

Physics Front Ends with Small Learned Heads for Detecting Dispersed Bursts in CHIME/FRB Baseband Voltages

Marcel Crasmaru
crasmarum@gmail.com

October 24, 2026

Abstract

We evaluate physics-constrained and generic learned detectors of dispersed radio bursts in complex CHIME/FRB baseband voltages, using synthetic bursts injected into real noise and interference from 29 events, with per-event thresholds at a nominal 1% false-alarm rate and seeds chosen on validation events. A hybrid that keeps a bank of 16 coherent dedispersion filters as an analytic complex front end and reads the dedispersed power with a small non-linear head detects 93% of bursts at an energy of 200 noise units with 13k parameters, and 98% with 47k, against 64–73% for generic convolutional networks of 42k–296k parameters, 51% for dedispersion at the true dispersion measure with a boxcar search, and 53% for the same filters with a linear read-out (65 parameters). It reaches the same detection rate at about half the burst energy of the generic networks, holds on the three test events not used during development, and is the only detector that handles the two test events with heavy interference. On a frozen set of ten real bursts, models with the physics front end localise 5–6, as many as dedispersion at the true DM (6), against 4–5 for generic CNNs trained on the same broad burst family (2–4 on the narrower one) and 1 for an energy detector; the test cannot yet separate the hybrid from the linear filter bank, so its gain on injected bursts is not yet shown on real ones. Matched-size complex and real CNNs perform similarly. We also find that CHIME baseband channels have the reverse frequency orientation to the textbook dispersion formula, and provide a native implementation of the filter layers in the CNet complex-valued framework with numerical gradient checks and CPU/GPU parity. Separately, an exploratory phase-jump experiment is consistent with a decayed Born-attention streaming model learning the period of the Vela pulsar in real Parkes baseband.

1 Introduction

Radio telescopes and receivers record complex voltages: each sample carries an amplitude and a phase. A fast radio burst (FRB) reaches the telescope dispersed by free electrons along its path: lower frequencies arrive later, by $\tau(f) \propto \text{DM}/f^2$, where the dispersion measure (DM) counts the electrons. Within one narrow frequency channel this dispersion is a chirp, a frequency sweep that is pure phase structure. The textbook way to undo it is coherent dedispersion [8]: an all-pass filter with quadratic phase that recompresses the chirp into a short pulse. A detector that discards phase, for example one that thresholds power, cannot use this structure, and loses sensitivity accordingly.

This makes burst detection a clean test of a broader question: when the information is in the phase, does a network that computes in complex numbers do better at a fixed size than one that treats the in-phase and quadrature (I/Q) channels as two real inputs [12]? Deep learning for FRBs usually works on detected, real-valued dynamic spectra [1, 4]. We work on baseband voltages

instead: complex samples before any power detection, after the instrument’s channelisation and beamforming. We call these raw in that sense only. We use models from a handful of parameters to 296k, and ask what a physics-structured model buys against generic networks, whether complex networks detect more than real networks of the same size, and whether a model trained only on synthetic bursts transfers to real ones.

Contributions, in order of weight:

1. A protocol on 29 CHIME/FRB baseband events: real noise and interference, injected dispersed bursts at controlled energies, splits by event, seed selection on validation events, per-event thresholds calibrated on separate noise windows, and three classical references.
2. A hybrid detector, an analytic complex dedispersion front end with a small non-linear head (13k–47k parameters), that detects 93–98% of injected bursts against 64–73% for generic CNNs up to 296k parameters, including on interference-heavy events and on test events not used during development.
3. A comparison of the same front end with a linear read-out (65 parameters, the chirp bank) and with generic complex and real CNNs (eight seeds each): the chirp bank outperforms the energy and trial-DM baselines and performs similarly to true-DM dedispersion; the CNNs detect more, mainly on events with heavy interference.
4. An ablation of the chirp bank: a trial-DM grid alone gives less than true-DM dedispersion; the learned read-out over trial DMs and widths provides the rest.
5. A matched-size complex-versus-real comparison on I/Q: tied at 42k parameters, a small and steadier complex advantage at 296k.
6. The CHIME baseband orientation finding, with the check that establishes it.
7. A native CNet implementation (**ChirpFilter**, **BoxcarMax**) with exact Wirtinger gradients, tested by finite differences and GPU-against-CPU parity.

Real-burst localisation, a label-free streaming detector and a test of period memory in a streaming Born-attention model on Vela pulsar baseband (Section 5.8) are reported as exploratory results.

2 Datasets

Both sources are public archives of complex radio-telescope voltages before any power detection or averaging (for CHIME, after channelisation and beamforming). That is rare: most public radio data are images, visibilities or detected power. Table 2 summarises them and what we derive from them.

CHIME/FRB baseband catalogue. The CHIME/FRB collaboration released beamformed baseband data for the 140 bursts of its first catalogue with recorded baseband [2, 3], on the CANFAR data publication service under DOI 10.11570/23.0029 (about 300 GB in total; files of 0.14 to 6.7 GB). Each event is one HDF5 file. Its main array, `tiedbeam_baseband`, holds complex voltages of shape (channels, polarisations, time): up to 1,024 channels of 390.625 kHz between 400 and 800 MHz, two linear polarisations, $2.56 \mu\text{s}$ per sample, recorded as 4+4-bit integers and stored as complex floats, already calibrated and beamformed to the burst position. Alongside are the coherently dedispersed power (`tiedbeam_power`, at the catalogue DM), the channel frequencies, a good-channel

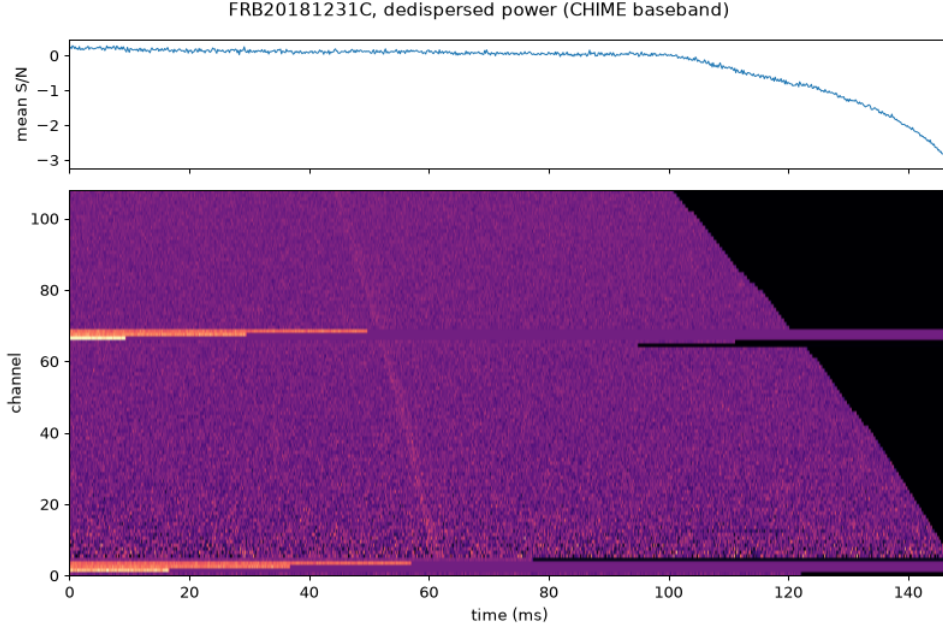


Figure 1: Coherently dedispersed power for FRB20181231C (DM 556, 108 channels from 401 to 446 MHz), as released. The burst is the faint sloped streak near 50 ms. Bright horizontal stripes are interference. The black triangle is padding where a channel has no data.

mask (absent in some files) and each channel’s start time. The windows of different channels start at different times: each holds about 147 ms (57,490 samples) placed along the dispersion sweep, so that the dedispersed burst lands at roughly the same index in every channel. Samples outside a channel’s window are padding (NaN), about 19% in the smallest file. Figure 1 shows one event.

We use 29 of the 140 events, chosen as the smallest files (up to 3 GB, 53 GB in all) for computational feasibility. This is a computationally selected subset, not a random sample of the archive, and event difficulty and interference are central to our conclusions. The events form two splits by event (Table 1). The *main split* assigns the two events of the first experiments and three random others to testing, five random events to validation (used only to choose seeds) and the remaining 19 to training. The *small split* is the first set of experiments: 10 events, 8 for training and 2 for testing. The release states no licence; we treat it as research data, and models trained on it as research results.

Breakthrough Listen open data. The Breakthrough Listen archive [10] serves raw baseband in GUPPI format, queryable by target through a public API. A GUPPI file is a sequence of blocks, each an ASCII header of 80-character cards followed by 8-bit complex samples ordered (channel, time, polarisation). For the Vela pulsar (J0835–4510) the archive holds one Parkes observation from 2018-03-21 recorded with the 13-beam receiver: 25 files of 7.4 GB, one per beam and sub-band (centred at 1,284.5 and 1,438.5 MHz), each with 44 coarse channels of 3.5 MHz, two polarisations and $0.286 \mu\text{s}$ per sample, about 12 s per file (42 million samples per channel). The file’s beam must be checked: the target name marks Vela as placed in beam 3, and only the two files with `BEAM.ID = 3` (compute nodes blc05 and blc06) contain it. A second observation listed under Vela, split into three consecutive files of 24 GB in total, was recorded by a beam pointing away from it. The data paper is published under CC BY 4.0; we have not found separate terms for the archive

Event	DM	Ch.	Win.	Main	Small	Event	DM	Ch.	Win.	Main	Small
FRB20181209A	329	760	2000	val	train	FRB20181231B	197	765	2000	test	–
FRB20181213A	679	763	2000	test	test	FRB20181231C	556	97	132	train	train
FRB20181214C	633	759	2000	train	–	FRB20190102A	699	162	512	train	train
FRB20181215B	494	754	2000	train	–	FRB20190102B	367	745	2000	val	–
FRB20181219C	648	755	2000	test	–	FRB20190103C	1,349	297	1754	test	test
FRB20181220A	210	742	2000	train	–	FRB20190106B	317	738	2000	train	–
FRB20181221B	1,395	516	2000	val	train	FRB20190110A	473	805	2000	train	–
FRB20181222E	328	714	2000	train	–	FRB20190110C	222	684	2000	train	train
FRB20181223C	113	751	2000	val	–	FRB20190111B	1,337	486	2000	val	train
FRB20181224E	582	772	2000	train	–	FRB20190115B	748	736	2000	train	–
FRB20181226A	349	895	2000	train	–	FRB20190117A	393	889	2000	train	–
FRB20181226D	385	762	2000	test	–	FRB20190118A	225	727	2000	train	–
FRB20181226E	309	311	1454	train	train	FRB20190122C	690	791	2000	train	–
FRB20181229A	956	750	2000	train	–	FRB20190124F	255	668	2000	train	–
FRB20181231A	1,377	443	2000	train	train						

Table 1: The 29 CHIME/FRB events: DM (pc cm^{-3}), good channels, noise windows, and role in the main and the small split. Both splits are by event: no channel of a test event is seen in training.

Dataset	Derived from	Content	Used in
Noise windows, main split	CHIME, 29 events	34,098 train / 10,000 val / 9,754 test windows of 16,384 samples	Sec. 5.1
Noise windows, small split	CHIME, 10 events	12,098 train / 3,754 test windows	Sec. 5.2–5.5
Injected bursts	on the fly	dispersed bursts at DM 100–1,000, width 0.1–2 ms, energy E	Sec. 5.1–5.5
Real-burst streams	CHIME, 10 held-out events	full channel windows, all good channels and polarisations	Sec. 5.6
Dedispersed token streams	CHIME, 10 events	64-sample tokens, 17 log-power features (16 trial DMs + raw)	Sec. 5.7
Vela byte streams	Breakthrough Listen, beam 3	8 channel groups $\times$ 2 files; 4,096-sample tokens (76.4 per period) or 256 (1,222)	Sec. 5.8
Synthetic pulse trains	generated	noise bytes with a pulse every 76.4, 300 or 1,222 tokens; 2.4–4.8M tokens	Sec. 5.8

Table 2: Datasets derived for this work. All CHIME splits are by event; the Vela streams are split by time (70% train, 10% validation, 20% test).

data.

3 Conventions and task

Channel orientation. Within a channel, a dispersed burst is the chirp $D(f) = \exp(\pm i \phi(f) \text{DM})$ with $\phi(f) = 2\pi k_{\text{DM}} f^2 / (f_c^2 (f_c + f))$, where f is the offset from the channel centre f_c and $k_{\text{DM}} = 4.148808 \times 10^{15} \text{ s Hz}^2$. Which sign is correct depends on how the channel’s spectrum is oriented. We fixed it empirically: we dedispersed the raw voltages with both signs and correlated the resulting power with the coherently dedispersed power the archive itself provides. Over 30 channel-polarisations of FRB20181213A the median correlation is 0.988 at zero lag for one sign and 0.06 for the other. CHIME baseband therefore has the *reverse* orientation to the textbook form. A model trained with the wrong sign learns filters that sweep backwards for real data. Our first real-burst test failed for exactly this reason, and passed after the fix (Section 5.6). Parkes data from Breakthrough Listen has the same issue at a coarser level: its band is inverted although the header gives a positive bandwidth (Section 5.8).

Task and injection. An example is one window of $L = 16,384$ complex samples (41.9 ms) from one channel and polarisation, normalised to unit mean power. Noise windows are tiled from each channel outside that channel’s own dispersion sweep around the real burst, up to 2,000 per event. Positives are noise windows with a synthetic burst added: complex Gaussian noise under a Gaussian

envelope of width 0.1 to 2 ms, dispersed within the channel at a DM drawn uniformly from 100 to 1,000, placed so that the dispersed burst fits the window, and scaled to a total energy E in noise units. Training energies are log-uniform between 50 and 3,000.

Metric and calibration. Detection probability P_d at a nominal 1% false-alarm rate. The noise windows of each validation and test event are split at random into a calibration half and a scored half. In the main split, each event’s threshold is the 99th percentile of a detector’s scores on that event’s calibration half without bursts, as a search that calibrates each observation would do; we also report a single threshold pooled over the events’ calibration halves, which is stricter on quiet events. Positives are the scored windows with one injected burst each, identical for every detector. We report the realised false-alarm rate and P_d pooled over the events of a set and per event. In the small split, the threshold is pooled over its two test events (1,877 calibration windows, 1,877 scored), with Wilson 95% intervals that treat windows as independent and therefore understate between-event variation. We report P_d at $E = 100, 200$ and 400 ; at $E = 200$ the classical detectors are in their transition, which separates models best.

Seed selection. Complex networks are sensitive to the seed (Section 5.4). In the main split each model trains eight seeds; we report the mean over seeds and the test score of the seed with the best pooled validation P_d , which is the number a practitioner who ships one model would obtain.

An earlier protocol. The first size sweeps on the small split (Sections 5.3 and 5.4) set the threshold on the same 1,500 noise windows that were scored, and those were the first in file order, all from FRB20181213A. Those tables measure one held-out event with an in-sample threshold. We keep them because they compare many sizes, seeds and training changes under identical conditions, and label them as such.

4 Models

References. The *energy detector* takes the maximum over positions and over widths $w \in \{256, 1024, 4096, 16384\}$ of the normalised window power $(\sum_w |x|^2 - w)/\sqrt{w}$; it uses no phase. *Dedispersion at the true DM* applies the exact filter $\bar{D}$ for the injected DM, then runs the same boxcar search over $w \in \{32, \dots, 1024\}$. It knows the answer to the hardest part of the problem, but it is not an optimal detector, because the maximum over many widths and positions costs sensitivity. Two further references isolate what the learned models add. The energy detector with narrow widths added ($w \in \{32, \dots, 16384\}$) tests whether the phase-blind baseline was handicapped by its widths. A *trial-DM grid search* dedisperses at the same 16 DMs as the chirp bank’s initial grid (100 to 1,000) and takes the unweighted maximum over DMs, widths ($\{32, 128, 512\}$ or $\{32, \dots, 1024\}$) and positions: the chirp bank without learning.

Complex chirp bank. K filters, each the Fourier-domain product $y_k = \mathcal{F}^{-1}(\bar{D}_k \cdot \mathcal{F}x)$ with the filter’s DM as its only parameter and the channel frequency f_c as a known input. This is the FFT–Hadamard pattern of CNet with the frequency response constrained to a quadratic phase. Power $|y_k|^2$ passes through a boxcar search over widths $\{32, 128, 512\}$, and a linear read-out maps the $3K$ values to a logit. With $K = 16$ the model has 65 parameters (16 DMs, plus 48 read-out weights and a bias). Figure 2 shows the pipeline.

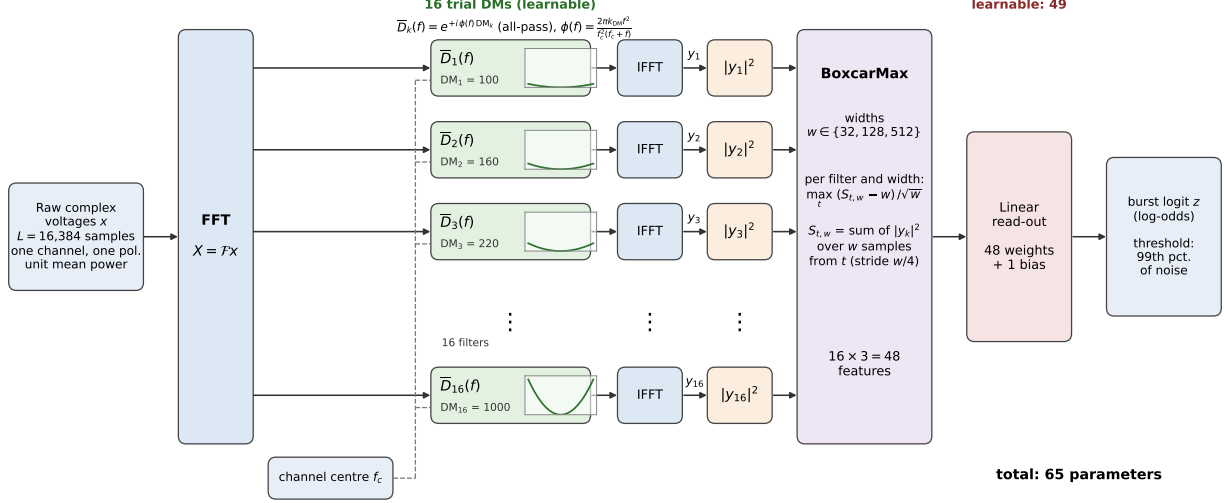


Figure 2: The 65-parameter chirp bank. Each filter is an all-pass dedispersion filter at one trial DM; the insets plot its phase $\phi(f)$ DM_k across one 390.625 kHz channel at $f_c = 600$ MHz, on a common scale. The DMs shown are the initial grid (100 to 1,000); they stay there in PyTorch and move in CNet (Section 5.5). The channel frequency f_c enters only the filters. The read-out sees the 48 boxcar features.

Hybrid: chirp front end with a small non-linear head. The same 16 fixed-grid dedispersion filters, but instead of boxcar maxima and a linear read-out, the dedispersed power $|y_k|^2$ is summed over 32 samples and normalised to unit-variance noise per bin, giving a map of trial DM against time (16×512), optionally with the undispersed power $|x|^2$ as a DM-0 channel. A small real CNN reads the map: three 1-D convolutions over time with the trial DMs as channels (kernel 5, strides 1, 2, 2, GELU), a maximum over time, and a linear head that also sees the channel frequency. With width 32 it has 12,946 parameters (13,106 with the DM-0 channel); with width 64, 46,674. The physics front end performs the coherent recompression; the head learns how to combine the evidence and what interference looks like. Figure 3 shows the pipeline.

Free filter bank. The same pipeline with $K = 4$ unconstrained complex frequency responses (131,085 parameters).

Convolutional networks. Five strided complex convolutions (kernel 9, stride 4) with split-GELU activations, the modulus, a maximum over positions and a linear head, against the same stack in real arithmetic on two input channels (I, Q). Widths are chosen so that each complex/real pair has matched parameters (about 4.7k, 42k and 295k). Both heads also see the channel frequency (Figure 4).

Every learned model outputs one real number per window, a burst logit z : the log-odds that the window contains a burst, so $\sigma(z)$ is the model’s burst probability under the training mix (half the windows with a burst). We never threshold it at $\sigma(z) = 0.5$; the detection threshold is the 99th percentile of z on noise-only calibration windows (Section 3), which fixes the false-alarm rate. All models train with Adam (learning rate 3×10^{-3} , cosine schedule, batch 64) with binary cross-entropy: 3,000 steps for filter banks and 10,000 for convolutional networks and hybrids (with warm-up and clipping where stated). Experiments ran in PyTorch on a single GPU; Section 5.5 repeats the main

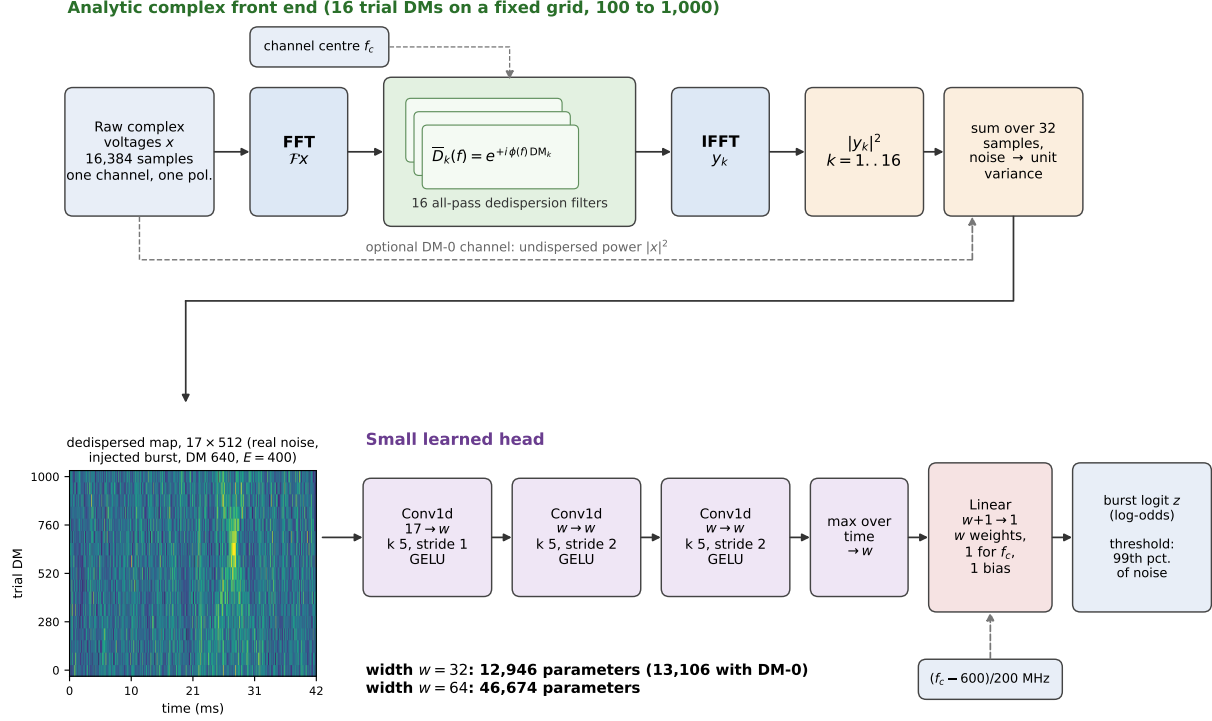


Figure 3: The hybrid detector. Top: the analytic complex front end, the same 16 fixed dedispersion filters as the chirp bank, followed by power and summation over 32 samples, with an optional undispersed DM-0 channel. Bottom: the map the learned head reads, computed with this front end from a real CHIME noise window (FRB20181213A) with one injected burst (DM 640, $E = 400$); the burst is the narrow streak strongest at the grid DMs next to 640. The head is three 1-D convolutions over time with the trial DMs as channels, a maximum over time and a linear layer that also sees the channel frequency.

model in CNet.

5 Results

5.1 Main result: 29 events

Table 3 gives the main result. **The hybrid detectors are the best models by a wide margin:** 0.93 at $E = 200$ with 13k parameters and 0.98 with 47k, against 0.73 for the best generic CNN with 296k, with a spread over eight seeds below 0.01. At $E = 100$ the 47k hybrid detects 0.51, against 0.19 for the large CNN, so it reaches a given detection rate at roughly half the burst energy, about 3 dB more sensitive. It is the only detector that keeps 0.97–0.98 on the two interference-heavy events, and on the three events not used during development it scores 0.97 against 0.68 (Table 4). Realised false-alarm rates are 0.9–1.6% pooled, as for the other detectors, and a stronger classical reference, true-DM dedispersion with boxcars up to 4,096 samples, still scores 0.51. The DM-0 channel makes no difference (0.93 with and without it), so the head does not simply veto undispersed interference. The gain comes from reading the whole dedispersed map, trial DM against time, with a small non-linear network instead of a linear read-out over 48 maxima: the physics front end recompresses

CConv (two real kernels W_r, W_i , no bias):

$$(W_r + iW_i) * (x_r + ix_i) = (W_r * x_r - W_i * x_i) + i(W_r * x_i + W_i * x_r)$$

$w = 24$: 41,930 **params**

$w = 64$: 296,130 **params**

CGELU (split activation): $\text{GELU}(\text{Re } z) + i \text{GELU}(\text{Im } z)$

receptive field 2,729 samples (7.0 ms)

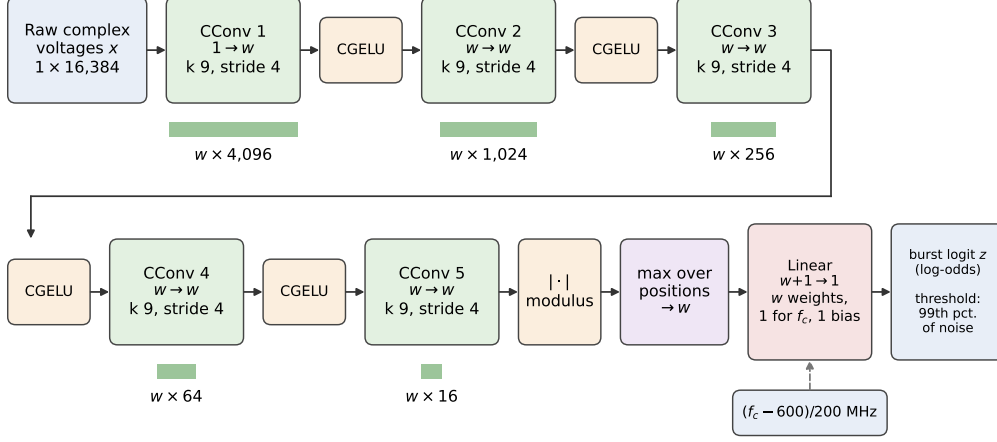


Figure 4: The complex CNN baseline. Five strided complex convolutions (no bias) with split-GELU activations between them shorten the window from 16,384 to 16 positions; the modulus and a maximum over positions give w features, and a linear head adds the normalised channel frequency. Unlike in the chirp bank, f_c enters only the head. The real twin is the same stack with real convolutions on two input channels (I, Q) and $|\cdot|$ after the last layer; $w = 34$ and 90 match the complex networks’ parameter counts (42,264 and 293,312).

the burst coherently, and the head learns to weigh the evidence across DMs and widths and to discount interference. The unweighted maximum of the classical search, by contrast, is dominated by the heavy noise tails of its narrow windows and by interference, which force a high threshold.

The chirp bank outperforms the energy detector (0.53 against 0.20 at $E = 200$) and the trial-DM grid (0.33), so the learned read-out adds 20 points to the same filters, and its eight seeds lie within 0.52–0.54. Against dedispersion at the true DM (0.51) the difference is small and not robust (Table 4). Generic networks detect more: 0.64 at about 42k parameters, 0.70–0.73 at about 295k, and 0.78 for the seed chosen on validation. With a single threshold pooled over the five events, which is stricter on the quiet ones, the order is the same. Realised false-alarm rates for the nominal 1% are 0.6–2.5% per event and 0.9–1.3% pooled (for the chirp bank 1.3%, 95% interval 1.0–1.7%).

Test-set provenance. Two of the five test events, FRB20181213A and FRB20190103C, were the test events of the earlier small split (Section 5.2), which informed our model and training choices. They were never trained on, but they are not untouched. Table 4 therefore reports the three newly held-out events separately, with an event-level bootstrap over the five. The ranking of CNNs above the chirp bank, and of the chirp bank above the energy and trial-DM baselines, holds on the new events. The chirp bank’s small lead over true-DM dedispersion does not: it comes from the two reused events, and on the three new ones the chirp bank is lower (0.42 against 0.48). With five events, the effective number of independent observations is about five, not the 4,877 scored windows, and the bootstrap intervals are correspondingly wide.

Detector	Params	P_d , mean over seeds			Val-chosen seed, $E=200$	Per test event, $E=200$					Pooled thr. $E=200$
		$E=100$	$E=200$	$E=400$		13A	19C	26D	31B	03C	
Energy, widths ≥ 256	0	0.03	0.20	0.69	–	.32	.32	.15	.02	.17	0.12
Trial-DM grid, widths 32–1,024	0	0.05	0.33	0.93	–	.37	.44	.25	.17	.45	0.30
True-DM dedispersion + boxcar	0	0.11	0.51	0.99	–	.57	.58	.46	.39	.57	0.52
Complex chirp bank, $K=16$	65	0.10	0.53 ± 0.01	0.94	0.54	.74	.76	.32	.21	.68	0.44
Complex CNN	41,930	0.14	0.64 ± 0.06	0.97	0.72	.87	.86	.81	.37	.71	0.59
Real CNN on I/Q	42,264	0.14	0.64 ± 0.08	0.96	0.74	.87	.86	.79	.39	.77	0.60
Complex CNN	296,130	0.19	0.73 ± 0.03	0.98	0.78	.91	.91	.79	.45	.83	0.72
Real CNN on I/Q	293,312	0.17	0.70 ± 0.05	0.98	0.78	.95	.93	.88	.35	.81	0.68
Hybrid, width 32	12,946	0.35	0.93 ± 0.01	1.00	0.93	.95	.94	.92	.91	.96	0.93
Hybrid, width 32, DM-0 channel	13,106	0.36	0.93 ± 0.01	1.00	0.94	.97	.96	.90	.90	.98	0.94
Hybrid, width 64, DM-0 channel	46,674	0.51	0.98 ± 0.01	1.00	0.98	.98	.97	.98	.97	.99	0.98

Table 3: Main split: 19 training, 5 validation and 5 test events; per-event thresholds at a nominal 1% false-alarm rate (realised 1.0–1.6% pooled), P_d pooled over the five test events (4,877 scored windows). Trained models: mean $\pm$ population standard deviation over eight seeds; the val-chosen seed is the one with the best validation P_d , and the per-event columns are for that seed. Events: FRB20181213A, 20181219C, 20181226D, 20181231B, 20190103C. Last column: one threshold pooled over the five events’ calibration noise. CNNs and hybrids use warm-up and clipping (10,000 steps), complex CNNs also the variance-matched initialisation.

Where the chirp bank loses. The per-event columns locate the gap. On FRB20181213A, 20181219C and 20190103C the chirp bank detects 0.68–0.76, 0.13–0.21 below the large networks. On FRB20181226D and FRB20181231B it detects 0.32 and 0.21, below dedispersion at the true DM (0.46, 0.39). These two events are the ones whose noise produces the most false alarms for every detector under a pooled threshold (for the energy detector 0.8% and 3.1%, against at most 0.1% elsewhere), which points to interference. The chirp bank’s read-out is a linear function of 48 boxcar maxima: it can weigh evidence across trial DMs and widths, but a bright interference spike raises the same maxima that a burst does. The CNNs keep 0.79–0.88 on FRB20181226D; a plausible reason, which we have not tested, is that their non-linear layers see the shape of the signal around a peak and learn to discount interference-like shapes. Neither model handles FRB20181231B well (0.21–0.45).

Complex versus real. At about 42k parameters the two families are tied (0.64 each, val-chosen 0.72 and 0.74). At about 295k the complex network has the slightly higher mean (0.73 against 0.70) and a smaller spread over eight seeds (0.03 against 0.05; ranges 0.68–0.78 and 0.62–0.78); the val-chosen seeds tie at 0.78. Under the architectures and training recipe tested here, the larger complex CNN had a higher mean detection probability and lower observed variation across eight seeds than its parameter-matched real counterpart. Parameter matching does not make the two matched in compute, optimisation geometry or inductive bias, so we do not attribute the difference to complex arithmetic alone, and eight seeds on one task do not estimate the best attainable performance. The clearest finding is the negative one: once warm-up and clipping are used, the difference is small.

How much the held-out data matter. On the small split’s two test events the chirp bank led all models on average (Section 5.2). Both of those events are in the main test set, and on them the chirp bank now scores 0.74 and 0.68 against 0.84 and 0.72 before, partly because its read-out is trained on different events; the reversal comes from adding interference-heavy test events and 11 more training events. A comparison of detectors on one or two held-out events can rank them in the wrong order.

Detector (P_d at $E=200$)	All 5 events	Event bootstrap (95%)	3 new events	2 reused events
Energy, widths ≥ 256	0.20	0.10–0.29	0.17	0.24
Trial-DM grid	0.34	0.24–0.43	0.29	0.41
True-DM dedispersion + boxcar	0.51	0.44–0.57	0.48	0.57
Complex chirp bank, $K=16$	0.53	0.32–0.73	0.42	0.70
Complex CNN, 42k	0.64	0.45–0.78	0.59	0.71
Real CNN, 42k	0.64	0.46–0.77	0.58	0.73
Complex CNN, 296k	0.73	0.53–0.86	0.68	0.81
Real CNN, 293k	0.70	0.51–0.84	0.65	0.78
Hybrid, 13k	0.93	0.90–0.96	0.91	0.96
Hybrid, 13k, DM-0 channel	0.93	0.91–0.96	0.92	0.96
Hybrid, 47k, DM-0 channel	0.98	0.96–0.99	0.97	0.99

Table 4: Main split by event subset: per-event thresholds, P_d averaged over events (and over the eight seeds for trained models). New events: FRB20181219C, 20181226D, 20181231B; reused: FRB20181213A and 20190103C, the test events of the small split. Bootstrap: resampling the five events.

Detector	Params	Win/s	FA	P_d , $E=100$	P_d , $E=200$	13A / 03C, $E=200$	Seeds, $E=200$	P_d , $E=400$
Energy, widths ≥ 256	0	15,400	.010	0.04	0.25 [.23,.27]	.33 / .13	–	0.79
Energy, widths 32–16,384	0	14,300	.005	0.01	0.05 [.04,.06]	.08 / .03	–	0.48
Trial-DM grid, widths 32/128/512	0	3,300	.013	0.06	0.39 [.37,.41]	.41 / .39	–	0.96
Trial-DM grid, widths 32–1,024	0	2,100	.013	0.06	0.42 [.40,.44]	.44 / .43	–	0.99
True-DM dedispersion + boxcar	0	31,400	.007	0.12	0.56 [.54,.58]	.56 / .56	–	1.00
Complex chirp bank, $K=16$	65	4,500	.009	0.16	0.77 $\pm$ 0.00	.84 / .72	.78 .77 .77	1.00
same, in CNet (one seed)	208	–	.011	0.17	0.77	.82 / .72	–	1.00
Complex CNN, warm-up + clipping	41,930	17,600	.012	0.13	0.65 $\pm$ 0.14	.70 / .59	.52 .84 .58	0.98
Real CNN on I/Q, warm-up + clipping	42,264	48,600	.011	0.11	0.59 $\pm$ 0.02	.64 / .51	.60 .56 .60	0.97

Table 5: Small split, calibrated protocol (Section 3): threshold from 1,877 separate noise windows, P_d and realised false-alarm rate (FA) on 1,877 scored windows of the two held-out events, pooled, with per-event P_d for FRB20181213A / FRB20190103C. Brackets: Wilson 95% intervals (per window, see text). Trained models: mean $\pm$ population standard deviation over three seeds; FA and per-event values are seed means. Win/s: windows of 16,384 samples per second on one GTX 1080 Ti in PyTorch, batch 64; one window is 41.9 ms of one channel-polarisation.

5.2 Small split: two cleaner test events

The small split came first, and it is where we isolated where the chirp bank’s gain comes from. Its two test events are relatively clean, and on them the 65-parameter chirp bank detects 77% of bursts at $E = 200$, with no spread over seeds, against 25% for the energy detector, 42% for the non-learned trial-DM grid and 56% for dedispersion at the true DM. It leads on both held-out events. FRB20190103C is harder for every detector, most of all for the energy detector (0.13), which bright interference in that recording pushes to a high threshold. Realised false-alarm rates on the scored windows lie between 0.5% and 1.4% for the nominal 1%.

The phase-blind baseline was not handicapped. Adding narrow widths to the energy detector makes it worse (0.05 against 0.25). Within one 390 kHz channel the dispersion spreads a burst over thousands of samples, so narrow boxes capture little of its energy, while each extra width adds trials that raise the noise threshold. The wide-width detector is the stronger phase-blind reference, and it is the one we use elsewhere.

Model	Params	P_d , $E=100$	P_d , $E=200$	Seeds, $E=200$	P_d , $E=400$
Energy detector (no phase)	0	0.05	0.35	–	0.88
Dedispersion at true DM + boxcar	0	0.12	0.58	–	1.00
Complex chirp bank, $K=4$	17	0.18	0.76 ± 0.01	.77 .75 .75	1.00
Complex chirp bank, $K=16$	65	0.20	0.82 ± 0.00	.83 .82 .82	1.00
Free complex filter bank, $K=4$	131,085	0.05	0.29	(one seed)	0.75
Complex CNN	4,762	0.11	0.60 ± 0.07	.53 .70 .59	0.98
Real CNN on I/Q	4,567	0.09	0.53 ± 0.01	.53 .54 .52	0.97
Complex CNN	41,930	0.16	0.73 ± 0.11	.59 .87 .72	0.99
Real CNN on I/Q	42,264	0.11	0.61 ± 0.03	.59 .66 .59	0.98
Complex CNN	296,130	0.13	0.49 ± 0.36	.89 .01 .57	0.66
Real CNN on I/Q	293,312	0.10	0.56 ± 0.03	.54 .54 .60	0.97

Table 6: Earlier protocol: detection probability at a 1% false-alarm rate with the threshold set on the same 1,500 noise windows that are scored, all from FRB20181213A (mean $\pm$ population standard deviation over three seeds unless stated). CNNs here are trained without warm-up or clipping; Table 7 adds them.

Where the gain comes from. The trial-DM grid is the chirp bank with its filters at their initial DMs and an unweighted maximum in place of the learned read-out. It scores 0.39–0.42, below dedispersion at the true DM (0.56), as expected: a grid DM is never exactly right, and every extra trial raises the threshold. The learned 49-number read-out lifts the same filters from 0.39 to 0.77. It learns a weighting over the three widths and over adjacent trial DMs: a burst between two grid DMs lights up both neighbouring filters, while their noise is less correlated, so a weighted sum separates burst from noise better than the maximum. The channel frequency enters only inside the filters, so each trial DM is exact for every channel. In short, the physics grid makes the model small and stable; the learned combination of evidence makes it better than the classical search it is built from.

Generic networks. On these two events, convolutional networks of about 42k parameters average below the chirp bank (0.65 complex, 0.59 real), though one complex seed reaches 0.84. With 19 training events and five test events, including two with heavy interference, the order reverses (Section 5.1). Parameter count is not compute: the chirp bank runs 17 FFTs of length 16,384 per window and processes 4,500 windows per second in this unoptimised PyTorch form, fewer than the CNNs, which use strided convolutions. That is still about 190 channel-polarisations in real time on one consumer GPU, without overlapping windows.

5.3 Size sweep on the small split (earlier protocol)

The sweeps below used the earlier protocol (Section 3): one held-out event, FRB20181213A, with an in-sample threshold. They compare many sizes and seeds under identical conditions; absolute values are higher than in Table 5 and should be read relative to each other.

Table 6 shows the size sweep. On this one clean event the chirp bank is the best model on average at every energy and essentially seed-independent. With only 4 filters (17 parameters) it already matches or beats the mean of every generic network; single complex CNN runs can do better (Section 5.4). Physics structure, not size, carries the result: the unconstrained version of the same pipeline (131k parameters) ends below the energy detector after the same training. Without warm-up and clipping, larger generic networks did not help on average; with them, and with more

Model	Training change	Seeds at $E=200$	Mean
Complex CNN, 42k	none (Table 6)	.59 .87 .72	0.73
Complex CNN, 42k	initialisation scaled by $1/\sqrt{2}$	.71 .55 .56	0.61
Complex CNN, 42k	+ warm-up (500 steps) and gradient clipping	.67 .88 .74	0.76
Complex CNN, 42k	+ RMS normalisation per position	.01 (first seed)	–
Real CNN, 42k	warm-up and clipping	.65 .65 .69	0.66
Complex CNN, 296k	warm-up and clipping	.48 .97 .87	0.77
Real CNN, 293k	warm-up and clipping	.85 .57 .86	0.76

Table 7: Stabilisation (earlier protocol). Warm-up and clipping help both families. At 42k parameters the complex network is ahead on average (0.76 against 0.66); at about 295k the means are equal, and the best complex run is higher (0.97 against 0.86; at $E = 100$, 0.43 against 0.23).

training events, they do (Section 5.1).

A note on what is learned. With Adam at learning rate 3×10^{-3} and the DM as a raw parameter (values 100 to 1,000), the filters’ DMs never left their initial grid (100, 160, ..., 1000). The PyTorch chirp bank is therefore, in effect, a fixed physics grid of 16 dedispersion filters plus a learned read-out of 49 numbers. The reason is scale: Adam’s step is about the learning rate in parameter units, so in 3,000 steps a DM can move by at most about 9 units out of hundreds, and the loss varies rapidly with DM, so its gradient sign is not consistent over such moves. The CNet version parametrises $DM = 100d$, which makes the same Adam step 0.3 DM units, and the DMs do move (Section 5.5).

5.4 Complex versus real networks, and training stability

This subsection uses the small split and the earlier protocol, except where it cites Table 5. The complex CNNs in Table 6 average higher than the real ones, and single runs reach the level of the chirp bank (0.87, 0.89), but their spread over seeds is 3 to 12 times larger, and one 296k run collapsed. We tested three remedies on the 42k complex network, then gave the real networks the same training (Table 7).

Three observations. First, matching the complex initialisation’s variance to a real layer (each of the real and imaginary kernels is initialised independently, so the complex weight starts with twice the variance) did not help, so excess signal growth is not the main cause. Second, warm-up and clipping raised both families. Third, per-position RMS normalisation across channels destroyed the task entirely: the loss stayed at $\ln 2$. A burst *is* extra power, and that normalisation removes power. It is not an instability fix for an energy-detection problem.

Under the calibrated protocol on the small split (Table 5) the 42k pair gives complex 0.65 ± 0.14 (seeds 0.52, 0.84, 0.58) against real 0.59 ± 0.02 . With eight seeds on the main split (Section 5.1) the two are tied at this size, and the complex network is slightly ahead and steadier at 296k. Warm-up and clipping are therefore what made the complex networks trainable here; once they are used, the remaining difference between complex and real arithmetic is small.

5.5 The model in CNet

We implemented the chirp bank natively in CNet [6]. `ChirpFilter` takes $[x | f_c | d]$ (signal, channel frequency in MHz as a per-example input, and K parameters with $DM_k = 100 \text{Re } d_k$) and returns K dedispersed signals. The map is linear in x , so with $g = \partial L / \partial y$ and $h = \partial L / \partial \bar{y}$ the Wirtinger

gradients [9] are

$$\frac{\partial L}{\partial x} = \sum_k \mathcal{F}(\bar{D}_k \mathcal{F}^{-1} g_k), \quad \frac{\partial L}{\partial \bar{x}} = \sum_k \mathcal{F}^{-1}(D_k \mathcal{F} h_k),$$

and, since only the real part of d_k enters, $\partial L / \partial d_k = \partial L / \partial \bar{d}_k = (S_1 + \bar{S}_2)/2$, where $S_1 = \sum_f (\mathcal{F}^{-1} g_k)_f V_f$, $S_2 = \sum_f (\mathcal{F}^{-1} \bar{h}_k)_f V_f$ and $V = \partial_{d_k} (\bar{D}_k) \mathcal{F} x$. The imaginary part of d_k never moves. Here $\mathcal{F}$ is the unnormalised forward DFT and $\mathcal{F}^{-1}$ its inverse including the $1/L$, the same pair as in the forward pass $y_k = \mathcal{F}^{-1}(\bar{D}_k \mathcal{F} x)$, so $\partial y_k / \partial d_k = \mathcal{F}^{-1} V$ and $\sum_t g_t (\mathcal{F}^{-1} V)_t = \sum_f (\mathcal{F}^{-1} g)_f V_f$ (the DFT matrix is symmetric). **BoxcarMax** returns, per filter and width, the maximum normalised window sum and sends its gradient to the winning window. The GPU versions batch all examples and filters through cuFFT.

Tests: the forward pass with DM 0 is the identity, and it undoes a DM-400 dispersion to 8×10^{-8} . Finite differences match the analytic gradients to 10^{-4} (signal) and 10^{-3} (DM) relative error through a smooth loss; through **BoxcarMax** the check is only piecewise valid, since a step that changes the winning window breaks it (about 2% worst case). The GPU matches the CPU for all outputs and gradients at window lengths up to 16,384. The network **ChirpFilter**($K=16$) $\rightarrow |\cdot|^2 \rightarrow$ **BoxcarMax** $\rightarrow$ linear $\rightarrow$ Born-rule cross-entropy has 208 learned real parameters: 16 DMs (real; only $\text{Re } d_k$ moves) and a 48×2 complex linear head (192 reals), since the Born-rule loss uses two complex output classes where PyTorch used one real logit. The linear layer has no bias. Under the protocol of Table 5 it reaches $P_d = 0.17, 0.77$ and 1.00 at $E = 100, 200$ and 400 (one seed; per event 0.82 and 0.72 at $E = 200$; realised false-alarm rate 1.1%), matching the PyTorch model ($0.17, 0.77, 1.00$). This is a functional replication of the detection performance, not a parameter-matched comparison: the two-class complex head makes the CNet model 208 real parameters against 65.

The DMs learn in CNet but not in PyTorch. This is the one substantive difference between the two implementations. In PyTorch the DM is a raw parameter (values 100 to 1,000) and stays on its initial grid (Section 5.3); CNet parametrises $\text{DM} = 100d$, so the same Adam step moves a DM 100 times further, and the DMs leave the even grid: after training they are 94, 174, 204, 274, 344, 408, $\dots$, 961, 986. That the two reach the same P_d suggests that, at 16 filters over this DM range, the read-out matters more than where exactly the filters sit. Training takes about 8 minutes on a GTX 1080 Ti.

5.6 Localising the real bursts (offline)

This test asks whether a model trained only on injected bursts responds to real ones. It is not a blind detection test: the bursts are known to be in these recordings, and the normalisation below uses the whole recording. We slid a 16-filter chirp bank over the recordings. It is a separate model from those of the benchmark tables: trained on the small split’s training events, only on injected bursts with the CHIME sign, and with DMs from 100 to 1,500 (not 1,000) to cover the test events over every good channel and polarisation of the two held-out events, in 16,384-sample windows with stride 4,096. Each window is first scaled to unit mean power, as in training. For each channel-polarisation we then standardise the window scores by their mean and standard deviation over the *whole* recording (offline, not causal), and sum the standardised scores over channels, as a multi-beam search would combine evidence. The energy detector (widths ≥ 256 , the stronger variant in Table 5) is processed identically. Figure 5 shows the timelines for the two test events of the small split, as an example.

Frozen evaluation on ten real bursts. Two events cannot establish how a model behaves on real bursts, so we froze an evaluation on the real bursts of all 10 events that no model was trained

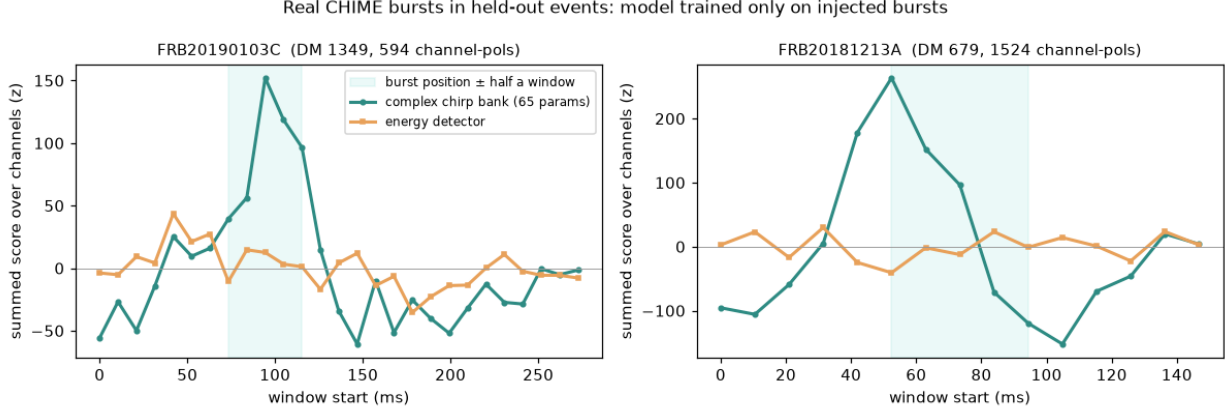


Figure 5: Channel-summed standardised scores for the two test events of the small split (offline normalisation over each recording). The chirp bank peaks at the real burst in both; the energy detector does not.

on: the main split’s five test and five validation events, with DMs from 113 to 1,395 (Table 8). Each model is trained on the 19 training events, on injected bursts with DMs from 100 to 1,500, with the settings of Table 3 and no selection on these bursts. We train each model twice: on the Gaussian bursts used so far, and on a broader family in which each burst has, at random, a Gaussian envelope, a one-sided exponential scattering tail (0.5 to 4 widths) or two components (1.5 to 5 widths apart, amplitude ratio 0.3 to 1), occupies only part of the channel’s band with probability 0.5 (a Gaussian spectral window covering 15 to 60% of it), and has a linear spectral tilt. We record whether the timeline peaks in a window that contains the burst, and the margin of the best burst window over the strongest other window in units of the other windows’ spread, so that models with different score scales compare. As a ceiling for this test we add dedispersion at each event’s catalogue DM followed by the boxcar search, processed identically.

Dedispersion at the true DM localises 6 of the 10 bursts. The four it misses (FRB20181219C, 20181226D, 20181231B and 20190102B) do not stand out in this single-channel, 42-ms-window search even with the right DM, so this test cannot score any model on them; they include the two interference-heavy test events. Against this ceiling, the models with the physics front end localise 5 or 6 bursts, essentially everything the true DM finds: the chirp bank finds all three bursts with DM above 1,300, and the hybrid two of them. The generic CNNs localise fewer. With Gaussian training the real CNN finds 4 and the complex CNN 2; with broad training, over three seeds each, they find 4, 5, 4 (real) and 5, 5, 4 (complex), so broad training makes them consistent, and complex and real do not differ meaningfully. All six broad-trained CNN runs miss FRB20190111B (DM 1,337), which the true DM, the chirp bank and the hybrid all localise, and they find FRB20190103C (DM 1,349) in only three of six runs. Over the high-DM cases (DM above 1,300) the physics-front-end runs find 10 of 12 and the CNN runs 10 of 24 (9 of 18 with broad training), consistent with the CNNs’ receptive field of 2,729 samples (7 ms) being shorter than a high-DM burst’s dispersion smear within one channel. On bright, low-DM bursts the CNNs have the largest margins (for FRB20181213A, up to 9.2 against 4.2 for the chirp bank). The energy detector localises 1. Training on the broader burst family changes little for the physics-front-end models and makes the CNNs more consistent. So on real bursts the physics front end matters, but this test cannot separate the hybrid from the linear chirp bank, since both reach the ceiling: the hybrid’s large advantage on injected bursts is neither confirmed nor refuted on real ones. A more sensitive real test, for example one that combines

Event (FRB2018.../2019...) DM (pc cm ⁻³)	1209A 329	1213A 679	1219C 648	1221B 1,395	1223C 113	1226D 385	1231B 197	0102B 367	0103C 1,349	0111B 1,337	Found	Margin
Energy detector (no phase)	✓	·	·	·	·	·	·	·	·	·	1/10	-1.3
Dedispersion at the true DM	✓	✓	·	✓	✓	·	·	·	✓	✓	6/10	+0.5
Chirp bank (65), Gaussian	✓	✓	·	✓	·	·	·	·	✓	✓	5/10	+0.2
Chirp bank (65), broad	✓	✓	·	✓	·	·	·	·	✓	✓	5/10	+0.1
Hybrid (47k), Gaussian	✓	✓	·	✓	✓	·	·	✓	·	✓	6/10	+0.2
Hybrid (47k), broad	✓	✓	·	✓	✓	·	·	·	·	✓	5/10	+0.1
Real CNN (293k), Gaussian	✓	✓	·	·	✓	·	·	·	·	✓	4/10	-0.8
Real CNN (293k), broad, seed 0	✓	✓	·	✓	✓	·	·	·	·	·	4/10	-0.4
seed 1	✓	✓	·	✓	✓	·	·	·	✓	·	5/10	+0.2
seed 2	✓	✓	·	✓	✓	·	·	·	·	·	4/10	-1.1
Complex CNN (296k), Gaussian	✓	✓	·	·	·	·	·	·	·	·	2/10	-0.8
Complex CNN (296k), broad, seed 0	✓	✓	·	✓	✓	·	·	·	✓	·	5/10	-0.1
seed 1	✓	✓	·	✓	✓	·	·	·	✓	·	5/10	-0.8
seed 2	✓	✓	·	✓	✓	·	·	·	·	·	4/10	-1.1

Table 8: Frozen real-burst evaluation on the 10 events never trained on (the main split’s test and validation events). ✓: the channel-summed score timeline peaks in a window that contains the burst. Found: number of events. Margin: median over events of the best burst window minus the strongest other window, in standard deviations of the other windows. All learned models are trained on the 19 training events with DMs 100–1,500, on Gaussian or broad injected bursts, with fixed settings; nothing is selected on these bursts. Each row is one training run (seed 0 unless stated).

channels coherently, or many more real bursts, is needed for that.

5.7 Exploratory: a label-free streaming detector

This section is exploratory and not part of the main claims. Towards long-running monitors, we trained a model that never sees a burst label. Each channel stream is coherently dedispersed at 16 fixed trial DMs (100 to 1,500), and power is averaged over tokens of 64 samples (164 μ s), giving 17 log-power features per token (the trial DMs plus raw power). A small complex Born-attention model (two blocks, width 32 complex, four heads, per-head decay with half-lives of about 6 to 44 tokens; 36k real parameters) predicts each next token as a diagonal Gaussian. The 17 real features enter as the real part of a complex vector ($x + i0$) through a complex linear embedding, and the mean and log-variance are read from the real and imaginary parts of the last block’s output. Born attention scores with the squared magnitude of a complex inner product and runs exactly as a fixed-size recurrence [5], so the same model streams in constant memory. Training used 4,000 steps on the streams of the eight training events.

How surprise is scored decides the result. The total negative log-likelihood over all 17 features ranked the burst regions *below* average (606th of 1,058 token positions and 1,087th of 1,848): ordinary fluctuations in the features a burst does not touch dominate the total. Scoring instead the largest *rise* of any feature above its prediction, in predicted standard deviations, ranks the burst first in both events (Figure 6). On FRB20190103C it also beats the raw best-DM power, which is fooled by a bright feature elsewhere (burst rank 54): the learned model has seen that feature as normal. The margin there is narrow (a second peak at about 0.9 of the burst). We chose the one-sided score after seeing the first score fail on these same two events, so this test is not blind, and two events are too few to support a detection claim. A blind test on fresh events is future work.

5.8 Exploratory: period memory in a streaming Born model (Vela)

A constant-state model reads an input of any length, but reading is not remembering [7]. A pulsar is a clean test of memory: Vela emits a dispersed pulse every 89.4 ms, so predicting the next pulse

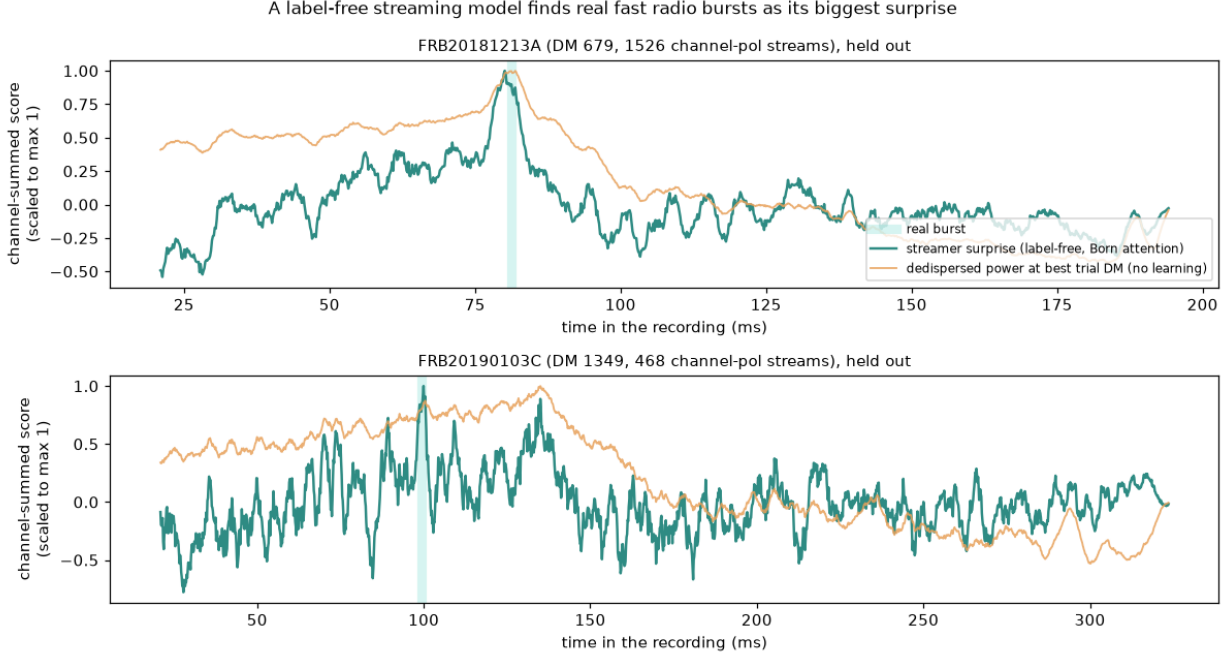


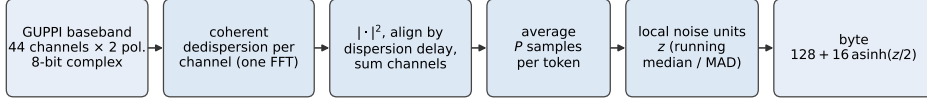
Figure 6: Label-free streaming surprise (one-sided, per trial DM), summed over channels, on the two held-out events. The first 21 ms are excluded as model warm-up.

requires remembering the phase across a period. We test this with the CNet streaming byte model (Figure 7).

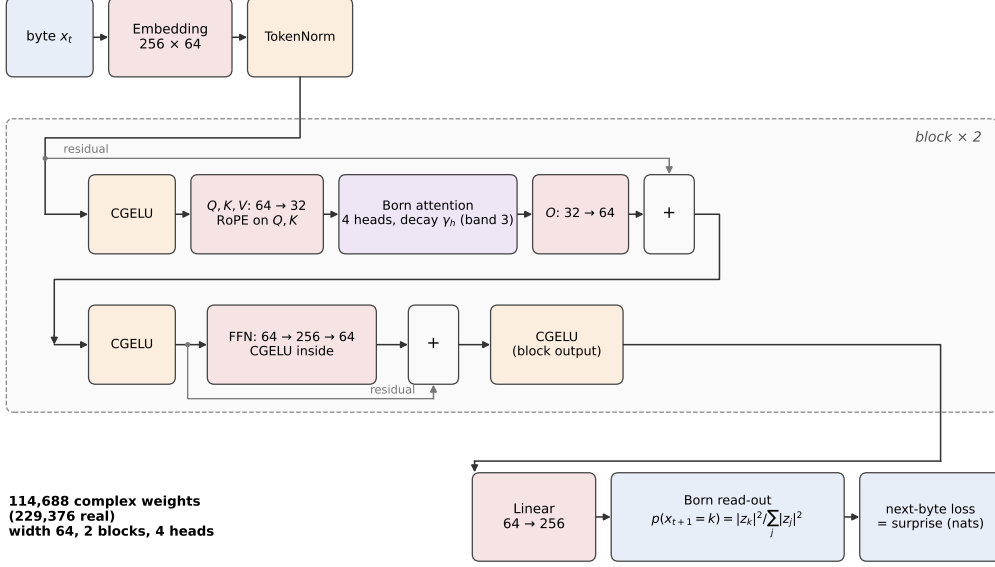
Data. The Breakthrough Listen Parkes baseband of Section 2 holds 12 s of Vela per file, in two sub-band files. The band is inverted: dedispersing with the header’s channel order folds to a signal-to-noise ratio of about 1, the inverted order to about 4,000 after coherent dedispersion within channels. Power is averaged over P samples per token, put in local noise units z (running median and $1.4826 \times$ median absolute deviation over 4,097 tokens) and coded as the byte $\text{clip}(\text{round}(128 + 16 \operatorname{asinh}(z/2)), 0, 255)$; noise ($|z| < 3$) occupies about 16 levels around 128 and clipping affects only pulses above $z \approx 600$. The model reads one byte per token through its embedding and predicts the next byte. With $P = 4,096$ (1.17 ms) a period is 76.4 tokens; with $P = 256$ it is 1,222. To get more pulses from the same seconds we split each file’s 44 coarse channels into 8 interleaved groups, each dedispersed and coded separately (16 streams, folded signal-to-noise 700–1,470 each), and split every stream by time. As a control with abundant data we generate synthetic pulse trains: Gaussian noise with a pulse every 76.4, 300 or 1,222 tokens (non-integer periods, so the phase drifts), 2.4–4.8M tokens. All training files begin with the 256 byte values in order, which fixes CNet’s byte vocabulary, so weights transfer between synthetic and real streams.

Phase-jump test. The model is scored with its state reset every W tokens ($W = 2,048$ on synthetic and 1,024 on Vela streams at a 76-token period; 4,096 or 8,192 at longer periods). Each window is built from W plus a few times SHIFT consecutive test tokens by physically removing SHIFT tokens at fixed positions inside the window (at 400, 800, 1,200, 1,600 on synthetic and 300, 600, 900 on Vela; for the controls, one position J), so the model sees an ordinary contiguous byte stream in which the pulse phase jumps by SHIFT tokens. SHIFT is half a period (phase broken) or

1. Raw baseband to byte tokens (one stream per channel group)



2. Network (CNet, complex weights throughout)



3. Born attention with per-head decay (linear, constant-size state)

$$O_t = \frac{\sum_{m \leq t} \gamma_h^{t-m} |(q_t, k_m)|^2 v_m}{\sum_{m \leq t} \gamma_h^{t-m} |(q_t, k_m)|^2 + \epsilon}$$

exact recurrence with a fixed-size state
 $S_t = \gamma_h S_{t-1} + (k_t \otimes \hat{k}_t) v_t^T$;
 constant memory over any stream length

$\gamma_h = 1 - 2^{-(b+sh)}$, heads $h = 0..3$
 $b = 6, s = 1.5$: half-lives $\approx 44, 125, 355, 1,004$ tokens (periods: 76, 300, 1,222)

Figure 7: The streaming model for the pulsar experiments: raw baseband to byte tokens (one stream per channel group), the CNet network (2 blocks of width 64, 114,688 complex weights), and Born attention with a fixed per-head decay, computed as an exact recurrence with a constant-size state. The decay shown ($b = 6, s = 1.5$) is the one of the curriculum; the 76-token experiments use $b = 5, s = 1$ (half-lives 22–177 tokens).

a full period (phase kept, the control); both conditions use the same windows and positions. We report the loss in nats on the first pulse onset after each removal (a token with $z > 10$ after one without). A model that tracks the phase should be surprised by that onset after a half-period jump and not after a full-period one; a model that has only adapted to the stream’s noise level and pulse brightness should not care. Standard errors in Table 9 are per onset; the controls below cluster by stream.

Results. Table 9 summarises the test. On synthetic data with a 76-token period, a half-period jump makes the next pulse cost 6.38 nats against 3.14 after a full-period jump, and warm onsets cost less than noise: the model expects each pulse. On real Vela the same holds, and is strongest for the model pretrained on synthetic pulses and fine-tuned on Vela: 9.41 against 4.28 nats. After a half-period jump the next pulse costs even more than a pulse right after a reset (7.6), because the model confidently expected it elsewhere; later pulses re-lock. Fine-tuning from synthetic data also

Data (period)	Model	Half-period jump	Full-period jump	Warm onsets
Synthetic (76)	trained on synthetic	6.38 ± 0.16	3.14 ± 0.06	2.59 ± 0.03
Vela, 16 streams (76)	trained on Vela	7.93 ± 0.27	5.91 ± 0.27	6.07 ± 0.18
Vela, 16 streams (76)	synthetic, fine-tuned on Vela	9.41 ± 0.35	4.28 ± 0.12	4.68 ± 0.14
Synthetic (300)	curriculum from 76	6.27 ± 0.25	2.63 ± 0.06	2.75 ± 0.07
Synthetic (1,222)	from scratch	5.74 ± 0.37	6.13 ± 0.39	6.24 ± 0.49
Synthetic (1,222)	curriculum 76 $\rightarrow$ 300 $\rightarrow$ 1,222	5.44 ± 0.36	4.54 ± 0.28	4.70 ± 0.33
Synthetic (1,222)	curriculum, slower decay, 8k steps	5.97 ± 0.30	3.23 ± 0.12	3.53 ± 0.18
Synthetic (1,222)	same, context grown to 8,192	5.32 ± 0.35	2.92 ± 0.07	3.23 ± 0.14

Table 9: Loss (nats, $\pm$ standard error) on the first pulse onset after a phase jump of half a period or a full period, and on onsets after the first period of a window and before any jump (warm). Onset counts per cell: 29–436 after jumps, 40–461 warm. Noise tokens cost about 3.4 nats. A large half-minus-full gap means the model predicts where the pulse should be.

gives the best Vela validation loss (3.306 against 3.364 from scratch).

Two further findings on synthetic data. Without decay the memory works only within the training length: scored with windows eight times longer than training, onset loss rises to 6.25 nats, while fixed and slow decay keep it at 2.6–2.8. And gated linear attention over complex amplitudes did not learn the period at all (onset loss 3.95 after the first period against 3.89 within it); on this task the fixed decay is the better choice.

Controls. Onsets within a stream are not independent, so we repeat the test with one removal per window at position J and compute the half-minus-full effect per stream, with the standard error across the 16 Vela streams (or 16 time slices of the synthetic stream). For the fine-tuned Vela model the effect is -1.23 ± 0.26 nats with half a period of history before the jump, $+5.67 \pm 0.32$ with 1.6 periods and $+6.11 \pm 0.49$ with 3.9 periods, positive in 16 of 16 streams once there is history (from scratch: -0.45 , $+2.77$, $+2.46$). The effect therefore needs previous pulses. On synthetic data it is already $+2.70$ at half a period, where about half the windows hold a pulse before the jump, and $+3.0$ to $+4.1$ with more history. In a second control we rebuild each test stream from chunks of random length (0.5 to 1.5 periods) separated by random gaps, which keeps pulse shapes, amplitudes and noise but destroys the periodicity: warm onsets then cost 2.21 ± 0.35 nats more for the fine-tuned Vela model (15 of 16 streams), 0.78 ± 0.23 more from scratch, and 4.13 ± 0.05 more on synthetic data. These results are consistent with learned periodic-phase tracking: the advantage requires earlier pulses, depends on their periodicity, and is broken by a half-period phase jump but not by a full-period one.

Longer periods. Memory is harder to learn as the period grows. At 300 tokens it is still strong. At 1,222 tokens, training from scratch learns none; a curriculum that starts at 76 and grows the period gives a partial effect (5.44 against 4.54, about two standard errors). Giving all four heads long half-lives ($\gamma_h = 1 - 2^{-(8+h)}$, about 180 to 1,400 tokens) and training the last stage $3\times$ longer gives a clear one on synthetic data (5.97 against 3.23), with warm onsets close to the noise level. Growing the context of the last stage from 4,096 to 8,192 tokens, by warm-starting the 4,096-token model at the longer length (CNet’s weights do not depend on the context length when there is no learned positional table), gives the same picture (5.32 against 2.92), with onsets after a full-period jump below the noise level. On the full-band Vela streams at this period, models trained from scratch or fine-tuned from these synthetic models all point the same way (for example 3.68 against 3.25 nats), but each cell has only 6 or 7 onsets after a jump, too few to conclude. On real Vela at 1,222

tokens per period we see no phase effect, but there the per-token pulse signal of one channel group is weak (onsets cost about as much as noise), so the test has little to detect. Our first attempts at this period also failed for a simpler reason: with pulse onsets at about 0.15% of tokens, perfect memory would lower the mean loss by only about 0.005 nats per token, too little to learn from. Shorter tokens per period, more pulses, synthetic pretraining and a period curriculum are what made the effect measurable.

6 Limitations

- **Hybrid on real signals.** The hybrid’s advantage is measured on injected bursts. On ten real bursts it reaches the ceiling of our localisation test, as does the linear chirp bank (Section 5.6), so the test cannot tell them apart; a more sensitive real test or many more real bursts are needed.
- **Computationally selected events.** The 29 events are the smallest files of a 140-event release, not a random sample; harder or longer recordings may behave differently.
- **Test-set provenance.** Two of the five main test events were used in earlier exploratory experiments (Table 4). A fully untouched test set would strengthen the main comparison.
- **Classical reference.** Dedispersion at the true DM with a boxcar search is a strong reference but not an optimal detector; a matched-filter detector for the injected pulse family is a natural follow-up.
- **Five test events.** The main result rests on five held-out events, and the ranking depends on which events are held out (Section 5.1): two of the five carry most of the gap between the chirp bank and the CNNs. Windows within an event are dependent, so per-window intervals understate the uncertainty.
- **Synthetic bursts.** The benchmark tables use noise-like bursts with a Gaussian envelope, a uniform DM in 100–1,000 and a width of 0.1–2 ms. Real FRBs have richer morphology; Section 5.6 adds a broader family (scattering tails, two components, partial band, spectral tilt) for training, and its real-burst test covers ten known events with offline per-recording normalisation.
- **One, moderate false-alarm rate.** A nominal 1% rate, calibrated on about 1,000 noise windows per event. Search-grade rates (10^{-6} and below) need far more noise data and an analysis of the score tails.
- **Single channels.** Each window is one channel and polarisation. Real searches combine channels and beams; our real-burst test sums standardised scores across channels only afterwards.
- **Parameters are not compute.** The chirp bank has 65 parameters but runs 17 FFTs of length 16,384 per window; Table 5 gives measured throughput. Its cost is close to the non-learned grid search, and above the energy detector’s.
- **Seeds and architectures.** Eight seeds per model in the main split, three in the small split; one CNN design at two sizes and one 1-D convolutional head on the dedispersed map. The real-burst evaluation uses one training run per model, three for the broad-trained CNNs.

- **Streaming memory.** One pulsar, 24 s of real data in two sub-bands, strong pulses, a small model; memory at 1,222-token periods is shown only on synthetic data; on real data at that period there are too few onsets to conclude.
- **Licences.** The CHIME release states no licence; we treat models trained on it as research results.

7 Related work

Coherent dedispersion is standard in pulsar and FRB processing [8]. Deep-learning FRB classifiers usually operate on real-valued dynamic spectra after incoherent dedispersion, for example [1, 4]. The closest design to our hybrid is FETCH [1], whose CNNs classify candidates from frequency-time and DM-time images after incoherent dedispersion. Our hybrid instead dedisperses the complex voltages coherently, inside a differentiable network trained end to end, and detects rather than classifies candidates; we also compare complex and real networks at matched size. Complex-valued networks have a long history [12]; we use exact Wirtinger gradients [9] throughout. Born attention, its exact linear form and its decay follow [5, 11]; the distinction between streaming and memory follows [7].

8 Conclusion

On CHIME/FRB baseband voltages with injected bursts and 29 events, the best detector combines both approaches: an analytic complex dedispersion front end with a small non-linear head detects 93–98% of bursts with 13k–47k parameters, against 64–73% for generic networks up to 296k parameters, including on interference-heavy events and on events not used during development. On ten real bursts it localises as many as dedispersion at the true DM, as does the linear chirp bank, while generic CNNs miss high-DM bursts (this is offline localisation of known bursts, not blind discovery); whether the learned head adds to the physics front end on real signals is still to be shown. The same front end with only a linear read-out (65 parameters) is small and seed-independent; it outperforms the energy and trial-DM baselines and performs similarly to dedispersion at the true DM. That chirp bank is not a strong detector on its own: generic CNNs of 42k to 296k parameters detect more, mainly on events with heavy interference, where the chirp bank’s linear read-out over boxcar maxima cannot separate bursts from interference. On a smaller split with two cleaner test events, the same comparison pointed the other way, which is a caution for small evaluations. Complex and real CNNs of matched size were tied at 42k parameters, with a small and steadier complex advantage at 296k in our recipe. Two of the five test events were used in earlier experiments; the conclusions above hold on the three that were not, except the comparison with true-DM dedispersion. Getting the orientation of the data right mattered more than any modelling choice.

Future work: training the front end’s DMs jointly with the head (CNet’s DM parametrisation lets them move, PyTorch’s raw one does not), and coherent multi-channel tests, to see whether the hybrid’s gain on injected bursts transfers to fainter real bursts; more held-out events; search-grade false-alarm rates; multi-channel models; a blind test of the streaming detector; and streaming memory at long periods on real data (Section 5.8).

Reproducibility. Code, results and figures are in the author’s repository (`rf_product/chime/`): `prep.py` (windows), `sweep.py` (models, training, the earlier protocol), `baselines2.py` (the protocol of Table 5), `split.py` and `rescore.py` (the main split and Table 3; the hybrid is `HybridBank`

in `sweep.py`, trained by `run_hybrid.sh`), `real_bursts.py` with `run_realeval*.sh` and `realeval_summary.py` (the frozen real-burst evaluation of Table 8), `stream_r7.py`; for Section 5.8 `vela_tokens.py`, `synth_pulsar.py`, `phase_eval.py` and the `run_vela4.sh`, `run_phase.sh` and `run_curriculum*.sh` scripts. The CNet layers and tests are `impl/chirp.h`, `gpu/chirp-gpu.cpp`, `examples/test_chirp_fd.cpp`, `examples/test_chirp-gpu.cpp` and `examples/chime_detect.cpp`. Appendix A lists the splits, seeds and settings.

References

- [1] D. Agarwal, K. Aggarwal, S. Burke-Spolaor, D. R. Lorimer, and N. Garver-Daniels. FETCH: A deep-learning based classifier for fast transient classification. *Monthly Notices of the Royal Astronomical Society*, 497:1661, 2020.
- [2] CHIME/FRB Collaboration. The first CHIME/FRB fast radio burst catalog. *The Astrophysical Journal Supplement Series*, 257:59, 2021.
- [3] CHIME/FRB Collaboration. Updating the first CHIME/FRB catalog of fast radio bursts with baseband data. *The Astrophysical Journal*, 2024. Data release DOI 10.11570/23.0029.
- [4] L. Connor and J. van Leeuwen. Applying deep learning to fast radio burst classification. *The Astronomical Journal*, 156:256, 2018.
- [5] M. Crasmaru. Born attention is linear: Exact sub-quadratic attention and constant-state inference from quantum measurement, 2026. Preprint, https://phasefield.ai/papers/born_attention_is_linear.pdf.
- [6] M. Crasmaru. CNet: A complex-valued deep learning framework with wirtinger autodifferentiation and FFT–Hadamard convolution, 2026.
- [7] M. Crasmaru. Streaming is not memory: Constant-state Born-attention byte models on PG-19, 2026. Preprint, https://phasefield.ai/papers/streaming_is_not_memory.pdf.
- [8] T. H. Hankins and B. J. Rickett. Pulsar signal processing. In *Methods in Computational Physics*, volume 14. Academic Press, 1975.
- [9] K. Kreutz-Delgado. The complex gradient operator and the CR-calculus. *arXiv preprint arXiv:0906.4835*, 2009.
- [10] M. Lebofsky et al. The Breakthrough Listen search for intelligent life: Public data, formats, reduction, and archiving. *Publications of the Astronomical Society of the Pacific*, 131:124505, 2019.
- [11] Y. Sun, L. Dong, S. Huang, S. Ma, Y. Xia, J. Xue, J. Wang, and F. Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.
- [12] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. J. Pal. Deep complex networks. In *International Conference on Learning Representations*, 2018.

A Reproducibility manifest

Item	Setting
Data	CHIME/FRB baseband release, DOI 10.11570/23.0029; the 29 events of Table 1
Main split	by event (<code>split.py</code> , Python <code>random.Random(0)</code>): 19 train (34,098 windows), 5 val (10,000), 5 test (9,754)
Small split	by event: 8 train (12,098 windows), 2 test (2,000 + 1,754 windows)
Windows	16,384 samples, one channel and polarisation, unit mean power, tiled outside the channel's own burst sweep
Calibration / scored	random half of each val/test event (<code>NumPy default_rng(0)</code> , events in sorted order)
Injection	noise-like burst, Gaussian envelope 0.1–2 ms, DM $U(100, 1000)$, CHIME sign, energy E
Training energies	log-uniform 50–3,000; label 1 with probability 0.5
Evaluation bursts	PyTorch generator seed 123 (same bursts for every detector); calibration noise unchanged
Training	Adam, lr 3×10^{-3} , cosine schedule, batch 64; 3,000 steps (chirp banks), 10,000 (CNNs and hybrids, with warm-up and clipping)
Real-burst evaluation	the 10 main-split test and validation events; models trained on the 19 training events with DMs 100–1,500, Gaussian or broad bursts (<code>sweep.INJECT</code>), seed 0 (CNNs with broad bursts also 1, 2)
Training seeds	0–7 (main split), 0, 1, 2 (small split): <code>torch.manual_seed</code> and the injection generator
Flags	<code>i</code> : variance-matched complex initialisation; <code>w</code> : 500-step warm-up and gradient clipping at norm 1
Hardware	main split: one NVIDIA GH200 (PyTorch 2.7); small split: one GTX 1080 Ti (PyTorch 1.12), CNet on the same GPU

Table 10: Settings behind Tables 3 and 5.