

# *Skeptical review: Dynamic, Weighted, Hierarchical Graph Analysis for Predicting Peptide Aggregate Instability and Identifying Molecular Determinants*

---

## Summary

The manuscript proposes a hierarchical, time-resolved graph framework to analyze peptide aggregate stability and fragmentation in MD simulations of self-assembling KYFIL pentapeptides. For each frame, the authors build (i) a coarse-grained (CG) peptide-peptide contact graph (30 nodes) and (ii) a fine-grained (FG) residue-residue contact graph (150 nodes) derived from heavy-atom contacts ( $< 4.5 \text{ \AA}$ ), then compute multiple graph descriptors (density/weights, Laplacian spectrum incl. Fiedler value, centralities, Louvain communities/modularity, connected components; Sec. 2.2–2.4). “Splitting events” are detected via changes in connected components of the CG graph (Sec. 2.5), and metrics in 1 ns pre-event windows are statistically compared to “stable” control windows (Sec. 2.6). The main empirical claims are that splitting events are frequent (Sec. 3.3) and are preceded by decreased CG connectivity (e.g., CG Fiedler value / density) and reduced FG “bridging” contacts between the two future fragments, with hydrophobic/aromatic residues (F/I/L/Y) implicated at interfaces (Sec. 3.4–3.5). The multiscale graph idea and the bridging concept are promising, but the current manuscript is undermined by (i) internal inconsistencies in timeline/frame counts and aggregate statistics, (ii) under-specified and potentially noise-sensitive split definitions at 20 ps resolution, (iii) insufficiently described dependence-aware statistics/control selection (with likely  $p$ -value inflation), and (iv) ambiguous or inconsistent metric definitions (notably weighted density  $> 1$  and unclear FG objects for Fiedler/bridging). Addressing these items would substantially improve reproducibility, physical interpretability, and the “bigger-picture” value of the proposed framework.

## Strengths

- Conceptually strong multiscale design linking peptide-level (CG) and residue-level (FG) contact networks, enabling mechanistic drill-down from fragmentation events to residue-pair interactions (Sec. 2.2–2.3, Sec. 3.5).
- Rich, time-resolved panel of graph descriptors (density/weights, Laplacian spectrum/Fiedler, centralities, Louvain communities/modularity) that can, in principle, capture connectivity and reorganization beyond simple cluster-size tracking (Sec. 2.4, Sec. 3.2).
- Automated event-centric analysis pipeline (define events, align pre-event windows, compare to controls) that could be broadly useful for other self-assembling systems if made robust to contact intermittency and dependence (Sec. 2.5–2.6, Sec. 3.3–3.4).

- The FG “bridging strength” idea—quantifying residue-level inter-fragment coupling prior to splitting—is an intuitively meaningful bridge between network measures and physical detachment mechanisms (Sec. 2.7, Sec. 3.4.2).
- Residue-level decomposition suggests specific interaction classes (hydrophobic/aromatic) weakening at interfaces prior to split, which aligns with known qualitative design principles and provides a plausible mechanistic narrative (Sec. 3.5).
- Many core graph-theoretic definitions are stated explicitly (e.g., weighted Laplacian), and the overall implementation appears feasible with standard Python graph tooling, suggesting scalability if computational details are clarified (Sec. 2.4, Sec. 2.8).

## Major issues

1. **Reproducibility is currently blocked by missing MD provenance and internal inconsistencies in trajectory length, frame counts, and analyzed time windows (Sec. 2.1 vs Sec. 3.1–3.3).** The manuscript variously reports  $\sim 65,000$  frames ( $1.3 \mu\text{s}$  at  $20 \text{ ps}$ ), an analysis slice of frames  $5000\text{--}64999$  ( $60,000$  frames  $\approx 1.2 \mu\text{s}$ ), but also  $66,771$  frames and a  $100\text{--}1435.42 \text{ ns}$  window, and later  $66,770$  inter-frame transitions. In addition, essentially no MD setup details are given (force field, water/ions, thermostat/barostat, PBC handling, equilibration), preventing assessment or replication.

*Recommendation:* In Sec. 2.1, provide a single consolidated MD description: peptide termini/protonation, force field (and version), solvent model, ions/concentration, box size/shape, ensemble,  $T/P$ , thermostat/barostat parameters, constraints, PME/cutoffs, equilibration and production lengths, and confirm periodic boundary conditions were used in contact calculations. Then reconcile the timeline by explicitly tabulating: total simulated time; frame stride ( $20 \text{ ps}$ ) and number of stored frames; analysis slice start/end frame indices (inclusive/exclusive) and corresponding times; and the resulting number of transitions ( $= \text{analyzed\_frames} - 1$ ). Update Sec. 3.1–3.3 so all reported statistics, figures, and event counts refer to the same, consistent slice.

2. **Core descriptive results about aggregate integrity are internally contradictory, casting doubt on downstream event counts and pre-splitting signatures (Sec. 3.1 vs Sec. 3.2–3.3).** Sec. 3.1 claims the system remains a single 30-peptide aggregate with LCC size  $30.0 \pm 0.0$  and aggregate count  $1.0 \pm 0.0$ , but Sec. 3.2.1 reports average connected components  $1.17 \pm 0.40$  and average LCC size  $28.89 \pm 2.72$ , and Sec. 3.3 reports 1184 splitting events—these cannot all be simultaneously true under a single graph/contact definition and time window.

*Recommendation:* Audit and reconcile the definitions and computations used in Sec. 3.1 vs Sec. 3.2–3.3. Explicitly state whether: (i) “aggregate count” uses a different connectivity criterion than “connected components” (e.g., filtering small detachments), (ii) different time ranges were used, or (iii) earlier summary numbers are incorrect. Ensure figures and captions (notably Fig. 1/2) explicitly state whether statis-

tics are computed for the largest aggregate only or for all aggregates per frame, and that the narrative about frequent transient splitting is consistent with the reported LCC/component time series.

- 3. The operational definition of “splitting events” is under-specified and likely overcounts topological flicker caused by intermittent contacts at 20 ps resolution and a binary edge rule (“any contact  $< 4.5 \text{ \AA}$  implies connected”) (Sec. 2.5, Sec. 3.3).** With such a criterion, momentary loss of a single marginal contact can disconnect the CG graph, producing many short-lived ‘splits’ that may not correspond to physically meaningful fragmentation. This threatens the central claim that pre-splitting graph signatures are specific and mechanistic rather than tautological consequences of noisy edge disappearance.

*Recommendation:* In Sec. 2.5, precisely define the production split criteria (minimum parent size, minimum daughter sizes, treatment of single-peptide detachments, handling of  $> 2$  fragments, and merge/split disambiguation). Add persistence/hysteresis: e.g., require fragments to remain disconnected for  $\geq K$  consecutive frames (or  $\geq \tau$  time), and/or require CG connectivity via edges with weight  $\geq w_{\min}$  (minimum contact count), and/or smooth contact weights (EMA/rolling window). In Sec. 3.3 (or Appendix), run a sensitivity analysis varying (i) distance cutoff (e.g., 4.0/4.5/5.0  $\text{\AA}$ ), (ii) weight threshold  $w_{\min}$ , and (iii) persistence  $K$ , and report how event counts, event durations, and key pre-splitting trends (Sec. 3.4) change. Also quantify how many splits rejoin within 0.1–1 ns to separate transient flicker from sustained fragmentation.

- 4. Statistical comparisons between pre-splitting windows and control windows are insufficiently specified and likely invalidate reported extreme  $p$ -values due to dependence, overlap, and unclear control selection (Sec. 2.6, Sec. 3.4).** The manuscript does not clearly define the control-window algorithm, whether windows overlap across events (highly likely with 1184 events), whether tests are one-/two-sided, how normality was assessed, whether multiple comparisons were corrected, or how temporal autocorrelation and repeated events from the same evolving aggregate are handled.

*Recommendation:* Expand Sec. 2.6 into a fully specified protocol: define event windows (length, alignment time, overlap rules) and control windows (how ‘stable’ is defined quantitatively, required split-free duration, size matching, time matching, and exclusion buffers around events). Use dependence-aware inference: e.g., event-level aggregation (one value per event), enforce a refractory period between events, and/or use block bootstrap / permutation respecting temporal correlation. Report effect sizes (e.g., Cohen’s  $d$  / Cliff’s delta) and confidence intervals alongside  $p$ -values, and apply multiple-hypothesis correction (FDR/Bonferroni) across metrics. In Sec. 3.4, report sample sizes (number of events retained after de-overlap, number of control windows) and robustness to window length (e.g., 0.5/1/2 ns).

5. **Several key metrics are ambiguously defined or mathematically inconsistent with reported values, undermining interpretability and reproducibility**—most notably weighted density exceeding 1 (Sec. 2.4.1 vs Sec. 3.2.1) and unclear conventions about undirected symmetry and double-counting of weights (Secs. 2.2–2.4). Weighted density is described as a normalized quantity bounded by 1, yet CG weighted density is reported around  $2.65 \pm 0.32$ . Additionally, shortest-path-based centralities in weighted graphs depend on whether weights are treated as ‘strengths’ or ‘costs’, which is not clarified (Sec. 2.4).

*Recommendation:* Provide the exact formulas used in code for: (i) weighted density (including whether sums are over  $i < j$  edges or over all  $i \neq j$  adjacency entries, and what is used as the denominator), and (ii) any centrality that depends on shortest paths (betweenness/closeness): clarify whether weights are inverted to convert strengths into distances/costs. If the computed quantity is not bounded by 1, rename it (e.g., mean weight per possible edge) or correct the normalization and recompute affected results/figures (Sec. 3.2–3.4). State explicitly that graphs are undirected with symmetric adjacency and zero diagonals, and document how isolated nodes/components are handled in each metric.

6. **The FG analyses central to the mechanistic claim are under-specified: it is unclear which FG node/edge set is used for FG Fiedler calculations and how ‘bridging strength’ is defined and tracked backward in time** (Sec. 2.3.2, Sec. 2.4.2, Sec. 2.7, Sec. 3.4.2–3.5). This ambiguity also affects interpretation of the reported ‘counterintuitive’ increase of FG Fiedler value before splitting, which could be a selection effect (e.g., LCC becomes smaller/denser when peripheral residues disconnect).

*Recommendation:* In Sec. 2.3.2 and Sec. 2.4.2, explicitly define the FG graph object used in each analysis: whole-system FG graph vs FG restricted to residues belonging to the parent CG aggregate; whether metrics are computed on the FG largest connected component (and how it is identified) or the full FG graph; and whether the combinatorial or normalized Laplacian is used (with explicit formula and isolated-node convention). In Sec. 2.7, formalize bridging strength: define how the two ‘future fragments’ are identified at the split frame (e.g., two largest CG components), how each residue is assigned to a future fragment at earlier times, and exactly which FG edges are summed (inter-fragment only; inter-peptide only; weight aggregation). In Sec. 3.4.2, report FG LCC size/weight changes alongside FG Fiedler aligned to splits to distinguish genuine compaction from component-selection artefacts; include at least one illustrative example graph/snapshot.

7. **The manuscript demonstrates ‘predictive signatures’ mainly via distribution shifts and  $p$ -values, but does not quantify practical predictive value or compare against simpler baselines, leaving the bigger-picture utility un-**

**clear (Sec. 3.4).** Relatedly, bridging-contact decline may be partially tautological if it is too directly coupled to the split definition (connectivity/contact loss).

*Recommendation:* Augment Sec. 3.4 with an explicit prediction assessment: classify windows as ‘split within next 1 ns’ vs ‘no split’ using thresholds or a simple model (logistic regression / survival model), and report ROC-AUC / PR-AUC, precision/recall at meaningful operating points, and lead-time distributions. Compare graph metrics (CG  $\lambda_2$ , density, bridging strength) to baseline structural descriptors (e.g., total inter-peptide contacts, number of CG edges, aggregate  $R_g$ , SASA, mean peptide degree/contact number). Use event-level cross-validation or blocked time splits to avoid leakage. Explicitly discuss to what extent each metric provides information beyond ‘contacts are decreasing’ and where the graph framework adds mechanistic value.

- 8. Claims about robustness and generality are overstated given evidence from a single peptide sequence and a single simulation setup, with limited discussion of force-field/contact-definition dependence and finite sampling (Sec. 3.6, Sec. 4).**

*Recommendation:* Tone down general statements in the Abstract, Sec. 3.6, and Sec. 4 to clearly separate (i) what is shown for KYFIL under the specific MD/contact definition from (ii) what is proposed as a general framework. Add a Limitations paragraph noting dependence on: force field/solvent model, concentration/box conditions, finite 1.3  $\mu$ s sampling, the 4.5 Å cutoff and time resolution, and the use of undirected distance-based contacts (no orientation/energetics). Outline concrete next validations (other sequences, variants, conditions, force fields; experimental or replicate simulations).

## Minor issues

1. Control-window selection is described only qualitatively in places, and the reader cannot verify comparability between controls and pre-splitting windows (Sec. 2.6, Sec. 3.4).

*Recommendation:* Even if Sec. 2.6 is expanded as requested above, also add a concise algorithm box/pseudocode (main text or Appendix) detailing control selection, size-matching tolerance, exclusion buffers, and the exact number of controls per event or per stable period. Summarize control distributions (aggregate size, time-of-trajectory) to demonstrate matching.

2. Weighted density uses a per-frame maximum observed edge weight as a normalizer, which is non-standard and can introduce artefactual fluctuations and hinder comparisons across time/systems (Sec. 2.4.1–2.4.2).

*Recommendation:* Justify this design choice and supplement it with at least one standard alternative reported in parallel (e.g., mean node strength, total inter-peptide contact weight, or normalization by a fixed global maximum across the trajectory).

Indicate whether conclusions in Sec. 3.2 and Sec. 3.4 are robust across normalizations.

3. Community detection (Louvain) details and randomness/sensitivity are not documented, yet reported community counts/modularity could depend on resolution/seed (Sec. 2.4.1–2.4.2, Sec. 3.2).

*Recommendation:* Report the exact implementation (library/version/function), resolution parameter, and how randomness is handled (fixed seed; multiple runs with averaging). Provide a small robustness check on a subset of frames or clearly state this as a caveat.

4. Laplacian variant usage (combinatorial vs normalized) is inconsistently labeled, and the normalized Laplacian definition is not fully provided (Sec. 2.4.1–2.4.2; Sec. 3.2–3.4).

*Recommendation:* Define both Laplacians explicitly (including isolated-node handling) and use consistent notation throughout (e.g.,  $\lambda_2(L)$  vs  $\lambda_2(L_{\text{norm}})$ ). Ensure every figure/table caption specifies which variant is plotted.

5. Residue-pair ‘molecular determinant’ analysis may be biased by differing heavy-atom counts and interface size, and uncertainty is not clearly quantified (Sec. 3.5; Fig. 10).

*Recommendation:* Clarify whether values are per-event, per-residue-pair, or per-interface totals; provide uncertainty (bootstrap CI or SE) for heatmap entries or top-ranked pairs. Consider normalizing contacts by possible atom–atom pairs, residue exposure, or reporting both contact frequency and conditional contact multiplicity to reduce heavy-atom-count bias.

6. Graph construction details are incomplete in ways that can change contact graphs (Secs. 2.2–2.3): treatment of periodic boundary conditions, termini charge/capping, and whether intra-peptide residue contacts are included/excluded in FG analyses used for bridging.

*Recommendation:* State explicitly: PBC minimum-image convention used for distances; termini/capping/protonation; and which intra-peptide edges are included in each FG metric. If bridging is meant to quantify inter-fragment coupling, ensure it is explicitly inter-fragment (and likely inter-peptide) only.

7. Computational cost/feasibility is not discussed despite potentially expensive all-to-all heavy-atom distance computations over  $\sim 60,000$  frames (Sec. 2.8).

*Recommendation:* Report approximate runtime and hardware; describe acceleration strategies (neighbor lists/cell lists, MDAnalysis distance search, cutoff-based sparse evaluation) rather than implying naive  $\mathcal{O}(N^2)$  `cdist` per frame. This will strengthen the ‘scalable framework’ claim.

8. Figures tied to the main claims need clearer definitions, sample sizes, and alignment/uncertainty descriptions (Figs. 1, 7–9; Sec. 3.2–3.4). In particular, Fig. 1 appears dominated by the 30-mer state and may obscure the rare-but-central split events, and Figs. 7–9 need explicit control definitions and statistical annotations.

*Recommendation:* For each relevant figure: define exactly what is averaged and over what sample (frames/events), specify  $N$ , clarify uncertainty bands (SE vs SD vs CI), and show controls/baselines on the same axes where appropriate. For histograms of discrete states, use integer bins and consider log-scale or inset to reveal rare states. Ensure resolution and fonts are publication-quality.

9. Related work and citations are not well situated in the biomolecular/peptide aggregation literature, and parts of the manuscript metadata appear inherited from an unrelated template (Sec. 1; Keywords/References).

*Recommendation:* Add a focused Related Work subsection in Sec. 1 covering residue interaction networks/protein contact graphs, dynamic network analysis in MD, and network approaches to amyloids/peptide aggregates, and cite relevant peptide self-assembly/design literature when interpreting F/I/L/Y roles (Sec. 3.5, Sec. 4). Replace non-domain keywords with appropriate terms; ensure author/affiliation and reference list are aligned with the submission’s scientific domain.

## Very minor issues

1. Typographical/formatting inconsistencies reduce polish (Sec. 1–4): broken words/line breaks (e.g., “Tradi\n tional”), stray underscores in terms, inconsistent section heading formatting (e.g., stray ‘#’), and inconsistent unit spacing/LaTeX ( $\mu\text{s}$ , ns, Å).

*Recommendation:* Do a full proofreading/format pass: standardize section headers, remove stray symbols/underscores, and use consistent unit formatting (e.g., ‘1.3  $\mu\text{s}$ ’, ‘< 4.5 Å’). Round overly precise statistics to sensible significant digits and ensure captions are self-contained but not repetitive.

2. Terminology/notation for residue indices vs residue types vs node IDs is sometimes ambiguous (Secs. 2.2.2, 3.5; Fig. 10).

*Recommendation:* Define separate symbols for residue position (1–5), residue identity (K/Y/F/I/L), and node IDs in the 150-node FG graph, and state how residue-pair aggregation (e.g., PHE–PHE) is computed from node-level contacts.

3. Figure styling inconsistencies (fonts, tick precision, redundant titles, color/contrast) may impair readability in print (Figs. 1, 7–9).

*Recommendation:* Standardize typography and tick precision, remove redundant in-plot titles, use colorblind-safe palettes plus line styles, and export as vector/high-DPI images.

## Key statements and references

- **✘** Molecular dynamics simulations provide atomistic trajectories that capture peptide self-assembly and dissociation processes, but traditional analyses based mainly on global properties such as aggregate size or static contact maps struggle to extract quantitative, predictive metrics of aggregate instability and to identify the specific molecular interactions responsible for breakdown from these high-dimensional datasets (Huang & Shuai, 2013; Liang et al., 2025).
- *Reference(s)*: Huang & Shuai, 2013, Liang et al., 2025
- *Justification*: Huang & Shuai, 2013 performs MD of zinc-bound A $\beta$ 1–16 (and a rat dimer) and analyzes RMSD, radius of gyration, coordination bonds, H-bonds, and SASA; it does not study peptide self-assembly/dissociation or discuss limitations of aggregate-size/contact-map analyses. Liang et al., 2025 introduces the NEP89 potential with broad MD applications (including a protein–ligand example) but does not address peptide aggregation or challenges extracting predictive instability metrics from high-dimensional MD data. Thus, the statement is not supported by these papers.
- **✘** In constructing the dynamic coarse-grained and fine-grained graphs, the total number of atomic contacts between peptides or residues is used as an interaction-strength edge weight, following recent graph-based representations of molecular systems where contact counts or similar proximity-based metrics define weighted edges between molecular entities (Airale et al., 2025, 2025; Coupette et al., 2025).
- *Reference(s)*: Airale et al., 2025, 2025, Coupette et al., 2025
- *Justification*: Neither Airale et al., 2025, 2025 nor Coupette et al., 2025 describe constructing dynamic coarse- or fine-grained molecular graphs with interaction-strength edge weights based on total atomic contact counts. Airale et al., 2025, 2025 focus on simple-path structural encodings for graph transformers on existing benchmark graphs (e.g., ZINC, Peptides) without defining new weighted edges by contact counts, and Coupette et al., 2025 introduce a dataset-perturbation framework (RINGS) evaluating datasets as provided, not constructing contact-count-weighted graphs.
- **△** Representing each simulation frame as a dynamic graph and then analyzing time series of graph-theoretical properties such as Laplacian eigenvalues (including the Fiedler value), density, centrality, and community structure builds directly on prior work that applied similar graph measures to evolving astrophysical or cosmological systems to quantify connectivity and clustering in complex particle or halo ensembles (Yang and Yu, 2023; Roncoli et al., 2024; Pavlou et al., 2023).
- *Reference(s)*: Yang and Yu, 2023, Roncoli et al., 2024, Pavlou et al., 2023

- *Justification:* Yang and Yu, 2023 construct graphs of halo systems from simulations and analyze graph spectra (adjacency and normalized Laplacian) and other graph metrics, and Pavlou et al., 2023 build similarity graphs of galaxies and apply community detection and spectral-clustering using Laplacian eigenvectors/modularity/conductance—thus supporting use of graph-theoretic measures (including Laplacian spectra) to study connectivity/clustering in astrophysical systems. However, none of the three works analyze dynamic graphs or time series across simulation frames, nor do they specifically emphasize the Fiedler value, graph density, or centrality measures; Roncoli et al., 2024 uses GNNs rather than classical graph metrics. Therefore, the claim is only partially supported.
- **✓ The use of the Louvain algorithm to detect communities in the peptide-level graph and the interpretation of the resulting modularity as a measure of the quality and strength of internal sub-structures within the aggregate are grounded in prior network-science studies that established Louvain-based community detection and modularity as standard tools for identifying and characterizing communities in complex weighted graphs (Reid et al., 2012; Prouteau et al., 2024).**
- *Reference(s):* Reid et al., 2012, Prouteau et al., 2024
- *Justification:* Reid et al., 2012 explicitly describe modularity as a measure of community quality and evaluate the widely used Louvain modularity-maximization method (“Many of the most popular CFAs are based on the modularity maximisation approach... The modularity function measures community quality... Modularity maximization algorithms, such as the fast ‘Louvain’ method...”). Prouteau et al., 2024 adopt Louvain for community detection, note it is extensively used, and discuss modularity and its multi-resolution parameter on (un)weighted graphs. Together, these papers establish Louvain-based community detection and modularity as standard tools for identifying/characterizing communities, supporting the stated interpretation.
- **✗ The overall computational workflow—loading long molecular dynamics trajectories, constructing large dynamic graphs, computing Laplacian spectra and centrality measures, and storing time series for later analysis—is implemented using Python libraries (MDAnalysis, NumPy, SciPy, igraph, multiprocessing, and HDF5 I/O) whose suitability for large-scale numerical, graph-theoretic, and time-series analyses has been demonstrated in recent astronomy and data-analysis toolkits (Wang, 2025; Zheng et al., 2025; Boniol et al., 2025).**
- *Reference(s):* Wang, 2025, Zheng et al., 2025, 2025, Boniol et al., 2025
- *Justification:* Zheng et al., 2025 describes an astronomy CMD-fitting package using Python (NumPy, SciPy, PyTorch, etc.) but does not mention MDAnalysis, igraph, HDF5 I/O, multiprocessing, Laplacian spectra, centrality, or molecular dynamics. Boniol et al., 2025 presents a time-series clustering method (k-Graph) implemented in

Python with PCA, k-Means, and spectral clustering, but likewise does not use or validate MDAnalysis, igraph, HDF5, multiprocessing, or compute Laplacian spectra/centrality, nor does it involve molecular dynamics or dynamic graph pipelines. Thus, the claimed workflow and specific library choices are not supported by the cited papers; only general Python/NumPy/SciPy use is shown.

## Mathematical consistency audit

This section audits **symbolic/analytic** mathematical consistency (algebra, derivations, dimensional/unit checks, definition consistency).

**Maths relevance:** substantial

The paper's core technical content is mathematical in the sense of defining time-dependent weighted graphs from contact-count data and analyzing them via Laplacian spectra, densities, centralities, connected components, and event definitions (splits). The main internal-consistency problems are not deep algebra errors but (i) contradictory bookkeeping of frames/time windows and (ii) a weighted-density definition that should be bounded yet is reported  $> 1$ , plus (iii) incompatible statements about always-single-aggregate behavior vs fluctuating component structure and many splitting events.

### Checked items

1. ✓ **Total frame count from timestep** (Sec. 2.1, p.3)
  - **Claim:**  $1.3 \mu\text{s}$  with  $20 \text{ ps}$  per frame corresponds to  $65,000$  frames.
  - **Checks:** arithmetic consistency
  - **Verdict:** PASS; confidence: high; impact: minor
  - **Assumptions/inputs:**  $1 \mu\text{s} = 10^6 \text{ ps}$ , Frames are stored at constant  $20 \text{ ps}$  intervals
  - **Notes:**  $1.3 \mu\text{s} = 1,300,000 \text{ ps}$ ;  $1,300,000/20 = 65,000$  frames, consistent with the statement.
2. ✓ **Equilibrium start frame index** (Sec. 2.1, p.3)
  - **Claim:**  $100 \text{ ns}$  corresponds to frame index  $5000$  for a  $20 \text{ ps}$  frame interval.
  - **Checks:** arithmetic consistency
  - **Verdict:** PASS; confidence: high; impact: minor
  - **Assumptions/inputs:** Indexing is 0-based or 1-based does not affect the division shown; claim is about the quotient
  - **Notes:**  $100 \text{ ns} = 100,000 \text{ ps}$ ;  $100,000/20 = 5,000$ .
3. ✓ **Analysis slice frame count** (Sec. 2.1, p.3)
  - **Claim:** Frames  $5000$  to  $64999$  total  $60,000$  frames and cover  $1.2 \mu\text{s}$ .
  - **Checks:** arithmetic consistency
  - **Verdict:** PASS; confidence: high; impact: minor

- **Assumptions/inputs:** Endpoints are included in the count
- **Notes:** Inclusive count:  $64999 - 5000 + 1 = 60,000$  frames. Duration:  $60,000 \times 20 \text{ ps} = 1,200,000 \text{ ps} = 1.2 \mu\text{s}$ .

#### 4. ✘ Results analysis window frame/time consistency (Sec. 3.1, p.5)

- **Claim:** The analysis window spans 100 ns to 1435.42 ns with 66,0771 frames at 20 ps per frame.
- **Checks:** arithmetic consistency, cross-section consistency (Methods vs Results)
- **Verdict:** FAIL; confidence: high; impact: critical
- **Assumptions/inputs:** Frames are contiguous at 20 ps intervals, Time span is end – start
- **Notes:** 100 → 1435.42 ns is a 1335.42 ns span; at 0.02 ns/frame that implies 66,0771 frame intervals, but this contradicts Methods stating a 1.3 μs (1300 ns) total trajectory and an analyzed slice of exactly 60,000 frames. Either the trajectory is longer than 1.3 μs, the frame interval is not 20 ps, or the stated end time/frame count is wrong.

#### 5. ✘ Number of transitions analyzed (Sec. 3.3, p.8)

- **Claim:** 66,0770 inter-frame transitions were analyzed.
- **Checks:** arithmetic consistency, cross-section consistency (Methods vs Results)
- **Verdict:** FAIL; confidence: high; impact: critical
- **Assumptions/inputs:** Number of transitions = number of analyzed frames – 1
- **Notes:** Methods imply 60,000 analyzed frames, hence 59,999 transitions. The reported 66,0770 transitions aligns with 66,0771 frames (Results Sec. 3.1) but conflicts with the Methods slice.

#### 6. ✔ Contact-based edge weight definition (peptide graph) (Sec. 2.2.1 and 2.3.1, pp.3–4)

- **Claim:** Inter-peptide edge weights are the number of heavy-atom contacts within 4.5 Å between peptides  $i$  and  $j$ .
- **Checks:** definition consistency
- **Verdict:** PASS; confidence: medium; impact: minor
- **Assumptions/inputs:** Contacts are counted over pairs of heavy atoms across peptides, Weights are nonnegative integers
- **Notes:** Definition is coherent as a weight construction. Symmetry ( $w_{ij} = w_{ji}$ ) is implied by pairwise distances but not explicitly stated.

#### 7. ✔ FG graph node count (Sec. 2.3.2, p.4)

- **Claim:** Fine-grained graph has 150 nodes representing 30 peptides  $\times$  5 residues.
- **Checks:** counting consistency
- **Verdict:** PASS; confidence: high; impact: minor
- **Assumptions/inputs:** Exactly 30 peptides and 5 residues per peptide in KYFIL
- **Notes:**  $30 \times 5 = 150$ .

8. ✓ **Weighted Laplacian construction** (Sec. 2.4.1, p.4)

- **Claim:** The weighted Laplacian is  $L = D - W$ , with  $D_{ii} = \sum_j W_{ij}$  (weighted degree/strength).
- **Checks:** algebraic/formula correctness, notation consistency
- **Verdict:** PASS; confidence: high; impact: moderate
- **Assumptions/inputs:**  $W$  is the weighted adjacency matrix for the (sub)graph being analyzed
- **Notes:** This definition is internally consistent and correctly describes the (combinatorial) weighted Laplacian.

9. ✓ **Fiedler value interpretation** (Sec. 2.4.1, p.4)

- **Claim:** The second-smallest Laplacian eigenvalue  $\lambda_2$  measures connectivity;  $\lambda_2 = 0$  indicates disconnection.
- **Checks:** conceptual consistency within the paper
- **Verdict:** PASS; confidence: medium; impact: minor
- **Assumptions/inputs:**  $L$  is the (combinatorial) Laplacian of an undirected graph with nonnegative weights
- **Notes:** Within the paper's stated conventions, this is consistent (they also state they compute  $\lambda_2$  on the LCC to ensure connectivity).

10. ✗ **Weighted density normalization vs reported CG values** (Definition: Sec. 2.4.1, p.4; Reported: Sec. 3.2.1, p.6)

- **Claim:** Weighted density =  $\frac{\text{sum of edge weights}}{\text{maximum possible sum of weights for a complete graph with same } n}$ , using max observed contact count in that frame as max edge weight; reported CG weighted density is  $2.65 \pm 0.32$ .
- **Checks:** normalization/bounds check, definition-to-result consistency
- **Verdict:** FAIL; confidence: high; impact: critical
- **Assumptions/inputs:** Graph is simple undirected with  $W_{ii} = 0$  and  $W_{ij} \geq 0$ , 'Sum of all edge weights' refers to a sum over unique edges ( $i < j$ ), or if over adjacency entries, the convention is stated
- **Notes:** Given the stated normalization, the ratio should be  $\leq 1$  (actual weight-sum cannot exceed max\_weight  $\times$  number\_of\_possible\_edges). The reported mean 2.65 violates this bound, indicating either the formula is mis-

stated, the computation double-counts/uses a different denominator, or the quantity is not a density.

11. ✓ **Aggregate definition as connected components** (Sec. 2.5, p.4)

- **Claim:** Aggregates are connected components of the CG graph at each frame.
- **Checks:** definition consistency
- **Verdict:** PASS; confidence: high; impact: moderate
- **Assumptions/inputs:** Edges represent peptide-peptide contacts in that frame
- **Notes:** Definition is clear and operational.

12. ✓ **Splitting event definition across consecutive frames** (Sec. 2.5, p.4)

- **Claim:** A split occurs if a component's peptide set at frame  $t$  is distributed across  $\geq 2$  components at frame  $t + 1$  (with size thresholds).
- **Checks:** logical consistency
- **Verdict:** PASS; confidence: medium; impact: moderate
- **Assumptions/inputs:** Component membership is computed per frame from the CG graph, Thresholds are applied consistently
- **Notes:** Event logic is coherent as a discrete-time definition.

13. ✓ **Pre-splitting window length** (Sec. 2.6, p.5)

- **Claim:**  $N_w = 50$  frames corresponds to 1 ns.
- **Checks:** unit/time consistency
- **Verdict:** PASS; confidence: high; impact: minor
- **Assumptions/inputs:** 20 ps per frame throughout the analysis
- **Notes:**  $50 \times 20 \text{ ps} = 1000 \text{ ps} = 1 \text{ ns}$ .

14. ✗ **Internal consistency of 'always 30-peptide aggregate' vs later CG stats and splitting counts** (Sec. 3.1, p.5; Sec. 3.2.1, p.6; Sec. 3.3, p.8)

- **Claim:** System is consistently a single 30-peptide aggregate (mean aggregates  $1.0 \pm 0.0$ ; LCC size  $30.0 \pm 0.0$ ), yet later reports average components  $1.17 \pm 0.40$ , average LCC  $28.89 \pm 2.72$ , and 1184 splitting events.
- **Checks:** cross-section consistency, logical consistency
- **Verdict:** FAIL; confidence: high; impact: critical
- **Assumptions/inputs:** All these summaries refer to the same CG graph definition and the same analysis window (100 ns onward)
- **Notes:** If LCC size is identically 30 and number of components identically 1 across the window, then splits as defined cannot occur and LCC/component averages cannot deviate from (30, 1). The paper needs to clarify which state-

ments are incorrect or whether these metrics were computed on different subsets/with different filtering.

15.  $\triangle$  **Normalized Laplacian / normalized Fiedler value usage** (Sec. 3.2.1–3.2.2, pp.6–7)

- **Claim:** Normalized Laplacian Fiedler values are reported alongside unnormalized ones.
- **Checks:** definition completeness
- **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
- **Assumptions/inputs:** Some normalized Laplacian was used
- **Notes:** The normalized Laplacian formula (and any handling of isolated nodes / subgraph extraction details) is not provided, so the normalized eigenvalue computations cannot be audited symbolically.

16.  $\triangle$  **Symmetry / double-counting risk in weight sums** (Sec. 2.3.1–2.4.1, pp.3–4)

- **Claim:** Quantities like 'sum of all edge weights' and densities are computed from adjacency matrices.
- **Checks:** definition ambiguity check
- **Verdict:** UNCERTAIN; confidence: medium; impact: moderate
- **Assumptions/inputs:** WCG and WFG are stored as full matrices
- **Notes:** If sums are taken over all  $i, j$  entries of a symmetric adjacency matrix, weights are double-counted relative to sums over unique edges. The paper does not specify the convention, and this ambiguity is especially relevant given the weighted-density inconsistency.

## Limitations

- The paper contains very few explicit, fully written formulas beyond the Laplacian and some simple time/frame arithmetic; many key metrics (e.g., exact unweighted density formula used, modularity expression, exact weighted betweenness/eigenvector centrality conventions) are referenced conceptually without explicit equations, limiting symbolic verification.
- Several central quantitative claims depend on definitions that are described in prose but not formalized (e.g., the exact computation of 'bridging strength' and whether it double-counts residue-residue edges across fragments). Without explicit formulas, these can only be partially audited for logical consistency.

## Numerical results audit

This section audits **numerical/empirical** consistency: reported metrics, experimental design, baseline comparisons, statistical evidence, leakage risks, and reproducibility.

Out of 13 candidate numeric checks, 11 PASSED, 1 FAILED (cross-section frame-count inconsistency between Methods and Results), and 1 was UNCERTAIN (FG node count/dimension check not executed due to an unsupported check type). Key unit conversions ( $\mu\text{s}/\text{ns}/\text{ps}$ ) and frame/transition arithmetic were internally consistent within sections; however, the Methods-vs-Results disagreement on the number of analysis frames (60,000 vs 66,1771) is material and should be resolved.

### Checked items

1. ✓ **C1\_frames\_total\_from\_duration\_and\_timestep** (p.3, Methods 2.1 (Total trajectory comprised 65000 frames ( $1.3 \mu\text{s} / 20 \text{ps}$ )))
  - **Claim:** The total trajectory comprised 65000 frames ( $1.3 \mu\text{s} / 20 \text{ps}$ ).
  - **Checks:** unit\_conversion\_and\_division
  - **Verdict:** PASS
  - **Notes:** Computed frames= $\text{total\_time}/\text{dt}=65000.000000000001$ ; matches reported under rounding.
2. ✓ **C2\_frame\_index\_for\_100ns** (p.3, Methods 2.1 (100 ns corresponds to frame index ... = 5000))
  - **Claim:** Given the 20 ps frame interval, 100 ns corresponds to frame index 100,000 ps / 20 ps = 5000.
  - **Checks:** unit\_conversion\_and\_division
  - **Verdict:** PASS
  - **Notes:** Index computed as  $\text{time\_offset}/\text{frame\_interval}$ .
3. ✓ **C3\_analysis\_window\_frame\_count\_from\_indices** (p.3, Methods 2.1 (frames from index 5000 to 64999, totaling 60000 frames))
  - **Claim:** Analysis window included frames from index 5000 to 64999, totaling 60000 frames.
  - **Checks:** index\_range\_count
  - **Verdict:** PASS
  - **Notes:** Count computed as end-start+1 (inclusive).
4. ✓ **C4\_analysis\_window\_duration\_from\_frame\_count\_and\_step** (p.3, Methods 2.1 (60000 frames ... covering  $1.2 \mu\text{s}$ ))
  - **Claim:** Frames 5000 to 64999 total 60000 frames covering  $1.2 \mu\text{s}$  of simulation time.
  - **Checks:** multiplication\_and\_unit\_conversion
  - **Verdict:** PASS
  - **Notes:** Best-matching convention:  $N \times dt$  ( $60000 \times 20 \text{ps} = 1.2 \mu\text{s}$ ).
5. ✓ **C5\_atoms\_per\_peptide\_consistency\_total\_vs\_heavy** (p.3, Methods 2.1 (Each peptide consisted of 76 atoms, with 38 heavy atoms.))

- **Claim:** Each peptide consisted of 76 atoms, with 38 heavy atoms.
  - **Checks:** difference\_check
  - **Verdict:** PASS
  - **Notes:** Implied hydrogen count is  $76 - 38 = 38$  (non-negative integer).
6. ✓ **C6\_heavy\_atoms\_per\_residue\_sum\_to\_total\_heavy\_atoms** (p.3, Methods 2.1 (Lys 8, Tyr 11, Phe 11, Ile 4, Leu 4; total heavy atoms 38))
- **Claim:** Heavy atoms per residue: Lys 8, Tyr 11, Phe 11, Ile 4, Leu 4; each peptide has 38 heavy atoms.
  - **Checks:** sum\_to\_total
  - **Verdict:** PASS
  - **Notes:**  $8 + 11 + 11 + 4 + 4 = 38$  matches the reported total heavy atoms.
7. △ **C7\_FG\_node\_count\_from\_peptides\_times\_residues** (p.3-4, Methods 2.2.2 and 2.3.2 (30 peptides, 5 amino acids each  $\Rightarrow$  150 nodes; mentions  $150 \times 150$ ))
- **Claim:** FG graph has 150 nodes (30 peptides  $\times$  5 residues per peptide) and a  $150 \times 150$  adjacency matrix.
  - **Checks:** multiplication\_and\_dimension\_check
  - **Verdict:** UNCERTAIN
  - **Notes:** Marked UNCERTAIN due to execution-side unsupported check type.
8. ✓ **C8\_pre\_split\_window\_time\_from\_frames\_and\_dt** (p.5, Methods 2.6 ( $N_w = 50$  frames (1 ns); dt=20 ps earlier))
- **Claim:** A pre-splitting window of  $N_w = 50$  frames corresponds to 1 ns.
  - **Checks:** multiplication\_and\_unit\_conversion
  - **Verdict:** PASS
  - **Notes:** Best-matching convention:  $N \times dt$  ( $50 \times 20$  ps = 1 ns).
9. ✓ **C9\_analysis\_window\_duration\_from\_reported\_end\_time\_minus\_start** (p.5, Results 3.1 (analysis window 100 ns to 1435.42 ns; described as 1335.42 ns window later))
- **Claim:** Analysis window spans from 100 ns to 1435.42 ns, i.e., 1335.42 ns.
  - **Checks:** difference\_check
  - **Verdict:** PASS
  - **Notes:**  $1435.42 - 100.00 = 1335.42$  matches.
10. ✓ **C10\_frames\_from\_analysis\_window\_duration\_and\_dt\_vs\_reported\_66771** (p.5, Results 3.1 (100 ns to 1435.42 ns (66,!771 frames at 20 ps per frame)))
- **Claim:** The analysis window spanning 100 ns to 1435.42 ns corresponds to 66,!771 frames at 20 ps per frame.
  - **Checks:** duration\_to\_frame\_count

- **Verdict:** PASS
  - **Notes:** 1335.42 ns/0.02 ns per frame = 66771.0 exactly.
11. ✘ **C11\_inconsistency\_between\_methods\_framecounts\_and\_results\_framecounts** (p.3 Methods 2.1 vs p.5 Results 3.1)
- **Claim:** Methods state analysis frames 5000–64999 totaling 60,000 frames; Results state 66,771 frames in analysis window.
  - **Checks:** cross\_section\_consistency
  - **Verdict:** FAIL
  - **Notes:** Direct equality check between sections fails (60000 vs 66771).
12. ✔ **C12\_interframe\_transitions\_from\_frames** (p.8, Results 3.3 (Over the 66,770 inter-frame transitions analyzed...) and p.5 Results 3.1 (66,771 frames))
- **Claim:** Over the 66,770 inter-frame transitions analyzed, ... (implying 66,771 frames).
  - **Checks:** off\_by\_one\_check
  - **Verdict:** PASS
  - **Notes:** 66,771 – 1 = 66,770 matches.
13. ✔ **C13\_splitting\_events\_rate\_per\_transition** (p.8, Results 3.3 (1184 splitting events over 66,770 transitions))
- **Claim:** A total of 1184 splitting events were detected over 66,770 inter-frame transitions.
  - **Checks:** rate\_computation
  - **Verdict:** PASS
  - **Notes:** Computed  $\text{event\_fraction} = 1184/66770 = 0.0177325146$ , within  $[0, 1]$ .

## Limitations

- Only parsed text was available; numeric values embedded solely in figures/plots were not extracted or checked.
- Many reported averages/standard deviations and all statistical-test outcomes require underlying per-frame or per-event data not present in the PDF text, so they cannot be recomputed with fast code.
- Some quantities depend on conventions (e.g., whether a window duration uses  $N \times dt$  vs  $(N - 1) \times dt$ ; inclusive vs exclusive frame endpoints), so checks may need to evaluate multiple plausible conventions.