



## OPEN ACCESS

## EDITED BY

Zizhang Sheng,  
Columbia University Irving Medical  
Center, United States

## REVIEWED BY

Chaim A. Schramm,  
National Institute of Allergy and  
Infectious Diseases (NIH),  
United States  
Victor Greiff,  
University of Oslo, Norway  
Juliana Silva Bernardes,  
Sorbonne Universités, France

## \*CORRESPONDENCE

Siavash Mirarab  
smirarab@ucsd.edu

## SPECIALTY SECTION

This article was submitted to  
B Cell Biology,  
a section of the journal  
Frontiers in Immunology

RECEIVED 08 August 2022

ACCEPTED 26 October 2022

PUBLISHED 06 December 2022

## CITATION

Zhang C, Bzikadze AV, Safonova Y and  
Mirarab S (2022) A scalable model for  
simulating multi-round antibody  
evolution and benchmarking of clonal  
tree reconstruction methods.  
*Front. Immunol.* 13:1014439.  
doi: 10.3389/fimmu.2022.1014439

## COPYRIGHT

© 2022 Zhang, Bzikadze, Safonova and  
Mirarab. This is an open-access article  
distributed under the terms of the  
[Creative Commons Attribution License  
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or  
reproduction in other forums is  
permitted, provided the original  
author(s) and the copyright owner(s)  
are credited and that the original  
publication in this journal is cited, in  
accordance with accepted academic  
practice. No use, distribution or  
reproduction is permitted which does  
not comply with these terms.

# A scalable model for simulating multi-round antibody evolution and benchmarking of clonal tree reconstruction methods

Chao Zhang<sup>1</sup>, Andrey V. Bzikadze<sup>1</sup>, Yana Safonova<sup>2</sup>  
and Siavash Mirarab<sup>3\*</sup>

<sup>1</sup>Bioinformatics and Systems Biology, University of California, San Diego, San Diego, CA, United States, <sup>2</sup>Computer Science and Engineering Department, University of California, San Diego, San Diego, CA, United States, <sup>3</sup>Electrical and Computer Engineering Department, University of California, San Diego, San Diego, CA, United States

Affinity maturation (AM) of B cells through somatic hypermutations (SHMs) enables the immune system to evolve to recognize diverse pathogens. The accumulation of SHMs leads to the formation of clonal lineages of antibody-secreting B cells that have evolved from a common naïve B cell. Advances in high-throughput sequencing have enabled deep scans of B cell receptor repertoires, paving the way for reconstructing clonal trees. However, it is not clear if clonal trees, which capture microevolutionary time scales, can be reconstructed using traditional phylogenetic reconstruction methods with adequate accuracy. In fact, several clonal tree reconstruction methods have been developed to fix supposed shortcomings of phylogenetic methods. Nevertheless, no consensus has been reached regarding the relative accuracy of these methods, partially because evaluation is challenging. Benchmarking the performance of existing methods and developing better methods would both benefit from realistic models of clonal lineage evolution specifically designed for emulating B cell evolution. In this paper, we propose a model for modeling B cell clonal lineage evolution and use this model to benchmark several existing clonal tree reconstruction methods. Our model, designed to be extensible, has several features: by evolving the clonal tree and sequences simultaneously, it allows modeling selective pressure due to changes in affinity binding; it enables scalable simulations of large numbers of cells; it enables several rounds of infection by an evolving pathogen; and, it models building of memory. In addition, we also suggest a set of metrics for comparing clonal trees and measuring their properties. Our results show that while maximum likelihood phylogenetic reconstruction methods can fail to capture key features of clonal tree expansion if applied naively, a simple post-processing of their results, where short branches are contracted, leads to inferences that are better than alternative methods.

## KEYWORDS

antibody evolution, clonal trees, joint tree and sequence evolution, birth-death models, method benchmarking

# 1 Introduction

Immune response to new pathogens relies heavily on the Affinity maturation (AM) process. AM follows the binding of immunoglobulin (IG) molecules to antigens and improves the affinity (i.e., binding ability) of B cells to the antigen (1, 2). The AM process involves many aspects, including the activation of naive B cells that have not been exposed to an antigen, clonal expansion of cells that increases the pool of antibodies, somatic hypermutations (SHMs) (3) that alter the structure of antibodies and their ability to bind, and a regulatory mechanism that plays the role of natural selection. The AM process creates memory and plasma B cells; memory B cells can be reactivated and can undergo the AM process again (4), while plasma B cells can secrete massive levels of neutralizing antibodies. Over time, the AM process leads to the formation of clonal lineages within a given antibody repertoire, where each clonal lineage is formed by descendants of a single naive B cell. The evolutionary history of each of these clonal lineages can be represented by a clonal tree, where each vertex corresponds to a B cell, and a directed edge connects each B cell to all its immediate descendants.

New sequencing technologies have enabled high-throughput scanning of antibody repertoires (AIRR-seq) and have opened up new avenues for studying adaptive immune systems (5–9). AIRR-seq technologies enabled AM analysis of antibody repertoires responding to antigens of various diseases, such as flu (10, 11), HIV (12, 13), hepatitis (14, 15), multiple sclerosis (16, 17), rheumatoid arthritis (18). Such analyses allow biologists to identify broadly neutralizing antibodies and reveal antigen-specific and general mutation patterns (11, 19).

Due to the short time frame of clonal expansion, inferred clonal trees have unique properties (20). Some sequenced nodes may belong to the internal nodes of the tree instead of the tips. Also, inferred clonal trees are often not even close to bifurcating. Thus, unlike traditional phylogenetics, perhaps Steiner trees (which can put observations at some of the internal nodes) or spanning trees (that put an observation at all internal nodes) should be preferred for reconstructing antibody sequences (Figure 1A). Various reconstruction methods have been developed attempting to recover clonal trees from antibody sequences [e.g., (13, 21–26)]. Some of these methods use simple clustering methods [e.g., (21)], while others formulate the problem as a Steiner tree problem (13, 22, 24, 26) or maximum-likelihood (ML) phylogenetic tree reconstruction under models of sequence evolution (23, 25).

In order to evaluate methods proposed for reconstructing clonal trees, we need models for antibody sequence evolution and clonal tree expansion that can be used for simulation. This modeling step is challenging for several reasons. (i) Selection, which is an integral part of AM, needs to be modeled directly; otherwise, the shape of the resulting trees will not be realistic. Traditional phylogenetics simulations first simulate a tree of

sampled taxa and then evolve sequences down the tree. This two-step approach simplifies simulations but misses the dependency between the clonal tree shape and the antibody sequences. A better approach is to co-evolve the tree and sequences. The challenge in co-evolving is to design a principled model for how sequences impact evolution and to develop a scalable simulation algorithm that can generate large numbers of cells. (ii) Literature suggests that there are hotspots and cold spots of SHMs [e.g., (27, 28)]. However, traditional models of sequence evolution assume each site evolves independently and will miss the context dependence. (iii) Different antibody cell types (e.g., activated and memory cells) have very different mutational and selection behaviors, and these distinctions need to be modeled.

There have been several attempts at designing statistical models of AM clonal expansion [e.g., (20, 29–32)]. As the AM process is complex, these models have taken different routes. For example, determining affinities of sequences to hypothetical antigens is difficult, as affinity binding is a complicated chemical process, and each method models affinity differently. Nevertheless, all these methods have limitations, which we will return to in our discussion section. Two factors worth pointing out are that they do not scale to very large numbers of cells, and they allow for simulating one round of infection (as opposed to an evolving pathogen and recurring infections); some also avoid differentiating different types of B cells.

In this paper, we propose a scalable and flexible simulation framework that can be instantiated in many ways. We introduce a general birth, death, and transformation (BDT) model and describe how BDT can be instantiated to create a model of AM that simultaneously co-evolves the clonal tree and antibody sequences. We then introduce a scalable sampling algorithm for our model that enables generating large trees. With the simulator (called DIMSIM) at hand, we note that comparing clonal trees and characterizing their properties require care. We refine existing metrics and define new ones for characterizing properties (e.g., balance) of clonal trees and for comparing them. Finally, we perform extensive simulation studies (Figure 1B) under various parameters using DimSim. We study how the parameters of the AM model impact properties of clonal trees and benchmark the performance of several reconstruction methods.

## 2 Methods

### 2.1 Statistical models

We first define a general Birth/Death/Transformation (BDT) model and then present an efficient algorithm for sampling trees from that model. We then instantiate the general model for simulating AM processes and move on to describe specific choices we made in our simulations.

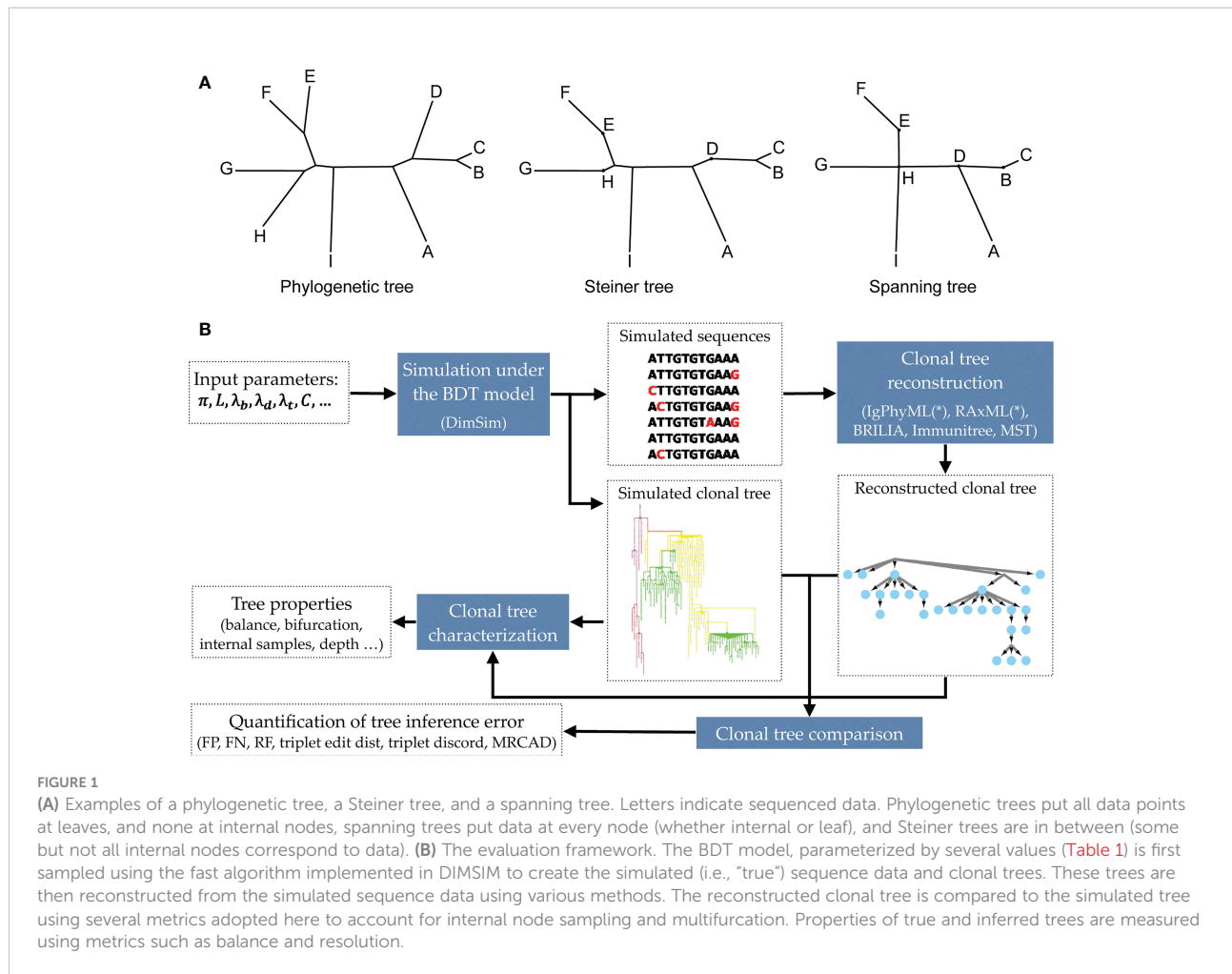


FIGURE 1

(A) Examples of a phylogenetic tree, a Steiner tree, and a spanning tree. Letters indicate sequenced data. Phylogenetic trees put all data points at leaves, and none at internal nodes, spanning trees put data at every node (whether internal or leaf), and Steiner trees are in between (some but not all internal nodes correspond to data). (B) The evaluation framework. The BDT model, parameterized by several values (Table 1) is first sampled using the fast algorithm implemented in DIMSIM to create the simulated (i.e., "true") sequence data and clonal trees. These trees are then reconstructed from the simulated sequence data using various methods. The reconstructed clonal tree is compared to the simulated tree using several metrics adopted here to account for internal node sampling and multifurcation. Properties of true and inferred trees are measured using metrics such as balance and resolution.

### 2.1.1 BDT model

Forward-time birth-death models are used extensively in macro-evolutionary modeling (33), whereas microevolution simulations often use coalescent models that are easier to sample. We propose a general forward-time model that can allow realistic microevolutionary simulations by ensuring that birth and death rates are not constant and instead change with the properties of evolving units (e.g., cells).

In the BDT model, a set of entities continuously undergo birth (B), death (D), and transformation (T) events. Each entity  $i$  has a list of properties  $\mathbf{x}_i \in \mathbb{R}_+^N$ . At each point in time, the system contains a set  $S$  of  $n$  active entities, and each active entity  $i \in S$  undergoes birth, death, and transformation events according to independent Poisson point processes. In the birth event, an entity  $i$  is removed from  $S$  and new entities  $j$  and  $k$ , with properties  $\mathbf{x}_j$  and  $\mathbf{x}_k$ , are added to  $S$ ; properties  $\mathbf{x}_j$  and  $\mathbf{x}_k$  are drawn from a distribution determined by  $\mathbf{x}_i$  and model parameters. In the event of the death of an entity  $i$ , it is removed from  $S$ . In the transformation event, an entity  $i$  is removed from  $S$  and a new entity  $j$  with properties  $\mathbf{x}_j$ , drawn from a distribution determined by  $\mathbf{x}_i$ , is added to  $S$ . Starting from

a single node and continuously applied, this process defines a rooted tree where nodes are all entities that ever existed (including those that died); birth events create bifurcations, transformation events create nodes with one child, and death events create leaves. The tree can be subsampled subsequently.

For each entity  $i \in S$ , the birth, rate, and transformation rates are thoroughly determined by its properties  $\mathbf{x}_i$  and the sum of properties over all entities  $S = \sum_{j \in S} \mathbf{x}_j$ . We let  $\Lambda_B(\mathbf{x}_i, S)$ ,  $\Lambda_D(\mathbf{x}_i, S)$ , and  $\Lambda_T(\mathbf{x}_i, S)$  denote the birth, death, and transformation rates, respectively. In the time interval between two events for any two entities in the system, we assume a memoryless process. Thus, these rates remain constant between any two events but can change when an event happens. The ratio between the birth rate and the death rate, both of which are functions of the entity properties, can be thought of as the factor controlling the selective pressure, which can be time-variant.

Because of the memoryless property, the time until the next BDT event always follows the exponential distribution with rates  $\Lambda_B(\mathbf{x}_i, S)$ ,  $\Lambda_D(\mathbf{x}_i, S)$ , and  $\Lambda_T(\mathbf{x}_i, S)$  for each event type. The time until any event for any entity follows an exponential distribution with  $\lambda = \sum_{i \in S} (\Lambda_B(\mathbf{x}_i, S) + \Lambda_D(\mathbf{x}_i, S) + \Lambda_T(\mathbf{x}_i, S))$ . The probability

of the next event being a specific event  $E \in \{B, D, T\}$  for a particular entity  $i$  is  $\Lambda_E(\mathbf{x}_i, \mathbf{S})/\lambda$ . Specifying the rate functions and the distribution of properties at the initial state fully specifies the model.

The BDT model can be efficiently sampled if rate functions have certain (very general) properties. We leave all the mathematical details for [Appendix 2.1](#). In short, the memoryless property of the model makes it possible to perform efficient sampling despite the fact that rates change with the tree. The main innovations of the sampling algorithm are 1) rewriting rate functions as polynomial functions of other parameters, which enable finding the time to the next event in constant time, and 2) using an interval tree data structure to store partial sums needed for normalization. With our proposed [Algorithm S1](#) (in [Appendix 4](#)), a tree on  $k$  nodes drawn from the distribution defined by the BDT process can be sampled in  $O(k \log(k))$  time. Thus, the BDT model can be efficiently sampled to create trees with millions of nodes.

## 2.1.2 Antibody affinity maturation model

We now define a specific instance of the BDT model designed for AM. Simulations according to this AM model are implemented in a C++ tool called Dynamic IMMuno-SIMulator

(DIMSIM). The model has many parameters reflecting immune system properties ([Table 1](#)), which we define throughout this section ([Appendix 1](#) defines our particular usage of terms commonly used in immunology). The use of birth death models for AM is not new [e.g., (32)] but particular choices of our model are different from prior work.

### 2.1.2.1 Rounds and stages

We model the evolution of antibody-coding sequences in response to  $r$  rounds of infections by an evolving antigen (e.g., SARS-Cov2 or flu). Each round consists of two stages, an infected stage, where a set of new antigens initiate a response that activates the B cells being modeled, and a dormant stage, where the B cells being modeled are not actively involved in an immune response. Both stages used the same BDT model but are parameterized differently. The switch between the two stages happens through user-defined rules (e.g., rules that reflect infection progression as described below). During the infected stage of round  $i$ , we target amino-acid sequence  $\zeta_i = (\zeta_i^{(1)}, \dots, \zeta_i^{(L)})$  of length  $L$  without any stop codon, defined as the best possible antibody-coding sequence that can bind to the present antigen. The target can change across rounds, and we describe two specific ways of choosing targets in [Section 2.1.3](#).

TABLE 1 Parameters of the AM model.

Param.	Default	Parameter description
$\lambda'_d$	1/402	Rate (inverse life time) of cell death for memory cells (days <sup>-1</sup> )
$\lambda_b$	6	Rate of cell division for activated B cells (days <sup>-1</sup> )
$\lambda_d$	10 <sup>4</sup>	Rate of cell death during dormant stage (day <sup>-1</sup> ).
$\lambda_t$	0.01	Rate of activation of a typical responsive memory cell
$\rho_p$	1/100	Portion of activated B cells that turn into plasma cells per cell division
$\rho_m$	1/4	Portion of activated B cells that turn into memory B cells per cell division
$\mu$	$5 \times 10^{-4}$	Rate of SHMs per base pair per generation
$\mathbf{K}^5$	<a href="#">Appendix 2</a>	Empirical 5-mer mutation frequencies per generation
$L$	125	Length of the amino acid antibody-coding sequence (assuming the length is fixed)
CDR	31–35, 50–65, 98–114	Positions of the three CDR regions (amino acid coordinates)
$\delta(i, j)$	<a href="#">Table S2</a>	BLOSUM matrix defined on a pair of amino-acids $i$ and $j$
$\Delta_0$	-120	BLOSUM score of a typical memory B cell antibody-coding sequence to target
$\Delta'_0$	-75	BLOSUM score of activated B cell antibody-coding sequences that leads to cure
$w_f$	1/3	BLOSUM score multiplier of non-CDR positions (i.e., FRs)
$\kappa$	2	BLOSUM score ratio of antibody-coding sequences to antigen sequences
$A$	0.1	Selective pressure: factor connecting sequence similarity and log binding affinity
$\rho_a$	1/2	Factor connecting log affinity and B cell activation (sensitivity to affinity level $A$ )
$C$	10 <sup>5</sup>	Carrying capacity limited by total resources (see text for meaning)
$M$	$Ce^{A\Delta'_0}$	The threshold of the sum of affinity for a stage change
$r$		Rounds of viral infections
$\hat{\psi}$	<a href="#">Appendix 2</a>	Nucleotide sequence of the initial B cell
$\zeta_1 \dots \zeta_r$	<a href="#">Appendix 2</a>	Target amino acid sequences for viral infections in each round
$\eta_1 \dots \eta_r$	<a href="#">Appendix 2</a>	Flu sequences assumed as antigens in the simulation
$t_1 \dots t_r$	<a href="#">Appendix 2</a>	Starting time of each infected stage (day)

### 2.1.2.2 Cell properties

Since only memory B cells can be repeatedly activated by an encounter with antigens, we only simulate memory B cells. Plasma B cells do not undergo SHMs and represent terminal states of the clonal lineage development and thus can be sampled from the leaves of the simulated tree. We will refer to a B cell that has just encountered an antigen and moved to a germinal center (GC) as an activated B cell (or “activated cell” for short) (Figure 2A).

In the AM model, each entity  $i$  represents a B cell with the property vector  $\mathbf{x}_i = (g_i, s_i, t_i, g_i/a_i, g_i a_i)$  with five values, among which the last three are derived from two other quantities:  $g_i$  and  $a_i$  (only saved in the derived form). We keep derived properties as part of  $\mathbf{x}_i$  because they allow us to define  $\Lambda_E(\mathbf{x}_i, \mathbf{S})$  as polynomials of saved properties; this, in turn, enables the use of our fast sampling algorithm. To follow the BDT model, we assume the properties of each cell are fixed in between B/D/T events, ignoring possible short-term temporal changes (34). We next define  $g_i, s_i, t_i,$  and  $a_i$ .

- The binary property  $g_i$  indicates whether a cell  $i$  is an activated B cell (1) or is a memory B cell outside lymph nodes, which we call a “memory cell” for simplicity (0).
- The  $s_i$  property stores the DNA sequence of B cell  $i$  coding for the variable region of the heavy chain with a fixed length  $3L$ . We focus on simulating the heavy chain sequences only because most existing AIRR-seq studies focus on sequencing heavy chains only [e.g., (11, 16, 35, 36)]. For the sake of simplicity, we assume the fate of the cell depends only on the variable region of the heavy chain. Each cell has a fixed sequence, and mutations occur at the time of a cell birth, which happens only for activated cells in the infected stage. After a birth event for cell  $i$ , sequences  $s_j$  and  $s_k$  of child cells  $j$  and  $k$  are

chosen independently and identically at random [ignoring the G1 origin of mutations (37)]. While any sequence evolution model could be incorporated in the DIMSIM framework, in Section 2.1.3, we describe a 5-mer-based model used in our analyses.

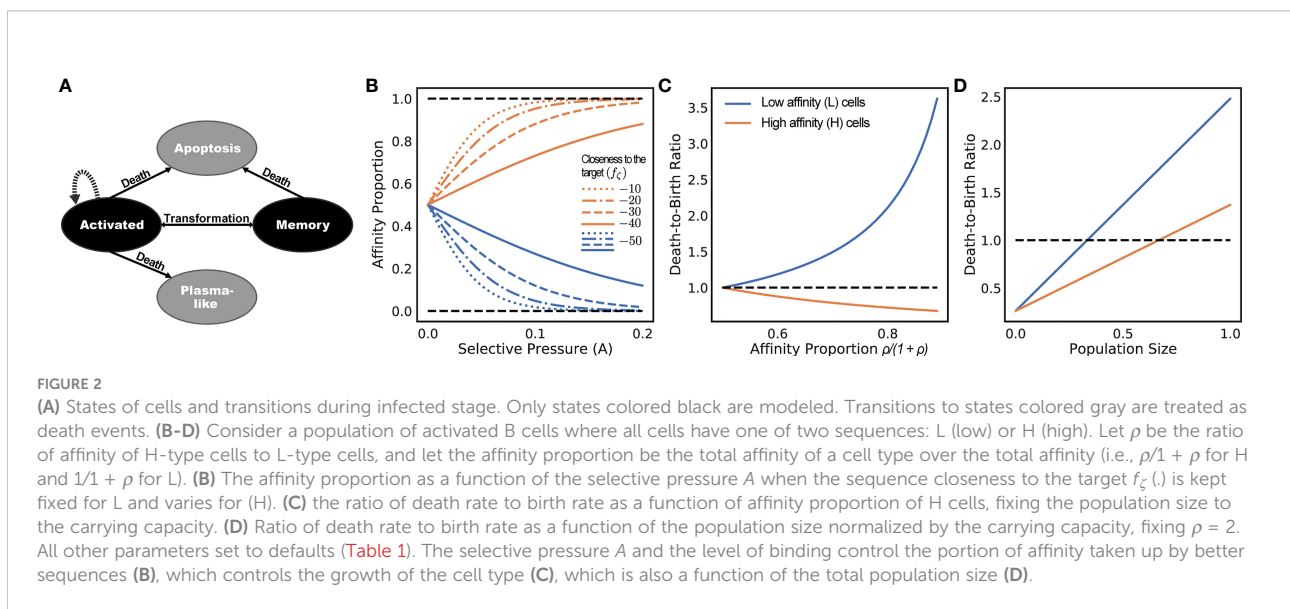
- Property  $t_i$  denotes the rate of transformation, which means the activation of a memory cell ( $g_i: 1 \rightarrow 0$ ) in response to an antigen, or the maturation of an activated cell into a memory cell ( $g_i: 0 \rightarrow 1$ ). Transformations, which only happen during the infected stage, flip  $g$  but keep the sequence  $s$  intact.
- Property  $a_i$  denotes the strength of affinity binding of the Ig receptor of the cell  $i$  to the antigen.

We let  $\sigma$  denote the total affinity of activated cells and note  $\sigma = \sum_{i \in S} g_i a_i$  is the last element of the vector  $\mathbf{S}$ . Then,  $a_i/\sigma$  is the fraction of total affinity assigned to a cell. Both  $t_i$  and  $a_i$  are derived and are set based on the sequence of  $i$  and the target.

### 2.1.2.3 Sequence affinity and birth, death, and transformation rates

Affinity  $a_i$  is only defined and used during the infected stage where the target is available and is function of the cell sequence  $s_i$  and the target sequence  $\zeta$ . The closer the sequence to the target, the higher its affinity should be, a fact that other simulators have also incorporated (30, 32). The exact relationships between the sequences and affinity are not known. For the purpose of benchmarking methods, we propose a simple formula. Let  $f_\zeta(s_i)$  be a measure of the closeness of the sequence to the target in the affinity space, we set

$$a_i \doteq e^{A f_\zeta(s_i)} \tag{1}$$



where  $A$  is a constant factor used to calibrate the selective pressure. As sequences get closer to the target, the affinity grows gradually with a speed controlled by  $A$  (Figure 2B). We will describe our particular choice of function  $f_{\zeta}(s_i)$  using BLOSUM similarity in Section 2.1.3.

The event rates are functions of cell properties and the stage (Table 2). During the dormant stage, there are no births or transformations; cells only die with a very high uniform rate  $\lambda_d$  for activated cells and a low uniform rate  $\lambda'_d$  for memory cells.

During the infected stage, we adjust the death rates of cells based on their affinities but keep the birth rates constant; this interplay is used to simulate the selective pressure. Note that we do not claim that a fixed birth rate and changing death rate is biologically realistic [e.g., see (38)]. However, in terms of the dynamics of our model, what matters is the ratio of the birth and death rates, which enable us to make this simplifying choice. In our model, an activated cell can undergo cell division at a uniform rate  $\lambda_b$ , differentiate into a memory cell at a uniform rate  $t_i = \rho_m \lambda_b$  or a plasma-like cell at a uniform rate  $\rho_p \lambda_b$ , and undergo apoptosis (i.e., die). We do not model plasma-like cells; instead, both differentiation into plasma-like cells and apoptosis are treated as death events (Figure 2A). The rate of apoptosis of an activated cell  $i$  is modeled as inversely proportional to the amount of resources (antigens and FDCs) to which the cell  $i$  has access when competing against other activated cells. Thus, the proportion of resources available to the cell  $i$  is modeled by the affinity proportion  $a_i/\sigma$  (i.e., the affinity of the cell to the antigen normalized by the current sum of the affinity of all activated cells). This affinity proportion is impacted by the parameter  $A$ . The lower the  $A$ , the more uniform these proportions become, modeling low selective pressure; conversely, as  $A$  increases,  $a_i/\sigma$  values further diverge between low affinity and high-affinity cells (Figure 2B). Thus,  $A$  can be used to control the strength of the selective pressure.

The memory cells undergo apoptosis at a uniform rate  $\lambda'_d$ . They can also be activated by helper T cells to enter the germinal center with the transformation rate

$$t_i \doteq \lambda_t e^{\rho_a A (f_{\zeta}(s_i) - \Delta_0)} = \lambda_t e^{-\rho_a A \Delta_0} a_i^{\rho_a} \quad (2)$$

Note that the activation rate of memory cells increases monotonically with their affinity to the target, according to  $a_i^{\rho_a}$  where  $\rho_a$ , set by default to 1/2, is the sensitivity of B cell activation to affinity. This dependency on affinity models the increased propensity of the memory cells to activate when presented by

helper T cells with familiar antigens. The default choice  $\rho_a = 1/2$  is motivated by the fact that although memory cells with higher binding strengths to the antigen are more likely to be activated, the interaction between a helper T cell and a memory B cell is a one-time event and thus less sensitive to binding strength.

As an example, consider a system with two cell types: L and H, each type with its own unique sequence (Figures 2B-D). Assume all cells are activated cells, the number of L and H are the same at one point in time, and H cells have a higher affinity than L cells by a factor of  $\rho$ . For ease of exposition, here, we include the mutation rate as part of the death rate because mutation events also decrease cell count. Let's assume the total number of cells equals the carrying capacity  $C$ . If L and H have the same affinity (i.e.,  $\rho = 1$ ), then the birth and death rates are identical for all cells. As the affinity of H cells is increased (i.e.,  $\rho > 1$ ), the death rate of L cells increases linearly, whereas the death rate of H cells decreases (Figure 2B). Thus, H cells will have higher birth rates than death, will be selected for, and will expand. If we fix  $\rho = 2$  and increase the population size, the death rates of both L and H cells increase, but at different rates (Figure 2D). When the population size is small compared to  $C$ , both types of cells have more birth than death. After a threshold ( $C/3$  in this example), the death rate of L type surpasses its birth rate (thus, its population starts to shrink) while the population of H cells continues to grow. However, as the population size increases ( $2C/3$  here), both sets of cells start to shrink (i.e., higher death rates than birth), because the population size is by definition bounded by  $C$ .

### 2.1.3 Specific modeling choices

Several steps of our simulations are flexible and can be changed by the user. We next describe the set of choices we have implemented and used in the experiments below. We have performed two sets of experiments with different settings; where the two simulations differ, we explain both choices.

*Switching between stages.* The system enters dormant stage when antigens are neutralized by the antibodies. We have implemented two options for defining neutralization. *i)* Switch to the dormant stage after a certain duration from the beginning of the infected stage. *ii)* Switch to the dormant stage when the total affinity of antibodies produced by plasma-like cells reaches a certain threshold; here, we switch when the sum of affinities of activated cells ( $\sigma$ ) reaches a predefined constant  $M$ .

*Sequence evolution.* In our experiments, we use an empirical 5-mer-based model inspired by the Yaari et al. (39) model. Let  $s(p)$  be

TABLE 2 Birth, death, and transformation rates.

Rate functions	Infected stage	Dormant stage
$\Lambda_B(x_i, S)$	$g_i \lambda_b + (1 - g_i) \times 0$	0
$\Lambda_D(x_i, S)$	$g_i \left( \frac{\lambda_b (1 - \rho_p - \rho_m) \sigma}{C} \frac{1}{a_i} + \rho_p \lambda_b \right) + (1 - g_i) \lambda'_d$	$g_i \lambda'_d + (1 - g_i) \lambda'_d$
$\Lambda_T(x_i, S)$	$t_i = g_i \rho_m \lambda_b + e^{-\rho_a A \Delta_0} a_i^{\rho_a} (1 - g_i)$	0

See Table S1 for polynomial forms.

the nucleotide on the  $p$ -th position of the nucleotide sequence of the cell  $i$ . Each  $s_j^{(p)}$  or  $s_k^{(p)}$  is independently set to  $s \in \{A, C, G, T\}$  with probability:  $Pr(s_j^{(p)} = s) = Pr(s_k^{(p)} = s) = f(s, s_i^{(p-2)}, s_i^{(p-1)}, s_i^{(p)}, s_i^{(p+1)}, s_i^{(p+2)})$  where  $f: \{A, C, G, T\}^6 \rightarrow [0, 1]$  denotes an empirically determined 5-mer frequency model based on the Yaari et al. (39) model and recomputed based on newer datasets including non-synonymous mutations (Appendix 2). Note that the rate of the mutation of each position changes as the sequence around it changes (e.g., from a low rate 5-mer to a high rate 5-mer).

**Sequence affinity function.** While various methods can be imagined for measuring the closeness of the sequence to the target, we used a simple approach: measuring sequence similarity according to the BLOSUM matrix and appropriate scaling of numbers. We assume each amino-acid position contributes to the binding strength to the target and the stability of the structure of the Ig-receptor independently. We model affinity proportionally to the product of the effect of each amino-acid position. This simple model ignores the 3D structure of proteins for the most part but should be sufficient for creating benchmarking datasets as none of the reconstruction methods consider 3D structure either. However, because complementarity-determining regions (CDRs), which include the binding sites, tend to accumulate more SHMs compared to framework regions (FRs) (19, 40), we do differentiate those. When  $s_i$  includes a stop codon, we simply set  $a_i = 0$ . Otherwise, we define the BLOSUM score of an amino acid sequence  $\xi = (\xi^{(1)}, \dots, \xi^{(L)})$  with respect to target  $\zeta$  as

$$\Delta_{\zeta}(\xi) = \sum_{p \in \text{CDR}} (\delta(\xi^{(p)}, \zeta^{(p)}) - \delta(\zeta^{(p)}, \zeta^{(p)})) + w_f \sum_{p \in \{1, \dots, L\} \setminus \text{CDR}} (\delta(\xi^{(p)}, \zeta^{(p)}) - \delta(\zeta^{(p)}, \zeta^{(p)})) \quad (3)$$

where  $\delta(\dots)$  gives the BLOSUM score between two amino acids (Table S2), and  $w_f$  is a constant used to calibrate the importance of CDRs versus FRs in the affinity and transformation processes. We then simply set  $f_{\zeta}(s_i) = \Delta_{\zeta}(\xi(s_i))$  where  $\xi(\cdot)$  translates from DNA to AA.

**Choosing targets.** One target sequence per round needs to be selected. The extent of the change in targets across rounds impacts the patterns of the immune response and hence the shape of the resulting clonal trees. To define targets across rounds in our experiments, we explore two options. *i*) We simply use the antibodies known to neutralize a specific disease as the target. Existing databases such as CoV-AbDab (41) provide excellent sources of such candidates. *ii*) Evolutionary approach: We seek a set of sequences with an evolutionary trajectory that reflects the evolutionary history of a set of real antigens (e.g., influenza virus). Let the known amino-acid sequences of an antigen (e.g., flu) sampled through time be denoted by  $\eta_1, \dots, \eta_r$ , and let each sequence have the fixed length  $L_{\eta}$ . To choose the targets, we first select an arbitrary naive B cell, here chosen from datasets of Ellebedy et al. (35), and set  $\hat{\Psi}$  to the nucleotide sequence of the variable region of its heavy chain.

Then, we simply set  $\zeta_1$  to the amino-acid translation of  $\hat{\Psi}$ . In other words, in the first round, we use the naive cell as the target, and therefore, the first couple of rounds of the simulation should be treated as dummy rounds and should be discarded. Let  $\kappa$  be a positive constant that controls the rate of change in the target relative to the rate of change in the antigen sequences. To define the remaining targets, we seek to find the set of  $r - 1$  sequences that minimize:

$$\sum_{i,j \in [r]} |\kappa \sum_{p \in \text{CDR}} \delta(\zeta_i^{(p)}, \zeta_j^{(p)}) - \delta(\zeta_i^{(p)}, \zeta_j^{(p)}) - \sum_{q=1}^{L_{\eta}} (\delta(\eta_i^{(q)}, \eta_j^{(q)}) - \delta(\eta_i^{(q)}, \eta_j^{(q)})|. \quad (4)$$

Thus, a set of target sequences across  $r$  rounds are preferred if their pairwise distance matrix maximally matches the pairwise distance matrix of all antigen sequences over the same rounds (with a scaling). To account for conserved regions, we arbitrarily chose to keep all the non-CDR regions invariable in all target sequences (this choice can be easily changed). Thus, we seek to make the distance between two target sequences from two rounds similar to the distances of antigen sequences, scaled by a factor of  $\kappa$ . We approach this NP-hard problem using a greedy search heuristic (Algorithm S3). The heuristic starts with arbitrary  $\zeta_2, \dots, \zeta_r$  and replaces one symbol of one sequence at a time to reduce the objective function; it repeats until reaching a local minimum where no such replacement is possible.

## 2.2 Benchmarking setup

We use two sets of experiments, one focused on SARS-CoV2 and one focused on influenza. The two sets of simulations use different settings to demonstrate the flexibility of the tool. In both experiments, unless otherwise specified, we used the default settings for the various parameters of Table 1.

### 2.2.1 SARS-CoV2 simulations

We performed 3–5 rounds of infections by SARS-CoV2. For SARS-CoV2, the potential antibodies neutralizing the virus are available, providing us with a natural way to choose the targets. We selected all heavy chain sequences of human antibodies with IGHV1-58 and IGHJ3 from the Coronavirus Antibody Database (41) that neutralize some variants of SARS-CoV2 and have 16 amino acids in their CDR3. We kept only one sequence per upload date by choosing the antibody that neutralizes the most variants of SARS-CoV2 and obtained 14 sequences (Table S3). We then simulated 3–5 (randomly selected) infections with targets randomly selected from the sequences in Table S3 and the infection start date set to be the upload date. We ensured that gaps between the start dates of any two infections were more than 50 days. Each round of infections is set to last 50 days. We repeat this process for 50 replicate individuals.

For each individual, at the end of simulations, we sample  $\zeta = 50, 100, 200, 500$  antibody-coding nucleotide sequences from cells in the system (i.e., from the last round of infection) and built their clonal tree. We also examine scaling of a subset of methods by testing  $\zeta \in \{50, 100, 200, 500\}$ .

## 2.2.2 Flu simulations

We performed several simulations of a series of  $r = 56$  rounds of flu, using sequences of hemagglutinin (HA) protein. HA found on the surface of the influenza viruses is the primary target of neutralizing antibodies. High mutation rates of influenza genome change the sequence of HA and allow the virus to escape from the immune pressure, thus making flu a recurring seasonal infection. The NCBI Influenza Virus Resource (42) contains 961 HA sequences from influenza B virus collected around the world. Each HA sequence is labeled with a year and a location. For simulation purposes, we extracted 59 HA sequences corresponding to flu infections in Hong Kong and selected 56 out of 59 HA sequences with the same length (584 aa). The selected HA sequences were detected in Hong Kong from 1999 to 2010. Notice that HA sequences could be replaced with other widely available antigen sequences (e.g., Coronavirus).

We used the evolutionary approach described earlier to choose the target amino-acid sequences. Each round corresponds to one season, starts at the infected stage with a given target sequence  $\zeta_b$ , and ends when  $\sigma = M$ . At that point, we assume the infection is overcome, and the system switches to dormant, where we stay until the next round starts (times of flu outbreaks are known in our dataset). When the  $r = 56$  rounds of infections end, we sample  $\zeta = 200$  antibody-coding nucleotide sequences  $\Psi_1, \dots, \Psi_\zeta$  from cells in the system (i.e., from the round  $r$ ) and built their clonal tree. While it is unrealistic for a person to get infected with flu these many times, this procedure allows us to test the impacts of a large number of infections.

Here, we set up four experiments, varying one or two parameters in each experiment (Table 3) and setting the remaining ones to default values (Table 1). The central experiment contains 19 conditions, changing the selective pressure ( $A$ ) and the rate of hypermutation ( $\mu$ ). We vary  $A$  from  $1/8\times$  of default value (0.1) to  $2\times$  and vary  $\mu$  s from  $1.25 \times 10^{-4}$  to  $2 \times 10^{-3}$  per base-pair per generation. In six combinations,

the selective pressure is not high enough to overcome random mutations; in these cases, the affinity values do not increase, and as a result, the carrying capacity is never reached. Thus, we exclude these conditions. We also study three other parameters. We vary the weight multiplier of FRs ( $w_f$ ) from  $1/5$  to  $2$ . We vary the carrying capacity ( $C$ ), which is the germinal center size or the amount of antigens FDCs hold in the context of B cell maturation, from 12500 to 400000. The value of this parameter can impact the speed of novel mutations arising and may change the properties of simulated trees. We also vary the mean lifetime of memory cells from 0.5 year to 16 years to study the impact of the extent of memory cell activation during recurrent infections.

## 2.2.3 Compared methods of clonal lineage reconstruction

We compare eight tools: minimum spanning tree, BRILIA (22), IgPhyML (23), RAxML (43), Immunitree (13), Dnapars (44), BEAST (45), and GcTree (26). We also include an alternative “contracted” version for four methods (IgPhyML\*, RAxML\*, BEAST\*, Dnapars\*). We note this is not an exhaustive list, as many other tools also exist [e.g., SAMM (32) and IgTree (46)] that we did not include. For all methods, we ran reconstructions using one CPU core and set a 24-hour wall time for each replicate.

MST(-like) methods: We implemented a simple minimum spanning tree method containing  $\Psi_1, \dots, \Psi_\zeta$  as well as  $\hat{\Psi}$  which is forced to be the root. We compute the nucleotide Hamming distance between all pairs of sequences and construct the minimum spanning tree (MST) using those distances. Besides the simple MST, we also test Immunitree (13), a tool that clusters antibody-coding sequences into lineages and builds clonal trees at the same time by optimizing a minimum spanning tree and Steiner tree-like problem. We took as input  $\Psi_1, \dots, \Psi_\zeta$  and used Immunitree to build a set of clonal trees. We then added vertex  $\hat{\Psi}$  as the root and let the roots of the clonal trees to be immediate children of  $\hat{\Psi}$ .

Brilia clusters antibody-coding sequences into lineages and builds clonal trees at the same time. We took as input  $\Psi_1, \dots, \Psi_\zeta$  and used Brilia v3.5.4 to build a set of clonal trees. We then added vertex  $\hat{\Psi}$  as the root and added roots of the clonal trees as children of  $\hat{\Psi}$ .

TABLE 3 Experiment setup.

Experiment	Parameters	Parameter values	Parameter units
Selective pressure vs. rate of hypermutation	$A \times \mu$	(2, 2), (2, 1), (2, 1/2), (2, 1/4), (2, 1/8), (1, 2), (1, 1), (1, 1/2), (1, 1/4), (1, 1/8), (1/2, 1), (1/2, 1/2), (1/2, 1/4), (1/2, 1/8), (1/4, 1), (1/4, 1/2), (1/4, 1/4), (1/4, 1/8), (1/8, 1/4), (1/8, 1/8)	$A: 10^{-1}$ $\mu: 10^{-3}$
Framework weight	$w_f$	2, 1, 1/2, 1/3, 1/5	1
Germinal center size	$C$	4, 2, 1, 1/2, 1/4, 1/8	$10^5$
Memory cell life	$1/\lambda_D'$	16, 8, 4, 2, 1, 1/2	year (365 days)



**Phylogenetic methods:** We tested ML phylogenetic reconstruction tool RAxML v8.2.12 under the GTR+GAMMA model and IgPhyML v1.1.5, an ML method tuned specifically for immune cells. We also tested maximum parsimony (MP) trees using Dnapars from the PHYLIP package v3.697 (44) and Bayesian MCMC using BEAST v1.10.4 (45). We ran BEAST under the GTR+GAMMA model and uncorrelated relaxed clock with log-normal distribution; we set the tree prior model to Bayesian Skyline, the length of MCMC chain to be 108, and the burn-in rate to 10%. For RAxML, Dnapars, and BEAST, we took as input  $\Psi_1, \dots, \Psi_\zeta$  and  $\hat{\Psi}$  to obtain an unrooted phylogenetic tree and rerooted at  $\hat{\Psi}$ . For Dnapars, we reported the greedy consensus tree when Dnapars outputted multiple trees. We also report the majority consensus tree and refer to it as Dnapars\*. For IgPhyML, we took as input  $\Psi_1, \dots, \Psi_\zeta$  and provided  $\hat{\Psi}$  as root to obtain a rooted phylogenetic tree. RAxML, IgPhyML, and BEAST produce fully binary trees, while Dnapars can produce non-binary trees.

**Contracted phylogenetic methods:** As previously suggested (26, 32), contracting short or low support branches is one way of addressing the limitations of methods that output fully binary trees. Since the length of each antibody-coding nucleotide sequence  $< 400$ , we can assume that both ends of any branch with length less than  $10^{-4}$  would correspond to the same sequence (if it was sampled). Therefore, we contracted branches of length less than  $10^{-4}$  for RAxML and IgPhyML, and call the resulting methods RAxML\* and IgPhyML\*. While the same logic does not hold for the Bayesian methods (because of the prior on branch lengths), for consistency, we applied the same procedure to BEAST to obtain BEAST\*.

GCTree is an approach that integrates sequence abundance information to break ties in an MP analysis (26). It allows sequences to be placed on internal nodes. We took  $\Psi_1, \dots, \Psi_\zeta$  as input to GCTree v4.04 and provided  $\hat{\Psi}$  as root to obtain a rooted phylogenetic tree. As  $\Psi_1, \dots, \Psi_\zeta$  often contain repetitive sequences, GCTree can utilize the sequence frequencies for lineage reconstruction.

#### 2.2.4 Evaluation metrics

The simulated and reconstructed histories of samples  $\Psi_1, \dots, \Psi_\zeta$  are represented as trees, where samples are uniquely labeled on some nodes, and the remaining nodes are left unlabeled. Labeled nodes represent sequences in the samples, and unlabeled nodes denote the ancestral sequences not present in the samples. We evaluate results in two ways, described in detail in Appendix 2.4. We use a set of metrics for characterizing properties of simulated trees in terms of their topology, branch length, and distribution of labeled nodes. We also compare the simulated trees to those inferred using each method (Table 4).

While metrics for comparing phylogenies exist, these metrics need to be amended for clonal trees that can have sampled ancestral nodes (32, 47). Many of the existing metrics can be generalized to

compare a simulated tree  $R$  and a reconstructed tree  $E$  (Table 4), both induced down to include all labeled nodes (i.e., removing unlabeled nodes if less than two of their children have any labeled descendants). Unlike traditional phylogenies, here, internal nodes can be labeled, and we define metrics based on rooted trees instead of unrooted trees. We refer to the set of labeled nodes under a node as a cluster. Note that singleton clusters (i.e., those with one labeled node) are trivial when all labeled nodes are placed at leaves; however, when labeled nodes can be placed at internal nodes, including or excluding singleton clusters can change the metrics. Thus, we also define many of the distances with and without singleton clusters. Some distances (i.e., FNR and FDR metrics) are already normalized. To normalize other distances, for each experimental condition, we create a control tree by randomly permuting labels of the true tree. We then normalize the errors of a reconstruction method by dividing it by the average score of replicates of the control method.

## 3 Results

### 3.1 Demonstration

Visualizing one replicate under default conditions for both SARS-Cov2 and flu simulations, we see similarities and differences between the two scenarios (Figures 3A, B versus Figures 3C–E). In both cases, during each round of infection, the affinity first decreases and then increases as long as the duration of the infection is long enough (Figures 3A, C). Thus, when the number of activated cells is low, and the selective pressure is low, a mutation is likely to lead to reduced affinity, whereas when the number of activated cells increases, the selective pressure begins to increase and select for higher affinity; these patterns are in concordance with the literature (48). In the SARS-Cov2 simulations with a fixed duration, the total number of activated cells converges to similar values across infections, but the final affinity changes (Figure 3A). In the case of flu, the duration of infections, the mean affinity at the end, and the total number of cells vary widely across different rounds (Figure 3C). When the affinity at the start of a round is low, the duration of infection is longer, and more activated cells and memory cells are generated (Figures 3C, S1A). This pattern can be justified: when the immune system already has a high affinity to the antigen, it can eradicate the antigen quickly and without much need for further evolution. To further quantify the pattern, we define the novelty of each target  $\zeta_i$  as the negation of the maximum BLOSUM score between that target and any previous target:  $-\max_{j < i} \{\Delta_{\zeta_i}(\zeta_j)\}$ . We observe that as novelty of the target increases, the average affinity of activated cells at the end of the infection tends to decrease ( $R^2 = 0.242$ ,  $p = 2.5 \times 10^{-4}$ ), whereas the number of

TABLE 4 Clonal tree properties and metrics for comparing the reference tree *R* to estimated tree *E*.

Property/Metric	Definition
Internal sample (%)	The percentage of labeled nodes that are internal nodes.
Bifurcation index	Ratio of the number of internal nodes to one less than labeled nodes; equals to 1 for bifurcating trees and $\approx 0$ for a star tree.
Sample depth	The average depth of labeled nodes.
Balance (cherry)	Half the sum over all leaves of the fraction of their siblings that are also leaves.
Single mutation branches (%)	The percentage of branches with length one.
Accumulated mutations (avg)	The average depth (path length to the root) of all labeled nodes.
Accumulated mutations (sum)	The summation of branch lengths of all branches.
Mutations per branch	The average branch length.
False Discovery Rate (FDR)	the percentage of clusters in <i>E</i> that are not in <i>R</i>
Negative Rate (FNR)	the percentage of clusters in <i>R</i> that are not in <i>E</i>
cluster distance (RF)	the number of clusters in either but not both trees
FDR*, FNR*, and RF*	*: similar to the previous metrics but with singletons excluded
Triplet discordance (TD)	the number of trees induced by triples of labeled nodes (leaf or internal) where the topology in the simulated tree and the reconstructed tree differ
Triplet edit distance (TED)	the sum of the RF cluster distance induced to each triplet of labeled nodes
MRCA Discordance (MD)	the summation of MRCA discordance <sup>†</sup> over all ordered pairs of labeled nodes.
Patristic Distance (PD)	the summation of the patristic discordance <sup>‡</sup> over all pairs of labeled nodes.

<sup>†</sup>MRCA discordance of two labeled nodes is the difference between the number of branches in the path between each of them and their MRCA.

<sup>‡</sup>Patristic discordance for a pair of labeled nodes is the difference between the number of branches in the path between the two nodes on the two trees *R* and *E*. See also Table S6.

activated cells at the end of the infection ( $R^2 = 0.248$ ,  $p = 2.0 \times 10^{-4}$ ) and the duration of infection ( $R^2 = 0.288$ ,  $p = 4.8 \times 10^{-5}$ ) both tend to increase (Figure 3D).

Memory cell counts fluctuate during each infection and across infections. In the SARS-Cov2 simulations, because the duration of the infection is fixed, after 15–30 days, the accumulation of memory cells stops; in fact, the buildup of memory cells starts to fade due to cell deaths, as one would expect. In the flu simulations, which have a much shorter duration, each round leads to a buildup in memory cells from the start to the end of the infection, and the amount of buildup depends on the duration and correlates with novelty ( $R^2 = 0.264$ ,  $p = 1.2 \times 10^{-4}$ ). However, the total number of memory cells reduces between rounds due to cell deaths (Figure S1C) and changes across rounds. In particular, a string of short-lived infections and large time spans between the flu seasons between 2002 and 2008 gradually lead to a depletion of the memory cells, which are then built up again in the subsequent rounds (Figure S1C). Finally, sequences from each round are not monophyletic in either simulation, as expected due to the use of memory of cells (Figures 3B, E). In both cases (especially SARS-Cov2), the antibodies evolved in later infections have more representations in the sample and are further away from the root.

## 3.2 Benchmarking methods

### 3.2.1 SARS-Cov2 simulations

#### 3.2.1.1 Default parameters

Under default parameters, GCTree failed to run on 2 out of 50 replicates, and we excluded those replicates from our

analyses. The relative accuracy of methods depends partially on the metric of choice (Figure 4). Using metrics that rely on cluster recovery (i.e., RF, FDR vs. FNR, and MD), the contracted phylogenetic ML methods (IgPhyML\* and RAxML\*) have the best accuracy. Judged by triplet-based metrics TED and TD, GCTree has the best accuracy (mean: 11% for TED and 12.4% for TD) followed closely by RAxML\* (mean: 11.2% for TED and 14.1% for TD). While RAxML\* is among the most accurate methods with all metrics, GCTree is very accurate according to triplet-based metrics but not with cluster-based methods. This discrepancy reflects different emphases of the metrics: TED and TD focus more on deeper branches in the tree that produce more triplets and are less sensitive to local branch contractions and resolutions. By contrast, cluster-based metrics treat every part of the tree equally and can be highly sensitive to local changes. The other methods have no obvious advantage to ML or GCTree. The MST-like methods have high precision, coming close to contracted ML methods, but also have much lower recall (FNR > 25%). Immunitree (which uses Steiner trees) is substantially better than a simple MST in terms of recall but not in terms of many metrics, including triplet-based measures (TED and TD). The MP method, Dnapars, was not more accurate than ML, and neither were BEAST or BEAST\*.

Branch contraction improves the accuracy of all phylogenetic methods under all metrics (Figure 4B). Normal phylogenetic ML methods (IgPhyML and RAxML), which produce fully binary trees with all samples at leaves, have the lowest FNR (highest recall), retrieving approximately 90% of the correct clusters. However, their FDR is predictably high (low precision): more than 30% of their clusters are incorrect. Luckily,

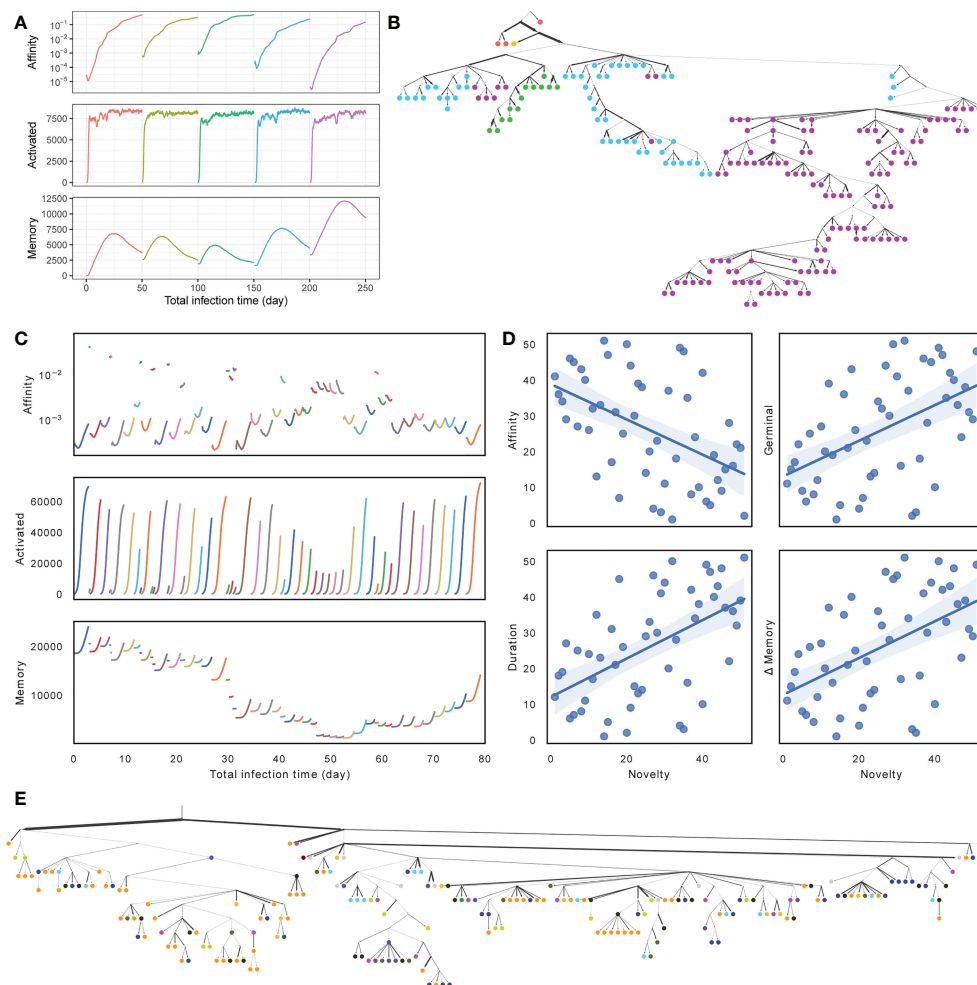


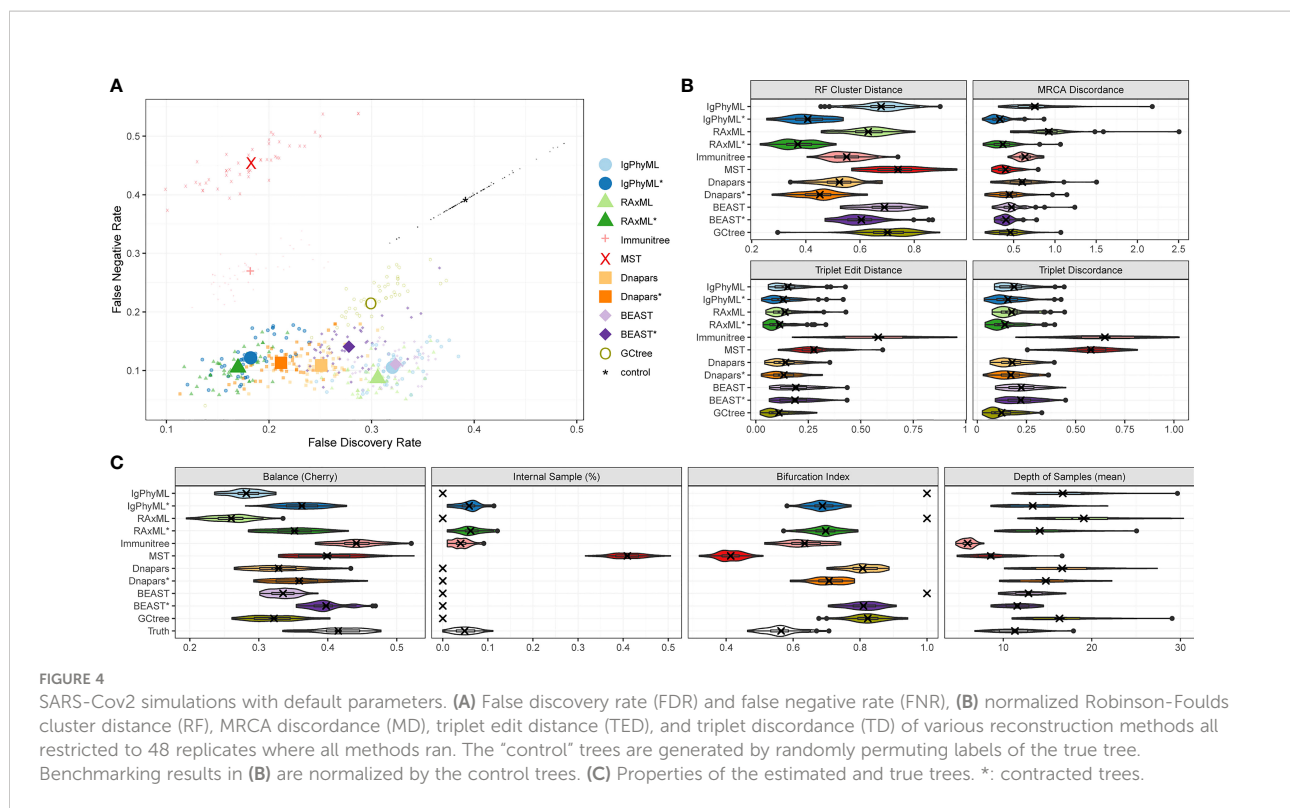
FIGURE 3

(A, C) Average affinity of activated cells to current infection target (log scale), the number of activated cells, and the number of memory cells by total time in infected stage across stages of infection (colors) for (A) SARS-Cov2 (discarding dormant stages) and (C) Flu (discarding the first 5 rounds and dormant stages). (B, E) Clonal tree of memory cells sampled from one simulation under default condition after all rounds for (B) SARS-Cov2 and (E) Flu. Nodes are colored by rounds when the memory cells emerge (A, C) (in Flu tree, gray color is used for rounds 1 through 46). Here, 11 internal nodes in SARS-Cov2 tree and 17 internal nodes in Flu tree are sampled and are indicated as circles. Edge weights denote the number of mutations of sequences denoted by adjacent nodes. (D) Impact of the novelty of the antigen on the outcome of the infection across the 56 rounds of influenza simulations. The novelty of rounds is measured by  $-\max_{i < j} (\Delta_c(\frac{c_j}{c_i}))$  and is ranked from less novel to more novel on the x axis. The y-axis shows ranking (from low to high) of average affinity of activated cells to the current infection target ( $R^2 = 0.242$ ,  $p = 2.5 \times 10^{-4}$ ) at the end of the infection, the number of activated cells ( $R^2 = 0.248$ ,  $p = 2.0 \times 10^{-4}$ ) at the end of the infection, the duration of infection ( $R^2 = 0.288$ ,  $p = 4.8 \times 10^{-5}$ ), and the change in memory cell count ( $R^2 = 0.264$ ,  $p = 1.2 \times 10^{-4}$ ) from the start to the end of the infection. See Figure S1 for more details.

contracted ML methods have only a slight increase in FN rate ( $< 2\%$  on average) but enjoy a dramatic improvement in precision. By simply contracting super-short branches, the FDR error reduces to less than 20% for IgPhyML\* and RAXML\*, which is better than all other methods. Similarly, normal phylogenetic methods perform poorly according to RF and MD metrics, which emphasize precision. Among the two phylogenetic ML methods, RAXML is slightly more accurate than IgPhyML in terms of RF, FNR, and FPR, while IgPhyML has slightly lower MD. The accuracy of MP and Bayesian methods also improve

with branch contraction, but the improvements are less dramatic than ML. Dnapars\* improves the precision of Dnapars by approximately 4% at the cost of only 0.4% decrease in recall. While BEAST has similar accuracy to RAXML before contraction, BEAST\* is far inferior to RAXML\* because it improves the precision over BEAST far less than RAXML\* versus RAXML.

We next compare the properties of the inferred trees and true trees (Figures 4C, S2). MST puts far too many labels at internal nodes ( $\approx 40\%$  instead of  $\approx 5\%$ ), while Immunitree and



contracted ML trees are very close to the true tree in terms of percent internal samples. Immunitree overestimates the tree balance, while other methods underestimate balance, especially before contracting short branches. Conversely, MST and Immunitree underestimate and other methods overestimate the depth of samples; note that contraction also reduced error in sample depth. Phylogenetic methods, by definition, overestimate the bifurcation index as 1; this overestimation is dramatically reduced but not fully eliminated by contracted phylogenetic methods and Immunitree. MST, on the other hand, underestimates bifurcation index.

### 3.2.1.2 Varying sample size

As the number of sequences sampled changes from 50 to 500, the accuracy and running time both change (Figures 5, S3). The impact of sample size on the accuracy of IgPhyML\* and Dnapars\* measured according to TED is insignificant ( $p = 0.62$ ,  $0.18$ , respectively, according to an ANOVA test). Increasing sample size does significantly reduce the accuracy of Immunitree and BEAST\* ( $p = 5 \times 10^{-6}$  and  $p = 0.0001$ , respectively) and significantly improves the accuracy of RAxML\* and MST ( $p = 0.006$  and  $p < 10^{-15}$ , respectively). The main impact of the sample size is on the running time. MST runs in seconds, and RAxML takes minutes to finish for all dataset sizes. All other methods were much slower. While Dnapars is the third fast method with 50-sequence input, its running time increases rapidly, and it fails to finish with  $\geq 200$  sample cases within 24 hours. In fact, Dnapars

had the worst running time scaling. We empirically estimate that the running time of other methods increases with sample size no worse than quadratically (proportionally to  $\zeta^{1.5}$ ,  $\zeta^{1.7}$ ,  $\zeta^{0.07}$ ,  $\zeta^{1.3}$ , and  $\zeta^{0.9}$  for IgPhyML, RAxML, Immunitree, MST, and BEAST respectively) while Dnapars increases with  $\zeta^{7.4}$  for the two data points where it did finish. Although we didn't directly include GTree, we can approximate the running time of GTree by Dnapars, as invoking Dnapars is the most time-consuming step of GTree. While BEAST scaled well, it had the highest running time and failed to finish 500 sample cases within the allotted 24 hours.

### 3.2.2 Flu simulations

Because Dnapars, GTree, and BEAST are much slower (and in the case of GTree, occasionally fail to run properly), we do not include them for the flu analyses (where  $\zeta = 200$ ).

#### 3.2.2.1 Default parameters

Under default parameters, Flu simulations exhibit very similar patterns to SARS-CoV2 simulations. Over all evaluation metrics, contracted ML methods (IgPhyML\* and RAxML\*) clearly have the best accuracy (Figure 6). By simply contracting super-short branches, the FDRs of IgPhyML and RAxML reduce to less than 15% from close to 35%. The MST-like methods still have low FDR and very high FNR. BRILIA consistently has a high error in our analyses. These patterns remain largely similar (but are magnified) when singletons are

removed from consideration (Figure S4). The main exception is that when singletons are excluded, Immunitree is no longer the second-best method according to the RF distance. When comparing the properties of the inferred trees and true trees (Figure 6C), BRILIA puts far too many labels at internal nodes ( $\approx 35\%$  instead of  $\approx 8\%$ ), overestimates the tree balance, underestimates the depth, and slightly underestimates the bifurcation index. On this dataset, MST is quite close to the correct levels of bifurcation.

### 3.2.2.2 Varying selective pressure

The reconstruction methods are all impacted as selective pressure ( $A$ ) changes, but some methods are more sensitive than others, and they are affected differently (Figures 7A, B). Contracted phylogenetic methods have the best accuracy across values of  $A$ . The ranking among other methods depends on the selective pressure, such that phylogenetic methods become the worst when  $A$  is high and become the best when  $A$  is low. As  $A$  increases, the error tends to increase for phylogenetic methods under all evaluation metrics except for the FNR; for example, the FDR of RAxML increases from 27% at the 1/4x selective pressure to 42% at the 2x level. In contrast, the error of Immunitree, MST, and BRILIA reduces with increased  $A$  according to FNR and RF. Contracted phylogenetic methods are relatively robust to the  $A$  and their error rates change only slightly across conditions. When singletons are removed from the metrics of comparison, patterns remain similar, though the impact of selective pressure becomes less pronounced (Figure S5A).

The reason behind these patterns becomes more apparent once we consider changes in tree properties (Figure 7C). As  $A$  increases, the fraction of internal samples tends to increase. This pattern can be explained: when selective pressure is high, cells with low affinity die off quickly, which results in shorter branch lengths. Since phylogenetic methods put all sequences at leaves,

they have reduced accuracy. In contrast, IgPhyML\*, RAxML\*, and Immunitree successfully assign sequences to internal branches; as a result, their percentages of internal samples match those of the true trees. Similarly, with increased  $A$ , the bifurcation index of the simulated tree tends to decrease, a pattern that is observed also in reconstructed trees from IgPhyML\*, RAxML\*, Immunitree, MST, and BRILIA. Again, phylogenetic trees, which produce binary trees, are unable to capture these patterns. As  $A$  increases, the depth of sampled nodes of the simulated tree tends to decrease, a pattern matched by IgPhyML\* and RAxML\* but not other methods. Finally, when  $A$  is high, trees are shorter (i.e., accumulate fewer mutations) and more branches are single mutation (Figure S6), both of which make phylogenetic inference more difficult. The reduced levels of depth, total change, and bifurcation make sense: higher pressure should result in fewer mutations needed to reach  $M$  because cells with unfavorable mutations are less likely to survive; this would produce shorter trees.

### 3.2.2.3 Varying rate of hypermutation

As the hypermutation rate ( $\mu$ ) increases, error decreases for normal phylogenetic methods (IgPhyML and RAxML) according to most metrics but stays relatively stable for contracted methods (Figures 7D, E). Increasing  $\mu$  results in simulated trees that are marginally less balanced, are more bifurcating, have fewer internal node samples, and have a higher depth for sampled nodes (Figure 7F). Thus, increasing  $\mu$  generates trees more similar to what traditional phylogenetic methods assume. Contracted phylogenetic methods and Immunitree designate the right percentage of nodes as internal, but both are slightly more bifurcating than true trees (Figure 7F). Overall, contracted phylogenetic methods are the most accurate across all values of  $\mu$ .

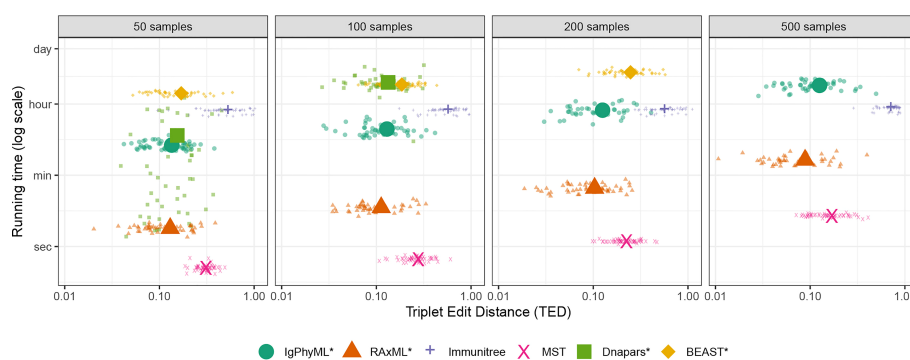
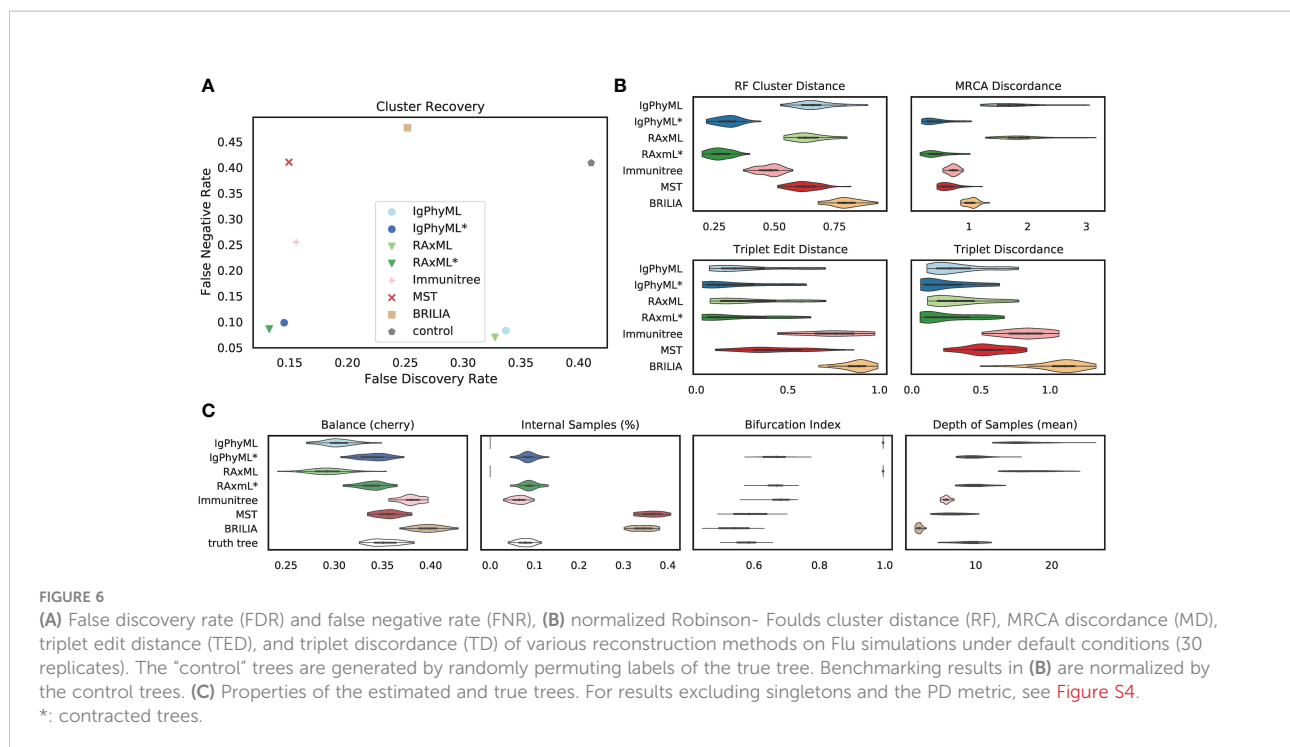


FIGURE 5

Running time and triplet edit distance (TED) of various reconstruction methods on SARS-CoV2 simulations under different sample sizes (50 replicates). Note that contracted methods have the same running time as non-contracted versions are hence are not shown. Also note that GCTree is not included but its running time increases similarly to Dnapars, which it runs internally.



### 3.2.2.4 Interplay between selective pressure and mutation rate

When we vary both  $A$  and  $\mu$ , we observe that increasing the mutation rate has similar effects on the error and tree properties as decreasing the selective pressure ([Figure 8](#)). Reassuringly, error patterns observed when fixing one variable and changing the other are consistent with patterns when both variables are changed ([Figures 8, S7](#)). The most difficult condition for phylogenetic methods is low mutation rates and high selective pressure, where close to 70% of the branches include only a single mutation, and the bifurcation index is only 43%. However, the contracted methods are impacted less in these conditions and are, in fact, improved according to the RF metric ([Figure S7](#)). In addition, we observe that antibody clonal trees become more phylogenetic-like – that is, more bifurcating (max: 0.74) and fewer internal samples (min: 20%) – with  $\mu = 10^{-3}$  and  $A = 1/4x$ . Increasing the mutation rate or decreasing the selective pressure beyond these values leads to combinations where the infection could not be overcome.

### 3.2.2.5 Other parameters

Beyond the main two parameters, we also studied changing six secondary parameters, most of which had relatively little impact on the results ([Figure 9](#)). As the weight of FRs regions in computing affinity ( $w_p$ ) increases, the error tends to increase slightly for all methods under many evaluation metrics ([Figure S8](#)). This pattern can be related to the slight increase in the number of single branch mutations and the reduction in the total number of substitutions across the tree. As germinal center

capacity ( $C$ ) increases, error increases or decreases slightly, depending on what measure is examined ([Figure S9](#)). Increasing  $C$  tends to reduce the number of internal samples and single mutation branches in the simulated tree, and tends to increase mutations per branch. As memory cell life-time ( $1/\lambda'_d$ ) increases, the error tends to increase for phylogenetic methods ([Figure S10](#)), including IgPhyML\* and RAxML\*, which nevertheless continue to be the best methods. Plasma cells conversion rate ( $\rho_p$ ) ([Figure S11](#)), rate of change in antibody target compared to antigen change ( $\kappa$ ) ([Figure S12](#)), and the threshold of total affinity for neutralization and stage change ( $M$ ) ([Figure S13](#)) have small and inconsistent impacts on tree inference error. In all conditions examined, IgPhyML\* and RAxML\* have the best accuracy ([Figure 9](#)).

## 4 Discussion

We introduced a general model of antibody evolution and instantiated it in two different ways to simulate antibody response to flu and SARS-Cov2. The two simulations were substantially different in their choice of models and model parameters and produced quite distinct scenarios ([Figure 3](#)). Nevertheless, they showed similar results in terms of benchmarking of methods. Below, we further comment on the implications of these results for reconstruction methods, evaluation metrics, and simulation procedures. We end by pointing out the limitations of the current work and directions for future work.

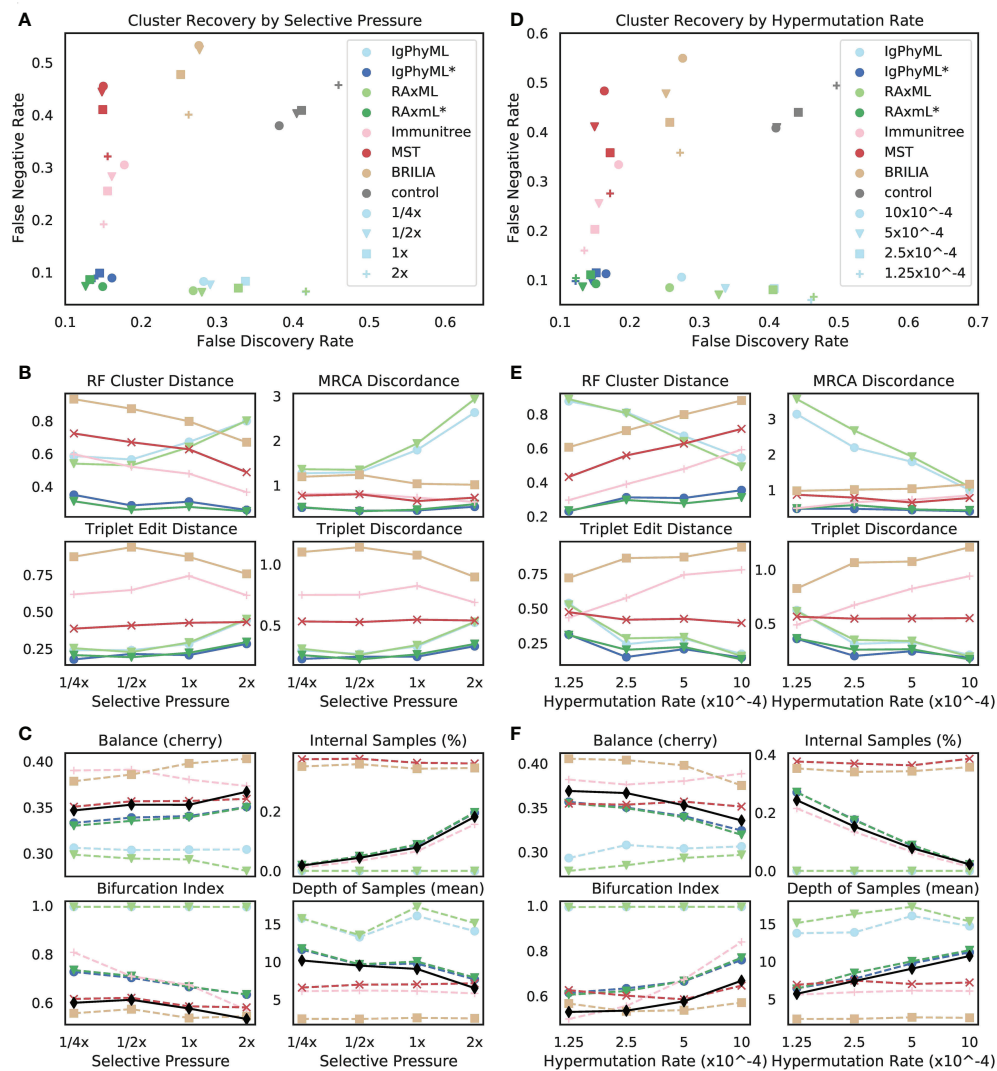


FIGURE 7

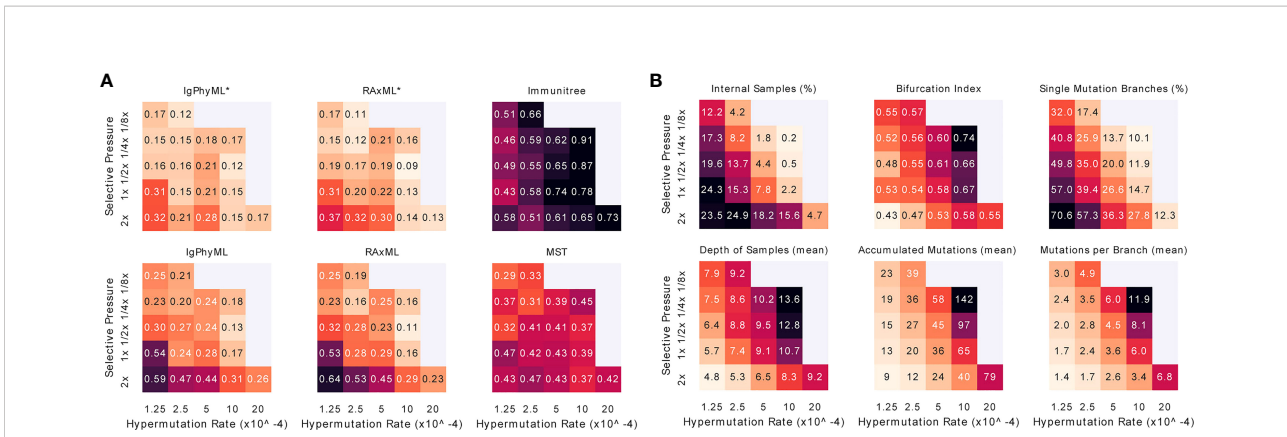
(A-C) Impact of selective pressure  $A$  and (D-F) mutation rate  $\mu$  on (A, B, D, E) tree inference error and (C, F) tree properties. We measure tree error by (A, D) FDR, FNR, (B, E) Robinson-Foulds cluster distance (RF), MRCA discordance (MD), triplet edit distance (TED), and triplet discordance (TD). Tree errors and tree properties are averaged over 30 replicates. (C, F) We show properties of true (black) and reconstructed trees.  $\mu = 5 \times 10^{-5}$  in (A-C) and  $A = 0.1$  in (D-F), which are all default values. \*: contracted trees.

## 4.1 Implications for reconstructing antibody evolution

Consistent with the literature, we found that ML methods need subsequent processing for inferring antibody clonal trees with high accuracy. Depending on the simulation condition, 1% to 20% of sampled sequences belonged to internal nodes, and the true trees are only 60% to 70% bifurcating. We observed that results of phylogenetic inference using ML, taken at face value, can have low accuracy. However, ML phylogenetic methods with the simple adjustment of contracting short branches can outperform the alternative methods. Despite the higher

accuracy of contracted phylogenetic methods compared to the available alternatives, we note that there is still substantial error. Under the default condition, while 90% of clusters of the true tree were recovered, about 15% of the recovered clusters were incorrect. In particular, the discrepancy between FNR and FDR was due to the fact that the inferred trees are somewhat more bifurcating than true trees (e.g.,  $\approx 70\%$  versus  $60\%$  in the default condition). Thus, while contracting super-short branches increases accuracy, these trees are still biased towards too much resolution.

Previous benchmarking of reconstruction methods has shown somewhat contradictory results in terms of the choice

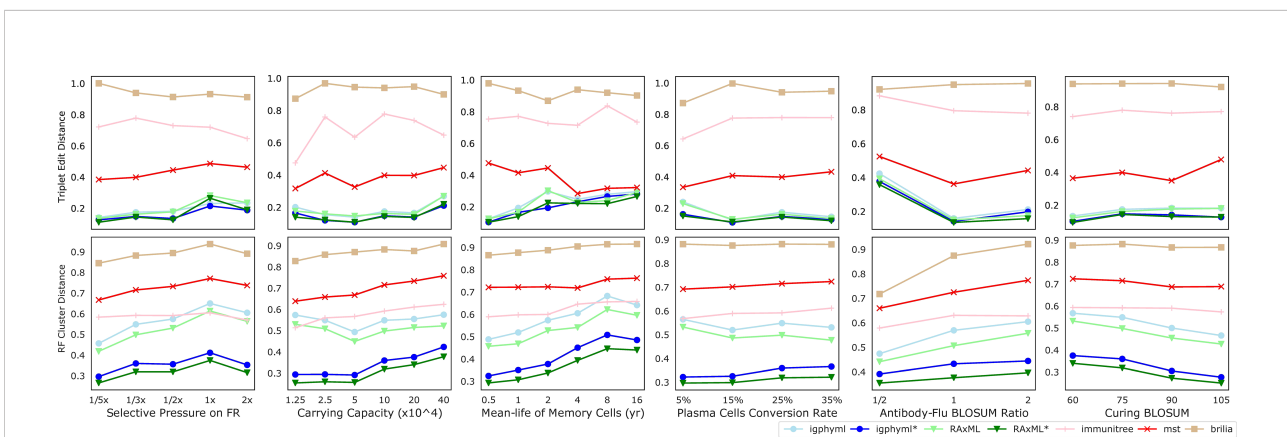


**FIGURE 8** For varying levels of selective pressure (A), rate of hypermutation ( $\mu$ ), and all reconstruction methods except BRILIA, we show (A) tree error measured by the triplet edit distance TED and (B) properties of the true tree. When the mutation rate is too high and the selective pressure is too low, the simulation never ends, meaning that the total affinity needed to overcome the antigen is never reached; these conditions are missing from the figure. For other evaluation criteria see Figure S7. \*: contracted trees.

of the best methods. Davidsen and Matsen (32) found parsimony-based methods GCTree and another post-processing method they introduced called SAMM to be superior to ML methods (they did not test Bayesian methods). In contrast, Yermanos et al. (49) concluded the Bayesian method BEAST is the best method overall, followed by ML and MP methods. The two studies used very different simulation approaches. In addition, Davidsen and Matsen (32) used the genotype-collapsing approach adopted from DeWitt et al. (26) to create contracted trees, while Yermanos et al. (49) did not. Our study, which has more similarities to Davidsen and Matsen (32) than Yermanos et al. (49), only partially confirms observations from these previous studies.

Just like Davidsen and Matsen (32), we observed that subsequent processing improves MP methods, as GCTree was better than Dnapars in our tests. Similar to them, we also ranked

GCTree highly when judged by triplet-based distances. However, Davidsen and Matsen (32), who used dnaml and IQ-TREE, found that ML methods are inferior to MP methods. Our results ranked RAxML\* (RAxML plus contraction) as the best or close to the best method in all simulations with all metrics. The exact causes of these disparities are not obvious. They could stem from differences in simulation procedures, evaluation metrics used (recall that with triplet metrics, GCTree is slightly better than RAxML\*), or even the choice of the ML inference method (dnaml and IQ-TREE versus RAxML). More consequentially, we used the  $\Gamma$  model of rate-across-sites heterogeneity, but it appears that Davidsen and Matsen (32) ran IQ-TREE with no model of rate heterogeneity, which may be problematic given that antibody sequences include hotspots and coldspots. A final interesting difference is the contraction strategy. In contrast to Davidsen and Matsen (32) who used ancestral reconstruction for contracting branches, we used a fixed branch



**FIGURE 9** Triplet edit distances and RF cluster distances by selective pressure on framework region, carrying capacity, mean-life of memory cells, plasma cell conversion rate, antibody-flu BLOSUM ratio (MARatio), stage change threshold (M).



length threshold that we selected based on the sequence length. It could be that difficulties in ancestral state reconstruction render the genotype-collapsing approach of DeWitt et al. (26) less effective than our simple rule of thumb. Note that statistical tests of whether a zero branch length null hypothesis can be rejected also exist (50–52) and are fast (53) and could be used in lieu of our simple heuristic, an option that should be tested in the future. Our results imply that phylogenetic methods that naturally model zero branch length [e.g., (54)] are also promising. In particular, the adaptive LASSO method of Zhang et al. (55) seems suitable for inferring antibody evolution and can be tested in the future.

It is harder to reconcile our results with Yermanos et al. (49). Unlike their study, our results did not find BEAST to be better (or much worse) than ML method RAxML. The main limitation of BEAST in our analyses was that it did not benefit from branch contraction nearly as much as ML methods do. This difference is likely because the prior distributions on branch length render our simple rule-of-thumb ineffective. For ML methods, branches with lengths much smaller than the inverse of the sequence length can be expected to correspond to no changes. This simple rule fails when the branch lengths is governed by priors as well as the likelihood. It is possible that Bayesian polytomy tests (54) could overcome this issue and produce accurate results. It is also likely that with different priors on branch length, our simple heuristic could become more effective. More broadly, the topological accuracy of BEAST may be impacted by priors (we attempted to use settings similar to Yermanos et al. (49), but other priors may increase accuracy). These uncertainties are not limited to our study and point to a practical difficulty with using Bayesian MCMC methods: given the limited signal in antibody sequences, the prior choice is consequential and needs more research.

Beyond accuracy, methods had substantially different running times in ways that may seem surprising. As expected, BEAST was slower than other methods, though its running time rose only linearly as the datasets became larger. However, RAxML was dramatically faster than the MP method Dnapars. While MP is a somewhat simpler problem than ML, the available tools for MP inference are not as advanced as ML methods. As a result, we did not see a trade-off between running time and accuracy; except for MST, which was extremely fast but inaccurate, RAxML\* had the best running time and the best accuracy according to most criteria. Larger samples can give a more complete picture of the evolutionary changes. Nevertheless, accuracy was not impacted dramatically by sampling (except for MST), and only 50 samples were enough to obtain reasonable accuracy, indicating that high sampling may not be always essential.

## 4.2 Implications for evaluation criteria

The ranking of reconstruction methods can change based on which of the ten evaluation criteria we choose, and these rankings only partially correlate (Figure S14). FDR and FNR are weakly anti-

correlated only when including singletons (mean Spearman's rank correlation coefficient across all tests  $\rho = -0.12$ ). RF distance, which combines both aspects, correlates moderately with both FDR ( $\rho = 0.5$ ) and FNR ( $\rho = 0.57$ ). The triplet-based metrics strongly agree with each other ( $\rho = 0.97$ ) and are mostly compatible with the RF distance ( $\rho \approx 0.75$ ), but are less similar to MD and PD metrics ( $\rho \leq 0.52$ ). Consistent with the observation that triplet metrics penalize false negatives more than false positives, they agree more strongly with FNR than FDR ( $\rho = 0.65$  vs  $0.26$ ). MD and PD are very similar to each other ( $\rho = 0.96$ ), have no correlation to FNR ( $\rho \leq 0.05$ ), but have moderately high correlation to FDR ( $\rho = 0.71$ ). Finally, we notice that singletons can matter: while FNR and FNR\* are highly correlated ( $\rho = 0.94$ ), RF correlates with RF\* less strongly ( $\rho = 0.71$ ), and FDR correlates with FDR\* only moderately ( $\rho = 0.61$ ).

The choice of the metric should depend on the downstream application of the clonal tree. While contracted phylogenetic methods are dramatically better than base methods based on most criteria, they are only slightly better according to the triplet-based criteria. The triplet metrics do not penalize trees heavily if they are more resolved than the true tree or if they move internal nodes to leaves. Thus, when downstream usage is robust to extra resolution and extra terminal edges, triplet metrics offer a good way to measure topological accuracy. On the other extreme, PD and MD are very sensitive to the tree resolution and internal placement, so much so that they often evaluate inferred phylogenetic trees to be much worse than random trees (Figure S7) because these trees generate fully resolved trees and put samples at leaves. Thus, we don't find PD and MD to be reliable metrics of topological accuracy. RF distance is in between: it penalizes extra resolution more than triplet metrics but less than path-based metrics. It does distinguish contracted and phylogenetic methods but rarely evaluates any methods to be worse than random (Figure S7). Overall, dividing the observed error along two (potentially contradictory) axes, such as FNR and FDR is recommended because this evaluation provides more insight into the reasons behind errors.

## 4.3 Comparison to other simulation models

Several simulation tools capable of benchmarking reconstruction methods have been developed. Some of these tools are not comparable to our effort because of various limitations. ImmuneSIM (56) generates mutations but does not model the clonal tree or the selection process. Methods of Amitai et al. (30) and Reshetova et al. (31) are based on the two-step simulation paradigm and only generate clonal trees under selection, leaving sequence generation to other methods. AbSim (49) models VDJ recombination and does not model memory cells; moreover, it appears that the tree shape evolves independently from sequences. The most relevant method to

ours are bcr-phylo (32) and gcdynamics (29), which simulate clonal trees of antibody-coding sequences under AM. Both bcr-phylo and gcdynamics have similarities and differences to our method (Table 5). For example, they both support multiple targets but only one round of simulations. Although our model is capable of multiple targets, for simplicity, DIMSIM uses one target per round of infection. However, unlike the two other methods that only simulate activated cells, DIMSIM also simulates memory cells; as a result, it can simulate multiple rounds of infection by an evolving pathogen with changing targets while considering memory built from previous infections. Moreover, DIMSIM simulates in continuous time, whereas the other tools simulate under discrete generations. All three methods use sequences to define affinity, albeit differently: DIMSIM using BLOSUM distance, bcr-phylo using hamming distance, and gcdynamics using random energy landscape. A main feature of DIMSIM is that its rates are polynomial fractions of individual and total affinity; this choice enables it to speed up the simulation, allowing it to scale up to large numbers of cells.

#### 4.4 Limitations of the study

Our study has limitations that should be kept in mind. While many of these limitations may impact the realism of the model, most will likely not impact the relative accuracy of reconstruction methods.

In our simulations, we did not add errors to sequence data used as input to clonal tree reconstruction methods. Real AIRR-seq samples undergo extensive PCR and thus might contain both sequencing and amplification errors. We assumed that error elimination is already performed (to perfection) prior to reconstruction using existing methods [e.g., (57–60)]. The efficacy of methods that simultaneously filter errors and build clonal trees [e.g., (22, 61)] should be the subject of future research. We also simulated only substitution SHMs but no insertions and deletions, leaving the latter to future work.

In our AM model, we made several simplifying assumptions. For example, we assumed the affinity grows gradually as the AA sequence becomes more similar to the target sequence. The idea that AM occurs by mutational diffusion along one or more preferred paths in the genotype space has been supported by Kepler et al. (62). Nevertheless, our i.i.d model is certainly a simplification without clear empirical support. More complex models of receptor binding exist and can be integrated in the future: Ymir (63) is a 3D structural affinity model which takes into account affinity jumps, cross-reactivity, and differential epitope accessibility. There are also machine learning attempts (64) to build predictive models of affinity, which can be incorporated. However, including such sophisticated models could drastically slow down simulations.

We also assumed the existence of a target antibody sequence. The literature has increasingly documented highly convergent immune responses to the same epitope across individuals and conditions (65, 66). This observation gives us reason to think the existence of target sequences is not a bad assumption; nevertheless, the choice of a single target may not be realistic. To model the change in the target as the viruses evolve, we used two approaches: in one, we used real antibody sequences, which seems like a reliable approach. In our second experiment, we chose targets with evolutionary divergence levels that mimic the divergence levels of the antigen, albeit with some scaling factor. In the latter case, it is conceivable that two antigens with high evolutionary distance are neutralized by similar antibodies or that antigens that are very similar require very distant antibodies. We modeled SHMs as affecting daughter cells independently, but it is arguably more realistic to make both daughter cells carry the same mutation due to the G1 origin of SHMs (37) (a simple change to the model). Finally, our 5-mer mutation model, while based on the empirical model of Yaari et al. (39), still fails to capture some complexities of the real antibody evolution. For example, we concentrated substitutions on the CDR region, but other regions are known to also accumulate mutations (61, 67, 68). Other B cell specific models [e.g., (69)] including those that seek to tease out the effects of

TABLE 5 A comparison of Most relevant tools for AM simulation.

	DIMSIM paper	bcr-phylo (32)	gcdynamics this (29)
Targets	Single-target (per round)	Multi-target (1 round)	Multi-target (1 round)
Rounds	Yes	No	No
Affinity	BLOSUM distance	Hamming distance	Random energy landscape
Mutation	Updated (39)	(39)	i.i.d
Scalability	Up to millions of cells	Thousands of cells	Thousands of cells
Cell type	Activated and Memory	Activated	Activated
Germinal Centers	Combined (single)	Combined (single)	Multiple (in competition)
Time	Continuous	Discrete generations	Discrete generations
Isotype	No	Yes	No
Birth/Death rate	Polynomial fraction of individual and total affinity	Neutral: independent of total affinity Kinetic: function of affinities	A function of affinity

selection from background mutations [e.g., (70)] and per-position mutability models (62) can be incorporated in the future.

For all these shortcomings, we offer several responses. Due to challenges in modeling antibody repertoire [e.g., (71)], the framework is designed to be flexible and can easily incorporate more complex models. Thus, our work should be considered a first step that will enable better modeling in the future. Also, our objective in simulations was to benchmark reconstruction tools; as long as our modeling choices did not distort the comparison of methods, some model misspecification can be tolerated. We observed that the choice of the best method was not sensitive to many parameter choices, and in fact, the two simulations resulted in similar conclusions.

Beyond model simplifications, we also chose to simulate parts of the complex immune system response but not others. For example, we simulated one clonal lineage involved in an immune response. As such, we ignored the important V(D)J recombination of IG loci (72) and sought to simply simulate a VDJ recombinant that is effective in fighting a specific antigen. Even then, we simulated only one clonal lineage at a time, a limitation that can be easily lifted in the future by starting from multiple root sequences with different VDJ settings and assigning to each a different target sequence. Note that our tool can be easily combined with methods of simulating VDJ recombination, such as IGoR (73). Neither did we simulate light chains, which are often not captured in AIRR-seq sequencing data. Finally, we did not simulate processes such as epitope focusing that produce broadly neutralizing antibodies (74).

## 4.5 Applications of the framework

Our framework for simulating clonal trees can be extended to other forms of microevolutionary scenarios. While the current implementation is geared towards AM simulations, our proposed algorithm enables forward-time simulation of very large numbers of entities under models that allow dependence between sequences and rates of birth, death, or transformation. The ability to simulate a very large number of entities combined with rates that change with properties of entities gives us the necessary ingredients to simulate under complex models of evolution that consider selective pressure. Thus, our framework can be adopted for other forms of microevolutionary simulation, such as the evolution of a virus within a host and accumulation of SHMs in tumor evolution. Such a possibility would become most intriguing if it can also model the co-evolution of different types of entities (e.g., antibodies and viruses). While we did not simulate co-evolution here, we believe the framework is capable of performing such simulations by simply creating entity types (just like we had cell types) and making the BDT rates a function of properties across different cell types. Another promising direction for extensions of this work is to integrate the sequence evolutionary models with network-based disease transmission models [e.g., (75, 76)] to enable more accurate simulations of disease spread and evolution.

## Data availability statement

The original contributions presented in the study are included in the article/[Supplementary Material](#). Further inquiries can be directed to the corresponding author.

## Author contributions

CZ designed and implemented the simulation algorithm. AB generated the empirical 5-mer mutation model. CZ, YS, and SM designed the experiments. CZ and SM designed evaluation metrics for benchmarking. CZ and SM analyzed the results. All authors contributed to the article and approved the submitted version.

## Funding

This work was supported by National Science Foundation (NSF) grant III-1845967 and National Institute of Health (NIH) grant R35GM142725. Computations were performed on the San Diego Supercomputer Center (SDSC) through XSEDE allocations, which is supported by the NSF grant ACI-1053575.

## Acknowledgments

We would like to thank Pavel A. Pevzner, Li-Fan Lu, and Jiawang Nie for insightful discussions on clonal tree reconstruction, affinity maturation, and optimization methods. We would also like to thank Yuan Wang for professional figure editing.

## Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

## Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

## Supplementary material

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fimmu.2022.1014439/full#supplementary-material>

## References

1. Tonegawa S. Somatic generation of antibody diversity. *Nature* (1983) 302:575–81. doi: 10.1038/302575a0
2. Neuberger MS, Milstein C. Somatic hypermutation. *Curr Opin Immunol* (1995) 7:248–54. doi: 10.1016/0952-7915(95)80010-7
3. Muramatsu M, Kinoshita K, Fagarasan S, Yamada S, Shinkai Y, Honjo T. Class switch recombination and hypermutation require activation-induced cytidine deaminase (AID), a potential RNA editing enzyme. *Cell* (2000) 102:553–63. doi: 10.1016/S0092-8674(00)00078-7
4. Mesin L, Schiepers A, Ersching J, Barbulescu A, Cavazzoni CB, Angelini A, et al. Restricted clonality and limited germinal center reentry characterize memory b cell reactivation by boosting. *Cell* (2020) 180:92–106.e11. doi: 10.1016/j.cell.2019.11.032
5. Georgiou G, Ippolito GC, Beausang J, Busse CE, Wardemann H, Quake SR. The promise and challenge of high-throughput sequencing of the antibody repertoire. *Nat Biotechnol* (2014) 32:158–68. doi: 10.1038/nbt.2782
6. Robinson WH. Sequencing the functional antibody repertoire—diagnostic and therapeutic discovery. *Nat Rev Rheumatol* (2015) 11:171–82. doi: 10.1038/nrrheum.2014.220
7. Yaari G, Benichou JIC, Vander Heiden JA, Kleinstein SH, Louzoun Y. The mutation patterns in b-cell immunoglobulin receptors reflect the influence of selection acting at multiple time-scales. *philosophical transactions of the royal society of London. Ser B Biol Sci* (2015) 370:20140242. doi: 10.1098/rstb.2014.0242
8. Watson CT, Glanville J, Marasco WA. The individual and population genetics of antibody immunity. *Trends Immunol* (2017) 38:459–70. doi: 10.1016/j.it.2017.04.003
9. Miho E, Yermanos A, Weber CR, Berger CT, Reddy ST, Greiff V. Computational strategies for dissecting the high-dimensional complexity of adaptive immune repertoires. *Front Immunol* (2018) 9:224. doi: 10.3389/fimmu.2018.00224
10. Laserson U, Vigneault F, Gadala-Maria D, Yaari G, Uduman M, Vander Heiden JA, et al. High-resolution antibody dynamics of vaccine-induced immune responses. *Proc Natl Acad Sci* (2014) 111:4928–33. doi: 10.1073/pnas.1323862111
11. Horns F, Vollmers C, Dekker CL, Quake SR. Signatures of selection in the human antibody repertoire: Selective sweeps, competing subclones, and neutral drift. *Proc Natl Acad Sci* (2019) 116:1261–6. doi: 10.1073/pnas.1814213116
12. Haynes BF, Kelsoe G, Harrison SC, Kepler TB. B-cell-lineage immunogen design in vaccine development with HIV-1 as a case study. *Nat Biotechnol* (2012) 30:423–33. doi: 10.1038/nbt.2197
13. Sok D, Laserson U, Laserson J, Liu Y, Vigneault F, Julien JP, et al. The effects of somatic hypermutation on neutralization and binding in the PGT121 family of broadly neutralizing HIV antibodies. *PLoS Pathog* (2013) 9:e1003754. doi: 10.1371/journal.ppat.1003754
14. Galson JD, Trück J, Clutterbuck EA, Fowler A, Cerundolo V, Pollard AJ, et al. B-cell repertoire dynamics after sequential hepatitis b vaccination and evidence for cross-reactive b-cell activation. *Genome Med* (2016) 8:68. doi: 10.1186/s13073-016-0322-z
15. Eliyahu S, Sharabi O, Elmedvi S, Timor R, Davidovich A, Vigneault F, et al. Antibody repertoire analysis of hepatitis c virus infections identifies immune signatures associated with spontaneous clearance. *Front Immunol* (2018) 9:3004. doi: 10.3389/fimmu.2018.03004
16. Stern JNH, Yaari G, Vander Heiden JA, Church G, Donahue WF, Hintzen RQ, et al. B cells populating the multiple sclerosis brain mature in the draining cervical lymph nodes. *Sci Trans Med* (2014) 6:107–248. doi: 10.1126/scitransmed.3008879
17. Lossius A, Johansen JN, Vartdal F, Holmøy T. High-throughput sequencing of immune repertoires in multiple sclerosis. *Ann Clin Trans Neurol* (2016) 3:295–306. doi: 10.1002/acn3.295
18. Elliott SE, Kongpachith S, Lingampalli N, Adamska JZ, Cannon BJ, Mao R, et al. Affinity maturation drives epitope spreading and generation of proinflammatory anti-citrullinated protein antibodies in rheumatoid arthritis. *Arthritis Rheumatol* (2018) 70:1946–58. doi: 10.1002/art.40587
19. Hsiao YC, Shang Y, DiCara DM, Yee A, Lai J, Kim SH, et al. Immune repertoire mining for rapid affinity optimization of mouse monoclonal antibodies. *mAbs* (2019) 11:735–46. doi: 10.1080/19420862.2019.1584517
20. Yermanos AD, Dounas AK, Stadler T, Oxenius A, Reddy ST. Tracing antibody repertoire evolution by systems phylogeny. *Front Immunol* (2018) 9:2149. doi: 10.3389/fimmu.2018.02149
21. Jiang N, He J, Weinstein JA, Penland L, Sasaki S, He XS, et al. Lineage structure of the human antibody repertoire in response to influenza vaccination. *Sci Trans Med* (2013) 5:171ra19. doi: 10.1126/scitransmed.3004794
22. Lee DW, Khavrutskii IV, Wallqvist A, Bavari S, Cooper CL, Chaudhury S. BRILIA: Integrated tool for high-throughput annotation and lineage tree assembly of b-cell repertoires. *Front Immunol* (2017) 7:681. doi: 10.3389/fimmu.2016.00681
23. Hoehn KB, Lunter G, Pybus OG. A phylogenetic codon substitution model for antibody lineages. *Genetics* (2017) 206:417–27. doi: 10.1534/genetics.116.196303
24. Horns F, Vollmers C, Croote D, Mackey SF, Swan GE, Dekker CL, et al. Lineage tracing of human b cells reveals the in vivo landscape of human antibody class switching. *eLife* (2016) 5:e16578. doi: 10.7554/eLife.16578
25. Lees WD, Shepherd AJ. Utilities for high-throughput analysis of b-cell clonal lineages. *J Immunol Res* (2015) 2015:323506. doi: 10.1155/2015/323506
26. DeWitt WS3rd, Mesin L, Victora GD, Minin VN, Matsen FA4. Using genotype abundance to improve phylogenetic inference. *Mol Biol Evol* (2018) 35:1253–65. doi: 10.1093/molbev/msy020
27. Rogozin I, Kolchanov N. Somatic hypermutagenesis in immunoglobulin genes. II. influence of neighbouring base sequences on mutagenesis. *Biochim Biophys Acta* (1992) 1171:11–8. doi: 10.1016/0167-4781(92)90134-L
28. Pham P, Bransteitter R, Petruska J, Goodman MF. Processive AID-catalysed cytosine deamination on single-stranded DNA simulates somatic hypermutation. *Nature* (2003) 424:103–7. doi: 10.1038/nature01760
29. Childs LM, Baskerville EB, Cobey S. Trade-offs in antibody repertoires to complex antigens. *Philos Trans R Soc B: Biol Sci* (2015) 370(1676). doi: 10.1098/rstb.2014.0245
30. Amitai A, Mesin L, Victora GD, Kardar M, Chakraborty AK. A population dynamics model for clonal diversity in a germinal center. *Front Microbiol* (2017) 8:1693. doi: 10.3389/fmicb.2017.01693
31. Reshetova P, van Schaik BD, Klarenbeek PL, Doorenspleet ME, Esveldt RE, Tak PP, et al. Computational model reveals limited correlation between germinal center b-cell subclone abundance and affinity: Implications for repertoire sequencing. *Front Immunol* (2017) 8:221. doi: 10.3389/fimmu.2017.00221
32. Davidsen K, Matsen FA. Benchmarking tree and ancestral sequence inference for b cell receptor sequences. *Front Immunol* (2018) 9:2451. doi: 10.3389/fimmu.2018.02451
33. Nee S. Birth-death models in macroevolution. *Annu Rev Ecol Evol System* (2006) 37:1–17. doi: 10.1146/annurev.ecolsys.37.091305.110035
34. Weisel FJ, Zuccarino-Catania GV, Chikina M, Shlomchik MJ. A temporal switch in the germinal center determines differential output of memory b and plasma cells. *Immunity* (2016) 44:116–30. doi: 10.1016/j.immuni.2015.12.004
35. Ellebedy AH, Jackson KJL, Kissick HT, Nakaya HI, Davis CW, Roskin KM, et al. Defining antigen-specific plasmablast and memory b cell subsets in human blood after viral infection or vaccination. *Nat Immunol*. (2016) 17(10):1226–34. doi: 10.1038/ni.3533
36. Magri G, Comerma L, Pybus M, Sintes J, Lligé D, Segura-Garzón D, et al. Human secretory IgM emerges from plasma cells clonally related to gut memory b cells and targets highly diverse commensals. *Immunity* (2017) 47:118–34. doi: 10.1016/j.immuni.2017.06.013
37. Sharbeen G, Yee CW, Smith AL, Jolly CJ. Ectopic restriction of DNA repair reveals that UNG2 excises AID-induced uracils predominantly or exclusively during G1 phase. *J Exp Med* (2012) 209:965–74. doi: 10.1084/jem.20112379
38. Gitlin AD, Shulman Z, Nussenzweig MC. Clonal selection in the germinal centre by regulated proliferation and hypermutation. *Nature* (2014) 509:637–40. doi: 10.1038/nature13300
39. Yaari G, Vander Heiden JA, Uduman M, Gadala-Maria D, Gupta N, Joel JN, et al. Models of somatic hypermutation targeting and substitution based on synonymous mutations from high-throughput immunoglobulin sequencing data. *Front Immunol* (2013) 4:358. doi: 10.3389/fimmu.2013.00358
40. Tanaka T, Nei M. Positive darwinian selection observed at the variable-region genes of immunoglobulins. *Mol Biol Evol* (1989) 6:447–4. doi: 10.1093/oxfordjournals.molbev.a008639
41. Raybould MIJ, Kovaltsuk A, Marks C, Deane CM. CoV-AbDab: the coronavirus antibody database. *Bioinformatics* (2021) 37:734–5. doi: 10.1093/bioinformatics/btaa739
42. Bao Y, Bolotov P, Dernovoy D, Kiryutin B, Zaslavsky L, Tatusova T, et al. The influenza virus resource at the national center for biotechnology information. *J Virol* (2008) 82:596–601. doi: 10.1128/jvi.02005-07
43. Stamatakis A. RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics* (2014) 30:1312–3. doi: 10.1093/bioinformatics/btu033

44. Baum BR. Phylip: Phylogeny inference package. version 3.2. joel felsenstein. *Q Rev Biol* (1989) 64:539–41. doi: 10.1086/416571
45. Drummond AJ, Rambaut A. BEAST: Bayesian evolutionary analysis by sampling trees. *BMC Evo Biol* (2007) 7:214. doi: 10.1186/1471-2148-7-214
46. Barak M, Zuckerman NS, Edelman H, Unger R, Mehr R. IgTree<sup>®</sup>: Creating immunoglobulin variable region gene lineage trees. *J Immunol Methods* (2008) 338:67–74. doi: 10.1016/j.jim.2008.06.006
47. DiNardo Z, Tomlinson K, Ritz A, Oesper L. Distance measures for tumor evolutionary trees. *Bioinformatics* (2020) 36:2090–7. doi: 10.1093/bioinformatics/btz869
48. Nakagawa R, Calado DP. Positive selection in the light zone of germinal centers. *Front Immunol* (2021) 12:661678/BIBTEX. doi: 10.3389/FIMMU.2021.661678/BIBTEX
49. Yermanos A, Greiff V, Krautler NJ, Menzel U, Dounas A, Miho E, et al. Comparison of methods for phylogenetic b-cell lineage inference using time-resolved antibody repertoire simulations (AbSim). *Bioinformatics* (2017) 33:3938–46. doi: 10.1093/bioinformatics/btx533
50. Jackman TR, Larson A, de Queiroz K, Losos JB, Cannatella D. Phylogenetic relationships and tempo of early diversification in anolis lizards. *System Biol* (1999) 48:254–85. doi: 10.1080/106351599260283
51. Walsh HE, Kidd MG, Mowm T, Friesen VL. Polytomies and the power of phylogenetic inference. *Evolution* (1999) 53:932–7. doi: 10.1111/j.1558-5646.1999.tb05386.x
52. Goldman N, Anderson JP, Rodrigo AG. Likelihood-based tests of topologies in phylogenetics. *System Biol* (2000) 49:652–70. doi: 10.1080/106351500750049752
53. Anisimova M, Gascuel O, Sullivan J. Approximate likelihood-ratio test for branches: A fast, Accurate, and Powerful Alternative. *Systematic Biology*. 55(4):539–552. doi: 10.1080/10635150600755453
54. Lewis PO, Holder MT, Holsinger KE. Polytomies and bayesian phylogenetic inference. *System Biol* (2005) 54:241–53. doi: 10.1080/10635150590924208
55. Zhang C, Dinh VU, Matsen FA 4th. Nonbifurcating Phylogenetic Tree Inference via the Adaptive LASSO. *J Am Stat Assoc* (2021), 116(534):858–873. doi: 10.1080/01621459.2020.1778481
56. Weber CR, Akbar R, Yermanos A, Pavlović M, Snapkov I, Sandve GK, et al. immuneSIM: tunable multi-feature simulation of b- and T-cell receptor repertoires for immunoinformatics benchmarking. *Bioinformatics* (2020) 36:3594–6. doi: 10.1093/bioinformatics/btaa158
57. Vander Heiden JA, Yaari G, Uduman M, Stern JN, O'Connor KC, Hafler DA, et al. pRESTO: a toolkit for processing high-throughput sequencing raw reads of lymphocyte receptor repertoires. *Bioinformatics* (2014) 30:1930–2. doi: 10.1093/bioinformatics/btu138
58. Safonova Y, Bonissone S, Kurpilyansky E, Starostina E, Lapidus A, Stinson J, et al. IgRepertoire- constructor: A novel algorithm for antibody repertoire construction and immunoproteogenomics analysis. *Bioinformatics* (2015) 31:i53–61. doi: 10.1093/bioinformatics/btv238
59. Bolotin DA, Poslavsky S, Mitrophanov I, Shugay M, Mamedov IZ, Putintseva EV, et al. MiXCR: software for comprehensive adaptive immunity profiling. *Nat Methods* (2015) 12:380–1. doi: 10.1038/nmeth.3364
60. Shlemov A, Bankevich S, Bzikadze A, Turchaninova MA, Safonova Y, Pevzner PA. Reconstructing antibody repertoires from error-prone immunosequencing reads. *J Immunol* (2017) 199:3369–80. doi: 10.4049/jimmunol.1700485
61. Safonova Y, Pevzner P. IgEvolution: clonal analysis of antibody repertoires. *BioRxiv* (2019), 1–18. doi: 10.1101/725424
62. Kepler TB, Munshaw S, Wiehe K, Zhang R, Yu JS, Woods CW, et al. Reconstructing a b-cell clonal lineage. II. mutation, selection, and affinity maturation. *Front Immunol* (2014) 5:170. doi: 10.3389/fimmu.2014.00170
63. Robert PA, Arulraj T, Meyer-Hermann M. Ymir: A 3d structural affinity model for multi-epitope vaccine simulations. *iScience* (2021) 24:102979. doi: 10.1016/j.isci.2021.102979
64. Robert PA, Akbar R, Frank R, Pavlović M, Widrich M, Snapkov I, et al. Unconstrained generation of synthetic antibody-antigen structures to guide machine learning methodology for real-world antibody specificity prediction. *BioRxiv* (2022). doi: 10.1101/2021.07.06.451258
65. Henry Dunand CJ, Wilson PC. Restricted, canonical, stereotyped and convergent immunoglobulin responses. *philosophical transactions of the royal society of London. Ser B Biol Sci* (2015) 370:20140238. doi: 10.1098/rstb.2014.0238
66. Robbiani DF, Gaebler C, Muecksch F, Lorenzi JCC, Wang Z, Cho A, et al. Convergent antibody responses to SARS-CoV-2 in convalescent individuals. *Nature* (2020) 584:437–442. doi: 10.1038/s41586-020-2456-9
67. Kirik U, Persson H, Levander F, Greiff L, Ohlin M. Antibody heavy chain variable domains of different germline gene origins diversify through different paths. *Front Immunol* (2017) 8:1433. doi: 10.3389/fimmu.2017.01433
68. Ovchinnikov V, Louveau JE, Barton JP, Karplus M, Chakraborty AK. Role of framework mutations and antibody flexibility in the evolution of broadly neutralizing antibodies. *eLife* (2018) 7:e33038. doi: 10.7554/eLife.33038
69. Elhanati Y, Sethna Z, Marcou Q, Callan CG, Mora T, Walczak AM. Inferring processes underlying b-cell repertoire diversity. *philosophical transactions of the royal society of London. Ser B Biol Sci* (2015) 370:15115. doi: 10.1101/015115
70. McCoy CO, Bedford T, Minin VN, Bradley P, Robins H, Matsen FA. Quantifying evolutionary constraints on b-cell affinity maturation. *philosophical transactions of the royal society of London. Ser B Biol Sci* (2015) 370:20140244. doi: 10.1098/rstb.2014.0244
71. Luo S, Perelson AS. The challenges of modelling antibody repertoire dynamics in HIV infection. *philosophical transactions of the royal society of London. Ser B Biol Sci* (2015) 370:20140247. doi: 10.1098/rstb.2014.0247
72. Kurosawa Y, Tonegawa S. Organization, structure, and assembly of immunoglobulin heavy chain diversity DNA segments. *J Exp Med* (1982) 155:201–18. doi: 10.1084/jem.155.1.201
73. Marcou Q, Mora T, Walczak AM. High-throughput immune repertoire analysis with IGoR. *Nat Commun* (2018) 9:561. doi: 10.1038/s41467-018-02832-w
74. Bonsignori M, Zhou T, Sheng Z, Chen L, Gao F, Joyce MG, et al. Maturation pathway from germline to broad HIV-1 neutralizer of a CD4-mimic antibody. *Cell* (2016) 165:449–63. doi: 10.1016/j.cell.2016.02.022
75. Ratmann O, Hodcroft EB, Pickles M, Cori A, Hall M, Lycett S, et al. Phylogenetic tools for generalized HIV-1 epidemics: Findings from the PANGEA-HIV methods comparison. *Mol Biol Evol* (2017) 34:185–203. doi: 10.1093/molbev/msw217
76. Moshiri N, Ragonnet-Cronin M, Wertheim JO, Mirarab S. FAVITES: simultaneous simulation of transmission networks, phylogenetic trees and sequences. *Bioinformatics* (2019) 35:1852–61. doi: 10.1093/bioinformatics/bty921