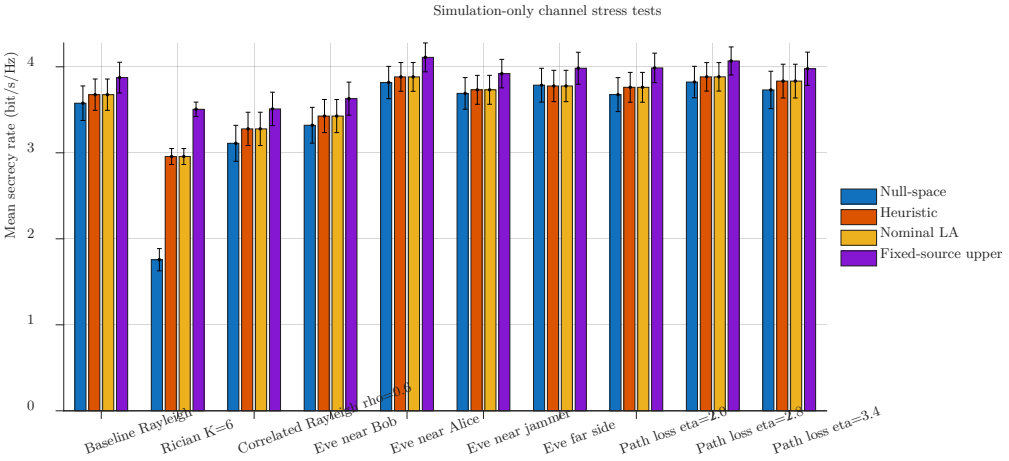

Figure B.11: Simulation-only channel stress tests over 150 realisations per scenario: Rician fading, correlated Rayleigh, Eve near Bob/Alice/jammer, farther Eve and path-loss exponents 2.0, 2.8, 3.4.

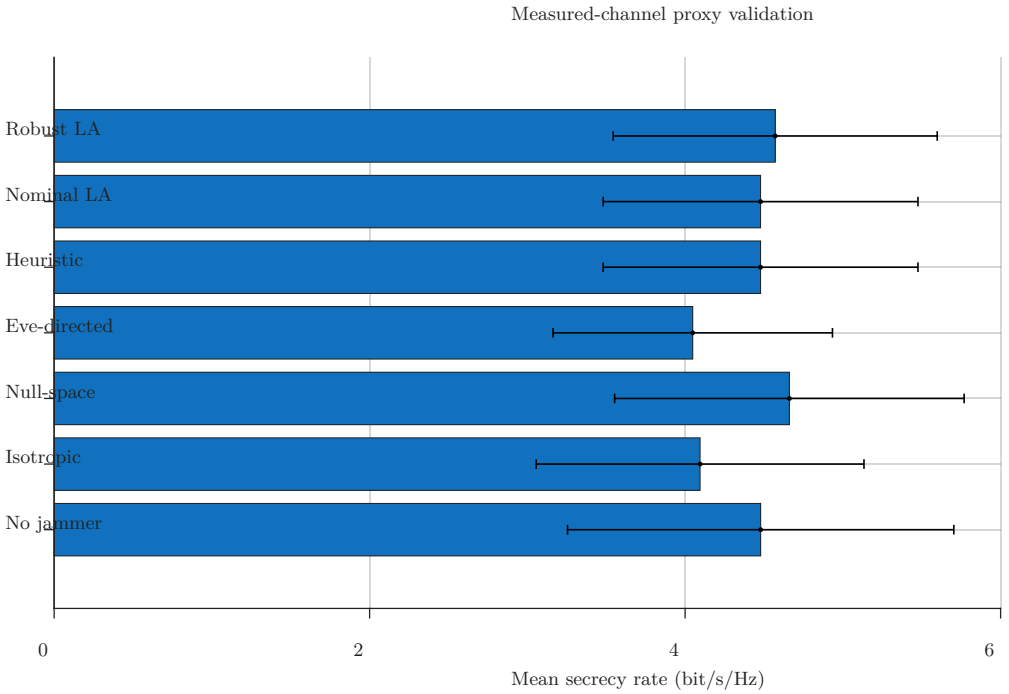

Figure B.12: Reduced measured-channel proxy using the public UPCT frequency-range-3 (FR3) VNA CFR dataset, DOI 10.82432/10317/21348 [39]. The 5×5 measured receiver URA is partitioned by reciprocity; central frequency window is normalised per scenario.

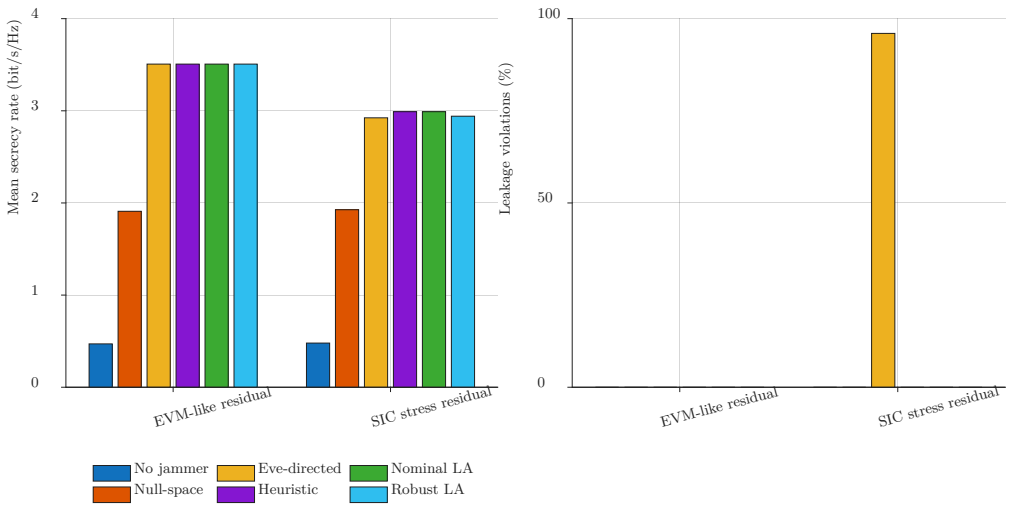

Figure B.13: Receiver-calibrated residual-leakage stress check over 50 Rician channels per setting, with EVM-like $\beta = 0.02$ and imperfect-SIC $\beta = 0.10$ at $L = 0.25\sigma^2$.

SCA Convergence Support

Razaviyayn et al.'s BSUM result is easy to misuse, so it is worth being explicit about what the surrogates here actually deliver. The result asks four things of a surrogate: local tightness at the current iterate, first-order consistency, a feasible set preserved across subproblems, and a bounded true-objective sequence to evaluate against [35]. The Cumanan-style MIMO SCA meets the first two by construction, since the tangent of a concave log-det is a global upper bound on its linearised Eve term. The other two I keep honest in code rather than on paper: my implementation replays the true secrecy rate after every subproblem, so the surrogate is checked against the underlying objective rather than taken on trust. The Cumanan 2017 GP-SCA reproduction does the same job with a different tool, the AGM monomial lower bound on the denominator posynomial, which is tight and first-order matched at the current power vector with the positive power-box constraints maintained throughout.

The diagonal Joint-SCA extension is genuinely weaker. Its Taylor terms are still tight at the current (Q_s, p), but constraining the jammer to diagonal powers walls off a slice of the design space by construction. The full-covariance solver lifts that restriction: the same lower- and upper-bound discipline then applies to Q_s and Q_j jointly, the power and leakage constraints survive every iteration, and the exact secrecy rate is evaluated after each step. Both Joint-SCA variants remain local and initialisation-dependent in principle; the BnB audit takes the edge off this for the 100 certified single-Eve channels, and the broad campaign is therefore reported as

empirical evidence rather than as a general optimality claim.

Single-Eve BnB certificate. For a single Eve, write

$$x = h_b Q_s h_b^H, \quad y = h_e Q_s h_e^H, \quad u = \beta g_b Q_j g_b^H, \quad v = g_e Q_j g_e^H,$$

so that the unclipped secrecy objective becomes

$$\phi(x, y, u, v) = \log_2\left(1 + \frac{x}{\sigma_B^2 + u}\right) - \log_2\left(1 + \frac{y}{\sigma_E^2 + v}\right).$$

The certificate proceeds by bounding ϕ on each BnB node through two independent SDPs, one over Q_s and one over Q_j, each pushed to its node-wise worst case.

Proposition. For any BnB node with $y \in [y_L, y_U]$ and $u \in [u_L, u_U]$, let x_{UB} be the optimum of

$$\max_{Q_s \succeq 0} \; h_b Q_s h_b^H \quad \text{s.t.} \quad \mathrm{Tr}(Q_s) \le P_s, \; y_L \le h_e Q_s h_e^H \le y_U,$$

and let v_{UB} be the optimum of

$$\max_{Q_j \succeq 0} \; g_e Q_j g_e^H \quad \text{s.t.} \quad \mathrm{Tr}(Q_j) \le P_j, \; \beta g_b Q_j g_b^H \le L, \; u_L \le \beta g_b Q_j g_b^H \le u_U.$$

Then every feasible covariance pair in the node has clipped secrecy rate at most

$$[\phi(x_{\mathrm{UB}}, y_L, u_L, v_{\mathrm{UB}})]^+.$$

Proof. The source SDP delivers $x \le x_{\mathrm{UB}}$ for every feasible Q_s in the node, and the jammer SDP delivers $v \le v_{\mathrm{UB}}$ for every feasible Q_j; these are independent upper-bound witnesses, not a jointly feasible covariance pair. The node itself fixes $y \ge y_L$ and $u \ge u_L$. With nonnegative received powers and positive noise variances, ϕ is increasing in x and v and decreasing in y and u, so $\phi(x, y, u, v) \le \phi(x_{\mathrm{UB}}, y_L, u_L, v_{\mathrm{UB}})$. The positive-part operator preserves the inequality, which gives the stated bound.

The proposition is a numerical certificate, not an analytic one, as it inherits the SDP solver tolerances and statuses, and applies only to the single-Eve branch boxes in (y, u).

Chapter C

Verification Code Blocks

The snippets below are shortened verifier extracts taken directly from the implementation, and not full software listings.

Bengtsson-Ottersten SDP

```
cvx_begin sdp quiet
    variable W(Nt,Nt,K) hermitian semidefinite
    minimize(sum_k trace(W(:,:,k)))
    subject to
        for i = 1:K
            signal = real(h(i,:)*W(:,:,i)*h(i,:)');
            interf = sum_{k ~= i} real(h(i,:)*W(:,:,k)*h(i,:)');
            signal >= gamma(i)*(interf + sigma2);
        end
cvx_end
```

Nominal leakage-aware jammer SDP

```
cvx_begin sdp quiet
    variable Qj(Nj,Nj) hermitian semidefinite
    variable t nonnegative
    maximize(t)
    subject to
        real(trace(Qj)) <= Pj;
        real(ge*Qj*ge') >= t;
        real(beta*gb*Qj*gb') <= leakageLimit;
```

```
cvx_end
```

Robust S-procedure jammer SDP

```
cvx_begin sdp quiet
    variable Qj(Nj,Nj) hermitian semidefinite
    variable t nonnegative
    variable lambdaB nonnegative
    variable lambdaE nonnegative
    maximize(t)
    subject to
        real(trace(Qj)) <= Pj;
        [Qj + lambdaE*eye(Nj), Qj*qe;
         qe'*Qj, real(qe'*Qj*qe) - t - lambdaE*rhoE^2] >= 0;
        [lambdaB*eye(Nj) - beta*Qj, -beta*Qj*qb;
         -beta*qb'*Qj, leakageLimit - beta*real(qb'*Qj*qb) ...
             - lambdaB*rhoB^2] >= 0;
cvx_end
```

Low-complexity leakage-aware proxy optimiser

```
uBob = gb'/norm(gb);
ePerp = ge' - uBob*(uBob'*ge');
r = norm(ePerp); c = abs(ge*uBob);
zLimit = leakageLimit/(beta*norm(gb)^2);
zEve = Pj*c^2/norm(ge)^2;
if beta == 0 || zLimit >= zEve
    x = sqrt(Pj)*ge'/norm(ge);
elseif r > 0
    uNull = ePerp/r;
    x = sqrt(Pj-zLimit)*uNull + ...
        sqrt(zLimit)*exp(-1i*angle(ge*uBob))*uBob;
```

```
else
    x = sqrt(min(Pj, zLimit))*uBob;
end
Qj = x*x';
```

Fixed-source perfect-cancellation upper bound

```
v = ge'/norm(ge);
Qj = Pj*(v*v');
Rupper = secrecy_rate_jammer_miso(w, Qj, hb, he, gb, ge, sigma2, 0);
```