

Minimum Flow Decomposition Guided by Saturating Subflows

Ke Chen¹  

Department of Computer Science and Engineering, The Pennsylvania State University,
University Park, PA, USA

Abhishek Talesara

Department of Computer Science and Engineering, The Pennsylvania State University,
University Park, PA, USA

Sanchal Thakkar 

Department of Computer Science and Engineering, The Pennsylvania State University,
University Park, PA, USA

Mingfu Shao¹  

Department of Computer Science and Engineering, The Pennsylvania State University,
University Park, PA, USA

Huck Institutes of the Life Sciences, The Pennsylvania State University, University Park, PA, USA

Abstract

Introduction. The minimum flow decomposition (MFD) problem asks to decompose a directed acyclic flow network (G, f) with a unique source s and a unique sink t into the fewest weighted s - t paths whose combined contributions exactly reproduce f . MFD underlies a broad class of multi-assembly tasks in bioinformatics: reference-based RNA assembly from splice graphs [24, 11, 23, 22, 16, 18, 15, 19, 29, 25], metagenomic assembly [21], and viral quasi-species inference [1].

MFD is strongly NP-hard [26] and hard to approximate within some fixed constant factor [12]. Exact solvers include an FPT algorithm whose runtime grows exponentially in the solution size [14] and a family of integer linear programming (ILP) formulations [5, 9] capable of handling extensions such as inexact flows [27, 4], safety and subpath constraints [28, 7, 13, 3], and graphs with cycles [6]. However, ILP remains unscalable on large practical instances.

The widely used greedy-width heuristic [26] is very efficient but can be exponentially worse than optimal in the worst case [2]. The state-of-the-art heuristic, catfish [20], substantially improves this efficiency-performance tradeoff by identifying linear equations among edge flow values – structural constraints implied by any optimal decomposition – and resolving them via safe graph transformations. On simpler instances catfish is highly effective, but three interrelated limitations degrade its performance on complex graphs: (1) it cannot distinguish *good* equations (arising from a true optimal decomposition) from *superficial* ones that distort the graph when resolved; (2) many good equations cannot be fully resolved due to the absence of suitable closed subgraphs, so catfish discards their information entirely; and (3) when no equation is resolvable catfish falls back to greedy-width, which performs poorly on entangled graphs.

Method. We introduce catfish-LP, which augments catfish with a lightweight linear programming (LP) formulation based on *saturating subflows*. For each edge $e \in E$ we define continuous variables $\{x_e(a) : a \in E\}$ modeling a valid s - t subflow that saturates e ; intuitively, $x_e(a)$ represents the amount of flow on e that must pass through edge a . Five base constraints enforce saturation, symmetry, flow validity, and flow conservation. Two additional equation constraints require that the aggregate subflow through the left-hand-side edges of an equation equals that through the right-hand-side edges. The full LP is polynomial-time solvable, adding only modest overhead over catfish.

The LP plays three complementary roles within a single unified framework: (1) *equation filtering*: if adding a candidate equation renders the LP infeasible, that equation cannot arise from any minimum decomposition and is discarded, preventing structurally invalid graph transformations;

¹ Corresponding authors.



(2) *safe edge merging*: a feasible LP solution reveals pairs of edges that must carry identical subflow and can therefore be safely contracted; (3) *informed greedy extraction*: when no further simplification is possible, rather than invoking greedy-width blindly, catfish-LP extracts from the LP solution the heaviest simple path consistent with all surviving equations, deferring the error-prone greedy step as long as possible.

Experimental Results. We compare catfish-LP against greedy-width, catfish, and the optimized ILP solver [9] on two benchmarks, using Gurobi [10] as the underlying LP/ILP engine. Table 1 reports decomposition quality on four datasets of biologically derived splice graphs, with abundances estimated by Salmon [17] or simulated with the Flux-Simulator [8]. Catfish-LP achieves the smallest excess among heuristics on the Salmon dataset and negative excess on the remaining three, matching ILP quality while being orders of magnitude faster. Figure 1 summarizes results on 1,440 simulated graphs spanning 72 complexity configurations. Catfish-LP consistently produces the smallest decompositions and recovers the most ground-truth paths among all heuristics, achieving near-ILP quality in a fraction of its runtime – including a threefold improvement over catfish on the hardest 498 instances where ILP times out on every instance.

Conclusion. Catfish-LP demonstrates that incorporating a polynomial-time LP oracle into a combinatorial heuristic yields substantial gains in decomposition quality with negligible scalability cost, addressing each of catfish’s core limitations in a unified manner. Future directions include strengthened LP formulations, domain-specific constraints for transcriptome assembly, and probabilistic interpretations of LP-guided decompositions.

2012 ACM Subject Classification Applied computing → Computational transcriptomics; Applied computing → Bioinformatics; Mathematics of computing → Network flows

Keywords and phrases flow decomposition, RNA assembly, linear programming, optimality gap

Digital Object Identifier 10.4230/LIPIcs.WABI.2026.31

Category Extended Abstract

Related Version *Full Version*: <https://doi.org/10.64898/2025.12.11.693570>

Supplementary Material *Software (Source Code)*: <https://github.com/Shao-Group/catfish-LP>
archived at `swb:1:dir:fb27825b2bba25a8f04b1ae7ff47369df018a4d3`

Funding This work is supported by the US National Science Foundation (2145171 to M.S.) and by the US National Institutes of Health (R01HG011065 to M.S.).

■ **Table 1** Decomposition results on perfect splice graphs (metric: $|P| - |P^\dagger|$, lower is better). $|P^\dagger|$ is the total number of ground-truth RNA isoforms, $|P|$ is the total decomposition size. For ILP, the parenthesised number is the ground-truth path count in timed-out instances.

Dataset	Graphs	$ P^\dagger $	$ P - P^\dagger $			
			greedy-width	catfish	catfish-LP	ILP
Salmon	13,300,893	32,889,145	1,330,137	86,355	15,093	−11,072 (43,476)
human	1,169,083	2,139,559	13,252	491	−142	−763 (169)
mouse	1,316,058	2,142,500	11,410	−1	−817	−1,332 (233)
zebrafish	1,549,373	2,140,200	2,928	−387	−386	−508 (16)

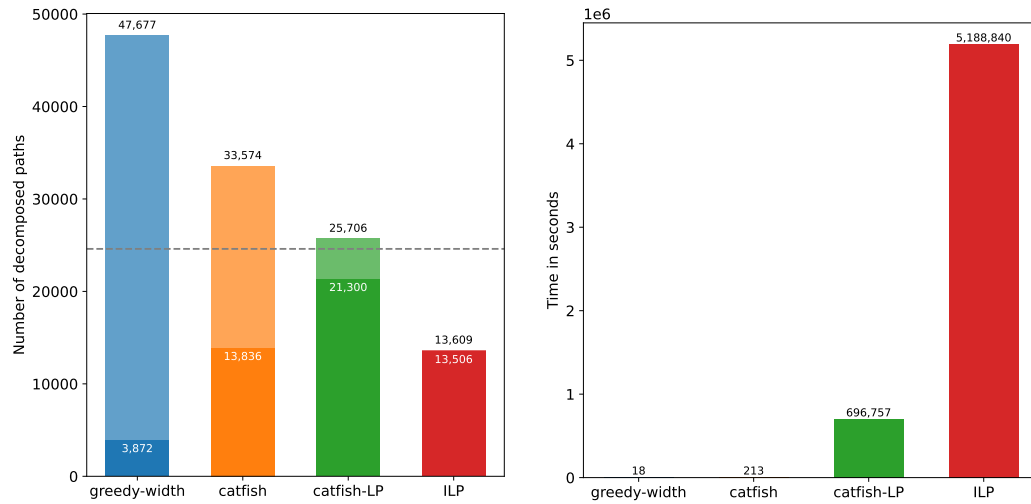


Figure 1 Decomposition quality and runtime on the full simulated dataset (1,440 graphs across 72 configurations). In the left subfigure, the dashed horizontal line marks the total number of ground-truth paths across all instances ($|P^\dagger| = 24,600$). For each method, the full bar represents the total number of paths produced, while the shaded portion denotes the number of decomposed paths that exactly match a ground-truth path. The ILP bar is heavily skewed because ILP failed to decompose 498 graph instances (35%) within the time limit.

References

- 1 Jasmijn A Baaijens, Leen Stougie, and Alexander Schönhuth. Strain-aware assembly of genomes from mixed samples using flow variation graphs. In Russell Schwartz, editor, *International Conference on Research in Computational Molecular Biology*, Lecture Notes in Computer Science, pages 221–222. Springer, 2020. doi:10.1007/978-3-030-45257-5_14.
- 2 Manuel Cáceres, Massimo Cairo, Andreas Grigorjew, Shahbaz Khan, Brendan Mumey, Romeo Rizzi, Alexandru I Tomescu, and Lucia Williams. Width helps and hinders splitting flows. *ACM Transactions on Algorithms*, 20(2):1–20, 2024. doi:10.1145/3641820.
- 3 Fernando HC Dias, Manuel Cáceres, Lucia Williams, Brendan Mumey, and Alexandru I Tomescu. A safety framework for flow decomposition problems via integer linear programming. *Bioinformatics*, 39:btad640, 2023. doi:10.1093/bioinformatics/btad640.
- 4 Fernando HC Dias and Alexandru I Tomescu. Accurate flow decomposition via robust integer linear programming. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 21:1955–1964, 2024. doi:10.1109/TCBB.2024.3433523.
- 5 Fernando HC Dias, Lucia Williams, Brendan Mumey, and Alexandru I Tomescu. Efficient minimum flow decomposition via integer linear programming. *Journal of Computational Biology*, 29(11):1252–1267, 2022. doi:10.1089/cmb.2022.0257.
- 6 Fernando HC Dias, Lucia Williams, Brendan Mumey, and Alexandru I Tomescu. Minimum flow decomposition in graphs with cycles using integer linear programming. *Journal of Global Optimization*, 93:1145–1176, 2025. doi:10.1007/s10898-025-01556-8.
- 7 Daniel Gibney, V Sharma, Srinivas Aluru, et al. Feasibility of flow decomposition with subpath constraints in linear time. In Christina Boucher and Sven Rahmann, editors, *22nd International Workshop on Algorithms in Bioinformatics (WABI 2022)*, volume 242 of *LIPIcs*, pages 17:1–17:16. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.WABI.2022.17.
- 8 Thasso Griebel, Benedikt Zacher, Paolo Ribeca, Emanuele Raineri, Vincent Lacroix, Roderic Guigó, and Michael Sammeth. Modelling and simulating generic RNA-Seq experiments with the flux simulator. *Nucleic acids research*, 40(20):10073–10083, 2012.

- 9 Andreas Grigorjew, Fernando HC Dias, Andrea Cracco, Romeo Rizzi, and Alexandru I Tomescu. Accelerating ILP solvers for minimum flow decompositions through search space and dimensionality reductions. In Leo Liberti, editor, *22nd International Symposium on Experimental Algorithms (SEA 2024)*, volume 301 of *LIPIcs*, pages 14:1–14:19. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.SEA.2024.14.
- 10 Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2024. URL: <https://www.gurobi.com>.
- 11 Mitchell Guttman, Manuel Garber, Joshua Z Levin, Julie Donaghey, James Robinson, Xian Adiconis, Lin Fan, Magdalena J Koziol, Andreas Gnirke, Chad Nusbaum, et al. Ab initio reconstruction of cell type-specific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs. *Nature Biotechnology*, 28(5):503–510, 2010.
- 12 Tzvika Hartman, Avinatan Hassidim, Haim Kaplan, Danny Raz, and Michal Segalov. How to split a flow? In Albert G. Greenberg and Kazem Sohraby, editors, *2012 Proceedings IEEE INFOCOM*, pages 828–836. IEEE, IEEE, 2012. doi:10.1109/INFOCOM.2012.6195830.
- 13 Shahbaz Khan, Milla Kortelainen, Manuel Cáceres, Lucia Williams, and Alexandru I Tomescu. Improving RNA assembly via safety and completeness in flow decompositions. *Journal of Computational Biology*, 29(12):1270–1287, 2022. doi:10.1089/cmb.2022.0261.
- 14 Kyle Kloster, Philipp Kuinke, Michael P O’Brien, Felix Reidl, Fernando Sánchez Villaamil, Blair D Sullivan, and Andrew van der Poel. A practical FPT algorithm for flow decomposition and transcript assembly. In Rasmus Pagh and Suresh Venkatasubramanian, editors, *2018 Proceedings of the Twentieth Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 75–86. SIAM, SIAM, 2018. doi:10.1137/1.9781611975055.7.
- 15 Sam Kovaka, Aleksey V Zimin, Geo M Pertea, Roham Razaghi, Steven L Salzberg, and Mihaela Pertea. Transcriptome assembly from long-read RNA-Seq alignments with StringTie2. *Genome Biology*, 20:1–13, 2019.
- 16 Juntao Liu, Ting Yu, Tao Jiang, and Guojun Li. TransComb: genome-guided transcriptome assembly via combing junctions in splicing graphs. *Genome Biology*, 17:1–9, 2016.
- 17 Rob Patro, Geet Duggal, Michael I Love, Rafael A Irizarry, and Carl Kingsford. Salmon provides fast and bias-aware quantification of transcript expression. *Nature methods*, 14(4):417–419, 2017.
- 18 Mihaela Pertea, Geo M Pertea, Corina M Antonescu, Tsung-Cheng Chang, Joshua T Mendell, and Steven L Salzberg. StringTie enables improved reconstruction of a transcriptome from RNA-Seq reads. *Nature Biotechnology*, 33(3):290–295, 2015.
- 19 Mingfu Shao and Carl Kingsford. Accurate assembly of transcripts through phase-preserving graph decomposition. *Nature Biotechnology*, 35(12):1167–1169, 2017.
- 20 Mingfu Shao and Carl Kingsford. Theory and a heuristic for the minimum path flow decomposition problem. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 16(2):658–670, 2017. doi:10.1109/TCBB.2017.2779509.
- 21 Jim Shaw, Jean-Sebastien Gounot, Hanrong Chen, Niranjan Nagarajan, and Yun William Yu. Floria: fast and accurate strain haplotyping in metagenomes. *Bioinformatics*, 40(Supplement_1):i30–i38, 2024. doi:10.1093/bioinformatics/btae252.
- 22 Li Song, Sarven Sabuncuyan, and Liliana Florea. CLASS2: accurate and efficient splice variant annotation from RNA-Seq reads. *Nucleic Acids Research*, 44(10):e98–e98, 2016.
- 23 Alexandru I Tomescu, Anna Kuosmanen, Romeo Rizzi, and Veli Mäkinen. A novel min-cost flow method for estimating transcript expression with RNA-Seq. *BMC Bioinformatics*, 14(S15):1–10, 2013. doi:10.1186/1471-2105-14-S5-S15.
- 24 Cole Trapnell, Brian A Williams, Geo Pertea, Ali Mortazavi, Gordon Kwan, Marijke J Van Baren, Steven L Salzberg, Barbara J Wold, and Lior Pachter. Transcript assembly and quantification by rNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology*, 28(5):511–515, 2010.

- 25 Laura H Tung, Mingfu Shao, and Carl Kingsford. Quantifying the benefit offered by transcript assembly with Scallop-LR on single-molecule long reads. *Genome Biology*, 20:1–18, 2019.
- 26 Benedicte Vatinlen, Fabrice Chauvet, Philippe Chrétienne, and Philippe Mahey. Simple bounds and greedy algorithms for decomposing a flow into a minimal set of paths. *European Journal of Operational Research*, 185(3):1390–1401, 2008. doi:10.1016/j.ejor.2006.05.043.
- 27 Lucia Williams, Gillian Reynolds, and Brendan Mumey. Rna transcript assembly using inexact flows. In Illhoi Yoo, Jinbo Bi, and Xiaohua Hu, editors, *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1907–1914. IEEE, IEEE, 2019. doi:10.1109/BIBM47256.2019.8983180.
- 28 Lucia Williams, Alexandru I Tomescu, and Brendan Mumey. Flow decomposition with subpath constraints. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20(1):360–370, 2022. doi:10.1109/TCBB.2022.3147697.
- 29 Qimin Zhang, Qian Shi, and Mingfu Shao. Accurate assembly of multi-end RNA-Seq data with Scallop2. *Nature Computational Science*, 2(3):148–152, 2022. doi:10.1038/s43588-022-00216-1.