

Faster and Efficient Density Decomposition via Proportional Response with Exponential Momentum

QUAN XUE and T-H. HUBERT CHAN*, The University of Hong Kong, China

Graphs are crucial for modeling complex relationships in fields like social networks and biology. A key aspect in graph theory is identifying dense subgraphs, with applications in various domains. Density decomposition refines this by analyzing a graph's global density structure. This concept has been independently rediscovered in research areas like graph mining, algorithm design, and economics.

Recent advancements in maximum-flow algorithms allow for nearly-linear time computation of the density vector, but they struggle with large real-world graphs. To address this, iterative first-order methods based on convex optimization, such as the Frank-Wolfe algorithm and momentum-based methods like accelerated FISTA, have been developed to approximate density vectors.

This work explores density decomposition through market dynamics, where edges represent buyers and nodes represent sellers in a Fisher market model. The iterative proportional response process, which converges to the Fisher market equilibrium, provides an alternative interpretation of density decomposition. In each step, agents allocate resources based on the proportion of benefit received, moving the system toward equilibrium.

Since each proportional response update can be seen as a gradient descent step, we investigate whether momentum methods can speed up convergence. Traditional momentum uses a linear combination of current and previous solutions, whereas our novel exponential momentum variant uses geometric interpolation, aligning better with proportional adjustments. Empirical evaluations on large-scale real-world and synthetic graphs confirm the effectiveness of our methods. Notably, the proportional response algorithm with exponential momentum outperforms existing methods, delivering improvements by several orders of magnitude in some cases. This advancement is significant, as the resulting density vector is crucial for many downstream tasks in graph mining and algorithm design.

CCS Concepts: • **Networks** → **Network algorithms**; **Network structure**; • **Theory of computation** → **Graph algorithms analysis**; **Graph algorithms analysis**; • **Mathematics of computing** → **Graph algorithms**.

Additional Key Words and Phrases: Densest Subgraph, Density Decomposition, Proportional Response, Momentum Methods

ACM Reference Format:

Quan Xue and T-H. Hubert Chan. 2025. Faster and Efficient Density Decomposition via Proportional Response with Exponential Momentum. *Proc. ACM Manag. Data* 3, 3 (SIGMOD), Article 161 (June 2025), 3 pages. <https://doi.org/10.1145/3725405>

*This research was partially supported by the Hong Kong RGC grants 17201823, 17203122 and 17202121.

Authors' Contact Information: Quan Xue, csxuequan@connect.hku.hk; T-H. Hubert Chan, hubert@cs.hku.hk, The University of Hong Kong, Hong Kong, China.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM 2836-6573/2025/6-ART161
<https://doi.org/10.1145/3725405>

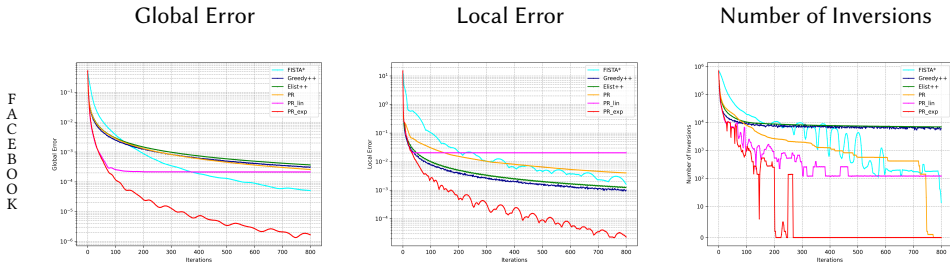


Fig. 1. Approximation Quality vs Number of Iterations: Selected Normal Graphs

Algorithm Time Comparison (seconds)

Algorithm	Max (10000)
FISIX	0.0000
Greedy++	0.0000
EIS++	0.0000
PR	0.0000
PR_in	0.0000
PR_out	0.0000

Algorithm	Size
FISIX	10.21 MB
Greedy++	10.21 MB
EIS++	10.21 MB
PR	10.21 MB
PR_in	10.21 MB
PR_out	10.21 MB

(a) Running Time and Memory Usage vs Graph Size on Normal Graphs

Algorithm Time Comparison (seconds)

Algorithm	Max (10000)
FISIX	0.0000
Greedy++	0.0000
EIS++	0.0000
PR	0.0000
PR_in	0.0000
PR_out	0.0000

Algorithm	Size
FISIX	10.21 MB
Greedy++	10.21 MB
EIS++	10.21 MB
PR	10.21 MB
PR_in	10.21 MB
PR_out	10.21 MB

(b) Running Time and Memory Usage vs Graph Size on Double Covers

Fig. 2

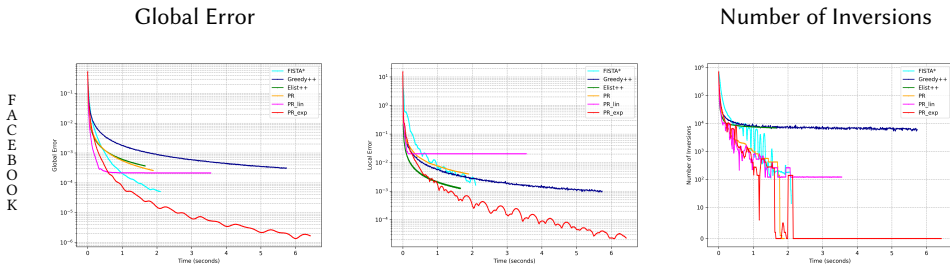


Fig. 3. Approximation Quality vs Simulated Wall Clock Time: Selected Normal Graphs

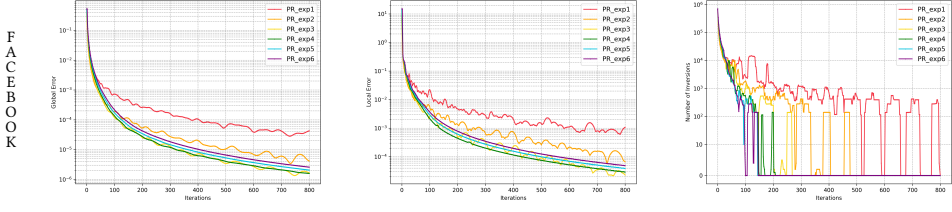


Fig. 4. Approximation Quality vs Number of Iterations for PR_EXP with Different C in $\gamma_t = 1 - \frac{C}{t+C}$, for $C = 1, 2, \dots, 6$

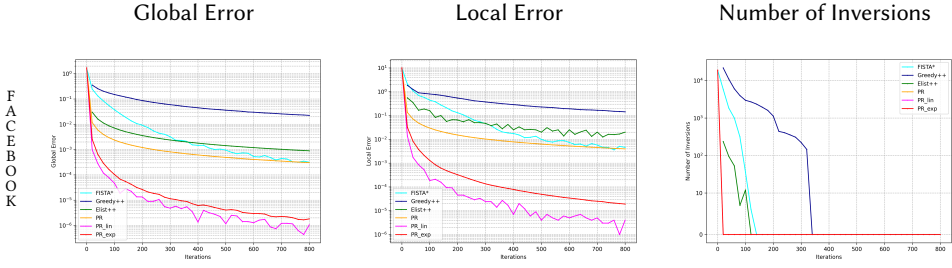


Fig. 5. Approximation Quality vs Number of Iterations: Selected Double Covers

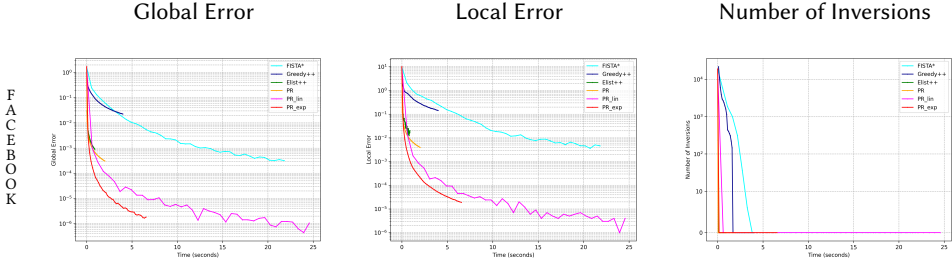


Fig. 6. Approximation Quality vs Simulated Wall Clock Time: Selected Double Covers