RETHINKING SPECTRAL CONDITIONING FOR NMR STRUCTURE ELUCIDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models for nuclear magnetic resonance (NMR) structure elucidation typically condition generation through a single global spectral representation, applying the same modulation to every atom and, in graph models, every bond. We study whether this uniform conditioning interface limits structure recovery and replace it with per-element spectral cross-attention, allowing each atom and candidate bond to query the spectral tokens independently. On simulated SpectraBase spectra, this raises top-1 exact recovery on NMR-DiGress from 43.26% for our same-protocol FiLM reproduction to 52.10%. The same conditioning change transfers to the coordinate-based ChefNMR architecture, where per-atom cross-attention improves over the released ChefNMR-S model by 7.24 percentage points under a common evaluation. Fine-tuning the cross-attention models with an auxiliary spectral reconstruction loss further reaches 54.38% on NMR-DiGress and 78.29% on ChefNMR. These results show that per-element spectral conditioning improves structure recovery across both graph and coordinate diffusion models.

1 INTRODUCTION

Structure elucidation from NMR spectra is the step of automated characterisation that still relies largely on expert interpretation to recover complete and correct topology from routine one-dimensional measurements, and computer-assisted structure elucidation remains underused (Aurich & Schymanski, 2026). Recently proposed methods instead use data-driven generative models that treat structure elucidation as a conditional generation problem. They either use sequential autoregressive models that generate molecular structures from spectra (Alberts et al., 2023; Hu et al., 2024; Han et al., 2026), or diffusion models that generate molecular graphs or atomic coordinates (Xiong et al., 2025; Yang et al., 2026b).

The diffusion-based approaches, among them ChefNMR, NMR-DiGress (Xiong et al., 2025) and DiffNMR (Yang et al., 2026b), share a common conditioning pipeline. A spectral encoder maps the spectrum to a sequence of latent tokens, and a global pooling step reduces the entire sequence to a *single* vector, which modulates the generative backbone through FiLM (Perez et al., 2018) or adaptive layer normalisation (adaLN) (Peebles & Xie, 2023). All atoms, and in graph models all bonds, therefore receive the *same* modulation. Here we study how the conditioning interface connects spectral tokens to generated structures. Cross-attention gives an autoregressive structure generator token-level access to spectral representations (Han et al., 2026), and it is standard in text-conditioned image diffusion, where image features attend to text tokens (Rombach et al., 2022). Spectra also form sequences of tokens with local meaning.

Our contributions are as follows.

- We identify uniform pooled modulation as a bottleneck of spectral conditioning in NMR diffusion models.
- We replace pooled spectral conditioning with per-element queries: separate node and edge queries in NMR-DiGress, and per-atom queries in place of the pooled spectral vector in ChefNMR. For a single conditioning step, uniform modulation is a strict special case of cross-attention under non-degeneracy conditions.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

- Cross-attention raises top-1 exact recovery from 43.26% for our FiLM reproduction to 52.10% on NMR-DiGress, and reaches 76.64% on ChefNMR, above the reported 72.04% of the three times larger adaLN-ChefNMR-L.

2 Related Work

Diffusion-based structure generation. Diffusion models generate the desired output by iteratively denoising corrupted representations. DiGress (Vignac et al., 2023) generates molecular graphs by denoising a discrete diffusion process (Austin et al., 2021) over node and edge categories, and later work extends this graph-diffusion backbone with hierarchical discrete diffusion (Jung et al., 2026). Xiong et al. (2025) adapt DiGress to NMR structure elucidation as NMR-DiGress, pairing it with their NMRConvFormer (NCF) spectral encoder: the pooled spectral embedding enters as a graph-level feature that DiGress injects into every node and edge update through FiLM layers (Vignac et al., 2023; Perez et al., 2018), so all atoms and bonds receive the same scale and shift. Our reproduction, FiLM-NMR-DiGress, is the baseline we modify. The same work introduces ChefNMR, a Diffusion Transformer that denoises 3D atomic coordinates and is conditioned by adaptive layer normalisation with zero initialisation (adaLN-Zero).

Conditioning. FiLM (Perez et al., 2018) conditions a network by predicting a scale and a shift from a conditioning vector and applying them to a feature map. Adaptive layer normalisation, used in diffusion transformers (Peebles & Xie, 2023), is the same idea composed with a normalisation step (Ba et al., 2016). Cross-attention conditioning comes from latent diffusion, where it conditions image synthesis on text and other structured inputs (Rombach et al., 2022). The per-token routing it induces is explicit enough to manipulate directly (Hertz et al., 2023). We test whether learned attention lets each atom and candidate bond select context from the unassigned spectrum.

Related families. Spectrum-to-structure methods fall into three families by how they generate structures. Input modality varies independently of the family. *Diffusion models* differ mainly in how the spectrum enters. DiffNMR (Yang et al., 2026b) conditions discrete graph diffusion through FiLM, with a peak-list encoder, contrastive pretraining and retrieval-seeded inference. DiffSpectra (Wang et al., 2025) generates 2D and 3D structure from multi-modal spectra through adaLN on a pooled vector. NMIRacle (Ottomano et al., 2025) conditions on IR and NMR jointly through a pooled latent embedding. DiSE (Chen et al., 2025) is the closest per-element design. It takes the molecular formula from mass spectrometry and attaches assigned chemical shifts to nodes and COSY correlations to edges as input features. *One-shot models* map a spectrum directly to a candidate structure. Early work applies sequence-to-sequence transformers to peak-list text (Alberts et al., 2023). Hu et al.'s multitask model (Hu et al., 2024), called NMR2Struct in benchmark papers, is also the source of the simulated SpectraBase dataset. Its follow-up extends de novo generation to molecules with up to 40 non-hydrogen atoms (Hu et al., 2026). Later models take two-dimensional spectra (Xue et al., 2025), peak lists (Yang et al., 2026a) or spectrum images (Han et al., 2026). *Search and ranking methods* assemble or match candidates instead. They use imitation learning (Jonas, 2019), retrieval (Jin et al., 2026a), multi-shot prediction with candidate ranking (Jin et al., 2026b), evolutionary search (Mirza et al., 2026), an LLM-backed search agent (Morales et al., 2026), or a language model applied directly (Su et al., 2026). Aurich & Schymanski (2026) survey the wider computer-assisted elucidation field.

3 The uniform conditioning bottleneck

We consider two diffusion models, NMR-DiGress and ChefNMR (Xiong et al., 2025). Both encode the spectrum into a token sequence $S \in \mathbb{R}^{L\times d}$ of L tokens of width d. The sequence retains position along the chemical-shift axis and is passed to a stack of generative blocks. Attention pooling collapses S to a single vector z, and the blocks access the spectrum only through a scale γ and shift β computed from z together with the timestep (and, in NMR-DiGress, graph-level statistics): FiLM in NMR-DiGress and adaLN-Zero in ChefNMR apply them to every element alike. Whatever regional structure S carried reaches every component of the molecule in the same form (Figure 1).

We illustrate this limitation on toluene, the molecule of Figure 1. Its ^{1}H spectrum has a sharp methyl singlet near 2.3 ppm, in the benzylic region, integrating to three protons, and an aromatic

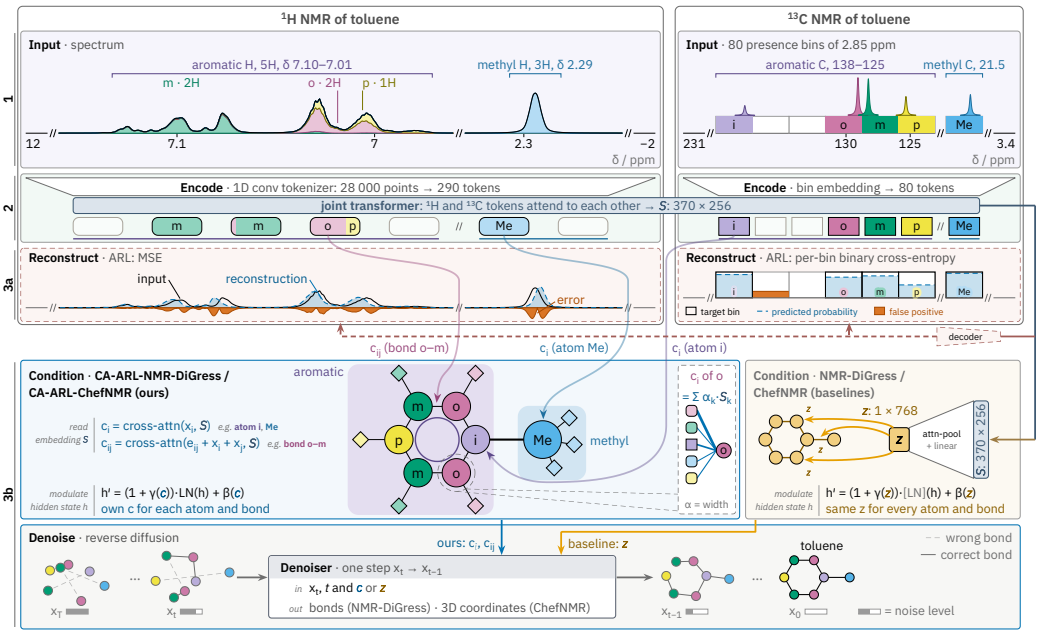


Figure 1: Method overview on toluene. (1) Simulated ^{1}H and ^{13}C spectra, with colours marking atom positions and brackets marking aromatic and methyl regions. (2) Each spectrum is tokenized separately. A joint transformer over all tokens produces the token sequence S. (3a) During training, a decoder reconstructs both spectra from S through the auxiliary reconstruction loss (ARL). (3b) Cross-attention lets every atom and bond query S for its own context. The baselines pool S into one vector z shared by all atoms and bonds. From z they compute a scale γ and a shift β. Reconstructions and attention weights are illustrative. The formulas show one conditioning step. Under pooling, γ and β also depend on the timestep. In NMR-DiGress they also depend on global graph features. ChefNMR has no bonds and conditions atoms only.

multiplet several ppm downfield, integrating to five. A chemist assigns the first to the methyl group and the second to the aromatic ring, and identifying the two regions separately constrains the bond structure. Under pooled conditioning, however, the model receives one vector summarising both, and the bond between the methyl carbon and the ring is modulated by exactly the same γ and β as every aromatic bond. The necessary information may be present in z, but within one conditioning step no mechanism lets a particular bond attend to the part of the spectrum that concerns it.

Within a single conditioning step, this limits what conditioning can express. Let $h_{e,ij}$ denote the hidden representation of the edge between atoms i and j, and $\odot$ the elementwise product. FiLM updates every edge as $(1+\gamma)\odot h_{e,ij}+\beta$, so the difference between the updates applied to two edges is $(1+\gamma)\odot(h_{e,ij}-h_{e,kl})$, which depends on the spectrum only through a common factor. Any differentiation between two edges therefore has to come from the edge representations $h_{e,ij}$ and $h_{e,kl}$ that the backbone had already computed, and at the first conditioning step those depend on the noisy graph and the timestep alone. FiLM can therefore rescale the differences the graph already expresses but cannot introduce a distinction the graph does not.

If routing rather than encoding is the binding constraint, neither a more faithful encoder nor an auxiliary objective that makes the encoder retain spectral detail should help much under pooled conditioning, since both still pass through the one-vector interface. Section 5 tests these predictions alongside the interface change itself.

4 Per-element cross-attention conditioning

To remove the bottleneck of a single pooled vector, we let every atom and every bond query the spectrum directly and derive its own conditioning (Figure 1). The NCF encoder is kept, and only its multi-head attention pooling head (Lee et al., 2019) is removed, so the conditioning reads the

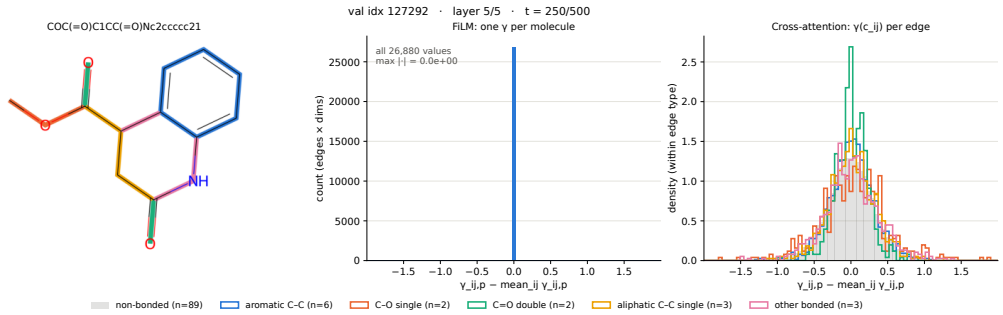


Figure 2: Per-edge modulation for validation molecule 127292 in the last of five layers at diffusion step $t = 250$ of 500. Left: the molecule with its bonds coloured by edge type. Centre: under FiLM every edge receives the same γ, so all 26,880 within-molecule-centred values (105 atom pairs times 256 channels) are zero. Right: under cross-attention, $\gamma(c_{ij})$ differs between edges, shown as within-edge-type densities of the centred values.

full token sequence $S \in \mathbb{R}^{L\times d}$. We fix the model names once. FiLM-NMR-DiGress is our reproduction of NMR-DiGress, while CA-NMR-DiGress replaces its conditioning with per-element cross-attention (CA). The infix ARL marks a model trained with the auxiliary spectral reconstruction loss, as in CA-ARL-NMR-DiGress, and FiLM-ARL-NMR-DiGress denotes the three-query FiLM variant with ARL. ChefNMR has two published sizes: adaLN-ChefNMR-S at 134 M and adaLN-ChefNMR-L at 462 M parameters. CA-ChefNMR-S and CA-ARL-ChefNMR-S add per-atom cross-attention to the small model.

In the baseline, each of the five blocks conditions twice, once on the edge stream and once on the node stream. CA-NMR-DiGress replaces both conditioning steps in every block with cross-attention over S. This removes the FiLM modulation of nodes and edges by DiGress's global feature vector. The encoder, backbone and training setup stay unchanged. For the widened edge tensor $h_{e,ij}$ of edge (i, j) (Appendix B), the two conditioning steps read

$$\text{FiLM:}\quad h'_{e,ij} = \big(1+\gamma_e(z)\big)\odot h_{e,ij} + \beta_e(z),$$
$$\text{CA:}\quad h'_{e,ij} = \big(1+\gamma(c_{ij})\big)\odot \mathrm{LN}(h_{e,ij}) + \beta(c_{ij}), \qquad c_{ij} = \mathrm{MHA}\big(q_{ij},\, SW_K,\, SW_V\big)\, W_O,$$

where z reaches γ_e and β_e through DiGress's global feature vector, which also carries the timestep and graph-level statistics, and MHA is n_{head}-head attention over the L spectral tokens (Vaswani et al., 2017). The edge query $q_{ij} = h_{e,ij}W_{Qe} + x_i W_{Qx} + x_j W_{Qx} + b_Q$ combines the edge with the block-input node features x_i and x_j of both endpoints. The node module has the same form with query $q_i = h_{x,i}W_Q + b_Q$ on $h_{x,i}$, the attention-weighted node value of node i (not the block-input feature x_i), and modulates $h_{x,i}$, so atom representations are conditioned before they feed the next block's edge updates. The two modules share no weights, and γ and β are initialised to zero, so each starts as a LayerNorm of its input. Specified in Appendix B, they add 2.3 M parameters to the 16.6 M baseline (14%).

ChefNMR diffuses atom coordinates and has no bond tensor, so the transfer to it is per-atom only. In adaLN-ChefNMR-S, every atom is modulated through adaLN-Zero by the same pooled spectral vector z. In our port, each atom instead queries S with its own representation, in the same form as the node module above, and derives its own γ and β. Figure 2 shows the resulting per-edge modulation for one validation molecule. Under FiLM, all 105 atom pairs of the molecule receive the same scale γ, so the centred values collapse to zero. Under cross-attention, γ varies from edge to edge, and its distribution differs between edge types, so each bond is modulated by the spectral context it reads rather than by one shared vector. Under FiLM, all 105 atom pairs of the molecule receive the same scale γ, so the centred values collapse to zero. Under cross-attention, γ varies from edge to edge, and its distribution differs between edge types, so each bond is modulated by the spectral context it reads rather than by one shared vector.

To formalise the bottleneck, consider a single conditioning step and let h_i denote the representation of element i. We define $\mathcal{F}_{\text{unif}}$ as the family of uniform FiLM transformations $(1+\gamma(z))\odot h_i + \beta(z)$, where the same conditioning signal z is applied to every element. Let $\mathcal{F}_{\text{CA}}$ denote the family of

transformations realisable when each element instead queries the spectral tokens through cross-attention, taken in the same unnormalised form, whereas the deployed module modulates $\mathrm{LN}(h_i)$.

Lemma 1 (Subsumption). $\mathcal{F}_{unif} \subseteq \mathcal{F}_{CA}$.

Cross-attention can therefore recover any conditioning transformation available to uniform FiLM conditioning within a single conditioning step.

Lemma 2 (Separation). *Under non-degeneracy conditions* $\mathcal{F}_{unif} \subsetneq \mathcal{F}_{CA}$.

Together, the lemmas show that within one realisable conditioning step, element-specific cross-attention expresses every uniform conditioning function and some that uniform conditioning cannot. The deployed modules also add pooling details and timestep inputs (Appendix A). Appendix B gives the module equations, Table 3 the dimensions and Appendix A the proofs and non-degeneracy conditions.

5 Experiments

5.1 Setup

Task and data. Given the ^{1}H and ^{13}C spectra of a molecule and its atom types, the model infers the bonds between those atoms. We use the simulated SpectraBase dataset (Hu et al., 2024) under its released random split by molecule: 114,332 training, 14,270 validation and 14,292 test molecules. The ChefNMR-backbone models are also evaluated on two sets of experimental spectra, SpecTeach (^{1}H and ^{13}C) and NMRShiftDB2 (^{13}C only), restricted to their C/H/O/N molecules (Appendix D.1).

Evaluation. The metric is top-1 exact structure recovery, the fraction of test molecules whose first valid sampled molecule matches the ground truth after RDKit canonicalisation. Because atom types are fixed, exact match compares molecular constitution. Stereochemistry is not scored. Top-k is pass@k over valid samples in sampling order without reranking, and a molecule without a valid sample counts as a miss. Validity is the fraction of generated molecules that RDKit accepts. Each backbone's cohort and sampling are given in its subsection and in Table 1.

5.2 The conditioning interface on the graph backbone

The graph backbone is DiGress (Vignac et al., 2023), a graph transformer with 5 layers, 8 heads and 256 hidden dimensions, run for 500 diffusion steps under a cosine schedule. The formula fixes the atom types, so only the five edge classes are corrupted and recovered. The NCF encoder (Xiong et al., 2025) reads a 28,000-bin ^{1}H spectrum (-2 to 12 ppm) and an 80-dimensional ^{13}C occupancy vector (3.42 to 231.3 ppm) as 370 spectral tokens of width 256 (Table 3). The baseline, FiLM-NMR-DiGress, pools these tokens with one learnable query. Following Xiong et al. (2025), the evaluation excludes the 3,231 test molecules containing aromatic nitrogens, leaving 11,061, and takes the first k valid of 20 samples per molecule without classifier-free guidance. All DiGress runs train with Adam at 10^{-4} under a cosine schedule on four A100 GPUs, with conditioning dropout $p = 0.2$ (Appendix B).

Our FiLM reproduction, FiLM-NMR-DiGress, trained from scratch, reaches 43.26% top-1, against 43.56% published for NMR-DiGress under an undisclosed sampling protocol (Xiong et al., 2025). Under the same setup, CA-NMR-DiGress, also trained from scratch, reaches 52.10%, a gain of 8.84 percentage points for 2.3 M additional parameters (14%). The two are evaluated at comparable training epochs, 2,593 and 2,565. The CA-NMR-DiGress value is the mean of three inference repeats (52.40%, 51.01% and 52.89%), and the reproduction is a single evaluation. Fine-tuning CA-NMR-DiGress with the auxiliary reconstruction loss gives CA-ARL-NMR-DiGress, which reaches 54.38%. The gain is that of the conditioning change as a whole rather than of the edge module alone (Appendix B). Figure 5 places every run against its parameter count.

Chemical validity falls with the gain, from 94.2% for FiLM-NMR-DiGress to 90.4% for CA-NMR-DiGress and 91.9% for CA-ARL-NMR-DiGress. Across the six pooled-conditioning runs with recorded validity in Table 8, validity spans 92.2% to 95.9%. We read the drop as consistent with freer per-edge and per-node bond placement producing invalid molecules that pooled smoothing would have suppressed. Appendix F analyses where each class of atom pair reads the spectrum.

Model	Params (M)	Top-1 (%)	Top-5 (%)	Top-10 (%)	Validity (%)
NMR-DiGress backbone: 11,061 test molecules, 20 samples, 500 steps					
FiLM-NMR-DiGress^{s}	16.6	43.26	61.31	67.35	94.20
CA-NMR-DiGress^{r}	18.9	52.10±0.98	67.87±1.16	72.23±1.13	90.39±0.02
CA-ARL-NMR-DiGress^{r}	18.9	54.38±0.49	70.11±0.67	74.90±0.97	91.85±0.08
Published (Xiong et al., 2025): ChefNMR protocol, different cohort					
Hu et al.†	–	45.24±0.18	62.37±0.08	67.38±0.05	–
NMR-DiGress†	14.4	43.56±0.30	62.47±0.20	68.39±0.35	–
adaLN-ChefNMR-S†	134	69.15±0.08	82.09±0.24	85.30±0.04	–
adaLN-ChefNMR-L†	462	72.04±0.02	85.24±0.10	88.20±0.07	–
ChefNMR backbone, ours: 14,137 test molecules, 10 samples, 50 steps, $\omega = 2$					
CA-ChefNMR-S^{r}	153.0	76.64±0.23	87.52±0.10	89.92±0.16	95.89±0.05
CA-ARL-ChefNMR-S^{r}	154.9	78.29±0.19	87.88±0.09	90.05±0.06	96.72±0.10

Table 1: Exact structure recovery on the SpectraBase test split. Mark ^{s} denotes a single evaluation, which carries no SD. Mark ^{r} denotes the mean of three inference repeats of one checkpoint. Mark †denotes published values (Xiong et al., 2025), reported as a mean of three sampling runs. The ± value is the sample SD over the three repeats (^{r}) or as published (†). Top-k is the first k valid of 20 draws for DiGress, top-k of 10 for our ChefNMR runs, and the first raw candidates for published ChefNMR. Cross-block comparisons use different cohorts and sampling protocols. Per-checkpoint results are in Table 11.

5.3 Transfer to a coordinate-based backbone

ChefNMR (Xiong et al., 2025) diffuses 3D atom coordinates with the atom types given by the formula, so it has no bond tensor. Its published models condition every atom on one pooled spectral vector through adaLN. In CA-ChefNMR-S, per-atom cross-attention replaces that pooled vector, and the model is trained from scratch. Its evaluation keeps aromatic-nitrogen molecules, scoring 14,137 test molecules, and takes top-k over 10 samples (Appendix E). At three checkpoints of one run, labelled 5.6k, 7.2k and 8.4k, top-1 recovery is 75.77%, 76.48% and 76.64%, and CA-ARL-ChefNMR-S reaches 78.29% (Table 11). With 153 M parameters, CA-ChefNMR-S exceeds both published ChefNMR models, the 134 M adaLN-ChefNMR-S at 69.15% and the 462 M adaLN-ChefNMR-L at 72.04%, which were evaluated on a different cohort and sampling pipeline. On one common pipeline, unmasked single-sample top-1 is 76.39% for CA-ChefNMR-S and 69.14% for the released adaLN-ChefNMR-S checkpoint, with CA-ChefNMR-S trained for 31% fewer optimiser steps (Appendix F).

On experimental spectra, CA-ChefNMR-S and CA-ARL-ChefNMR-S match the SpectraBase-trained adaLN-ChefNMR-S in exact recovery, at 42.9% and 42.3% top-1 against 41.2% on SpecTeach and 6.1% to 6.2% for all three on NMRShiftDB2, within the spread over sampling seeds, and generate more valid molecules on both sets (Table 9, Appendix D.1). The released adaLN-ChefNMR-S trained on simulated USPTO spectra recovers 46.6% and 17.9%, so on experimental input the training set moves accuracy more than the conditioning interface does.

5.4 Controls

Conditioning capacity. Pooling capacity recovers little of the gain. Three learnable pooling queries instead of one raise top-1 recovery from 43.26% for FiLM-NMR-DiGress to 44.73%, 1.47 percentage points for 0.13 M parameters, against the 8.84-point gain of cross-attention. The control also changes the encoder, since its three query tokens pass through all four encoder layers (Appendix B).

A separate encoder arm. A ViT spectral encoder reconstructs the input spectrum better than the convolutional encoder but does not recover more structures (Appendices B and C).

The auxiliary loss under each interface. The loss adds a ^{1}H and a ^{13}C spectral reconstruction term to the diffusion objective. Trained from scratch under single-query pooled conditioning, it reaches 40.21%, below the 43.26% of FiLM-NMR-DiGress. Under three-query pooling, fine-tuning

Spectra at inference	Top-1 (%)	Top-10 (%)	Validity (%)
^{1}H and ^{13}C	78.33	90.31	96.70
^{1}H only	68.51	83.16	95.42
^{13}C only	32.20	52.49	91.51
guidance -1	0.20	1.51	80.63

Table 2: ^{1}H carries most of the accuracy of CA-ARL-ChefNMR-S, and ^{13}C still contributes. One inference run was performed per row. The first row is a separate run from those in Table 11. The spectrum is removed through the conditioning-dropout path at drop probability 1. The last row removes neither spectrum and samples at guidance setting -1.

with the loss raises 44.73% to 47.29% (FiLM-ARL-NMR-DiGress). Under cross-attention, CA-ARL-NMR-DiGress reaches 54.38% against 52.10% for CA-NMR-DiGress, and on ChefNMR CA-ARL-ChefNMR-S reaches 78.29% against 76.64% for CA-ChefNMR-S. The positive cases add fine-tuning, while the negative one trains from scratch (Appendix B).

Removing a whole spectrum. Removing one spectrum at inference shows that CA-ARL-ChefNMR-S draws most of its accuracy from ^{1}H but still uses ^{13}C (Table 2). Top-1 recovery falls from 78.33% with both spectra to 68.51% without ^{13}C and to 32.20% without ^{1}H, and sampled at guidance setting -1 the model recovers 0.20% of structures. Each spectrum is removed through the conditioning-dropout path the model saw in training, so every arm is in distribution. Masking chemically defined regions of both spectra damages structures containing the matching chemistry in CA-ChefNMR-S and adaLN-ChefNMR-S. Relative to each model's own recovery, the two models do not differ (Appendix F).

6 Limitations

Chemical validity. Cross-attention lowers chemical validity on NMR-DiGress from 94.2% for our reproduction to 90.4%, so part of the accuracy gain trades against generating fewer valid structures. A validity-aware sampler or constraint is left for future work.

Single training runs. Every DiGress configuration is a single training run, and the CA-NMR-DiGress figure averages three inference repeats of one checkpoint. Two evaluations of one checkpoint differ by 1.62 points, so differences of one to two points indicate a direction only. Multi-seed training would measure training-run variability. The SpectraBase comparisons also use a random split by molecule, which does not test generalisation to unseen scaffolds.

Compute. Cross-attention reads all L spectral tokens for every atom and bond, which costs more per step than one pooled vector. CA-ChefNMR-S was trained for 31% fewer optimiser steps than the published adaLN-ChefNMR-S, and a matched-compute comparison is left for future work.

7 Conclusion

Diffusion models for NMR structure elucidation condition every atom and bond on one pooled spectral vector. We replaced this interface with per-element cross-attention, so that each atom and bond reads the spectral tokens that concern it, and showed that, in an idealised single conditioning step, cross-attention contains pooled modulation and is strictly more expressive. On NMR-DiGress, per-element conditioning lifts our reproduced FiLM baseline from 43.26% to 52.10% with 14% more parameters. On ChefNMR, the per-atom port reaches 76.64% and exceeds both published ChefNMR models, including the three times larger adaLN-ChefNMR-L. Auxiliary reconstruction fine-tuning adds to both, reaching 54.38% and 78.29%. This gain is larger than the published gain from scaling ChefNMR from 134 M to 462 M parameters, and it needs no additional data. The next steps are a scaffold-split evaluation and broader tests on experimental spectra.

REPRODUCIBILITY STATEMENT

Code, configurations and trained checkpoints will be released upon acceptance. We introduce no new data: the benchmark is the simulated SpectraBase set of Hu et al. (2024) with its released train, validation and test split. Appendix B gives the architecture, parameter counts, optimiser, schedule and evaluated epochs, Appendix D the complete results, and Appendix E the evaluation protocol of each backbone and of the published baselines. The cross-attention NMR-DiGress values are means over three inference repeats of one checkpoint.

AI USE STATEMENT

We used large language model assistants in four ways, matching the categories declared on the submission form. For research ideation and execution, they helped discuss design choices and implement, debug and review training, evaluation and analysis code, and checked experimental protocols, numerical results and proofs for errors. For retrieval and discovery, they helped identify related work and format references. They drafted sections of the paper and figure code, and they edited the text for clarity and consistency. The authors directed the research, checked the reported numbers against the underlying run records, revised all generated text, and take full responsibility for the content of this paper.

REFERENCES

Marvin Alberts, Federico Zipoli, and Alain C. Vaucher. Learning the language of NMR: Structure elucidation from NMR spectra using transformer models. In *AI for Accelerated Materials Design (AI4Mat), NeurIPS 2023 Workshop*, 2023. doi: 10.26434/chemrxiv-2023-8wxcz. URL `https://openreview.net/forum?id=akFqQokObE`. Called NMRFormer by Fang et al.

Dagny Aurich and Emma L. Schymanski. Advances and perspectives in computer-assisted structure elucidation: a review. *Digital Discovery*, 5(3):1023–1036, 2026. doi: 10.1039/d5dd00438a.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems 34 (NeurIPS 2021)*, pp. 17981–17993, 2021.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. arXiv:1607.06450, 2016.

Haochen Chen, Qi Huang, Anan Wu, Wenhao Zhang, Jianliang Ye, Jianming Wu, Kai Tan, Xin Lu, and Xin Xu. DiSE: A diffusion probabilistic model for automatic structure elucidation of organic compounds. arXiv:2510.26231, 2025.

Chao Han, Xiaolin Pan, and Yingkai Zhang. End-to-end molecular structure elucidation from multimodal NMR spectra images using vision transformers. *Chemical Science*, 17(26):13072–13082, 2026. doi: 10.1039/d6sc02352e.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023.

Frank Hu, Michael S. Chen, Grant M. Rotskoff, Matthew W. Kanan, and Thomas E. Markland. Accurate and efficient structure elucidation from routine one-dimensional NMR spectra using multitask machine learning. *ACS Central Science*, 10(11):2162–2170, 2024. doi: 10.1021/acscentsci.4c01132. Source of the SpectraBase dataset.

Frank Hu, Jonathan M. Tubb, Dimitris Argyropoulos, Sergey Golotvin, Mikhail Elyashberg, Grant M. Rotskoff, Matthew W. Kanan, and Thomas E. Markland. Pushing the limits of one-dimensional NMR spectroscopy for automated structure elucidation using artificial intelligence. *Journal of Chemical Information and Modeling*, 66(12):6992–7002, 2026. doi: 10.1021/acs.jcim.6c00905. Same author group as Hu et al. (2024); not used as a quantitative baseline here (different top-k metric and atom-count scope, protocol not established as comparable to SpectraBase).

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4198–4205. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.386.

Sarthak Jain and Byron C. Wallace. Attention is not explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 3543–3556. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1357.

Yongqi Jin, Jun-Jie Wang, Fanjie Xu, Xiaohong Ji, Zhifeng Gao, Linfeng Zhang, Guolin Ke, Rong Zhu, and Weinan E. NMR-Solver: automated structure elucidation via large-scale spectral matching and physics-guided fragment optimization. *Nature Communications*, 17(1):4740, 2026a. doi: 10.1038/s41467-026-71315-0.

Zheqi Jin, Grace Armitage, Richard Cox, Ben Honoré, Mohammad Golbabaee, and Craig Butts. Inverse-IMPRESSION: A graph-based platform for molecular structure elucidation from experimental NMR spectroscopic properties. arXiv:2607.09978, 2026b.

Eric Jonas. Deep imitation learning for molecular inverse problems. In *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, pp. 4991–5001, 2019.

Hojung Jung, Rodrigo Hormazabal, Jaehyeong Jo, Youngrok Park, Kyunggeun Roh, Se-Young Yun, Sehui Han, and Dae-Woong Jeong. MolHIT: Advancing molecular-graph generation with hierarchical discrete diffusion models. arXiv:2602.17602, 2026. Preprint.

Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7057–7075. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.574.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning (ICML)*, pp. 3744–3753, 2019.

Adrian Mirza, Luc Patiny, and Kevin Maik Jablonka. End-to-end multimodal structure elucidation from raw spectra combining contrastive learning and evolutionary algorithms. *Nature Communications*, 17(1):5013, 2026. doi: 10.1038/s41467-026-73846-y.

Irina Espejo Morales, Damon Hinz, Marvin Alberts, Geraud Krawezik, Haewon Jeong, and Shirley Ho. NMR elucidation as an agentic search problem, not a modeling problem. arXiv:2607.19406, 2026.

Federico Ottomano, Yingzhen Li, and Alex M. Ganose. NMIRacle: Multi-modal generative molecular elucidation from IR and NMR spectra. arXiv:2512.19733, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182, 2023. doi: 10.1109/ICCV51070.2023.00387. Introduces DiT and adaLN-Zero.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 3942–3951, 2018. doi: 10.1609/aaai.v32i1.11671.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685, 2022. doi: 10.1109/CVPR52688.2022.01042. Cross-attention conditions each spatial location on text tokens individually, replacing a single pooled embedding.

Yunyue Su, Jiahui Chen, Zao Jiang, Zhenyi Zhong, Liang Wang, Qiang Liu, and Zhaoxiang Zhang. SpectraLLM: Uncovering the ability of llms for molecular structure elucidation from multi-spectral data. In *International Conference on Learning Representations (ICLR)*, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30 (NeurIPS 2017)*, pp. 5998–6008, 2017.

Clément Vignac, Igor Krawczuk, Antoine Siraudin, Bohan Wang, Volkan Cevher, and Pascal Frossard. DiGress: Discrete denoising diffusion for graph generation. In *International Conference on Learning Representations (ICLR)*, 2023.

Liang Wang, Yu Rong, Tingyang Xu, Zhenyi Zhong, Zhiyuan Liu, Pengju Wang, Deli Zhao, Qiang Liu, Shu Wu, Liang Wang, and Yang Zhang. DiffSpectra: Molecular structure elucidation from spectra using diffusion models. arXiv:2507.06853, 2025.

Ziyu Xiong, Yichi Zhang, Foyez Alauddin, Chu Xin Cheng, Joon Soo An, Mohammad R. Seyedsayamdost, and Ellen D. Zhong. Atomic diffusion models for small molecule structure elucidation from NMR spectra. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025)*, pp. 128593–128629, 2025. doi: 10.52202/085713-3868. Introduces ChefNMR and the NMR-DiGress baseline.

Xi Xue, Hanyu Sun, Jingying Sun, Luc Patiny, Xiangying Liu, Kai Chen, Jingjie Yan, Liangning Li, Xue Liu, Shu Xu, Dongming Zhang, Yafeng Deng, Yingda Zang, Yaling Gong, Jie Ma, and Xiaojian Wang. NMRMind: A transformer-based model enabling the elucidation from multidimensional NMR to structures. *Analytical Chemistry*, 97(41):22603–22614, 2025. doi: 10.1021/acs.analchem.5c03783.

Liujia Yang, Zhuo Yang, Jiaqing Xie, Yubin Wang, Ben Gao, Xingjian Wei, Jiaxing Sun, Jiang Wu, Conghui He, Yuqiang Li, and Qinying Gu. NMRTrans: Structure elucidation from experimental NMR spectra via set transformers. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pp. 12621–12632. ACM, 2026a. doi: 10.1145/3770855.3818935.

Qingsong Yang, Binglan Wu, Xuwei Liu, Bo Chen, Wei Li, Gen Long, Xin Chen, and Mingjun Xiao. DiffNMR: diffusion models for nuclear magnetic resonance spectra elucidation. *Materials Futures*, 5(1):015601, 2026b. doi: 10.1088/2752-5724/ae301f.

Appendix

A PROOFS

Both results are stated for a single conditioning step applied to each element of the diffusion process, as set out in Section 4. Write $S \in \mathbb{R}^{L\times d}$ for the spectral token sequence, and h_i for the element representation entering the conditioning step. In the graph backbone the elements are candidate bonds and atoms, while in the coordinate backbone they are the atoms only. Write $z_{\text{pre}} \in \mathbb{R}^d$ for the vector that the baseline's multi-head attention pooling head produces from S at the token width d, using a learned query q_{CLS} and projections $W_Q^{\text{MAP}}, W_K^{\text{MAP}}, W_V^{\text{MAP}}$ and W_O^{MAP}, and z for the conditioning vector that uniform conditioning applies to every element. The results of this appendix hold for an idealised matched-head family. In this family the pooling head attends over the tokens of S alone, with q_{CLS} serving only as the query, and z is an affine image of z_{pre} with no normalisation or dropout in between. The cross-attention module has the same head count and head width as the pooling head and applies the same φ as the uniform step it is compared with. The deployed baseline differs from this family. Its pooling head also attends to the CLS token itself, and it passes z_{pre} through LayerNorm, dropout and a projection to the 768-wide conditioning vector $z \in \mathbb{R}^{768}$ (Appendix B). The results below therefore do not state exact equivalence with the deployed module. Uniform conditioning computes z and applies it to every element alike, so its parameters γ and β depend on S but not on i. Cross-attention computes a per-element context c_i from a query built from h_i, so its γ and β depend on the element as well. In the graph backbone, an edge query also carries features of both endpoints.

One convention, fixed here and matching Appendix B, is used throughout: a modulation step is written $h_i' = (1+\gamma)\odot\varphi(h_i)+\beta$, where φ is the identity under FiLM and layer normalisation under adaLN-Zero, and γ and β are produced by affine maps with free biases. Setting $\gamma = \beta = 0$ returns $\varphi(h_i)$, and on a single coordinate p the step is the affine map $y = (1+\gamma_p)\,x + \beta_p$ of $x = \varphi(h_i)_p$.

Lemma 1 (Subsumption), restated. $\mathcal{F}_{\text{unif}} \subseteq \mathcal{F}_{\text{CA}}$.

Proof. It suffices to construct, for any uniform conditioning function, a cross-attention module realising it. Take every query projection to be zero and set the query bias to the pooling query of the baseline encoder, $b_Q = q_{\text{CLS}} W_Q^{\text{MAP}}$. Every element then produces the same query, so the attention weights α_i are identical across elements regardless of architecture and the module collapses into uniform conditioning. Set the key and value projections to those of the pooling head, $W_K = W_K^{\text{MAP}}$ and $W_V = W_V^{\text{MAP}}$. Set the output projection to that of the pooling head, $W_O = W_O^{\text{MAP}}$, with zero biases on the key, value and output projections. Every element then receives the same context, $c_i = z_{\text{pre}}$. In this family z is an affine image of z_{pre}, so composing that map into the affine maps γ and β keeps them affine. The resulting module applies the same affine modulation to every element, which is exactly the uniform conditioning function, so $\mathcal{F}_{\text{unif}} \subseteq \mathcal{F}_{\text{CA}}$. □

The construction copies the pooling head head by head, so it uses the shared head count n_{head} and head width d_k that Appendix B records on each side. This match is a hypothesis of the construction. For one fixed S the vector z_{pre} is a single vector, but the construction must reproduce it for every S, and a single head does not suffice for this in general. Take S to consist of two tokens u and $-u$, with the remaining tokens zero, and rescale u. A single head with a fixed query weights the values by coefficients summing to one, so its output stays on one fixed line, whereas two pooling heads with different key and value projections can trace a curve that no line contains. Where the baseline adds the timestep to z before modulating, the same construction applies within one conditioning step: the cross-attention term is constant and equal to z_{pre}, and at a fixed timestep the added contribution is a constant that the free biases of γ and β absorb.

Three implementation properties carry the construction (Appendix B). First, the query projection has a bias term. Second, the pooling head and cross-attention module share head count n_{head} and head width d_k. Third, the map from z_{pre} to z is affine. This holds in the idealised family, while the deployed baseline applies LayerNorm before its projection. The first is load-bearing: with the query projections zeroed and no bias the query is identically zero, the attention weights are uniform, and the construction recovers mean pooling instead of the learned attention pooling the baseline uses.

Lemma 3. *Under uniform conditioning, the difference between the updates applied to two elements is a rescaling of the difference of their incoming representations, by a factor common to both:* $\Delta_i^{unif} - \Delta_k^{unif} = \gamma(z) \odot \big(\varphi(h_i) - \varphi(h_k)\big)$.

Proof. Write the uniform update as $\Delta_i^{\text{unif}} = h_i' - \varphi(h_i) = \gamma(z) \odot \varphi(h_i) + \beta(z)$. Since $\gamma(z)$ and $\beta(z)$ do not depend on the element, subtracting the corresponding expression for k cancels $\beta(z)$ and leaves the stated identity. □

Lemma 3 says that under uniform conditioning the spectrum can rescale a difference the incoming representations already express, through a factor common to both elements, but cannot introduce a distinction they do not carry. The lemma uses only that γ and β carry no element index, so it holds for either conditioning form and for either backbone.

Write p for a coordinate index, to keep it apart from the token width d, and d_c for the width of the context c_i.

Lemma 4. *Let* $h_1, \dots, h_m$ *be elements whose queries, restricted to a single attention head, are* linearly independent*, with* $m \le \min(d, d_k, d_c)$*. Fix any coordinate* p *and any real values* $g_1, \dots, g_m$*. There exist spectral tokens, projections and affine maps* γ, β *with* $\gamma_p(c_r) = g_r$ *for every* r*, and with the remaining coordinates of* γ*, and all of* β*, free to be chosen independently.*

Proof. The deployed module splits queries, keys and values across n_{head} heads of width d_k (Appendix B), and the softmax is taken per head. Carry the construction inside one head and set the value projections of the others to zero, so that only the chosen head contributes to the context. The queries in the statement are the components of q_r in that head. Take m linearly independent spectral tokens $S = [s_1, \dots, s_m]$, which is possible because $m \le d$. Set the remaining $L - m$ tokens ($m \le 32 < L = 370$) to zero. Their logits are zero, and their attention weights vanish as a grows. Because the q_r are linearly independent, the matrix Q with rows q_r has rank m, so a key projection W_K exists with $Q(SW_K)^\top = aI_m$ for any a. Linear independence is what this construction needs: $\operatorname{rank}(aI_m) = m$ forces $\operatorname{rank}(Q) = m$, so distinct queries alone would not let the logits be driven to the identity. Taking $a \gg 0$ drives the attention weights to the one-hot vectors $\alpha_r \to u_r$. The contexts then tend to $c_r \to s_r W_V W_O$, which are linearly independent for generic W_V and W_O. Independence is an open condition, so it also holds at large finite a. Writing $\gamma_p(c) = b_\gamma + w_\gamma^\top c$ with a free bias, the requirement $\gamma_p(c_r) = g_r$ for all r becomes the system $w_\gamma^\top c_r + b_\gamma = g_r$ of m linear equations in the $d_c + 1$ unknowns (w_γ, b_γ), whose coefficient rows $(c_r, 1)$ are linearly independent because the contexts c_r are, so a solution exists. The same system with any other right-hand side is solvable for the same reason, so the other output coordinates of γ may be set to any prescribed values, and all of β is unconstrained. □

Lemma 4 is the counterpart to Lemma 3. Cross-attention can give each element its own modulation, chosen independently of the others, which is exactly what a single shared affine map cannot do.

Lemma 2 (Separation), restated. Under non-degeneracy conditions $\mathcal{F}_{\text{unif}} \subsetneq \mathcal{F}_{\text{CA}}$.

The following non-degeneracy conditions are required: Let A, B and C be three elements whose incoming representations are *pairwise distinct* in some coordinate p of $\varphi(h)$, with $\varphi(h_B)_p \neq 0$, and whose queries are linearly independent within one attention head.

Proof. By Lemma 4 applied with $m = 3$ at coordinate p, cross-attention can assign $\gamma_p(c_A) = 1$, $\gamma_p(c_B) = -\frac{1}{2}$ and $\gamma_p(c_C) = 1$. The lemma leaves β free, so take $\beta \equiv 0$, and let $f^\star$ be the conditioning function this module realises, which puts $f^\star \in \mathcal{F}_{\text{CA}}$.

On coordinate p this function maps

$$f^\star(h_A)_p = 2\,\varphi(h_A)_p, \qquad f^\star(h_B)_p = \tfrac{1}{2}\,\varphi(h_B)_p, \qquad f^\star(h_C)_p = 2\,\varphi(h_C)_p. \tag{1}$$

Uniform conditioning applies one affine map to every element, so on coordinate p it acts as $y = (1+\gamma_p)\,x + \beta_p$ with γ_p and β_p shared by every element, and realising these three values would require the three points $(\varphi(h_A)_p, 2\varphi(h_A)_p)$, $(\varphi(h_B)_p, \frac{1}{2}\varphi(h_B)_p)$ and $(\varphi(h_C)_p, 2\varphi(h_C)_p)$ to be collinear. Since $\varphi(h_A)_p \neq \varphi(h_C)_p$, the unique line through the first and third points is $y = 2x$. The second

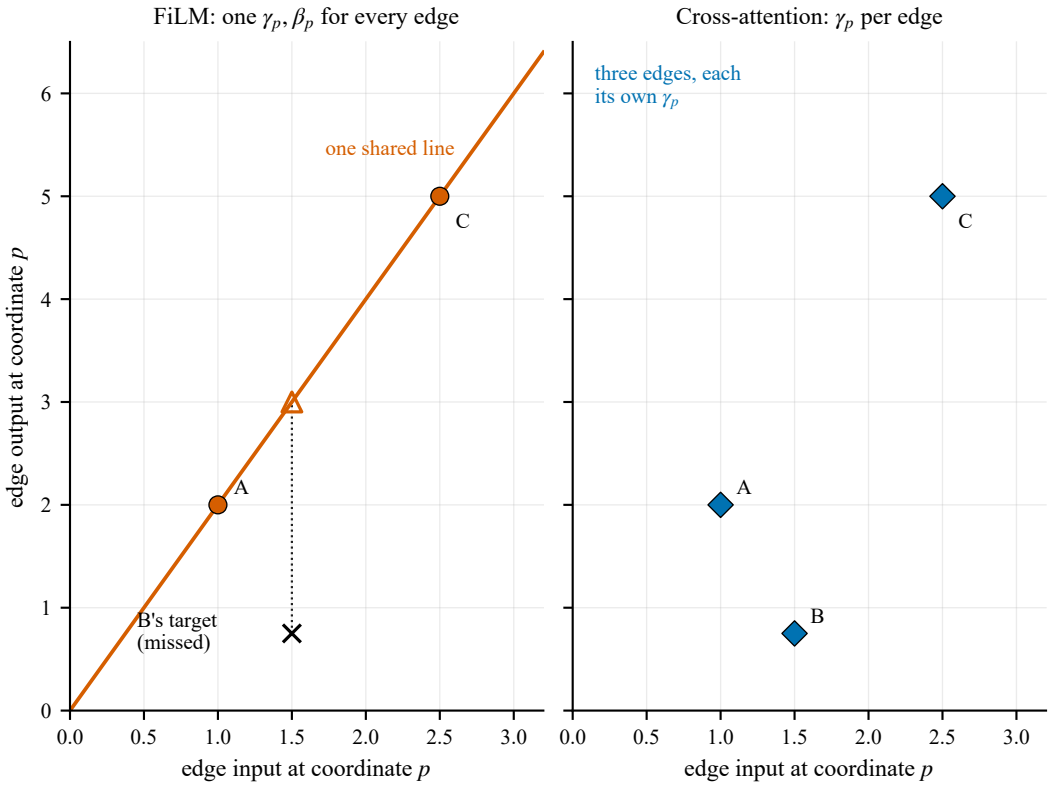


Figure 3: The construction above, at one coordinate p, for three elements A, B, C with $\varphi(h_A)_p = 1$, $\varphi(h_B)_p = 1.5$, $\varphi(h_C)_p = 2.5$ and target $f^\star(h_A)_p = 2\varphi(h_A)_p$, $f^\star(h_B)_p = \frac{1}{2}\varphi(h_B)_p$, $f^\star(h_C)_p = 2\varphi(h_C)_p$. Uniform conditioning (left) applies one shared affine map to every element: the line through A and C is $y = 2x$, and B's target does not lie on it. The cross marks where it would need to be, and the dotted segment shows the miss. Cross-attention (right) assigns each element its own γ_p and hits all three discrete points exactly, as Lemma 4 proves for three elements. This is not a claim about the function between the points, and it does not depend on the query-bias property Lemma 1 needs. The input values are illustrative, chosen only to satisfy the hypothesis (pairwise distinct, $\varphi(h_B)_p \neq 0$), not measured or fit to data.

point lies on that line only if $\frac{1}{2}\varphi(h_B)_p = 2\varphi(h_B)_p$, that is only if $\varphi(h_B)_p = 0$, which the hypothesis excludes. The three points are therefore not collinear, so no uniform conditioning function can realise these three values, and $f^\star \notin \mathcal{F}_{\text{unif}}$. □

The separation lemma requires linearly independent queries rather than merely distinct ones, since distinct queries may still satisfy a linear relation such as $q_3 = q_1 + q_2$. In ChefNMR, representations depend on continuously sampled diffusion noise, so such exact coincidences are non-generic. Likewise, the condition $\varphi(h_B)_p \neq 0$ excludes a degenerate case in the separating construction. We state both explicitly as non-degeneracy assumptions of the lemma.

The construction shown in Figure 3 carries the same caveat one step further. Its key, value and output projections are built from the queries of three particular elements, and its spectral tokens are chosen to suit them rather than produced by the encoder from a measured spectrum. A trained model instead shares one set of projections across every molecule and reads whatever tokens the encoder emits, so Lemma 2 establishes what the conditioning family can express and not that a trained model routes this way.

Scope. The lemmas of this appendix are conditional on the incoming representations h_i of one conditioning step, held fixed, and hold for either backbone. Lemma 3 holds whether or not these fixed h_i already carry spectral information. Its reading that uniform conditioning cannot introduce

new differentiation between elements concerns only the difference the step adds, $\gamma(z) \odot \big(\varphi(h_i) - \varphi(h_k)\big)$, and not the difference $\varphi(h_i) - \varphi(h_k)$ the step receives. The lemmas do not describe how the incoming representations arise, so they make no statement across layers or across steps of the diffusion process.

B ARCHITECTURE DETAILS

The backbone is a DiGress graph transformer with $n_{\text{layers}} = 5$ NodeEdgeBlock layers, DiGress's transformer block that updates node, edge and global features jointly within one layer, carrying a node stream, an edge stream and a global feature stream. The spectral encoder emits a sequence of L tokens $S \in \mathbb{R}^{L \times d}$, which the conditioning modules read through the key and value projections W_K and W_V. The baseline instead pools S to a single vector at width d with one attention layer and projects it up to a 768-wide conditioning vector. The three-query control of Section 5 prepends three learnable class tokens to the encoder input. They take part in all four encoder transformer layers. Their three 256-wide outputs are flattened to 768 and passed through LayerNorm and a 768-to-768 projection. The final conditioning width is unchanged, but the control changes the encoder as well as the pooling head. CA-NMR-DiGress conditions through the 256-wide tokens directly. The pooler splits its attention across the same n_{head} heads of width d_k as the cross-attention modules, 8×32 as listed in Table 3. Its projection to the 768-wide conditioning vector is affine, $\mathbb{R}^{256} \to \mathbb{R}^{768}$ with a bias term.

The baseline applies two FiLM conditioning steps per block, one modulating the edge stream and one modulating the node stream. CA-NMR-DiGress replaces both with cross-attention modules that read from S, an edge module and a node module, and both are applied at every layer. This is the mechanism the paper describes and evaluates throughout.

Conditioning is applied after the self-attention sublayer has widened the edge stream to d_x channels, not to the raw edge feature. That widened tensor $h_e \in \mathbb{R}^{N \times N \times d_x}$ holds the per-head attention scores, scaled and offset by the incoming edge features and concatenated across heads. Cross-attention reads h_e, and a linear map $\mathbb{R}^{d_x} \to \mathbb{R}^{d_e}$ returns to the edge width afterwards. On the node side the corresponding tensor is $h_x \in \mathbb{R}^{N \times d_x}$, the attention-weighted node values.

Edge cross-attention module. The query for edge (i, j) combines the widened edge feature with the features of both endpoints,

$$q_{ij} = h_{e,ij} W_{Qe} + x_i W_{Qx} + x_j W_{Qx} + b_Q \in \mathbb{R}^{d_x}, \qquad W_{Qe}, W_{Qx} \in \mathbb{R}^{d_x \times d_x},\ b_Q \in \mathbb{R}^{d_x}, \quad (2)$$

Applying one matrix W_{Qx} to both endpoint features is part of the implementation. The DiGress backbone also symmetrises the edge tensor with its transpose at its input and output projections, `(new_E + new_E.transpose(1,2))/2`. This final-output operation guarantees symmetric edge predictions. We do not use it to claim that all internal endpoint-aware queries are equal. The spectral tokens supply keys and values through $K = SW_K$ and $V = SW_V$, with $W_K, W_V \in \mathbb{R}^{d \times d_x}$. All three projections are split across n_{head} heads of width d_k. Each head attends over the L tokens, and the heads are concatenated and projected,

$$c_{ij} = \mathrm{Concat}_{h=1}^{n_{\text{head}}} \left[\mathrm{softmax}\left(\frac{q_{ij}^{(h)} (K^{(h)})^\top}{\sqrt{d_k}} \right) V^{(h)} \right] W_O, \qquad W_O \in \mathbb{R}^{d_x \times d_x}. \quad (3)$$

For the realisable family used in Appendix A, the context then conditions the widened edge tensor h_e in the same functional form as FiLM,

$$h'_{e,ij} = \big(1 + \gamma(c_{ij})\big) \odot h_{e,ij} + \beta(c_{ij}), \quad (4)$$

where $\gamma, \beta : \mathbb{R}^{d_x} \to \mathbb{R}^{d_x}$ are affine maps. The deployed path applies the modulation to $\mathrm{LayerNorm}(h_{e,ij})$ with γ and β initialised to zero, an adaLN-style form that starts as a LayerNorm of its input, $\mathrm{LayerNorm}(h_{e,ij})$, whereas the FiLM baseline modulates the unnormalised stream. The equation above is the realisable abstraction used for the lemma, not an exact specification of the normalised module. No reported run uses a global adaLN variant of the conditioning. The source revision behind each reported run has not been independently verified.

Node cross-attention module. The node analogue differs only in its query. Node i projects its own attention-weighted value alone, $q_i = h_{x,i}W_Q + b_Q$ with $W_Q \in \mathbb{R}^{d_x \times d_x}$ and $b_Q \in \mathbb{R}^{d_x}$, and reads the same spectral tokens through its own key and value projections. Multi-head attention over S followed by W_O returns a context $c_i \in \mathbb{R}^{d_x}$, and $h_{x,i}$ is updated by the same affine rule, applied after LayerNorm in the deployed path,

$$h'_{x,i} = \big(1 + \gamma(c_i)\big) \odot h_{x,i} + \beta(c_i). \tag{5}$$

Neither module shares weights with the other, although their computation graphs have the same shape. They differ in the query, N^2 endpoint-aware edge queries against N node queries, and in the edge module's projection back to d_e, which the node path has no counterpart for. At each of the n_{layers} blocks, a bond or an atom can therefore draw a different spectral context, rather than receiving a summary computed once for the molecule as a whole.

Table 3: Dimensions used by the CA-NMR-DiGress cross-attention modules. The number of spectral tokens is $L = 370$ (290 ^{1}H + 80 ^{13}C).

Symbol	Meaning	Value
d_x	node and internal feature width	256
d_e	edge feature width	64
d_y	global feature width	256
d	spectral token width	256
n_{head}	attention heads	8
d_k	head dimension, d_x/n_{head}	32
n_{layers}	graph transformer blocks	5
L	number of spectral tokens	370

Evaluated checkpoints lie between 2,565 and 4,175 training epochs (NCF 2,565–4,079, ViT 3,464–4,175). Full runs use Adam at 1×10^{-4} across four A100s with DDP (cosine annealing, 1,000-step linear warmup, gradient clipping at norm 1.0), at batch size 512, except the 38.12% and 40.89% runs, which resume at the full learning rate with the auxiliary loss at batch size 256. The two runs marked *Res* in Table 8 resume at 1×10^{-5}, at batch size 256 for the CA-ARL-NMR-DiGress run and 512 for the 47.29% run. Three further runs resume at the full learning rate, as listed in the table footnote. The auxiliary loss adds two spectral reconstruction terms to the diffusion objective: a mean squared error on the binned ^{1}H spectrum and a binary cross-entropy on the ^{13}C occupancy vector, weighted 0.0992 and 0.0008 against 0.9 for the diffusion term.

Why single runs and fine-tuning rather than matched from-scratch pairs. Time and compute availability, four A100s shared across configurations rather than a dedicated allocation, ruled out both a multi-seed average for every configuration and a from-scratch run of the auxiliary loss under three-query pooling and cross-attention. Those arms therefore resume from the corresponding trained checkpoint rather than starting fresh, which is why the auxiliary-loss comparisons cross training regimes. Table 8 labels each row by its Full or Res training regime.

What the training budget confounds. The budget confound does not reach the same-protocol comparison: FiLM-NMR-DiGress and CA-NMR-DiGress are both from-scratch runs at comparable evaluated epochs, 2,593 and 2,565, and the baseline figure is its best checkpoint while CA-NMR-DiGress's is its last, an asymmetry that favours the baseline. It does reach the auxiliary-loss numbers. Both cases where the auxiliary-loss model scores higher are Full-to-fine-tuned transitions, 44.73% to 47.29% under three-query pooling and CA-NMR-DiGress to CA-ARL-NMR-DiGress (54.38%) under cross-attention, while the case where it scores lower, 43.26% to 40.21%, compares two models trained from scratch. We have no from-scratch run with the auxiliary loss under either three-query pooling or cross-attention, so neither gain can be separated from the additional fine-tuning compute. Under pooled conditioning, the auxiliary-loss configuration underperforms the baseline whether trained from scratch or resumed from a checkpoint, at 40.21% and 38.12%. Extra compute does not rescue it. No pair in the table isolates the training regime, however. The 38.12% run resumes at the full learning rate rather than the reduced one, and carries a ^{13}C auxiliary weight ten times that of every other auxiliary-loss run, so its 2.09 percentage point gap against 40.21% confounds

initialisation with the auxiliary weighting itself. Every configuration is a single run, so the two positive auxiliary-loss effects are observational and not results controlled for training regime.

Single runs, evaluation movement, and a separate encoder arm. Every DiGress configuration reported here is a single training run, not a multi-seed average. Compute capacity ruled out multi-seed runs and from-scratch auxiliary-loss runs under three-query pooling and cross-attention. We also did not run a conv-stem ViT arm, which would hold depth fixed and vary only the ^{1}H tokenizer. That arm would isolate tokenization within the ViT family without decomposing the gap to the convolutional encoder. Evaluation itself moves results: two evaluations of the same FiLM-NMR-DiGress checkpoint, with 10 and 20 draws per molecule, give 41.63% and 43.26% top-1, a difference of 1.62 points of which the smaller draw budget explains at most about 0.18 through additional molecules without a valid draw. This is one contrast between two unpaired evaluations, not an interval, so it sets a scale for comparison rather than a bound on evaluation noise. The 1.47-point capacity-control change, the 0.30-point reproduction gap and the 0.09-point difference between the 40.98% and 40.89% ViT pair are all smaller than this 1.62-point difference. The auxiliary-loss changes above, 2.1 to 3.1 points, exceed it, as does the 8.84-point interface gain over our reproduction. The capacity control is descriptive and does not identify how much of that gain comes from added conditioning capacity. The three-query control carries 16.7 M parameters, against 16.6 M for the single-query baseline and 18.9 M for CA-NMR-DiGress, so both changes in that comparison are priced. The ViT encoder arm asks whether a more faithful spectral encoder rescues pooled conditioning. It does not, reaching 36.68% against 43.26%. The ViT FiLM arm carries 17.8 M parameters, more than the 16.6 M NCF FiLM baseline it loses to, so the shortfall is not explained by fewer parameters. Replacing its FiLM modulations with cross-attention brings the DiGress ViT arm to 20.6 M, the largest DiGress configuration reported here.

Higher reconstruction fidelity does not rescue pooled conditioning. The ViT arm uses a second encoder, not the convolutional encoder with a different front end. Its ^{1}H tokenizer linearly projects patches of width 200, while the convolutional encoder uses two 1D convolutions and max-pooling. It stacks six transformer blocks instead of four, at matched width and head count. It pools through a class token and a linear head rather than attention over a learnable query. It also resumes from a prior checkpoint, while the 43.26% baseline trains from scratch, and is evaluated after more training epochs, 4,175 against 2,593. The fidelity comparison therefore separates none of these differences from one another. The last two favour the ViT arm and it still loses, which is why we read the comparison as higher fidelity bringing no gain rather than as evidence about the tokenizer. These factors change together in every run we have, and we do not claim the list is exhaustive, so the comparison attributes nothing to any single difference and the supported reading is that higher reconstruction fidelity does not help in our setup, not that it actively harms. ChefNMR's encoder ablation addresses that narrower question, finding a convolutional tokenizer outperforming a patch tokenizer on 1D spectra (Xiong et al., 2025). It agrees with us in direction, but it is measured on a different backbone and generation approach, so only the direction carries over, and it does not license reading our own gap as a tokenizer effect. The DiGress results use one backbone on one dataset with 114,332 training molecules. The ChefNMR-backbone results extend the test to a coordinate-based model, using a test cohort and sampling protocol that differ from the published ChefNMR evaluation. Our FiLM baseline also uses the affine form throughout rather than the adaLN-Zero variant our codebase supports and ChefNMR itself uses, so part of the gap to ChefNMR's reported accuracy may trace to conditioning form rather than to the routing axis this paper tests.

The two backbones use different protocols. The ChefNMR-backbone rows score 14,137 test molecules, aromatic-nitrogen molecules included, with 10 samples, 50 sampling steps and classifier-free guidance. The DiGress rows score 11,061 with 20 samples, 500 steps and no guidance. Accuracies are therefore compared within a backbone, not across the two.

The controls cover the graph backbone only. The capacity, three-query and encoder controls are all DiGress runs, with no counterpart on the ChefNMR backbone. The auxiliary-loss gains on DiGress include the additional fine-tuning compute discussed above.

What the interface gain bundles. CA-NMR-DiGress replaces both FiLM modulations at once, the edge and the node one, so the gain is not attributable to the edge module alone. The replacement

also changes several things together: per-node and per-edge routing, modulation computed by attention over the spectral tokens, LayerNorm with zero-initialised modulation, the loss of global-feature modulation, and 2.3 M additional parameters. No run separates these components.

C THE RECONSTRUCTION PROBE

Probe design. The reconstruction probe measures how much spectral detail survives in a frozen encoder representation. A decoder is trained on top of the frozen embedding to measure achievable reconstruction fidelity under this probe. Without an optimality argument, the result is not an upper bound on all information recoverable by the conditioning pathway. Two decoder families are used, a convolutional stack and a transformer stack, and the probe is run in three target configurations: the ^{1}H spectrum alone, the ^{1}H spectrum and the 80-dimensional ^{13}C occupancy vector together, and the ^{13}C vector alone. The probe carries its own input resolution: its ^{1}H spectra are binned to 10,000 bins, distinct from the 28,000-bin spectra fed to the generation models of Section 5.

Fidelity against recovery. The probe compares encoder architectures, not training objectives. No row here is trained with the auxiliary reconstruction loss, so the probe does not measure how that loss changes reconstruction fidelity. Table 4 reports the ^{1}H configuration and carries that evidence: a ViT encoder trained from scratch reconstructs the input spectrum far better than the convolutional encoders, reaching a Pearson correlation of 0.909 against 0.675 for a scratch-trained ConvFormer of the architecture the generation runs use, while recovering no more structures end to end (Table 8). Reconstruction fidelity and structure recovery therefore come apart under a fixed pooled interface. This probe measures how faithfully the spectrum is encoded, and Section 5 measures what the interface then does with it. The two are coupled rather than interchangeable, so a gain in the first does not transfer to the second on its own.

Regional, joint and ^{13}C-only configurations. Table 5 breaks the same ^{1}H reconstruction down by chemical-shift region, and shows where each encoder spends its capacity. Table 6 reports the joint configuration and Table 7 the ^{13}C-only one. Neither is diagnostic of encoder quality on its own: the ^{13}C occupancy target is far easier than the 10,000-bin ^{1}H spectrum, and when it is the sole target the scratch-trained models saturate it. Under the joint objective the ^{13}C F1 of the scratch-trained ConvFormer rows falls to 0.302 and 0.441 while the ViT rows still saturate (Table 6). The ^{13}C channel therefore orders the encoders differently from the ^{1}H channel, and Table 4 remains the evidence on encoder quality.

An auxiliary-loss reconstruction. Figure 4 shows a reconstruction from the auxiliary-loss arm.

Caveats. Three caveats bound this reading. The two encoders compared here differ in tokenization, in transformer depth, six blocks against four, and in further respects Appendix B lists, so the supported claim is that higher fidelity brings no recovery gain under our training budget, not that it harms, and this comparison separates none of those differences from one another. The probe measures the frozen representation, so it reports performance under this decoder probe rather than an information-theoretic bound on what the conditioning pathway could carry. And the probe's 10,000-bin ^{1}H resolution differs from the 28,000-bin spectrum the generation encoders consume, so the 0.909-versus-0.675 fidelity ranking is measured under different conditions than the recovery numbers it is paired against, and need not carry over to the generation encoders' own resolution.

D COMPLETE ABLATION TABLE

Table 8 gives all ten of our DiGress runs, including the single-query auxiliary-loss run at 38.12%, which differs from the 40.21% row in initialisation and in auxiliary weighting rather than being a second seed of it (Appendix B). All rows carry the top-5, top-10 and maximum-Tanimoto columns omitted from the main figures.

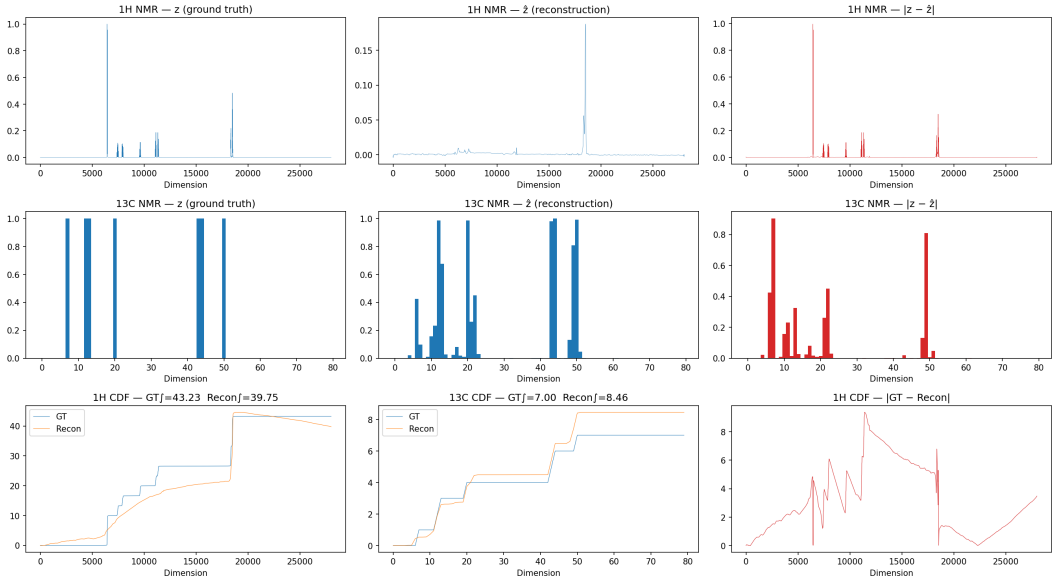


Figure 4: A reconstruction from the auxiliary-loss training arm (ViT encoder), reproduced as the model's raw raster output. This arm is trained with the auxiliary spectral reconstruction loss, unlike the frozen-encoder probe of Tables 4 to 7, which is trained without it. The example is a qualitative companion to those tables and supports no quantitative claim on its own.

Table 4: Reconstruction probe on ^{1}H spectra, measuring how much spectral detail survives in each encoder's frozen representation. The decoder is a convolutional or a transformer stack, not a linear map, so these scores are decoder-dependent probe results rather than information-theoretic upper bounds or linear-probe scores. *Dim 95* is the number of principal components needed to explain 95% of the variance across all embeddings, so it measures how many dimensions the encoder actively uses. *Collapse* is the mean pairwise cosine similarity across a sample of embedding pairs: values near 1 indicate representations that have collapsed together, values near 0 well-spread ones. *MAE* is the mean absolute error in *bin position* on the 10,000-bin grid between matched predicted and true peaks, so it is measured in bins rather than in ppm. The mean predictor is a constant baseline that ignores its input. *ConvFormer (paper)* is the checkpoint released with the source paper. *ConvFormer (scratch)* and *ViT (scratch)* are retrained from scratch here, as the generation runs are. Every row naming a scratch-trained encoder trains its own instance, so encoder statistics such as *Dim 95* and *Collapse* can differ between rows sharing one architecture. Higher is better for Pearson, cosine and peak F1. Lower is better for MAE and collapse. Entries marked *n/a* do not apply because a constant predictor has neither a decoder nor a learned embedding. Entries marked *n/r* are not reported. The ViT (scratch) / CNN row comes from a separate evaluation.

Encoder	Decoder	Pearson	Cosine	Peak F1	MAE	Dim 95	Collapse
Mean predictor	n/a	0.107	0.129	0.149	12.1	n/a	n/a
ConvFormer (paper)	CNN	0.281	0.290	0.260	14.0	27	0.938
ConvFormer (paper)	ViT	0.306	0.315	0.241	8.5	27	0.938
ConvFormer (scratch)	CNN	0.657	0.608	0.248	5.3	14	0.008
ConvFormer (scratch)	ViT	0.675	0.677	0.320	6.2	29	0.138
ViT (scratch)	CNN	0.853	n/r	0.620	3.6	84	n/r
ViT (scratch)	ViT	0.909	0.909	0.709	3.1	131	0.102

D.1 Experimental spectra

Table 9 gives the evaluation of the ChefNMR-backbone models on experimental spectra, run by collaborators on the C/H/O/N molecules that the SpectraBase-trained models can represent: 213 of the 238 SpecTeach molecules and 14,423 of the 23,457 NMRShiftDB2 molecules. All models

Table 5: Reconstruction quality by chemical-shift region, as the Pearson correlation between predicted and true ^{1}H intensity within each window. Regions R1 to R5 partition the 14 ppm ^{1}H axis into five equal windows of 2.8 ppm, R1 being the most upfield and R5 the most downfield. The mean predictor outperforms every learned model in R1 because nearly all molecules in the dataset share a similar profile there and a constant prediction is close to optimal. It also leads in R5, but every correlation in that window is low, so that ordering reflects weak prediction by all four models rather than the same effect. The learned encoders allocate capacity to R2, R3 and R4, which is the behaviour the probe is designed to detect.

Encoder	Dec	Overall	R1	R2	R3	R4	R5
Mean predictor	n/a	0.107	0.451	−0.010	0.145	0.125	0.121
ConvFormer (paper)	ViT	0.306	0.126	0.266	0.210	0.195	0.045
ConvFormer (scratch)	ViT	0.675	0.067	0.591	0.244	0.283	0.031
ViT (scratch)	ViT	0.909	0.038	0.805	0.453	0.506	0.017

Table 6: Reconstruction probe trained jointly on ^{1}H and ^{13}C targets. Pearson and peak F1 are measured on the ^{1}H channel and the last two columns on the ^{13}C occupancy vector, which is a binary target and so is scored by F1 and area under the ROC curve. *Dim 95* is the number of principal components explaining 95% of representation variance. The two ViT rows reach a perfect ^{13}C score, which reflects the low difficulty of the 80-dimensional binary occupancy target rather than a strong result.

Encoder	Dec	Pearson ↑	Pk F1 ↑	Dim 95	^{13}C F1 ↑	^{13}C AUC ↑
ConvFormer (paper)	CNN	0.281	0.277	27	0.697	0.962
ConvFormer (paper)	ViT	0.307	0.243	27	0.699	0.962
ConvFormer (scratch)	CNN	0.603	0.253	13	0.302	0.701
ConvFormer (scratch)	ViT	0.654	0.332	31	0.441	0.752
ViT (scratch)	CNN	0.690	0.397	83	1.000	1.000
ViT (scratch)	ViT	0.752	0.479	111	1.000	1.000

are sampled with ChefNMR's released settings for these benchmarks. On the full sets, the same pipeline gives the released USPTO-trained adaLN-ChefNMR-S 49.0% (SpecTeach) and 20.0% (NMRShiftDB2) top-1, against about 49% and 20% read from Figure 4 of Xiong et al. (2025).

On SpecTeach, CA-ChefNMR-S and CA-ARL-ChefNMR-S reach 42.9% and 42.3% top-1 against 41.2% for the SpectraBase-trained adaLN-ChefNMR-S, and 0.541 and 0.533 Tanimoto similarity against 0.520, while at top-10 the baseline leads with 58.5% against 57.4% and 56.0%. With 213 molecules, one molecule is worth 0.47 points. On NMRShiftDB2, which provides ^{13}C alone, the three SpectraBase-trained models tie at 6.1% to 6.2% top-1.

E Baseline parity with ChefNMR and NMR-DiGress

Which comparisons are like for like. Table 10 answers, axis by axis, how the models of Section 5 compare with NMR-DiGress and ChefNMR. Our own FiLM-NMR-DiGress and CA-NMR-DiGress runs are like for like with each other: same data, split, scored subset of 11,061 molecules, generation task and sampling protocol. Against the published NMR-DiGress figure the comparison is close but not exact: it also drops aromatic-nitrogen molecules, but its scored count, sampling protocol and guidance are not disclosed, and it reports 14.4 M parameters where our reproduction counts 16.6 M. Against ChefNMR it is not like for like, and the table names the axes. A cell marked *not established* means the axis is not stated on that side's disclosed record. It does not mean the two sides agree.

Protocol differences. Three differences bear directly on how Figure 5 should be read. ChefNMR applies classifier-free guidance at $\omega = 2$ at sampling time while our DiGress runs apply none, so part of its reported accuracy may be attributable to a sampling-time mechanism that is available to any of these models and is orthogonal to the conditioning question. The two sides also score different cohorts: our DiGress runs exclude aromatic-nitrogen molecules and score 11,061 test molecules,

Table 7: Reconstruction of the ^{13}C channel alone. The target is an 80-dimensional binary occupancy vector, which is a far easier prediction problem than the 10,000-bin ^{1}H spectrum, and the scratch-trained models saturate it. Table 4 carries the evidence on encoder quality. Lower is better for binary cross-entropy, higher for F1 and area under the ROC curve.

Encoder	Decoder	BCE ↓	F1 ↑	AUC-ROC ↑
ConvFormer (paper)	CNN	0.130	0.693	0.961
ConvFormer (paper)	ViT	0.126	0.704	0.963
ConvFormer (scratch)	CNN	0.00066	0.9993	1.000
ConvFormer (scratch)	ViT	0.00045	0.9992	1.000
ViT (scratch)	ViT	0.00052	0.9997	1.000

Table 8: Complete results on the SpectraBase test set with the @5, @10 and Tanimoto similarity columns. Abbreviations: *Enc* is the encoder, with NCF denoting NMRConvFormer. *Pool* is the pooling head. k CLS denotes multi-head attention pooling with k learnable queries, while None passes the full token sequence. *Cond* is the conditioning mechanism. XA denotes per-edge and per-node spectral cross-attention at every layer, and AUX adds the auxiliary spectral reconstruction loss. *Tr* is the training regime. Full uses the full learning rate, and Res resumes from a checkpoint. *S@k* is the maximum Tanimoto similarity over k candidates, computed with Morgan fingerprints of radius 2. *Val* is chemical validity. Top-k recovery and validity are percentages. Similarities are in $[0, 1]$. Row 1 is reported by Xiong et al. (2025), and row 2 is our reproduction. The cross-attention rows are single evaluations. Table 11 gives the means of CA-NMR-DiGress and CA-ARL-NMR-DiGress over three inference repeats. Validity is marked *n/r* (not reported) for row 1.

Enc	Pool	Cond	Tr	@1	@5	@10	S@1	S@5	S@10	Val
NCF	1 CLS	FiLM	Full	43.56	62.47	68.39	0.625	0.779	0.817	n/r
NCF	1 CLS	FiLM	Full	43.26	61.31	67.35	0.636	0.766	0.802	94.2
NCF	1 CLS	FiLM+AUX	Full	40.21	59.24	66.14	0.623	0.769	0.811	94.0
NCF	1 CLS	FiLM+AUX	Full	38.12	56.50	62.71	0.603	0.746	0.786	92.2
NCF	3 CLS	FiLM	Full	44.73	62.64	67.70	0.653	0.781	0.813	95.9
NCF	3 CLS	FiLM+AUX	Res	47.29	63.96	69.30	0.674	0.791	0.822	95.9
ViT	1 CLS	FiLM	Full	36.68	51.95	56.65	0.588	0.707	0.740	94.1
ViT	None	XA	Full	40.98	54.84	59.16	0.604	0.715	0.744	90.1
ViT	None	XA+AUX	Full	40.89	53.23	57.76	0.566	0.685	0.718	90.8
NCF	None	XA	Full	**52.89**	69.20	73.54	0.688	0.800	0.826	90.4
NCF	None	XA+AUX	Res	**54.87**	70.83	76.00	0.727	0.822	0.846	91.8

Three rows resume from a checkpoint at the full learning rate, so for them *Full* denotes the learning rate rather than from-scratch initialisation: the pooled ViT run at 36.68, the ViT cross-attention run with the auxiliary loss, which resumes from the ViT cross-attention checkpoint without it, and the single-query pooled row at 38.12, which resumes from the corresponding pooled checkpoint and in addition carries a ^{13}C auxiliary weight ten times that of every other auxiliary-loss run.

whereas ChefNMR applies no such exclusion and the CA-ChefNMR-S rows score 14,137. And the candidate protocols differ: our DiGress runs draw 20 candidates, discard the invalid ones and take the first k of the remainder in sampling order, the CA-ChefNMR-S rows take top-k over 10 draws, and ChefNMR scores its first raw candidates without resampling, so invalid candidates occupy positions rather than being discarded. The remaining difference is training compute, disclosed in Table 10.

Scope of the ChefNMR comparison. The CA-ChefNMR-S rows extend cross-attention to the coordinate backbone within the protocols studied. Each is a mean over three inference runs of one model, not independent training replicates. Neither the published 72.04% of adaLN-ChefNMR-L nor the published DiGress rows is a like-for-like accuracy comparison with our runs. This paper makes no state-of-the-art claim.

Table 9: Experimental spectra, C/H/O/N subsets: SpecTeach (213 molecules, ^{1}H and ^{13}C) and NMRShiftDB2 (14,423 molecules, ^{13}C only). Mean ± sample SD over three sampling seeds (7, 23, 42) of one checkpoint per model. CA-ChefNMR-S is the 8.4k checkpoint of Table 11, and CA-ARL-ChefNMR-S is the checkpoint of its row there, evaluated with EMA weights. The first three rows of each block are trained on simulated SpectraBase spectra, and the (USPTO) row is the released adaLN-ChefNMR-S trained on simulated USPTO spectra. Sampling follows ChefNMR's released settings: guidance $\omega = 1$, 50 steps, and 15 (SpecTeach) or 10 (NMRShiftDB2) samples per molecule. A molecule counts as recovered at top-k if any of its first k samples matches, and Tanimoto is the mean over molecules of the similarity of the first sample to the target. All values in %, except Tanimoto.

Model	Seeds	Top-1	Top-5	Top-10	Tanimoto	Validity
SpecTeach						
CA-ARL-ChefNMR-S	3	42.3 ± 2.6	53.1 ± 0.5	56.0 ± 0.5	0.533 ± 0.020	92.0 ± 0.5
CA-ChefNMR-S	3	42.9 ± 0.7	54.1 ± 0.7	57.4 ± 0.5	0.541 ± 0.006	91.6 ± 0.2
adaLN-ChefNMR-S	3	41.2 ± 0.3	53.5 ± 0.8	58.5 ± 1.4	0.520 ± 0.003	91.4 ± 0.3
adaLN-ChefNMR-S (USPTO)	3	46.6 ± 2.2	59.9 ± 1.4	63.4 ± 1.2	0.564 ± 0.013	90.9 ± 0.3
NMRShiftDB2						
CA-ARL-ChefNMR-S	3	6.2 ± 0.1	10.4 ± 0.1	12.3 ± 0.1	0.197 ± 0.001	77.5 ± 0.1
CA-ChefNMR-S	3	6.1 ± 0.2	10.4 ± 0.2	12.4 ± 0.1	0.194 ± 0.001	76.7 ± 0.1
adaLN-ChefNMR-S	3	6.2 ± 0.2	10.4 ± 0.0	12.3 ± 0.1	0.193 ± 0.001	75.5 ± 0.1
adaLN-ChefNMR-S (USPTO)	3	17.9 ± 0.2	27.6 ± 0.2	31.6 ± 0.2	0.336 ± 0.002	81.0 ± 0.2

Overlap of either set with SpectraBase training molecules: n/r (not reported).

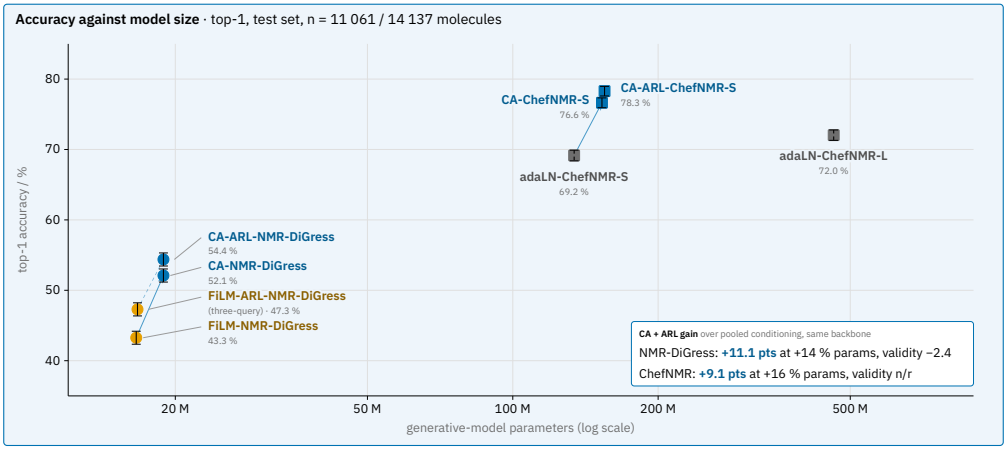


Figure 5: Top-1 exact structure recovery against generative-model parameters (ARL decoder excluded, log scale). Colour encodes conditioning and shape encodes backbone (circles NMR-DiGress, squares ChefNMR). Arrows join pooled and cross-attention conditioning within a backbone. Bars are 95 % Wilson intervals over 11,061 NMR-DiGress and 14,137 ChefNMR test molecules. Dashed bars mark the published adaLN-ChefNMR-S and adaLN-ChefNMR-L values, which assume $n = 14{,}292$ and use a different cohort and sampling protocol (Table 10).

F What the model attends to

Attention profiles. The attention analysis asks where each atom-pair class reads the spectrum. Both analyses in this appendix, attention profiles and masking, use test molecules. We register hooks on the edge cross-attention module in all five layers. We average each edge's attention over heads and reweight it by the norm of the corresponding value vector (Kobayashi et al., 2020). We then average over 1,000 test molecules, five diffusion timesteps and layers, grouping edges by atom-pair class. Figure 6 shows the row-normalised residual after subtracting the class-shared mean.

Table 10: Axis-by-axis comparison between our DiGress-backbone runs and the published comparators. Cells marked *not established* mean the axis is not stated on that side's disclosed record, not that the two sides agree. NCF is NMRConvFormer. The CA-ChefNMR-S rows of Table 11 are noted in the ChefNMR column.

Axis	FiLM-NMR-DiGress, CA-NMR-DiGress (ours)	NMR-DiGress (published)	ChefNMR-S / -L (published)
Train / val / test	114,332 / 14,270 / 14,292 SpectraBase, random split by molecule, released by Hu et al. (2024)	same split, aromatic-nitrogen molecules removed (132,710 molecules reported)	same split. 219 invalid ^{1}H spectra removed, 141,489 conformers retained
Data type	simulated	simulated	simulated
Compute	2,565–4,079 evaluated epochs, 4×A100 DDP, Adam 1×10^{-4}	4 A100 GPUs for 48 hours	10,000 epochs (S), 5,000 epochs (L) on SpectraBase
Parameters	18.9 M (16.6 M baseline +2.3 M, 14%)	14.4 M as reported. Our reproduction counts 16.6 M	134 M / 462 M. CA-ChefNMR-S 153.0 M and CA-ARL-ChefNMR-S 154.9 M as counted for our runs
Given vs. generated	atom types fixed by formula. Five edge classes recovered. No coordinates or stereochemistry	same	denoises 3D coordinates with atom types fixed by formula
Sampling protocol	draw 20, discard invalid, take the first k of the remainder in sampling order	not established	first raw candidates with no resampling. CA-ChefNMR-S rows use top-k over 10 draws in bf16 mixed precision
Guidance at sampling	none (conditioning dropout $p = 0.2$ during training only)	not established	classifier-free guidance, $\omega = 2$ on SpectraBase
Aromatic-nitrogen exclusion	applied. 3,231 of 14,292 test molecules are excluded before scoring, leaving 11,061.	applied (Xiong et al., 2025). Scored count not stated	not applied. CA-ChefNMR-S rows score 14,137 test molecules
Numbers re-run or quoted?	single training runs, ours. CA-NMR-DiGress also reports a mean of three inference repeats	quoted	quoted with a mean over three sampling runs. CA-ChefNMR-S rows report three inference runs of one model

C–C edges attend above the shared baseline to the aromatic ^{1}H region at 6.5 to 8.5 ppm, C–O edges most strongly below 3.5 ppm rather than in the oxygenated region, and N–O and O–O edges show profiles of their own. Of the classes with enough support to characterise, C–C's routing reaches the region most clearly associated with its own chemistry. This is consistent with selective routing, which a single pooled vector, identical for every pair, does not provide. C–O is an exception, and the remaining classes have no clean chemistry-associated reading.

What this analysis does not show. The attention analysis above is correlational, and we treat the attention weights as plausibility evidence rather than as a causal account of what the model routes (Jain & Wallace, 2019; Jacovi & Goldberg, 2020; Kobayashi et al., 2020). Establishing functional dependence requires an intervention, such as masking spectral regions at inference and measuring the effect on generation. We have not run a masking intervention on the DiGress model analysed

Table 11: Cross-attention results for two backbones under their own evaluation protocols. Parameters are in millions. The CA-NMR-DiGress and CA-ARL-NMR-DiGress rows use 11,061 test molecules, 20 samples per molecule and 500 diffusion steps without classifier-free guidance. The first and third rows report means ± sample SD over three inference repeats of one checkpoint. Validity has no dispersion estimate. The CA-ARL-NMR-DiGress parameter count excludes the training-only reconstruction decoder. The second row is one of the first row's three repeats, listed separately because it is the only repeat with stored per-molecule output. It has no dispersion estimate. The ChefNMR-backbone rows use 14,137 test molecules with aromatic nitrogens included, ten samples per molecule, top-10 over all ten draws, 50 sampling steps and guidance scale $\omega = 2$. Their ± values are sample SDs over three inference runs of one model. For the 5.6k row, two of the three runs share a seed. Labels 5.6k, 7.2k and 8.4k mark checkpoints saved at epochs 5,699, 6,999 and 8,249 of one from-scratch training run. The last checkpoint uses exponential-moving-average weights. CA-ARL-ChefNMR-S is fine-tuned with the auxiliary reconstruction loss. ChefNMR-backbone parameter counts are those of our CA-ChefNMR-S and CA-ARL-ChefNMR-S runs. Parity with the published ChefNMR rows is discussed in Appendix E.

Model	Params	Top-1	Top-10	Validity
CA-NMR-DiGress (mean of 3)	18.9	52.10 ± 0.98	72.23 ± 1.13	90.4
CA-NMR-DiGress (one repeat)	18.9	52.89	73.54	90.4
CA-ARL-NMR-DiGress (mean of 3)	18.9	54.38 ± 0.49	74.90 ± 0.97	91.9
CA-ChefNMR-S (5.6k)	153.0	75.77 ± 0.22	89.92 ± 0.10	95.86 ± 0.02
CA-ChefNMR-S (7.2k)	153.0	76.48 ± 0.09	89.82 ± 0.15	96.15 ± 0.04
CA-ChefNMR-S (8.4k)	153.0	76.64 ± 0.23	89.92 ± 0.16	95.89 ± 0.05
CA-ARL-ChefNMR-S (8.4k)	154.9	78.29 ± 0.19	90.05 ± 0.06	96.72 ± 0.10

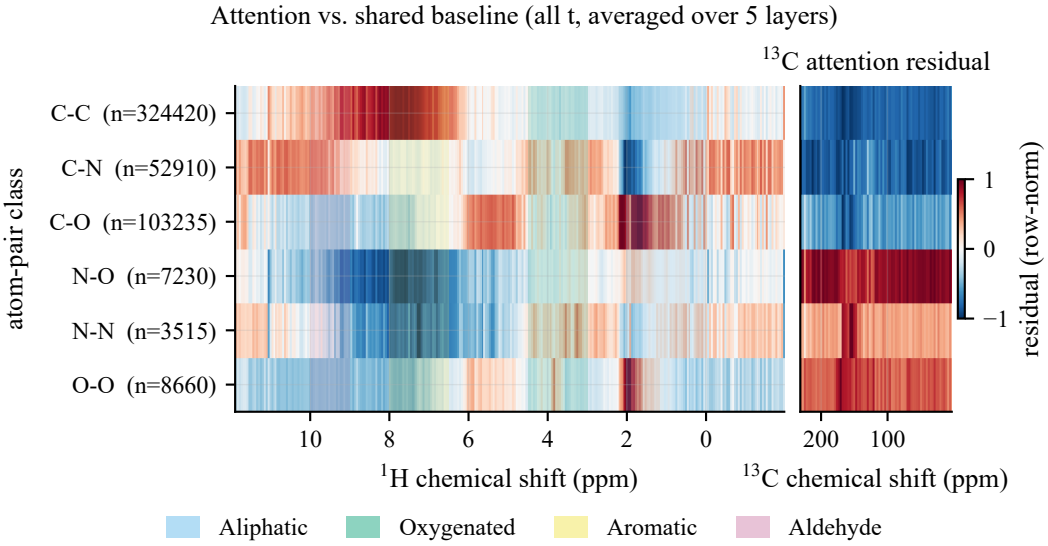


Figure 6: Attention profiles differ by atom-pair class. C–C most clearly reaches the spectral region associated with its chemistry. Profiles are averaged over 1,000 test molecules, five diffusion timesteps and layers, as the row-normalised residual after subtracting the shared mean. Red attends more and blue less in the ^{1}H (left) and ^{13}C (right) channels. Each class buckets every unordered atom pair $i < j$, bonded or not, as the dense edge tensor the model conditions on does, so the counts are pair observations, not bond counts. Class support varies widely, from 324,420 C–C pair observations down to 3,515 N–N in the sample, so sparser classes rest on thinner evidence.

above. The masking experiment below intervenes on the ChefNMR backbone, so it tests dependence on spectral regions rather than the routing shown in Figure 6.

Table 12: Full interaction matrix for spectral masking. Each entry is the selective drop in exact structure recovery, in percentage points, when masking the spectral region given by the row and stratifying molecules by the bond class given by the column. Positive values indicate greater damage to molecules containing that bond class. Negative values indicate greater damage to molecules without it. Bold entries are the chemically matched region–bond pairs. Results use top-1 with one sample per molecule and one sampling pass per condition. Each window includes 13,669 to 14,134 of 14,137 test molecules. The 95% bootstrap intervals over molecules are within ± 1.6 points of each bold value.

Window	aromatic C–C	alkyl C–O	C=O	aliphatic C–C
CA-ChefNMR-S				
aromatic	$\mathbf{+66.0}$	-7.1	-0.1	-23.2
oxygenated	-3.6	$\mathbf{+42.6}$	$+5.0$	$+9.1$
carbonyl	$+0.2$	-3.0	$\mathbf{+18.9}$	-4.0
aliphatic	-28.4	$+6.4$	$+0.1$	$\mathbf{+49.4}$
random 4	$+15.2$	-6.4	$+9.5$	-18.5
random 6	-14.2	$+5.6$	-0.1	$+34.1$
adaLN-ChefNMR-S				
aromatic	$\mathbf{+59.5}$	-5.1	$+1.9$	-19.2
oxygenated	-2.4	$\mathbf{+39.1}$	$+4.0$	$+8.2$
carbonyl	-0.4	-0.7	$\mathbf{+16.2}$	-4.2
aliphatic	-24.8	$+5.9$	$+0.2$	$\mathbf{+44.8}$
random 4	$+12.6$	-3.1	$+7.6$	-13.6
random 6	-13.9	$+5.2$	$+1.2$	$+32.7$

Spectral masking on the ChefNMR backbone. On the ChefNMR backbone, spectral masking provides an intervention on the model input. We jointly zero four chemically defined windows in the ^{1}H and ^{13}C spectra at inference: aromatic (6.5–8.5 ppm, 110–160 ppm), oxygenated (3.0–4.5 ppm, 50–90 ppm), carbonyl (9.0–10.0 ppm, 160–220 ppm), and aliphatic (0.8–2.0 ppm, 10–40 ppm). Both channels are masked simultaneously because either may contain evidence for the same chemical motif. The masked CA-ChefNMR-S checkpoint is saved at epoch 6,999 (1.55 M optimiser steps). adaLN-ChefNMR-S is the released checkpoint at epoch 10,099 (2.24 M steps), with the same batch size. Both are sampled once per molecule on all 14,137 test molecules with 50 steps and $\omega = 2$, under a seed rule that gives every molecule identical noise across conditions and models. Exact recovery is top-1: the canonical SMILES of the single sample equals the ground truth, and a failed perception counts as a miss. Unmasked, this protocol gives 76.39% for CA-ChefNMR-S and 69.14% for adaLN-ChefNMR-S, a difference of 7.24 points (95% CI 6.56 to 7.96). Molecules for which a mask removes every ^{13}C or ^{1}H signal, or for which renormalising the masked ^{1}H spectrum would amplify it more than 1000-fold, are excluded for that mask (3 to 468 of 14,137).

For each masked region r and bond class b, we measure the selective recovery drop

$$\Delta_{r,b} = \left(A^{b}_{\text{unmasked}} - A^{b}_{\text{masked}(r)}\right) - \left(A^{\neg b}_{\text{unmasked}} - A^{\neg b}_{\text{masked}(r)}\right),$$

where A^{b} denotes exact structure recovery on molecules containing bond class b, and $A^{\neg b}$ denotes recovery on molecules without it. Positive values therefore mean that masking region r disproportionately damages molecules containing bond class b.

Matched windows. For each chemically defined mask, the largest positive contrast occurs for its corresponding bond class in both models. The effect is strongest for the aromatic window, where the selective drop reaches 66.0 points for CA-ChefNMR-S and 59.5 points for adaLN-ChefNMR-S, followed by the aliphatic and oxygenated regions. The carbonyl effect is smaller but remains selective at 18.9 and 16.2 points, respectively.

The off-diagonal structure is also informative. In particular, aromatic and aliphatic masking produce opposing effects: masking the aromatic region yields a -23.2-point contrast for aliphatic C–C under CA-ChefNMR-S, while masking the aliphatic region yields a -28.4-point contrast for aromatic C–C. The same pattern appears in adaLN-ChefNMR-S at -19.2 and -24.8 points. The interven-

tion does not simply reduce recovery uniformly. Suppressing one spectral region changes recovery differently depending on the chemistry present in the molecule.

Random windows. The two random windows are drawn from eight candidates. Each has the ^{1}H and ^{13}C widths of a named window and is placed uniformly at random. The rule fixed before sampling keeps the windows that remove the smallest (random 4) and largest (random 6) mean ^{1}H fraction. Their contrasts track overlap with the chemical windows (Pearson $r = 0.85$ for CA-ChefNMR-S and 0.83 for adaLN-ChefNMR-S over the eight cells). The $+34.1$-point aliphatic C–C effect of random 6 is a partial aliphatic mask, not noise: random 6 covers 85% of the aliphatic ^{1}H window, and its interval is 32.2 to 35.9. Random windows that overlap no chemical window give negative contrasts, the same motif-balance effect as the negative off-diagonals above.

Comparing the two interfaces. The raw selective drop of CA-ChefNMR-S exceeds that of adaLN-ChefNMR-S in all four matched cells, by 2.7 to 6.5 points, with 95% intervals excluding zero. CA-ChefNMR-S, however, starts 7.24 points higher. Dividing each model's drop by its own unmasked recovery on the same molecules, the differences are $+0.75$, -0.17, $+1.44$ and $+0.08$ points (aromatic, oxygenated, carbonyl, aliphatic), and every interval contains zero. The masking experiment therefore shows region-specific dependence in both models and does not separate the two ChefNMR interfaces. All intervals come from a bootstrap over molecules from one sampling pass per condition. They omit sampling variance and are too narrow. The two models also differ in training budget.

What zero-masking measures. Zero-masking is not a neutral removal of information. It presents the model with a spectrum in which signal in the selected region is absent, so $\Delta_{r,b}$ measures the consequence of that intervention rather than the amount of information originally contained in the region.