Pairing T Cell Receptor α and β Sequences using Pooling and Min-cost Flows

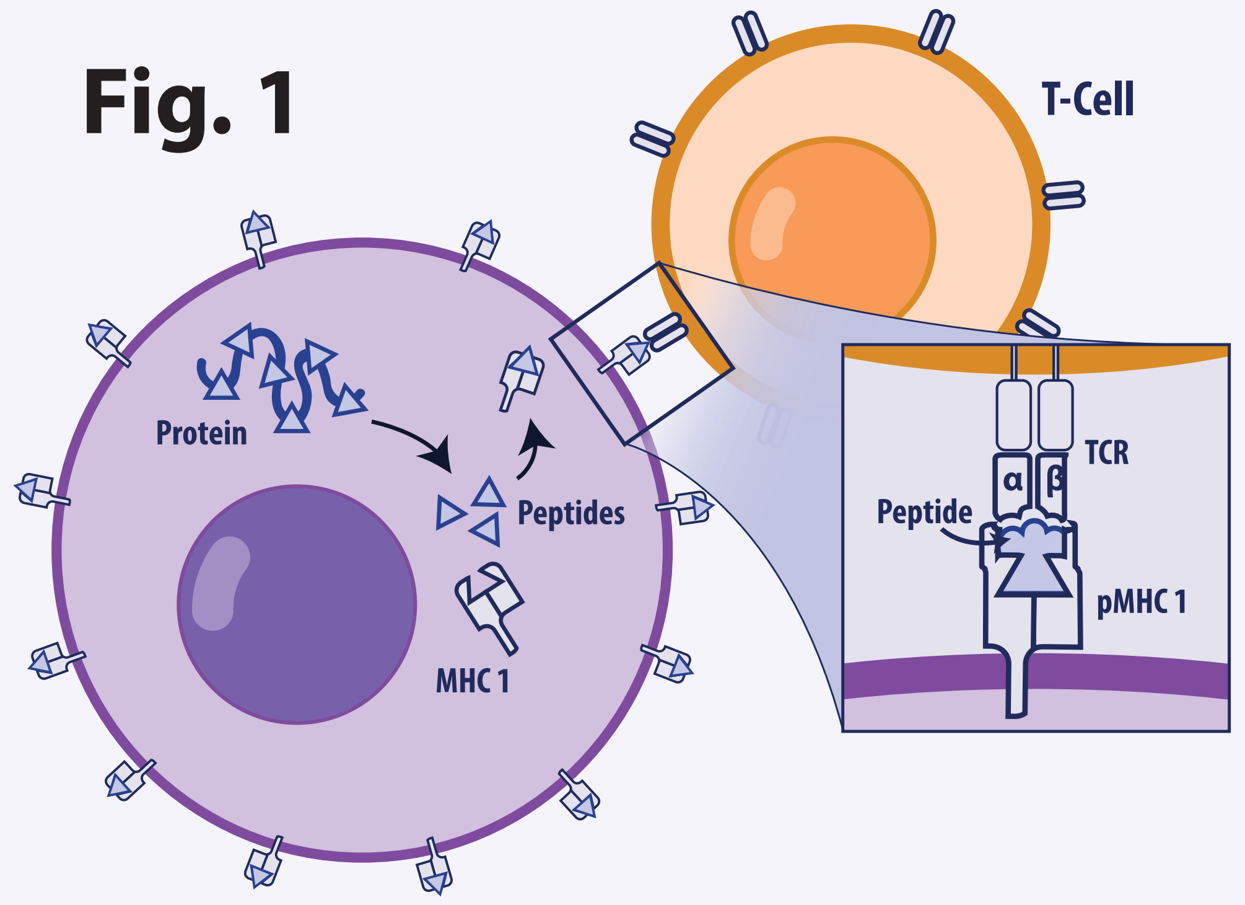
Tyler Daddio, Ion I. Măndoiu

Computer Science and Engineering Dept., University of Connecticut

Introduction

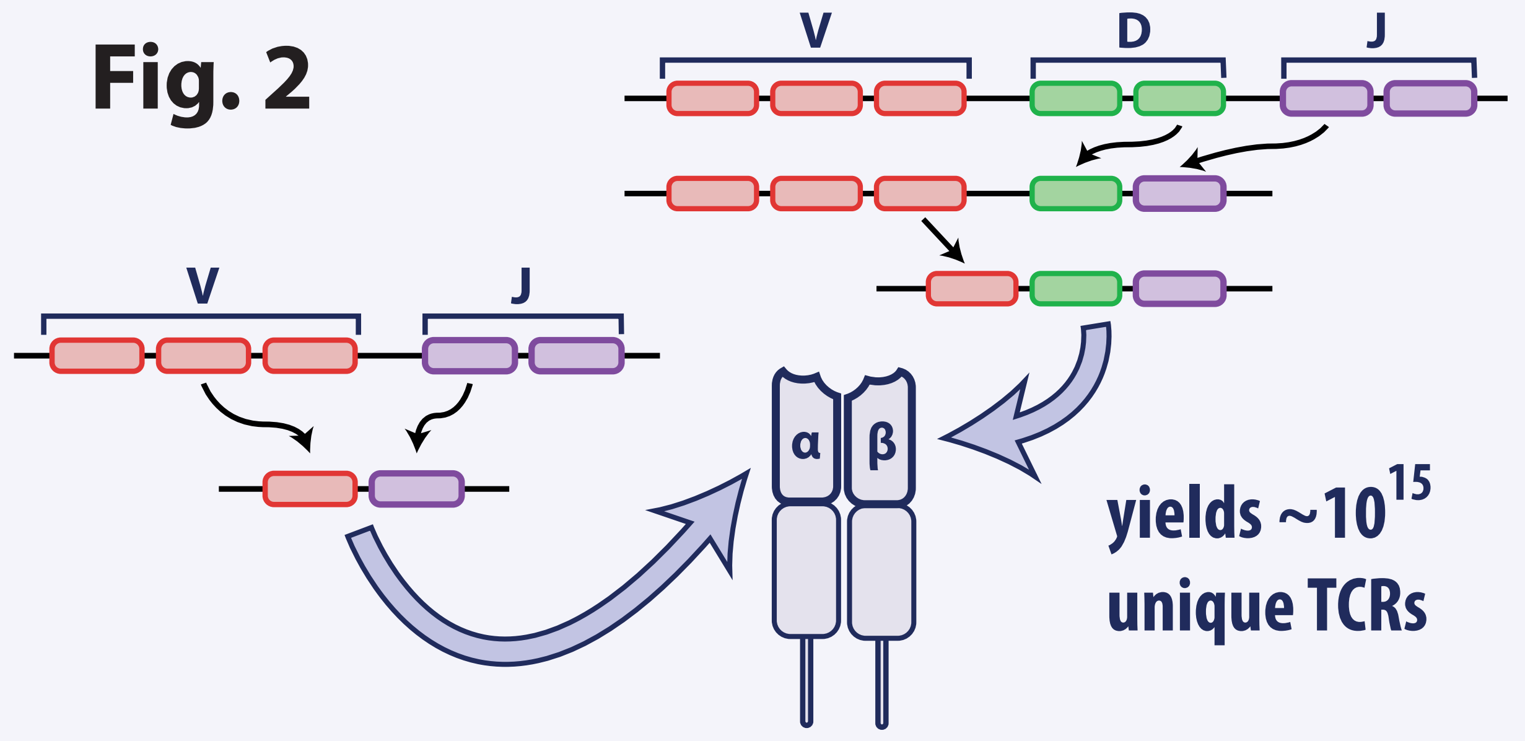
The T-cell receptor (TCR) is a protein heterodimer composed of an α (alpha) chain and a β (beta) chain, both of which contribute to the receptor's ability to recognize specific antigens **(Fig. 1)**. The α and β chains are encoded by genes that undergo somatic DNA recombination during T-cell development, a process that can potentially yield over 10^{15}distinct αβ combinations **(Fig. 2)**. This staggering diversity of the TCR repertoire is critical for T-cell differentiation between self and foreign antigens and mounting effective adaptive immune responses against infectious agents and cancer neoepitopes.

Fig. 1



Whereas independently sequencing the recombined α and β sequences has become routine, comprehensive characterization of the paired αβ repertoire remains challenging. De novo assembly of α and β sequences from single T-cell RNA-Seq has been demonstrated [1] but has low throughput and high cost. Emulsion-based sequencing of molecularly linked α and β amplicons generated in parallel from single T-cells promises much higher throughput [2], but protocols are still under active development. A cost-effective alternative relying on standard single-chain TCR sequencing is the pairSEQ protocol proposed in [3]. In this protocol millions of T-cells are distributed randomly across 96 wells. Barcoded PCR amplicons spanning the hyper-variable CDR3 regions of the α and β genes are then generated from each well and pooled for highthroughput sequencing **(Fig. 3)**.

Fig. 2



Methods

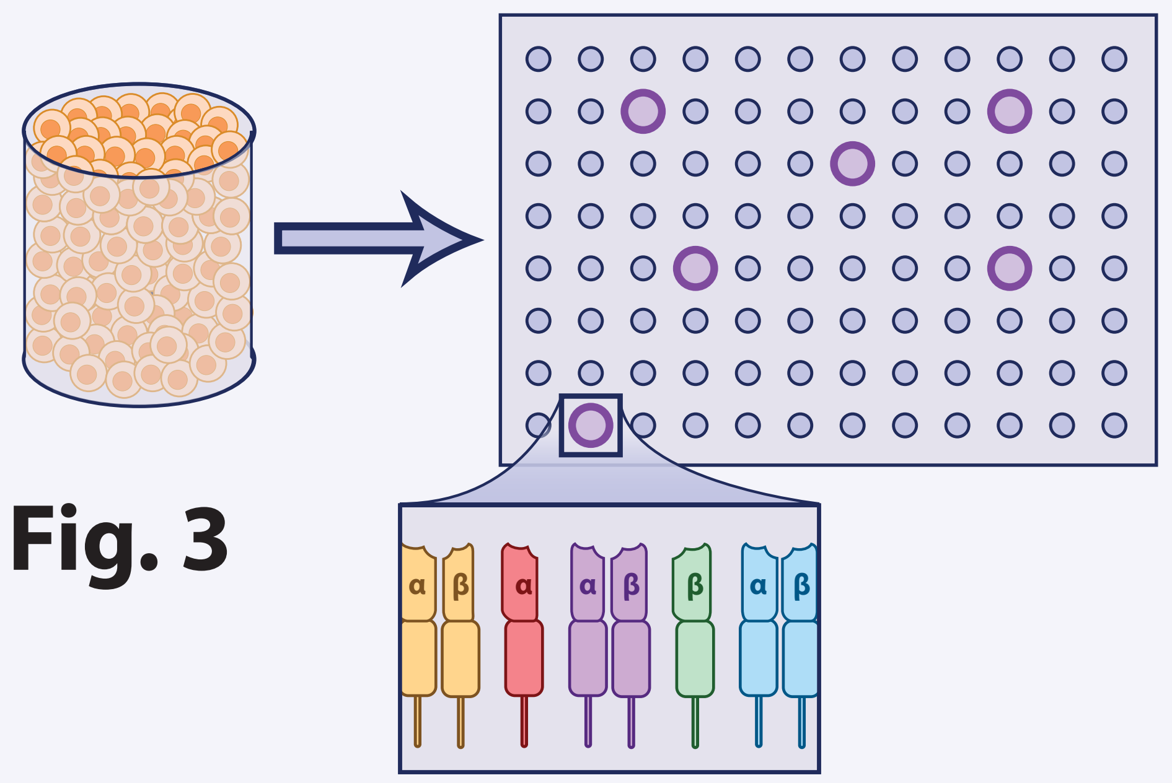


Fig. 3

In [3], putative pairs of α and β sequences were generated using a simple binomial model. In this work, we model αβ pairing as a perfect b-matching problem in a bipartite graph. An identical experiental design as in [3] is used.

For each α and β sequence we compute the set of wells it appears in. The resulting well sets are used as nodes in a bipartite graph, with additional "dummy" nodes added to allow for unmatched sequences. Well sets A and B corresponding to a pair of α and β sequences are connected by an arc whose weight is equal to the hamming distance between them.

To keep this approach computationally feasible, only a small subset of all possible arcs can be included in the graph. The most effective method of sparsification investigated thus far was ***relative radius sparsification***. Given a parameter r (namely, the *radius*), arcs are only added the final graph if the following equation holds:

$$|A \Delta B| \leq r \text{ x } \max(|A_i|, |B_j|)$$

This criterion will substantially reduce the size of the graph, but an efficient approach is still needed to identify these arcs and avoid searching the entire space. These arcs are generated using an efficient algorithm based on multi-index hashing [4]. The ensuing perfect b-matching problem was solved using the cost-scaling min-cost max-flow algorithm implemented in the LEMON library [5] **(Fig. 4)**. The results of this matching are discussed in the next section.

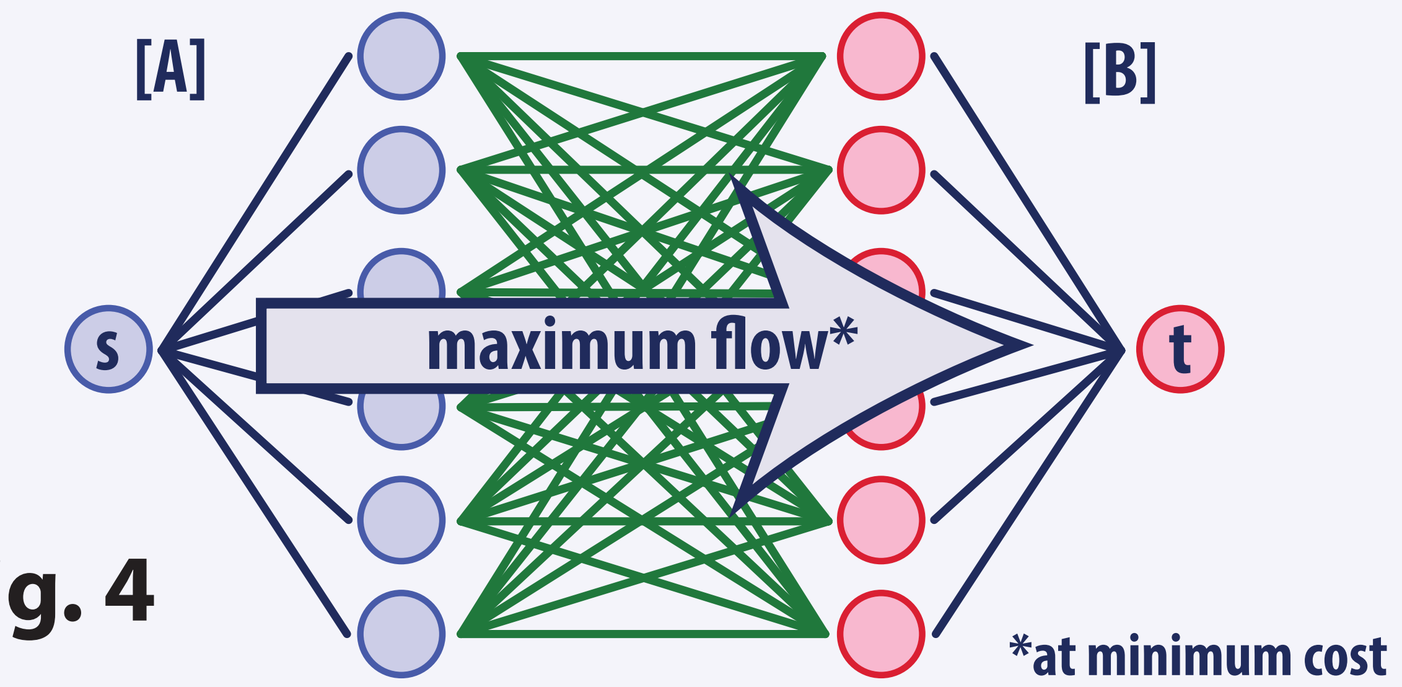
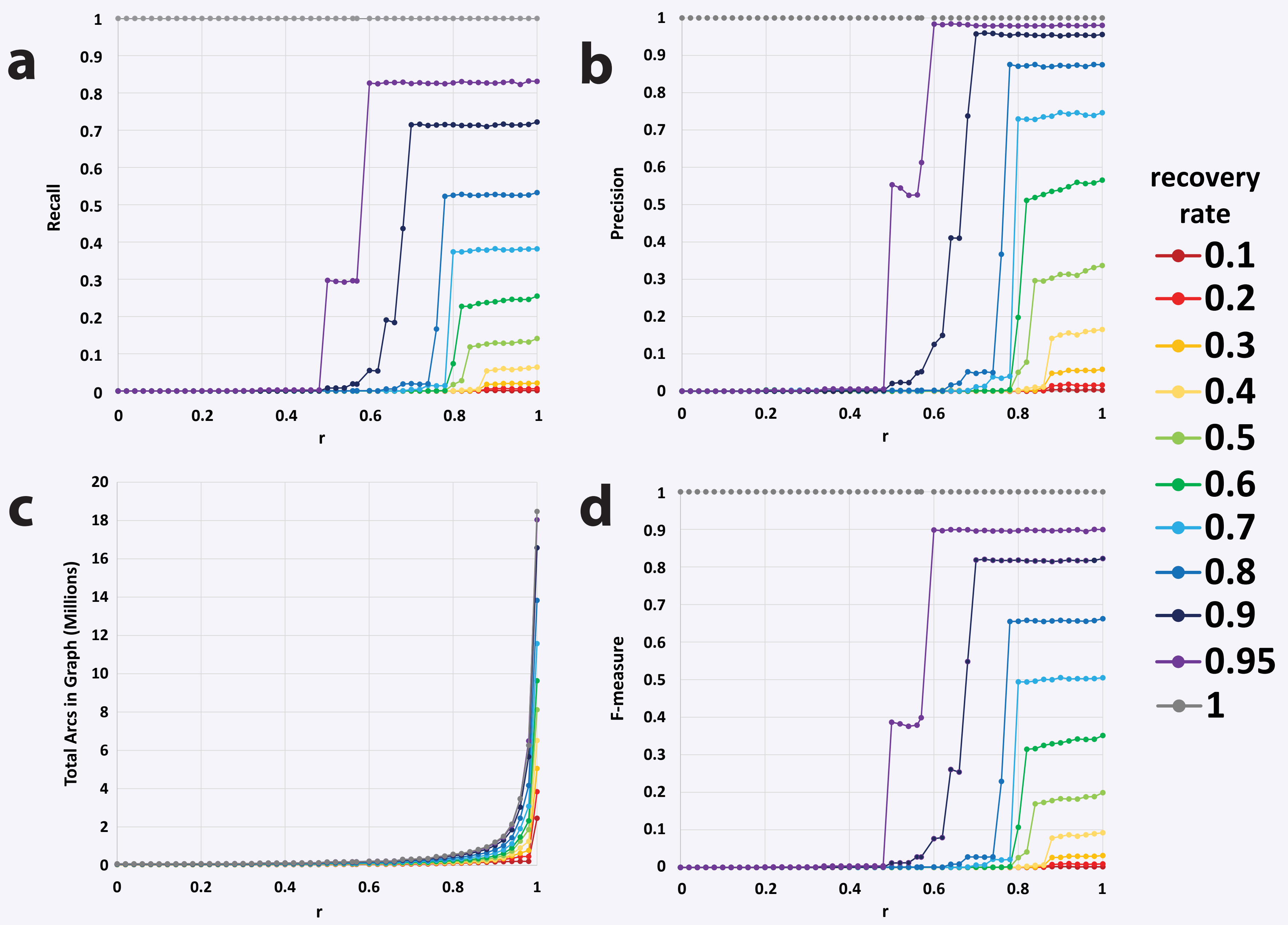


Fig. 4

*at minimum cost

Preliminary Results

Fig. 6



As shown in **Fig. 6**, the overall accuracy of our approach undergoes a phase transition. The reason for this is currently unknown, but it is reminiscent of the phase transitions observed in the well known Erdos-Renyi graphs. The value r at which this phase transition seems to vary with the simulated recovery rate, but once exceeding this crtical radius, the effectiveness of the method plateaus and no substantial improvement is observed. Although not included here, experiment 1 also experienced this behavior.

The effectiveness of the relative radius sparsification method is evident from **Fig. 6(c)**. At r=0.8, we see a 99.75% reduction in the number of arcs in the graph and a 99.5% reduction at r=0.9. As expected, both the recall and precision of our pair identification method depends largely on the recovery rate, but at r=0.9, this method is still maximally effective. These results are promising, but more work is still required to better understand the behavior of this method.

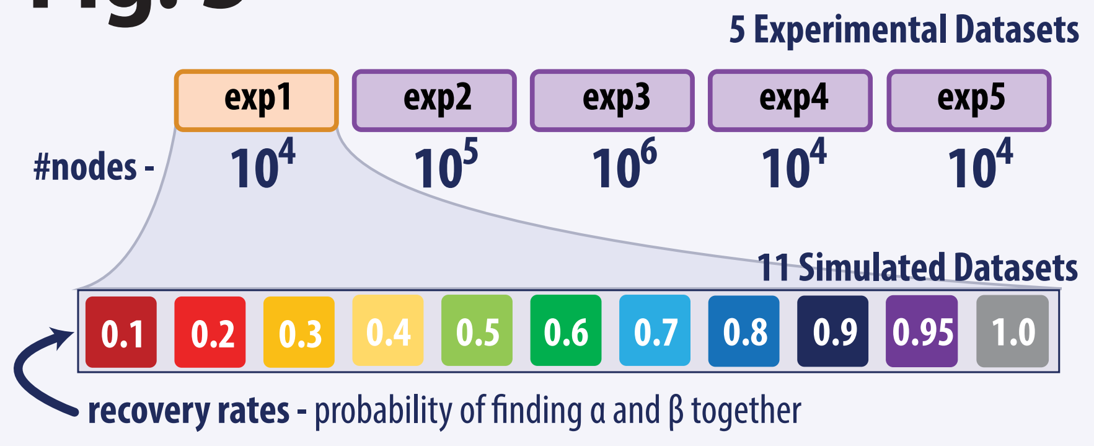
Future Work

1. Analyze remaining experimental datasets
2. Test method resilience to sequence distribution
3. Optimize sparsification using threading
4. Investigate the nature of this phase transition

Datasets

Experiments were conducted on simulated datasets generated by randomly pairing 1,495,345 α sequences with 1,566,719 β sequences extracted from pairSEQ experiment 1 in [3]. We simulated uniform sequencing ***recovery rates*** between 10–100% to reflect the fact that some receptor sequences are lost during PCR amplification and sequencing.

Fig. 5



recovery rates - probability of finding α and β together

References

[1] M. J. T. Stubbington, T. Lonnberg, V. Proserpio, S. Clare, A. O. Speak, G. Dougan, and S. A. Teichmann, "T cell fate and clonality inference from single-cell transcriptomes," Nat Meth, vol. 13, no. 4, pp. 329–332, 2016.

[2] M. A. Turchaninova, O. V. Britanova, D. A. Bolotin, M. Shugay, E. V. Putintseva, D. B. Staroverov, G. Sharonov, D. Shcherbo, I. V. Zvyagin, I. Z. Mamedov, C. Linnemann, T. N. Schumacher, and D. M. Chudakov, "Pairing of t-cell receptor chains via emulsion pcr," European Journal of Immunology, vol. 43, no. 9, pp. 2507–2515, 2013.

[3] B. Howie, A. M. Sherwood, A. D. Berkebile, J. Berka, R. O. Emerson, D. W. Williamson, I. Kirsch, M. Vignali, M. J. Rieder, C. S. Carlson, and H. S. Robins, "High-throughput pairing of T cell receptor α and β sequences," Science Translational Medicine, vol. 7, no. 301, pp. 301ra131–301ra131, 2015.

[4] M. Norouzi, A. Punjani, and D. J. Fleet, "Fast exact search in hamming space with multi-index hashing," CoRR, vol. abs/1307.2982, 2013.

[5] B. Dezs, A. J¨uttner, and P. Kov´acs, "LEMON - an open source C++ graph template library," Electron. Notes Theor. Comput. Sci., vol. 264, no. 5, pp. 23–45, Jul. 2011.