Accounting for partner notification in epidemiological birth-death models

Anna Zhukova[1,2*], Olivier Gascuel[3],

1 G5 Evolutionary Dynamics of Infectious Diseases, Institut Pasteur, Université de Paris, Paris, France
2 Bioinformatics and Biostatistics Hub, Institut Pasteur, Université de Paris, Paris, France
3 Institut de Systématique, Evolution, Biodiversité (ISYEB) - URM 7205 CNRS, Muséum National d'Histoire Naturelle, SU, EPHE & UA, Paris, France

* anna.zhukova@pasteur.fr

Abstract

Phylodynamics bridges the gap between classical epidemiology and pathogen genome sequence data by estimating epidemiological parameters from time-scaled pathogen phylogenetic trees. The models used in phylodynamics typically assume that the sampling procedure is independent between infected individuals. However, this assumption does not hold for many epidemics, in particular for such sexually transmitted infections as HIV-1, for which partner notification schemes are included in health policies of many countries.

We developed an extension of phylodynamic multi-type birth-death (MTBD) models with partner notification (PN), and a simulator to generate trees under MTBD and MTBD-PN models. We proposed a non-parametric test for detecting partner notification in pathogen phylogenetic trees. Its application to simulated data showed that it is both highly specific and sensitive. For the simplest representative of the MTBD-PN family, the BD-PN model, we solved the differential equations and proposed a closed form solution for the likelihood function. We implemented it in a program, which estimates the model parameters and their confidence intervals from phylogenetic trees. It performed accurate estimations on simulated data, and detected partner notification in HIV-1 B epidemics in Zurich and the UK. Importantly, we showed that not accounting for partner notification when it is present leads to bias in parameter estimation with the BD model, while BD-PN parameter estimator performs well both in presence and in absence of partner notification.

Our PN test, MTBD-PN tree simulator and BD-PN parameter estimator are freely available at github.com/evolbioinfo/treesimulator and github.com/evolbioinfo/bdpn.

Author summary

Phylodynamic models can estimate epidemiological parameters such as the number of secondary infections R_e from pathogen phylogenetic trees (i.e., genealogies, inferred from pathogen genomic sequences). These models do not account for partner notification (contact tracing), and instead assume that detection of new cases is independent between infected individuals. However, especially for sexually transmitted infections, partner notification plays an important role and is included in health policies of many countries.

We developed a phylodynamic model accounting for partner notification and a test for the detection of partner notification in pathogen phylogenetic trees. Our test and epidemiological parameter estimator showed good performance both on simulated and real data. We detected the presence of partner notification in the HIV-1 B epidemics in Zurich and the UK, and corrected the previous parameter estimates made without accounting for partner notification.

Introduction

The interaction of epidemiological and evolutionary processes leaves a footprint in pathogen genomes. Phylodynamics leverages this footprint to estimate epidemiological parameters [1, 2]. It relies on models that bridge the gap between traditional epidemiology and sequence data by estimating such parameters as the basic reproduction number, R_0, from topology and branch lengths of pathogen phylogenies (i.e., genealogies of the pathogen population, approximating the transmission trees) combined with metadata on the samples. Classical epidemiological methods make inferences from incidence curves, which in the presence of imported cases might be biased and give appearance of elevated local transmission. In contrast, phylodynamic approach can overcome such biases as it uses the transmission structure as one of its information sources [3]. Using genomic data with phylodynamic estimations can provide valuable insights for preventing epidemic spread.

Phylodynamic models can be classified into two main families: coalescent [4–6] and birth-death (BD) [7–10]. Coalescent models are often preferred for estimating deterministic population dynamics, however, for highly stochastic processes, such as the dynamics of emerging pathogens, BD models are better adapted [11]. In the classic BD model with incomplete sampling [9], births represent pathogen transmission events (happening at a constant transmission rate), while deaths correspond to becoming non-infectious (e.g., due to healing, self-isolation, starting a treatment, or death, modelled with a constant removal rate).

However, the spread and detection of an epidemic can be non-homogenous and depend on different factors, including governmental health policies (e.g., quarantine measures or development of pathogen detection tests), pathogen evolution over time (e.g., some SARS-CoV-2 variants are more transmissible than the others), differences between host individuals (e.g., due to their immune system particularities or to their behaviour).

To allow for heterogeneity at the population level, the multi-type birth-death (MTBD) extension of the classical BD model was developed by Stadler *et al.* [12]. The MTBD framework allows for different types of individual states (and changes between them, modelled with state change rates). MTBD models are phylodynamic analogies of compartmental models in classical epidemiology (e.g., SIR, Susceptible-Infectious-Removed). Examples of MTBD models include the birth-death exposed-infectious (BDEI) model, which was designed for pathogens featuring an incubation period between the moments of infection and of becoming infectious (e.g., Ebola and SARS-CoV-2), and the birth-death with superspreading model (BDSS) accounting for the fact that some individuals might spread the pathogen (e.g., HIV) more than the others [13].

To allow for heterogeneity over time, Stadler *et al.* [14] developed one-state Bayesian birth-death skyline plot (BDSKY) that divides the time into intervals and allows for different piecewise constant rates on them. Kühnert *et al.* [15] combined the MTBD model with the BDSKY to allow for both piecewise-constant rate changes over time and multiple individual types. In particular the skyline approach can be useful to account for changes in the sampling policies (e.g., before/after HIV discovery or before/after

PCR test spread for SARS-CoV-2).

Another important source of heterogeneity that none of these models accounts for is caused by non-independent sampling. One can set different sampling probabilities for different types of individuals in the MTBD framework, or change of the sampling probability between time intervals with the BDSKY, however, none of these models can account for correlation effects in sampling, in particular due to partner notification (PN). PN (or contact tracing) is a process in which contacts of detected infected individuals are identified and offered testing for infection. It has been applied to various infectious diseases, including syphilis, HIV, tuberculosis, Ebola and COVID-19, and shown to successfully reduce transmission [16]. For instance, for HIV, partner notification showed increased early referral and initiation of treatment, and several countries include it in their national HIV testing services policies [17].

In this study we propose an extension of MTBD models that allows for modeling non-random sampling due to partner notification, MTBD-PN. We describe the -PN extension and its assumptions, and present a transmission tree simulator under MTBD-PN models. We propose a non-parametric test for detecting partner notification in pathogen phylogenetic trees. For the simplest representative of this model family, the BD-PN model, we also derive the equations (and their closed-form solution) for its likelihood calculation and present a maximum-likelihood parameter estimator. We test our BD-PN parameter estimator on simulated data, and apply it to the study of HIV-1 B epidemics in Zurich and in the UK. Importantly, we show that not accounting for partner notification when it is present leads to bias in parameter estimation with the BD model, while BD-PN parameter estimator performs well both in presence and in absence of partner notification.

Materials and methods

In a pathogen transmission tree $\mathscr{T}$ (approximated by a time-scaled pathogen phylogeny) the tips represent sampled pathogens while bifurcations (i.e., internal nodes) correspond to pathogen transmissions. The tree branches are measured in units of time, and the time goes forward, from the tree root (representing the beginning of the (sub-)epidemic, at time $t = 0$) to the time of the last sampled tip (at $t = T$).

Extending Multi-Type Birth-Death models with Partner Notification

Birth-Death (BD) model with incomplete sampling

Under the basic birth-death (BD) model [9] all the individuals are in the same state. They can transmit with a constant rate λ, get removed with a constant rate ψ, and their pathogen can be sampled upon removal with a constant probability ρ.

Under this model we can calculate the likelihood density $L(\mathscr{T}|\Theta)$ of a tree $\mathscr{T}$ under parameter values $\Theta = \{\lambda, \psi, \rho\}$ (see Mathematical formulation of BD and BD-PN models). The likelihood density can then be used to estimate model parameters from a phylogenetic tree (e.g., by finding parameter values that maximize it, in the maximum-likelihood framework). In particular, it permits the estimation of the following epidemiological parameters:

- *effective reproduction number* $R_e = \frac{\lambda}{\psi}$, expected number of individuals directly infected by an infectious case;
- *infectious time* $= \frac{1}{\psi}$, expected time during which an infectious individual can further spread the epidemic.

Note that the BD model is asymptotically unidentifiable (see Remark 3.4 in [9]). To become identifiable it requires one of its parameters to be fixed. In practice, it is often the sampling probability ρ, as it may be approximated from epidemiological data (e.g., the proportion of sampled cases among the declared ones) or the infectious time $\frac{1}{\psi}$ (estimated from observations of infected cases).

Multi-Type Birth-Death (MTBD) models

Multi-Type Birth-Death (MTBD) models [12] add population structure to the BD model by allowing different individual states, transmissions between them and state changes. A general MTBD model with m individual states has $2m^2 + m$ parameters: An individual in state $k \in \{1, \ldots, m\}$ can be removed at a constant average rate ψ_k (with pathogen sampling probability ρ_k), change their state to state $l \in \{1, \ldots, m\}$ at a constant average rate μ_{kl} ($k \neq l$), and transmit their pathogen to an individual in state $r \in \{1, \ldots, m\}$ at a constant average rate λ_{kr}. The time between events is hence modeled with exponential distributions.

Partner Notification (PN) extension

We propose an extension of the MTBD models that adds partner notification (PN). In the partner notification process, an identified infected index patient notifies their potentially infected contacts (e.g., sexual or drug injection partners for HIV), either on their own (patient referral) or with the help of a health service provider (provider referral, when a health adviser notifies the partners without identifying the index patient; or contract referral, when if the index patient fails to notify partners within a specified time period, the health adviser proceeds to provider referral) [18]. The notified partners are then provided with appropriate tests and treatment. Therefore, a successful detection of a new case B via partner notification by an index patient A implies several conditions to be met: (i) A remembers the contact with B and either notifies B on their own or gives the B's contacts to a health advisor for assisted notification; (ii) B is successfully reached and notified; (iii) B agrees to take a test; (iv) B is indeed infected. For instance, according to the report from the Public Health England [19], in England in 2013 for HIV, reaching the stage (ii) had an estimated probability of about 50%; reaching the stage (iii) – of about 41% (as 83% of all the patients who attended a clinic through PN had a reported HIV test); and the stage (iv) – of less than 3%.

All these conditions are hardly identifiable separately from only sequence data, hence in our model we combine them into one, "an infected person B being successfully notified by A", and model it with a constant notification probability υ which is applied upon sampling of A.

For simplicity, a positive draw with probability υ models a slightly modified scenario: (i) A remembers the contact with B and either notifies B on their own or gives the B's contacts to a health advisor for assisted notification; (ii', similar to iv in the previous scenario) B is indeed infected, moreover, *the infection of either A or B is caused by the pathogen transmission during their contact*; and either (iii', same as ii) B is successfully reached and notified and (iv', same as iii) agrees to take a test (which at the end of the considered time period might not yet be available in the database); or (iii") *by the time of notification, B is already removed by other means* (already got sampled or stopped being infectious without sampling, e.g., died).

The condition (ii') in contrast to (iv) excludes the possibility of notifying partners if there was no pathogen transmission during the contact with them (and thus mostly focuses on regular partners). We additionally simplify the model by allowing to notify only the last such partner. From now on, we will use the term "partner" to denote a partner, during a contact with whom a pathogen transmission happened.

Upon notification, if not yet removed, the partner's pathogen is sampled almost instantaneously (modeled via a constant notified sampling rate $\phi >> \psi$). Upon sampling, the partner might notify in their turn.

BD-PN hence has 3 classical BD and 2 additional parameters:

- λ – constant transmission rate: an infected individual transmits their pathogen to a newly infected individual with this rate. This leads to an internal node in the transmission tree;
- ψ – constant removal rate: an infected individual stops being infectious at this rate. This leads to a tip in the transmission tree (observed or hidden);
- ρ – constant sampling probability: upon removal the individual's pathogen might get sampled with this probability. This leads to a tip in the transmission tree being observed;
- υ – constant notification probability: upon sampling, the sampled individual might notify their most recent partner with this probability. This leads to a removal rate change for the partner branch in the transmission tree;
- $\phi >> \psi$ – constant notified sampling rate: a notified individual (if not yet removed via the standard procedure) gets removed *and observed* with this rate. This leads to an observed tip in the transmission tree.

This model is a simplification of the notification process, in particular it makes 3 assumptions:

1. only observed individuals can notify (instead of any removed individual);
   This is a realistic hypothesis in which an individual is "observed" because he or she enters a diagnostic and follow-up process that includes partner notification. Without entering this process, he or she is not observed and does not notify.
2. successfully notified individuals get sampled (i.e., their pathogens are always observed upon removal, unlike the standard case where the observation happens with a sampling probability ρ);
   Realistically again, we assume here that the diagnostic and follow-up process works as expected, sampling notified patients.
3. only the most recent partner can get notified;
   In the case of sexually transmitted diseases, notifying only the most recent partner could roughly correspond to notifying the "main" partner (e.g., spouse). This assumption might need to be removed or relaxed (e.g., to up to a certain fixed number of most recent partners, or all the partners within a certain time interval) in the future work. It is however useful to make the mathematics and likelihood calculation under the model manageable.

   Note that the "most recent partner" relationship is not necessarily symmetrical. The most recent partner j of individual i might have transmitted their virus further (to individual k) after the contact with i. Hence, while j is the most recent partner of i, j's most recent partner is k, and, once notified, j might notify k.

These assumptions can be easily relaxed in a tree simulator (e.g., to be used with a deep learning estimator [20], which does not require likelihood calculation).

The -PN extension can be generalized from BD to MTBD models: at the moment of sampling, an individual of any type can notify their last partner with the probability υ, whose removal rate will then get replaced by a constant sampling rate $\phi >> \psi_k \ \forall k \in \{1, \ldots, m\}$. The -PN extension therefore adds two parameters to an MTBD model: ϕ and υ.

MTBD-PN tree simulator and test datasets

We implemented a tree simulator that generates sampled transmission trees under MTBD and MTBD-PN models (with m states). It is Gillespie-based, generates state change, transmission and removal times, and only reconstructs the sampled parts of the tree to save memory and increase speed. We give its algorithmic details in S1 Appendix. The results in section "PN test is highly specific and sensitive at detecting partner notification" show that the trees generated under MTBD and MTBD-PN models are different and easily distinguishable.

Using our simulator, we generated trees under three classical MTBD-family models: BD, BDEI (Birth-Death Exposed-Infectious [13]) and BDSS (Birth-Death with Super Spreading [12]), and under their -PN versions: BD-PN, BDEI-PN and BDSS-PN models. (One can see Fig. 1 in [20] for a graphical representation of BD, BDEI and DBSS models and their parameters).

For each model, we generated 100 trees with 500–1 000 tips. For each tree the parameter values were drawn randomly within the following bounds:

- $R_e = \frac{\lambda}{\psi} \in]1, 5]$,
- removal rate $\psi \in]0.05, 0.2]$,
- sampling probability $\rho \in]0.1, 0.9]$,
- *(for -PN models only)* ratio between the partner removal rate after notification and the removal rate $\frac{\phi}{\psi} \in]50, 500]$,
- *(for -PN models only)* partner notification probability $\upsilon \in]0.01/\rho, 0.9]$ (the lower bound was picked this way to ensure that the probability to be a notified partner is at least $0.01 = \rho\upsilon$),
- *(for BDEI models only)* becoming-infectious rate (at which an exposed individual becomes infectious) $\mu \in]0.2, 10]$,
- *(for BDSS models only)* fraction of superspreaders $f_{SS} = \frac{\lambda_{SS}}{\lambda_{SS}+\lambda_{SN}} \in]0.05, 0.2]$,
- *(for BDSS models only)* superspreading transmission ratio $x_{SS} = \frac{\lambda_{SS}}{\lambda_{NS}} = \frac{\lambda_{SN}}{\lambda_{NN}} \in]3, 10]$.

The common parameters were consistent between models, in the sense that the i-th tree under the BDSS model had the same R_e, ψ and ρ values as the i-th trees under BDSS-PN, BD(-PN) and BDEI(-PN) models; the i-th tree under the BDSS-PN model had the same ϕ and υ values as the i-th trees under BD-PN and BDEI-PN models, etc.

Testing for presence of Partner Notification

To access whether a given tree reveals partner notification, we developed a non-parametric test based on cherries (two-tip clades).

The intuition behind the test is that in the presence of partner notification the tree will contain more cherries whose tips are close in time. Indeed, in a transmission tree internal nodes correspond to transmissions while tips correspond to sampling events. Hence if a transmission tree is complete, a cherry corresponds to a transmission from an infected individual to their most recent partner, followed by sampling of both. If there is no partner notification, these two sampling events are independent, while in the presence of partner notification the first sampling might trigger the second one, and they will be close in time (e.g., the second cherry in Fig. 1a). For an incomplete transmission tree (where some infected individuals were not sampled), the difference

would be that some cherries might include hidden transmissions. The notification pattern should however still be observed for some of the cherries.

More formally, in the absence of partner notification (null hypothesis, H0), we assume that branch lengths of two cherry tips are independently drawn from the same distribution (which is the case for MTBD models). In the presence of partner notification (alternative hypothesis, H1), we assume that for some of the cherries (including notified partners) their branch lengths are not independent and are similar. Therefore, in the absence of partner notification two external branches taken randomly from the tree and two external branches belonging to the same cherry should be indistinguishable, if they all start at about the same time. Being close in starting time is important as the tree is censored by the end of the sampling period: $t \leq T$. Hence the external branches that start close to the end of the sampling period would be on average shorter (as unless sampled before T they will stay hidden) than those that start closer to the root [21].

Let us assume that there are N cherries in the tree. For each cherry i ($1 \leq i \leq N$) with tips $i0$ and $i1$, we calculate the difference between its tip times $d_i = |t_{i0} - t_{i1}|$. We then generate a collection of reshuffled cherries by (1) ordering the cherries by their root dates; (2) for each cherry i randomly selecting one of its tips $i* \in \{i0, i1\}$; (3) swapping the selected tips between the neighboring cherries: the tip $2i*$ gets swapped with the tip $2i-1*$ ($1 \leq i \leq N/2$) as illustrated in Fig. 1. (If the total number of cherries N is odd, instead of swapping the selected tips of the last two cherries, we swap the selected tips of the last three cherries in a cycle as in Fig. 1). We then calculate the time differences $d_{i'}$ between the tips of these reshuffled cherries. The test then compares d_i to $d_{i'}$: $sign(d_{i'} - d_i) = 1$ if $d_{i'}$ is larger than d_i; $= -1$ if it is smaller; or $= 0$ if they are equal (this could happen as in real datasets dates are often truncated, e.g., to a month or even a year). Note that if cherry tip branch lengths are independently drawn from the same distribution, $d_{i'}$ will be smaller than d_i for about half of the cherries. However, if some cherries are "notified" their real tip differences should be smaller than reshuffled ones. We therefore output the pvalue calculated by the sign test.

We implemented the PN-test in Python 3. It is available as a part of BD-PN parameter estimator (*pn_test*, see section "BD-PN parameter and CI estimator").

Mathematical formulation of BD and BD-PN models

Master equation and tree likelihood under BD model

The BD model can be described with master equations representing the likelihood densities of evolving as on the observed transmission tree, with time t going forward from the root ($t = 0$) till the time of the last sampled tip ($t = T$). These likelihood densities depend on the probability $U(t)$ of an unobserved transmission tree that started from one individual at time t and evolved till T, without this individual nor any of their induced cases being sampled:

$$\begin{cases} \dot{U}(t) = & \big(\lambda + \psi\big)U(t) \leftarrow \textit{no event in the next infinitesimal time } \Delta t \\ & -\lambda U^2(t) \leftarrow \textit{transmission, followed by unsampled evolutions of both subtrees} \\ & -\psi(1-\rho) \leftarrow \textit{removal without sampling} \\ U(T) = & 1 \leftarrow \textit{the probability to stay unsampled over time 0 is 1} \end{cases} \tag{1}$$

With the equation (1), we can write down the equation for the probability density $p^{(i)}(t)$ of evolving as on an observed tree branch i till time t_i, starting on this branch at time $t \leq t_i$. The master equations for the BD model were initially developed by Stadler *et al.* [9], however what we present here is their branch-specific formulation that we proposed in [22]: In our formulation, $p^{(i)}(t)$ describes an evolution along a tree branch i,

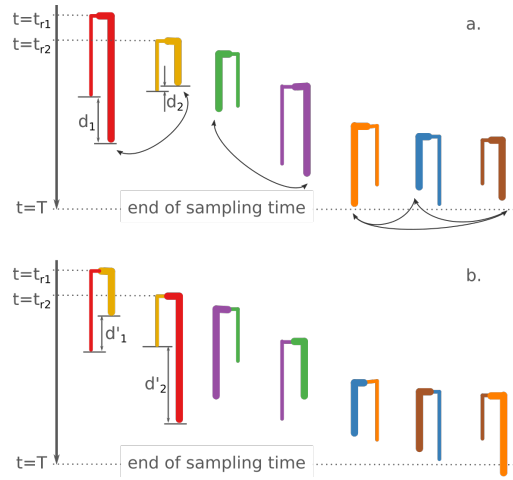

Fig 1. Reshuffled cherry generation. **a.** Cherries of the original tree sorted by their root times (times of the roots of the two oldest cherries, t_{r1} and t_{r2}, are shown on the left). The randomly selected tip is shown with a bold branch for each cherry (e.g., the tip on the right for the leftmost cherry). The tip sampling time differences are shown for the two leftmost cherries (d_1 and d_2). **b.** Reshuffled cherries, which were obtained from the original ones by swapping the selected tips between the neighbouring cherry couples (as shown with arrows in a.): the first and the second; the third and the forth. As the total number of cherries (7) is odd, the last three (instead of two) cherries exchanged their selected tips in a cycle. The reshuffled tip sampling time differences are shown for the two leftmost cherries (d'_1 and d'_2).

without taking into account the branch's subtree. t_i corresponds to the time of the equation's initial condition (typically at the end of the branch i).

$$\begin{cases} \dot{p}^{(i)}(t) = & \big(\lambda+\psi\big)p^{(i)}(t) \leftarrow \textit{no event in the next infinitesimal time } \Delta t \\ & -2\lambda p^{(i)}(t)U(t) \leftarrow \textit{transmission, where one of the subtrees stayed unsampled} \\ p^{(i)}(t_i) = & 1 \leftarrow \textit{the probability of no change over zero time is 1} \end{cases} \tag{2}$$

Using this equation, we can calculate the likelihood density $L(\mathscr{T}|\Theta)$ of a tree $\mathscr{T}$ under parameter values $\Theta = \{\lambda, \psi, \rho\}$ as:

$$\begin{aligned} logL(\mathscr{T}|\Theta) = & \sum_{i\in tips} log(\psi\rho) \leftarrow \textit{sampling of tips} \\ & + \sum_{i\in \substack{internal \\ nodes}} log\big(2\lambda p^{(i0)}(t_i)p^{(i1)}(t_i)\big) \leftarrow \textit{transmission events, followed by child} \\ & \textit{branch evolutions (each can be a donor)} \end{aligned} \tag{3}$$

Master equation under BD-PN model

Tree branches under BD vs BD-PN models. In the classical BD model with incomplete sampling, a branch of the tree corresponds to a virus evolution between the transmission corresponding to the node at the beginning of this branch and either a sampling event (if the branch is external) or a transmission corresponding to the node at the end of the branch (if it is internal). Note, that since sampling is incomplete, hidden transmissions (leading to fully unsampled subtrees) could occur along this branch, where the unsampled subtree could correspond to either donor or recipient. Hence, we do not know whether the individual at the end of the branch (at time t_i) is the same as

at any time $t < t_i$ along the branch. These two individuals could be different if a hidden transmission where the donor subtree stayed unobserved, occurred between times t and t_i. $p^{(i)}(t)$ in equation (2) describes an evolution along such a tree branch, integrating over all the possibilities, starting from no hidden transmission along the branch and including any number of hidden transmissions with either donor or recipient tree staying unsampled (and hence potentially different individuals at times t and the branch end t_i).

In trees generated by the BD-PN model, however, the individuals do not behave in the same way when they are notified or not and, hence, other types of branches are possible. As in a real tree we do not know who might have notified whom, we have to integrate over all the possibilities to calculate the tree likelihood. We calculate the likelihood density of a tree using a pruning algorithm [23], while considering all the combinatorial possibilities for each node. For instance, for each tree tip we consider four possibilities: whether the corresponding individual was notified or not by someone, and on top of that, whether the tip's individual notified or not their most recent partner upon sampling. Before describing the combinations behind tree likelihood calculation, we will introduce additional tree branch types.

Additional master equations under BD-PN model. Let us consider a person A who got sampled at time t_A and successfully notified their partner B, who as a consequence got sampled at time $t_B \geq t_A$. A and B hence correspond to tree tips, while their most recent common ancestor node AB corresponds to the transmission between A and B (see Fig. 2(1a)).

According to our model assumption 3, only the most recent partner can get notified. Hence, AB must be the parent node of A, and no hidden transmission is possible along the branch AB-A (as otherwise there would have been a partner of A who is more recent than B).

We will use a **notifier branch probability** $p_n^{(A)}(t)$ for such cases (when no hidden transmission occurred along the branch A, coloured violet in Fig. 2):

$$\begin{cases} \dot{p}_n^{(A)}(t) & = \left(\lambda + \psi\right) p_n^{(A)}(t) \leftarrow \textit{no event in the next infinitesimal time } \Delta t \\ p_n^{(A)}(t_A) & = 1 \end{cases} \tag{4}$$

As B corresponds to A's partner, the individual on the path between the transmission node AB and the tip B cannot change (is always B) and only transmissions from B to someone else are possible along the path leading from AB to B. These transmissions can be either hidden (where the recipient tree stayed unobserved) or observed: in both cases we know which subtree corresponds to the donor (the one that contains B). However, depending on whether B already got notified ($t \geq t_A$ in Fig. 2(1a)) or not yet ($t < t_A$ in Fig. 2(1a)), B's removal rate is different (is correspondingly ϕ or ψ). We will use a **partner branch probability before notification** $p_{pb}^{(x)}(t)$ for cases when the individual at the end of the branch (or considered time interval, here $t_x = t_A$) is the same as at time t, has not yet been notified, but might have transmitted to someone else (unobserved) along the way (coloured orange in Fig. 2):

$$\begin{cases} \dot{p}_{pb}^{(x)}(t) & = \left(\lambda + \psi\right) p_{pb}^{(x)}(t) \leftarrow \textit{no event in the next infinitesimal time } \Delta t \\ & \quad - \lambda p_{pb}^{(x)}(t) U(t) \leftarrow \textit{transmission with hidden recipient subtree} \\ p_{pb}^{(x)}(t_x) & = 1 \end{cases} \tag{5}$$

We will use a **partner branch probability after notification** $p_{pa}^{(i)}(t)$ for cases when the individual at the end of the branch is the same as at time t, by time t has already been notified, and might have transmitted to someone else (unobserved) along the way (coloured dark red in Fig. 2):

$$\begin{cases} \dot{p}_{pa}^{(B)}(t) & = \Big(\lambda+\phi\Big)p_{pa}^{(B)}(t) \leftarrow \textit{no event in the next infinitesimal time } \Delta t \\ & \quad - \lambda p_{pa}^{(B)}(t)U(t) \leftarrow \textit{transmission with hidden recipient subtree} \\ p_{pa}^{(B)}(t_B) & = 1 \end{cases} \tag{6}$$

Finally, once sampled, B could have notified their own most recent partner (represented by the most recent hidden subtree in Fig. 2(1a)). This would imply that between the moment of transmission between B and their most recent partner (t_x) and the moment of B's sampling (t_B) there was no event along the B's branch. Hence, if B is a notifier, the part of the B's branch after t_x should be modeled with a notifier probability instead (no event along the branch). Note that once B is notified, the removal rate becomes ϕ, and hence a **both notified and notifier branch probability** $p_b^{(i)}(t)$ applies:

$$\begin{cases} \dot{p}_{b}^{(B)}(t) & = \Big(\lambda+\phi\Big)p_{b}^{(B)}(t) \leftarrow \textit{no event in the next infinitesimal time } \Delta t \\ p_{b}^{(B)}(t_B) & = 1 \end{cases} \tag{7}$$

The equations 1, 2, 4-7 have the following closed form solutions:

$$\begin{cases} U(t) = \frac{\lambda+\psi}{2\lambda} + \frac{c_1}{2\lambda}\left(\frac{E(t)-1}{E(t)+1}\right) \\ p^{(i)}(t) = \left(\frac{E(t_i)+1}{E(t)+1}\right)^2 e^{c_1(t-t_i)} \\ p_n^{(i)}(t) = e^{(\lambda+\psi)(t-t_i)} \\ p_b^{(i)}(t) = e^{(\lambda+\phi)(t-t_i)} \\ p_{pb}^{(i)}(t) = \sqrt{p^{(i)}(t)p_n^{(i)}(t)} \\ p_{pa}^{(i)}(t) = e^{(\phi-\psi)(t-t_i)}p_{pb}^{(i)}(t) \end{cases}, \tag{8}$$

$$\textit{where } c_1 = \sqrt{(\lambda-\psi)^2 + 4\lambda\psi\rho},\ c_2 = \frac{c_1+\lambda-\psi}{c_1-\lambda+\psi}, E(t) = c_2 e^{c_1(t-T)}$$

Note that on top of the scenario we just described (shown in Fig. 2(1a)), two other scenarios are possible for a person A who once sampled (at t_A) notified their most recent partner B, who got sampled at time t_B:

1b. by the time t_A, B was already notified by someone else (e.g., at time $t_C < t_A$ in Fig. 2(1b)).

    This would lead to a shorter partner-before-notification part on the B's branch (finishing at t_C), and a longer notified-partner part (between t_C and t_B).

1c. by the time t_A, B was already sampled via a standard procedure (at time $t_B < t_A$, hence the branch B would finish before the branch A, as in Fig. 2(1c)).

    This would lead to the partner-before-notification part on the B branch finishing at t_B and no notified-partner part. Note that the branches between AB and B stay oriented as B remains the most recent partner of A.

Note that if A is a notifier, A is always observed, as notification only happens upon sampling. However, B could have stay unsampled: either because once notified (by A or another partner, who notified them before A did) B has not yet got sampled (Fig. 2(2a-b,3a-b)); or because B got removed via the standard procedure without sampling before any notification (Fig. 2(2c,3c)). Moreover, between the transmission AB and the moment of B's notification/removal, B might have transmitted to someone else (e.g., C in Fig. 2(2), BC denoting the transmission). If that happened and someone in BC's subtree got sampled, the branch AB-A would still correspond to the notifier A, while the other AB's child branch would partially correspond to B (between AB and

BC) and partially to C (starting at BC), as in Fig. 2(2). Otherwise (if there was no such transmission or all such subtrees stayed unobserved, as in Fig. 2(3)), the node AB will not be observed either, and A's branch will get extended with its parent's branch. Hence only the bottom part (after AB) of the A's branch would correspond to the notifier probability (see Fig. 2(3)).

To model the cases when B is not sampled, we will need to describe the probability of a **mixed branch** $p^{(i)}_{m(s1,s2)}(t, t_r)$ for branches that include a hidden partner subtree (AB-C branches in Fig. 2(2) and A's tip branch in Fig. 2(3)). $s1$ here denotes the type of the top of the branch (e.g., between AB and BC in Fig. 2(2), partner) and $s2$ denotes the type of the bottom of the branch (e.g., between AB and A in Fig. 2(3), notifier). Let us denote the first notification time of the hidden partner as t_r, and the time of the hidden partner subtree start as t_h, while as usual t_i stands for the time at the branch end. For instance, in Fig. 2(2a) $t_i = t_C$, $t_r = t_A$, $t_h = t_{BC}$; in Fig. 2(2b) $t_i = t_r = t_C$, $t_h = t_{BC}$; in Fig. 2(3c) $t_i = t_r = t_A$, $t_h = t_{AB}$.

$p^{(i)}_{m(s1,s2)}(t, t_r)$ combines three elements: the probability of the top part of the branch $p^{(h)}_{top=s1}(t)$, which finishes with a transmission to or from the hidden partner, the probability of the bottom part of the branch $p^{(i)}_{bottom=s2}(t_h)$ (without taking into account the event at its end) and the probability of the hidden partner subtree $U_p(t_h, t_r)$. We describe the details behind calculation of $p^{(i)}_{m(s1,s2)}(t, t_r)$ in S2 Appendix.

In real trees we do not know sampled individual's status (individual who did not notify, notifier, partner notified in time or not), and hence during the tree likelihood calculation, we will have to consider all these scenarios.

Tree likelihood under BD-PN model

Using the branch probability formulas described above, we can calculate the likelihood density of a tree using a pruning algorithm [23]. For each visited node i we calculate the likelihood density of its subtree including the branch ji connecting i to its parent node j for two cases depending on i's receiving-a-notification status (shown with the first superscript, $^{\not n}$ or ^{n}). In the first case, $l^{(\not n)}(i)$, the node i corresponds to an unnotified individual. In the second case, $l^{(n)}(i, r)$, the branch ji and the node i correspond to an (eventually) notified partner (notified by the individual corresponding to a tip r at time t_r). If i is a tip, we additionally consider whether there is an observed partner in our tree, whom the individual i notified (shown with the second superscript, ^{n} or $^{\not n}$). For instance, $l^{(\not n,n)}(i)$ represents the case where the tip i was not notified but notified their most recent partner (who is observed).

The tree $\mathscr{T}$ likelihood density $L(\mathscr{T}|\Theta)$ under BD-PN model with parameters $\Theta = \{\lambda, \psi, \rho, \phi, \upsilon\}$ can be calculated as $l^{(\not n)}(root)$, as the root can only correspond to an unnotified individual (since there was no observed transmission before it).

We describe the subtree likelihood density calculation case by case in S2 Appendix.

Extension to forests

As we previously described in [22], the likelihood calculation with MTBD models can be easily extended to forests. This extension applies also to the BD-PN model. Forests are useful for cases when a (sub-)epidemic started with several infected individuals (e.g., due to multiple pathogen introductions to a country of interest or due to a change of health policies leading to a change in parameter values). In this case the (sub-)epidemic leads to a forest $\mathscr{F}$ of f observed trees: $\mathscr{T}_1, \ldots, \mathscr{T}_f$. The forest $\mathscr{F}$ might also include a certain number u of unobserved trees, for which none of their tips got sampled. Forest likelihood formula hence combines the likelihoods of f observed and u hidden trees, and

can be represented in a logarithmic form (9). The tree likelihood formula is its special case, where $f = 1$ and $u = 0$.

$$\begin{aligned} logL(\mathscr{F}|\Theta, u) &= u\, logU_{hidden}(\Theta) + \sum_{j=1}^{f} logL(\mathscr{T}_j|\Theta), \\ &\textit{where } U_{hidden}(\Theta) = U(t_{start}), \\ &\quad \textit{and } t_{start} \textit{ is the (potentially averaged) start time of the hidden trees.} \end{aligned} \tag{9}$$

For given model parameter values Θ we can estimate the number of hidden trees u from the number of observed trees f as:

$$u = f\frac{U_{hidden}(\Theta)}{1 - U_{hidden}(\Theta)} \tag{10}$$

BD-PN parameter and CI estimator

We implemented a parameter estimator for the BD-PN model (which we called bdpn). It estimates the BD-PN model parameters $\Theta = (\lambda, \psi, \phi, \rho, \upsilon) \in \mathbb{R}^5$ for a forest $\mathscr{F}$ comprising $f \geq 1$ observed trees in the maximum-likelihood framework, where one of the BD parameters (λ, ψ or ρ) is fixed (for identifiability reasons). We also implemented the classical BD model estimator, bd.

Once the optimal parameter values are found, we calculate their confidence intervals (CIs) using Wilks' method [24]. For each non-fixed parameter $p \in \Theta$, we calculate its 95%-CI as $[\tilde{p} \in \mathbb{R} \mid logL(\mathscr{F}|\Theta_{opt|p=\tilde{p}}) > logL(\mathscr{F}|\Theta_{opt}) - \frac{\chi_1^2(0.95)}{2}]$, where $\chi_1^2(0.95) \approx 3.84$ is the value of chi-squared distribution with 1 degree of freedom corresponding to the significance level of 0.95, while $\Theta_{opt|p=\tilde{p}}$ corresponds to the maximum-likelihood value for the other non-fixed parameters when $p = \tilde{p}$.

Code availability

The BD-PN and BD parameter estimators are implemented in Python 3. They use ETE 3 framework for tree manipulation [25] and NumPy package for array operations [26].

They are available as a command-line program and a Python 3 package via PyPi (bdpn), and via Docker/Singularity (evolbioinfo/bdpn). The source code and the installation and usage documentation are available on GitHub at github.com/evolbioinfo/bdpn.

Zurich men-having-sex-with-men HIV-1 B tree and sampling probability

We applied our estimator to asses the partner notification for HIV-1 subtype B epidemic among men-having-sex-with-men (MSM) in Zurich. We used the phylogenetic tree of 200 samples reconstructed by Rasmussen *et al.* [27] from the sequences collected as a part of the Swiss Cohort Study [28] between 1988 and 2014. This tree was also previously analyzed by Voznica *et al.* [20] using a Birth-Death model with SuperSpreading (BDSS). Following Voznica *et al.* [20], we fixed the sampling probability for this tree at 25%: $\rho = 0.25$.

UK HIV-1 B forest reconstruction and sampling probability estimation

To asses the partner notification in HIV-infected patients in the UK, we used the phylogenetic tree from our recent study of HIV drug resistance in the UK [29]. The tree

tips represent viruses sampled from 40 055 individuals between 1996 and 2016. The samples used for the tree reconstruction were obtained from the UK HIV Drug resistance database [30]. The sampling date metadata for these samples were expressed in months (e.g., Aug 2015), hence potentially creating simultaneous sampling dates for cases that were sampled on different days of the month. To avoid this issue we randomly selected a date within the specified month for each sample, and time-scaled the tree with LSD2 [31] (v2.4.1, strict molecular clock with outlier removal). We repeated random date resolution and time-scaling 10 times, obtaining 10 trees with slightly different branch lengths and root dates between Oct 5 and Nov 11, 1962.

Between 2012 and 2015, in the UK antiretroviral treatment (ART) of HIV-infected individuals was initiated when their CD4 count dropped below 350 cells/mL [32]. However "if a patient with a CD4 cell count > 350 cells/mL wishes to start ART to reduce the risk of transmission to partners, this decision is respected and ART is started" [32]. Before 2012, treatment started with an even lower CD4 (i.e., later). The British HIV Association 2015 guidelines recommend all individuals with suspected or diagnosed primary HIV infection "are offered immediate ART" [33]. To have a more homogeneous setting in terms of access to ART, awareness of the HIV infection, etc., we cut the dated trees between 2012 and 2015.

To estimate the BD-PN parameter values, we needed to fix one of the parameters for model identifiability. We therefore estimated the base sampling probability ρ as follows. According to the "Towards elimination of HIV transmission, AIDS and HIV-related deaths in the UK" report, the estimated total number of people living with HIV in the UK in 2015 was 101 200. According to our estimate [29], 66.5% of the HIV-infected individuals in the UK were infected with HIV-B. We hence estimated that in 2015 67 298 people were living with HIV-B. Our trees contained 39 047 tips sampled by the end of 2015, and therefore account for about 58% ($= 39\,047/67\,298 \cdot 100\%$) of the total number of individuals living with HIV-B in the UK. We hence fixed the sampling probability ρ to 0.58.

Results

PN test is highly specific and sensitive at detecting partner notification

To illustrate the PN test performance, we applied it to the six 100-tree simulated datasets, corresponding to the three classical MTBD-family models: BD, BDEI (Birth-Death Exposed-Infectious [13]) and BDSS (Birth-Death with Super Spreading [12]), and to their -PN versions: BD-PN, BDEI-PN and BDSS-PN (see "MTBD-PN tree simulator and test datasets" for dataset details).

For 100/98/98 out of 100 trees in the BD/BDEI/BDSS dataset the PN test p-value was above 0.05 (i.e., 100%/98%/98% true negative and 0%/2%/2% false positive results). For 95/93/94 out of 100 trees in the BD-PN/BDEI-PN/BDSS-PN dataset the PN test p-value was below 0.05 (i.e., 95%/93%/94% true positive and 5%/7%/6% false negative results). These results are summarised in Table 1. Our test therefore showed both high specificity and sensitivity (99% and 94% respectively on the combined, 600-tree, data set).

Accounting for partner notification is crucial for accurate parameter estimation

We assessed the performance of our maximum-likelihood estimator for the BD-PN model bdpn on the 100 trees from the the BD-PN simulated data set (see "MTBD-PN tree simulator and test datasets" for dataset details).

Table 1. PN test performance on trees generated under BD, BDEI, BDSS, BD-PN, BDEI-PN and BDSS-PN models (100 trees per model). We report the number of trees for which the PN-test p-value was < 0.05, ≥ 0.05, and the mean p-value for each model dataset.

	p-value < 0.05	**p-value** ≥ 0.05	**mean p-value**
BD	0	100	0.538
BD-PN	95	5	0.007
BDEI	2	98	0.539
BDEI-PN	93	7	0.027
BDSS	2	98	0.515
BDSS-PN	94	6	0.022

We applied bdpn to each of the trees fixing the sampling probability ρ to its true value (for identifiability). As expected, the tree likelihoods for the estimated parameter values were higher than or equal to the tree likelihoods for the true parameter values for all the trees in both settings. This shows the consistent behavior of our likelihood calculation and optimization.

To highlight the importance of accounting for partner notification with the model, we also applied the classical BD model estimator bd to these trees. Additionally, we applied both bdpn and bd estimators to the dataset simulated under the BD model. The results are shown in Fig. 3. We compared the method accuracy by a paired z-test. **On the BD data set**, bd and bdpn estimators performed equally well in estimating R_0 and infectious time $\frac{1}{\psi}$, with an average relative error $\leq 5\%$, and average relative bias $\leq 1\%$. The partner notification probability υ estimated on these trees by bdpn was systematically low, with the maximum value of 0.015, and mean value of 0.001. Moreover, its true value (zero) was within the CIs for all the trees. **On the BD-PN data set**, bdpn accuracy was significantly better (p-value ≤ 0.01), with errors and biases $\leq 11\%$ for all the parameters (vs. $12 - 70\%$ for bd). bd estimator tends to significantly overestimate the removal rate (average relative bias of 70%), which could be explained by the fact that the BD model has only one removal rate to accommodate a non-homogeneous population containing both standard cases and notified partners with a higher removal rate ϕ. However, bd also systematically overestimates the transmission rate (average relative bias of 11%), while this parameter is homogeneous in the population.

The percentage of the trees for which true parameter values were within the CI estimates and relative CI widths ($\frac{CI_{97.5\%} - CI_{2.5\%}}{true\ parameter\ value}$) are shown in Table 2. For the CIs as for the point estimates, bdpn and bd estimators performed equally well on the BD trees, while on the BD-PN trees the bd estimator performed significantly worse (only 44% and 16% of CIs included respectively the true transmission rate λ and removal rate ψ values versus 94% and 81% for bdpn).

Finally, we performed a likelihood ratio test on the BD and BD-PN trees with the parameters estimated by the bd and bdpn estimators: $-2(l_{BD} - l_{BDPN}) \lessgtr 5.99$ (where 5.99 is the value of chi-squared distribution with 2 degrees of freedom corresponding to the significance level of 0.95). For all the 200 trees the test selected the correct model.

Application 1: Men-having-sex-with-men (MSM) HIV-1 B epidemic in Zurich

We applied our estimator to asses the partner notification for HIV-1 subtype B epidemic among men-having-sex-with-men (MSM) in Zurich, using a 200-sample time-scaled phylogenetic tree from the study by Rasmussen *et al.* [27], and fixing the sampling

Table 2. Percentage of simulated BD-PN and BD trees for which the true parameter values were withing the bdpn- and bd-estimated 95% CIs, and the median CI width: $\frac{CI_{97.5\%}-CI_{2.5\%}}{true\ parameter\ value}$. (The sampling probability ρ was fixed to its true value in all the settings.)

		bdpn estimator		**bd estimator**	
	parameter	% within CI	median CI width	% within CI	median CI width
BD-PN model	λ	94%	11.4%	44%	12.6%
	ψ	81%	25.6%	16%	29.4%
	ϕ	93%	35.0%	–	–
	υ	71%	24.5%	–	–
BD model	λ	97%	10.3%	97%	10.3%
	ψ	95%	23.2%	95%	23.1%
	$\upsilon(=0)$	100%	0.004*	–	–

*absolute width ($CI_{97.5\%} - CI_{2.5\%}$) is given instead of the relative one, as the true value is zero.

probability to 25%: $\rho = 0.25$ (see "Zurich men-having-sex-with-men HIV-1 B tree and sampling probability" for more details).

The PN test detected partner notification in this tree (p-value = 0.014). Our bdpn estimator inferred the partner notification probability $\upsilon = 0.09$ 95%-CI $[0.008 - 0.226]$, $R_e = 1.43$ $[1.26 - 1.66]$, infectious time $\frac{1}{\psi} = 8$ $[6 - 11]$ years, and 4.9 months [6 days – 1.4 years] between partner notification and their sampling. These results globally agree (CIs intersect) with the R_e estimate for this data using a coalescent model from Rasmussen *et al.* [27] (between 1.0 and 2.5), and the estimates using the BDSS model from Voznica *et al.* [20]: R_e of 1.6 and 1.7 and an infectious period of 10.2 and 9.8 years (depending on the tree representation used for the deep learner).

Note that this tree covers a long non-homogeneous time interval (between 1974, the root of the tree, and 2012), including times with no or limited access to treatment, when the infectious time corresponded to progression to AIDS (≈ 10 years), as well as later times, when antiretroviral treatment (ART) became available, and hence the infectious period would correspond to the time till ART start, as, when taken properly, it prevents transmission. Our estimate (8 years) represent a mix of these scenarios. This also holds for the other estimated parameters, whose values represent a somewhat averaged estimate over the time period. As this data set was already small, we did not cut it to obtain a more homogeneous treatment policy (as we did in Application 2).

For comparison, parameter estimation on the same tree with the BD model gave a slightly lower estimate of R_e and a slightly shorter infectious time, though with intersecting CIs ($R_e = 1.32$ $[1.19 - 1.47]$, $\frac{1}{\psi} = 7$ $[6 - 9]$ years), which is in line with underestimation of R_e and $\frac{1}{\psi}$ with the bd estimator on the BD-PN simulated data. The likelihood ratio test suggested the BD-PN model: loglikelihood under the BD-PN-estimated parameters was $l_{BDPN} = -1\,140$ vs. $l_{BD} = -1\,146$ under the BD parameters, which gives the ratio of $-2(l_{BD} - l_{BDPN}) = 12 > 5.99 \approx \chi^2_2(0.95)$.

Application 2: HIV-1 B epidemic in the UK

We also applied our estimator to asses the partner notification in HIV-1 subtype B infected individuals in the UK between 2012 and 2015. This time period was chosen to represent a homogeneous treatment policy (treatment start at CD4 count below 350 cells/mL [32]). We used 10 forests reconstructed from HIV-1 B sequences from the UK HIV Drug resistance database [30] (see "UK HIV-1 B forest reconstruction and sampling probability estimation" for details).

The PN-test values were below 0.05 for all the forests (p-values between $1 \cdot 10^{-18}$ and $2 \cdot 10^{-14}$), strongly suggesting the presence of partner notification. Using sampling probability $\rho = 0.58$ (see "UK HIV-1 B forest reconstruction and sampling probability estimation" for its estimation), we estimated the partner notification probability $\upsilon = 0.006 - 0.008$, $R_e = 1.2 - 1.3$, infectious time $\frac{1}{\psi} = 2.3 - 2.4$ years, and 7–11 days between partner notification and their sampling. The results were consistent between 10 forests and are shown in Table 3.

In the UK in 2015 the majority of infected population (72% [34]) was on antiretroviral treatment, which, when taken properly, prevents transmission. Hence, the average infectious time roughly represents the average time before starting treatment. Our estimate is approximately 2.3 years. According to the HIV report from the Public Health England [19], in England in 2013, less than 3% of sampled infected individuals were detected though partner notification. Our estimate is even lower, potentially because our model does not allow for notifying partners with whom there was no transmission, nor notifying multiple partners.

Table 3. BD-PN model parameters estimated for UK HIV-1 B epidemic between 2012 and 2015, assuming that our data represents 58% of infected cases ($\rho = 0.58$).

forest	num. tips	num. trees	R_e	infectious time $\frac{1}{\psi}$ [years]	partner removal t. $\frac{1}{\phi}$ [days]	υ
1	11 117	8 838	1.24 [1.22 – 1.27]	2.34 [2.28 – 2.40]	8 [5 – 12]	0.008 [0.005 – 0.012]
2	11 109	8 833	1.24 [1.21 – 1.27]	2.33 [2.28 – 2.40]	7 [4 – 11]	0.007 [0.003 – 0.011]
3	11 122	8 844	1.24 [1.22 – 1.27]	2.34 [2.28 – 2.39]	8 [5 – 11]	0.008 [0.005 – 0.012]
4	11 119	8 838	1.30 [1.26 – 1.32]	2.40 [2.27 – 2.43]	11 [3 – 13]	0.006 [0.002 – 0.016]
5	11 116	8 829	1.24 [1.22 – 1.27]	2.34 [2.28 – 2.39]	9 [6 – 13]	0.007 [0.004 – 0.011]
6	11 111	8 827	1.25 [1.22 – 1.27]	2.34 [2.28 – 2.40]	8 [5 – 12]	0.007 [0.004 – 0.011]
7	11 119	8 838	1.24 [1.21 – 1.27]	2.33 [2.28 – 2.39]	8 [5 – 12]	0.007 [0.004 – 0.011]
8	11 111	8 830	1.24 [1.21 – 1.27]	2.33 [2.28 – 2.40]	7 [4 – 11]	0.007 [0.003 – 0.011]
9	11 120	8 833	1.24 [1.21 – 1.27]	2.33 [2.28 – 2.39]	8 [4 – 12]	0.007 [0.003 – 0.011]
10	11 112	8 827	1.25 [1.22 – 1.28]	2.34 [2.28 – 2.40]	8 [5 – 11]	0.008 [0.004 – 0.012]

For comparison, parameter estimation on the same forests with the BD model gave similar results ($R_e = 1.2$, $\frac{1}{\psi} = 2.3$ years), however the likelihood ratio test suggests the BD-PN model. For instance, on forest 1, loglikelihood under the BD-PN-estimated parameters was $l_{BDPN} = -28\,984$ vs. $l_{BD} = -29\,002$ under the BD parameters, which gives the ratio $-2(l_{BD} - l_{BDPN}) = 36 > 5.99 \approx \chi^2_2(0.95)$.

Overall the estimates on the much smaller Zurich dataset show larger CIs than those on the larger UK one. The CIs estimated on the two data sets intersect for the effective reproductive number R_e, while the infectious time is larger for Zurich (CI of $6 - 10$ years vs $2.2 - 2.4$ years for the UK). This is however expected as the Zurich tree covers a longer, earlier and non-homogeneous time interval (between 1974 and 2012), including times with no or limited access to treatment, when the infectious time corresponded to progression to AIDS (≈ 10 years). The CIs for notification probability υ and partner removal time $\frac{1}{\phi}$ also intersect on the two data sets, though they do not have to, as the notification policy might differ between the two countries, and especially during the long and non-homogeneous time interval represented by the Zurich tree. The point estimates suggest stronger partner notification probability and longer notified partner removal time in Zurich than in the UK.

Discussion

We proposed an extension of the phylodynamic birth-death model that accounts for non-random sampling due to partner notification (BD-PN). In real-life epidemics, detected infected individuals might notify the people whom they think they might have infected or got infected by, and who in turn might get tested and detected much faster than via the standard procedure. Our model accounts for this scenario.

The -PN extension adds two constant parameters to the classical BD model: the probability of partner notification υ and the partner detection rate ϕ. It can also be applied to a more general, multi-type birth-death (MTBD) model framework, which accounts for heterogeneity in the infected population.

We developed a PN-test allowing to distinguish between trees generated under models with and without partner notification. Its application to simulated data showed high sensitivity and specificity.

We developed a mathematical framework for calculation of the likelihood density of a transmission tree (approximated by a time-scaled phylogenetic tree) under the BD-PN model, and implemented a maximum-likelihood BD-PN parameter estimator bdpn based on this framework. Tests on simulated data show that our estimator robustly infers model parameter values and their CIs in the setting when the base sampling probability ρ is fixed. Fixing one of the model parameters is needed for model identifiability and in practice, ρ may often be approximated from epidemiological data, e.g., as the proportion of sampled cases among the declared ones.

Importantly, our simulations show that in the presence of partner notification, the classical BD model estimator fails to accurately estimate epidemiological parameters (average relative errors $> 25\%$ for R_0 and infectious time $\frac{1}{\psi}$). On the contrary, in the absence of partner notification, the BD-PN model estimator correctly identifies R_0 and infectious time $\frac{1}{\psi}$ (as well as the BD model estimator), and estimates a very low partner notification probability υ, whose confidence intervals include zero.

We applied the PN-test and our estimator to the HIV-1 B MSM cluster in Zurich and to the HIV-1 B epidemic in the UK between 2012 and 2015. The test detected presence of partner notification in both cases. The obtained estimates of epidemiological parameters are in agreement with what we now know about the HIV-1 B epidemic (e.g., R_e between 1.2 and 1.7). We estimated a modest partner notification probability (9% in Zurich, 1% in the UK), and rapid notified partner detection (≈ 5 months vs ≈ 8 years for detection under the standard procedure in Zurich; and ≈ 8 days vs ≈ 2.3 years for detection under the standard procedure in the UK). Note that the notification probability represents "successful notification", which combines several conditions: the index patient remembering and agreeing to notify the partner, the partner being reached, agreeing to take a test, being indeed infected, and moreover the transmission happening during the contact between the index and the partner. This explains the seemingly low values.

The BD-PN model is the first step towards accounting for non-random sampling in phylodynamic models. In order to reduce its mathematical complexity (having closed-form solutions for its differential equations) and make parameter estimation fast, we combined several conditions of the complex partner notification process into one "successful notification" probability υ (see above), and made several additional assumptions: (i) only sampled individuals can notify their partners; (ii) only the most recent partner can get notified; (iii) once notified, partners are always sampled upon removal (however, potentially after the end of the considered sampling period). These assumptions may be too simple to fully grasp the real partner notification process and should be relaxed in the future work. Another needed extension should account for population heterogeneity (i.e., MTBD-PN). As it is much easier to implement a tree

simulator with a complex model (e.g., as we did for MTBD-PN models) than to derive its likelihood, a promising path to take for future -PN model developments would be using deep learning (e.g., adding new models to PhyloDeep [20]).

Finally, what we presented here is a specific case of an epidemiological model. However, it might be useful in other contexts as well. For instance, the State Speciation and Extinction (SSE) family of models [8, 35, 36] in the macroevolution domain are mathematically very similar to the epidemiological MTBD models [37]. In macroevolution, a detection of a new species might lead to faster detection of closely related ones. This process could be modeled in the same way as partner notification.

Supporting information

S1 Appendix MTBD-PN tree simulator. The appendix describes the tree simulator that we developed for generating sampled transmission trees under MTBD and MTBD-PN models.

MTBD-PN tree simulator

Our tree simulator generates sampled transmission trees under MTBD and MTBD-PN models (with m states). It is Gillespie-based, generates state change, transmission and removal times, and only reconstructs the sampled parts of the tree to save memory and increase speed.

The simulator iterates through occurring events till either time or sampled tip number limit is reached. It keeps updating an array $C = [c_1, \ldots, c_m]$ of counts of currently infected individuals (i.e., non-removed) in different states; an array $S = [s_1, \ldots, s_m]$ of counts of sampled individuals in different states; as well as a mapping M between states and ids of currently infected individuals in those states, and a mapping N between states and ids of sampled individuals in those states. In the beginning ($t = 0$) the only infected individual corresponds to the root: $c_k = 0\ \forall k \neq state(root),\ c_{state(root)} = 1;\ M_k = \emptyset\ \forall k \neq state(root)$, $M_{state(root)} = \{root\}$, and no individual is sampled: $s_k = 0,\ N_k = \emptyset\ \forall 1 \leq k \leq m$.

At each iteration the algorithm (1) calculates the time of the next event; (2) chooses the type of this event and the the individual involved in it; (3) updates the counts and the mappings according to the event and records its time.

At step 1, to calculate the time of the next event, we (i) calculate the total rate r as a sum of total state change, transmission and transition rates: $r = r^{(\mu)} + r^{(\lambda)} + r^{(\psi)}$, where $r^{(\mu)} = \sum_{k=1}^{m} c_k \sum_{r=1}^{m} \mu_{kr}$, $r^{(\lambda)} = \sum_{k=1}^{m} c_k \sum_{r=1}^{m} \lambda_{kr}$, and $r^{(\psi)} = \sum_{k=1}^{m} c_k \psi_k$; (ii) draw Δt from the exponential distribution with rate r; (iii) update the current time to $t + \Delta t$.

At step 2, to chose the type of the event, we draw a value w (uniformly) from an interval $[0, r[$. This interval can be seen as composed of subintervals corresponding to each possible event, where the width of each subinterval is defined by the corresponding rate and infected individual count, e.g., the subinterval corresponding to a transmission from k to l has a width of $\lambda_{kl} c_k$. The subinterval in which w is located defines the event type.

At step 3, we randomly draw an individual i of type k (selected at step 2) from $M[k]$, and proceed depending on the event type. If it is a state-change from k to l, then we decrease c_k by one, increase c_l by one, remove i from $M[k]$ and put i into $M[l]$. If it is a transmission from k to l, then we increase c_l by one, create a new id j for the recipient and put j into $M[l]$. If it is a removal, then we decrease c_k by one, remove i from $M[k]$, and draw a value p (uniformly) from an interval $[0, 1[$: if $p < \rho$, the individual gets sampled and we put i into $N[k]$.

For transmission and sampling events we also record their time and ids of the involved individuals. These values are used to reconstruct the sampled parts of the tree once the simulation is finished.

For the PN version of this simulator, we add additional m states for notified-partner versions of each state. $m+k$'s state-change rates are set to the k's state-change rates : $\mu_{m+k,m+l} = \mu_{k,l}\ \forall 1 \leq k, l \leq m$, while the state-change rates between initial and notified states are set to zero: $\mu_{m+k,l} = \mu_{k,m+l} = 0$. Only the transmissions to non-notified states are allowed (and the ones from notified states are set to the corresponding non-notified rates): $\lambda_{m+k,l} = \lambda_{k,l}$; $\lambda_{m+k,l+k} = \lambda_{k,m+l} = 0\ \forall 1 \leq k, l \leq m$. The removal rates of notified states are set to ϕ: $\psi_{m+k} = \phi >> \psi_k$, while their sampling probabilities are set to 1: $\rho_{m+k} = 1\ \forall 1 \leq k \leq m$. We keep a mapping of most recent partners P for each infectious individual, and update it for both the donor and the recipient at each transmission event. We also keep a mapping between each infectious individual and their state, and update it at each state-change event. At each removal event, if the removed individual i gets sampled upon removal, we draw a value p (uniformly) from an interval $[0, 1[$. If $p < \upsilon$, and if i's most recent partner ($j = P[i]$) is not yet sampled (not in $N[state(j)]$) and not yet notified ($j \leq m$), we change j's state to $m + j$ as in a state-change event.

Code availability

The simulator is implemented in Python 3. It uses ETE 3 framework for tree manipulation [25] and NumPy package for array operations [26].

It is available as a command-line program and a Python 3 package via PyPi (treesimulator), and via Docker/Singularity (evolbioinfo/treesimulator). Its source code and the installation and usage documentation are available on GitHub at github.com/evolbioinfo/treesimulator.

S2 Appendix. Calculation of a probability of a branch containing a hidden partner. The appendix describes the approximation of the probability of having a branch in the tree that contains one or two hidden partners.

Mixed branch probability

To model the cases when a partner B is not sampled, we need to calculate the probability of a **hidden partner subtree** $U_p(t, t_A)$, which started at time t on a partner branch and evolved till time T without any of its individuals being sampled, provided that the first of the partner's notifiers tried to notify them at time t_A (B's branch in Fig. 2(2a,c,3a,c)). This probability consists of two parts: either the partner got notified at time t_A but has not got sampled by the end of the sampling period T (like in Fig. 2(2a,3a)), $U_p^{(1)}(t, t_A)$; or the partner got removed via the standard procedure before the time t_A (like in Fig. 2(2c,3c)), $U_p^{(2)}(t, t_A)$:

$$U_p(t, t_A) = U_p^{(1)}(t, t_A) + U_p^{(2)}(t, t_A). \tag{11}$$

The first part can be easily expressed via the previously defined equations: partner branch evolution before notification till time t_A (if $t_A > t$) followed by partner branch evolution after notification till time T. Note, that neither $p_{pb}^{(x)}(t, t_A)$ not $p_{pa}^{(x)}(t_A)$ include an event at the end of the branch, and in this case the partner branch does not end at $t_x = T$:

$$U_p^{(1)}(t, t_A) = \begin{cases} p_{pb}^{(A)}(t) p_{pa}^{(x)}(t_A) & \textit{if } t_A > t \\ p_{pa}^{(x)}(t) & \textit{if } t_A \leq t \end{cases}, \textit{ where } t_x = T \tag{12}$$

The second scenario is only possible if $t_A > t$. To calculate it, we would need to integrate over all possible partner removal times $t_B \in [t, t_A[$, provided that in the interval between t and t_B the partner might have transmitted to someone else (potentially multiple times), whose subtree(s) stayed unobserved till T. Note that $p_{pb}^{(A)}(t)$ describes a partner branch evolution between t and t_A with any number of hidden transmissions, no removal along it, and no event at the end of the branch. What we need instead is a removal with no sampling at time t_B. To approximate $U_p^{(2)}(t, t_A)$ we will divide $p_{pb}^{(A)}(t)$ by the probability of no removal during the time $(t_A - t)$ (i.e., $e^{-\psi(t_A - t)}$), multiply it by the probability of removal during that time $(1 - e^{-\psi(t_A - t)})$, and by the probability of not being sampled upon removal $(1 - \rho)$:

$$U_p^{(2)}(t, t_A) \approx \begin{cases} p_{pb}^{(A)}(t) \frac{1 - e^{-\psi(t_A - t)}}{e^{-\psi(t_A - t)}} (1 - \rho) & \textit{if } t_A > t \\ 0 & \textit{if } t_A \leq t \end{cases} \tag{13}$$

We now describe the probability of a **mixed branch** $p_{m(s1,s2)}^{(i)}(t, t_r)$ for branches that include a hidden partner subtree (AB-C branches in Fig. 2(2) and A's tip branches in Fig. 2(3)). $s1$ here denotes the type of the top of the branch (e.g., between AB and BC in Fig. 2(2), partner) and $s2$ denotes the type of the bottom of the branch (e.g., between AB and A in Fig. 2(3), notifier). Let us denote the first notification time of the hidden partner as t_r, and the time of the hidden partner subtree start as t_h, while as usual t_i stands for the time at the branch end. For instance, in Fig. 2(2a) $t_i = t_C$, $t_r = t_A$, $t_h = t_{BC}$; in Fig. 2(2b) $t_i = t_r = t_C$, $t_h = t_{BC}$; in Fig. 2(3c) $t_i = t_r = t_A$, $t_h = t_{AB}$.

$p_{m(s1,s2)}^{(i)}(t, t_r)$ combines two elements: the probability of the top part of the branch $p_{top=s1}^{(h)}(t, t_r)$, which finishes with a transmission to or from the hidden partner at time t_h and the probability of the bottom part of the branch (without taking into account the event at its end) $p_{bottom=s2}^{(i)}(t_h)$.

Note that since the branch contains a hidden partner, this partner must have at least one notifier (who is observed as all the notifiers). The notifier could correspond to a tip in the branch's supertree: For example, tip A who is part of the C's supertree in Fig. 2(2a,c). If the branch is external, the notifier could also correspond to its own tip: For example, such notifiers correspond to tip A in Fig. 2(2b) and tip C in Fig. 2(3b). Having a notifier as the branch's tip implies that the bottom part of the branch corresponds to the notifier. Having a notifier in the supertree implies that the top part of the branch corresponds to the partner (already notified or not yet). Hence if the branch is internal, the top part can only correspond to the partner, while the bottom part can only correspond to a standard state (unnotified non-notifier), as the bottom part of such a branch corresponds to the partner's recipient (who could not be notified as the partner is unobserved and could not be a notifier as the branch is not external). Finally, if the branch is external, and the only partner's notifier corresponds to its tip, then there are two possibilities for its top part: (1a) either it fully corresponds to a standard state (unnotified non-notifier); or (1b) there is another partner's notifier in its supertree, and this other partner is also hidden along the top part of the same branch, hence the top part of the branch is partner-standard (see Fig. 4). If, on the other hand, the branch is external, and the partner's notifier(s) is(are) only found in its supertree, then there are two possibilities for its bottom part: (2a) either it fully corresponds to a standard state (unnotified non-notifier); or (2b) there is another partner's notifier at its tip, and this other partner is also hidden along the bottom part of the same branch, hence the bottom part of the branch is standard-notifier (see Fig. 4). Note that (1b) and (2b) represent the same case but from the point of view of two different hidden partners (B and their notifier A for the former, and D and their notifier C for the latter in Fig. 4).

We will first describe the one-hidden-partner case $p^{(i)}_{m(s1,s2)}(t,t_r)$ (where $s1 \in \{p,-\}$, $s2 \in \{n,-\}$ and $-$ stands for the standard (unnotified non-notifier) state), and then proceed with the two-hidden-partners case $p^{(i)}_{m(p,-,n)}(t,t_r)$, where t_r is the earliest notification time for the partner hidden at the top of the branch (e.g., $t_r = t_C$ in Fig. 4), the only notification time for the partner hidden at the bottom of the branch being t_i (e.g., $t_i = t_A$ in Fig. 4).

For the bottom branch part:

$$p^{(i)}_{bottom=s2}(t_h) = \begin{cases} p^{(i)}_n(t_h) & \textit{if } i \textit{ is a notifier tip } (s2 = n), \\ p^{(i)}(t_h) & \textit{otherwise } (s2 = -, \textit{ the standard state}) \end{cases} \tag{14}$$

We will approximate $p^{(h)}_{top=s1}(t)$ using the branch evolution probability of the type corresponding to $s1$ in the following way. For example, let us assume that $s1 = -$ (standard branch). $p^{(h)}(t)$ expresses a probability of evolving along this branch between the times t and t_h with no or any number of hidden transmissions, without taking into account the event at the end of it. In our case there must be at least one hidden transmission: to/from the partner at time t_h (where the partner tree stayed unsampled). If we remove the probability of no event during this time ($e^{-(\lambda+\psi)(t_h-t)}$) from $p^{(h)}(t)$, we obtain a probability of a branch with at least one hidden transmission somewhere between t and t_h (the probability of the corresponding hidden tree being included). $p^{(h)}_{top=-}(t)$ expresses almost the same thing, with the difference being that the last hidden transmission must have happened at time t_h and correspond to the hidden partner. We will hence approximate $p^{(h)}_{top=-}(t)$ as follows:

$$p^{(h)}_{top=-}(t,t_r) \approx p^{(h)}(t) - e^{-(\lambda+\psi)(t_h-t)} \tag{15}$$

We will treat the case when the branch is in the partner state ($s1 = p$) in a similar way:

$$p^{(h)}_{top=p}(t,t_r) \approx \begin{cases} p^{(h)}_{pb}(t) - e^{-(\lambda+\psi)(t_h-t)} & \textit{if } t_h \le t_r \\ p^{(r)}_{pb}(t)\Big(p^{(h)}_{pa}(t_r) - e^{-(\lambda+\phi)(t_h-t_r)}\Big) & \textit{if } t \le t_r \le t_h \\ p^{(h)}_{pa}(t) - e^{-(\lambda+\phi)(t_h-t)} & \textit{if } t > t_r \end{cases} \tag{16}$$

$$\textit{where } t_r \textit{ is the first notification time of the hidden partner.}$$

Finally, we will account for the fact that the hidden tree starting at time t_h corresponds to the hidden partner by dividing the expression by $U(t_h)$ (a standard hidden tree) and multiplying by $U_p(t_h,t_r)$ (partner hidden tree).

Putting everything together, and approximating the time of the start of the hidden partner tree t_h with the middle of the branch, we obtain:

$$\begin{aligned} p^{(i)}_{m(s1,s2)}(t,t_r) &= p^{(h)}_{top=s1}(t,t_r)\frac{U_p(t_h,t_r)}{U(t_h)}p^{(i)}_{bottom=s2}(t_h), \\ &\textit{where } t_h \approx t + \frac{(t_i - t)}{2} \textit{ is an approximation of the time} \\ &\qquad \textit{of the hidden partner subtree start,} \\ &\textit{where } t_r \textit{ is the earliest notification time of the hidden partner.} \end{aligned} \tag{17}$$

Finally, we approximate the mixed case with two hidden partners using similar

principals:

$$
\begin{aligned}
p^{(i)}_{m(p,-,n)}(t,t_r) \approx p^{(h1)}_{top=p}(t,t_r)\frac{U_p(t_{h1},t_r)}{U(t_{h1})}p^{(h2)}_{top=-}(t_{h1},t_i)\frac{U_p(t_{h2},t_i)}{U(t_{h2})}p^{(i)}_n(t_{h2}),\\
\textit{where } t_{h1} \approx t + \frac{1}{3}(t_i - t),\\
t_{h2} \approx t + \frac{2}{3}(t_i - t),\\
t_r \textit{ is the earliest notification time of the partner}\\
\textit{hidden at the top of the branch.}
\end{aligned}
\tag{18}
$$

S3 Appendix. Tree likelihood calculation. The appendix describes tree likelihood calculation case by case, accounting for all possible notifier-partner configurations.

Tree likelihood calculation under BD-PN model

Case 1: i is a tip

For a tip i with a parent node j the subtree likelihood density values can be calculated as:

$$
\begin{aligned}
l^{(\not{n},\not{n})}(i) \quad &= p^{(i)}(t_j)\psi\rho(1-\upsilon) \textit{ i.e., i was not notified and did not notify}\\
&\textit{(standard branch evolution followed by sampling and no notification)}\\
&\quad + p^{(i)}_{m(-,n)}(t_j,t_i)\psi\rho\upsilon \textit{ i.e., i was not notified but notified their partner who is hidden}\\
&\textit{(mixed standard-notifier branch evolution followed by sampling and notification)}
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
l^{(\not{n},n)}(i) \quad &= p^{(i)}_n(t_j)\psi\rho\upsilon \textit{ i.e., i was not notified but notified their most recent patner (observed)}\\
&\textit{(notifier branch evolution followed by sampling and notification)}
\end{aligned}
\tag{20}
$$

$$
l^{(n,\not{n})}(i,r) \quad = \begin{cases}
p_{pa}^{(i)}(t_j)\phi(1-\upsilon) \ \textit{if i got notified at time } t_r < t_j \textit{ and did not notify} \\
\textit{(notified partner branch evolution then notification-provoked sampling and no notification)} \\
\quad + p_{m(pa,-)}^{(i)}(t_j,t_r)\psi\rho(1-\upsilon) \ \textit{i.e., branch i contains a hidden partner notified by r} \\
\qquad\qquad\qquad\qquad \textit{at } t_r < t_j\textit{, i did not notify} \\
\quad \textit{(mixed partner-standard branch evolution followed by sampling and no notification)} \\
\quad + p_{m(pa,n)}^{(i)}(t_j,t_r)\psi\rho\upsilon \ \textit{i.e., branch i contains a hidden partner notified both by r and i} \\
\quad \textit{(mixed partner-notifier branch evolution followed by sampling and notification)} \\
\quad + p_{m(pa,-,n)}^{(i)}(t_j,t_r)\psi\rho\upsilon \ \textit{i.e., branch i contains a hidden partner notified by r} \\
\qquad\qquad\qquad\qquad \textit{at } t_r < t_j\textit{, i notified another hidden partner} \\
\quad \textit{(mixed partner-standard-notifier branch evolution then sampling and notification)} \\
\\
p_{pb}^{(i)}(t_j)\psi\rho(1-\upsilon) \ \textit{if i got notified at time } t_r > t_i \textit{ and did not notify} \\
\textit{(partner-before-notification branch evolution then standard sampling and no notification)} \\
\quad + p_{m(pb,-)}^{(i)}(t_j,t_r)\psi\rho(1-\upsilon) \ \textit{i.e., branch i contains a hidden partner notified by r} \\
\qquad\qquad\qquad\qquad \textit{at } t_r > t_i\textit{, i did not notify} \\
\quad \textit{(mixed partner-standard branch evolution followed by sampling and no notification)} \\
\quad + p_{m(pb,n)}^{(i)}(t_j,t_i)\psi\rho\upsilon \ \textit{i.e., branch i contains a hidden partner notified both by r and i} \\
\quad \textit{(mixed partner-notifier branch evolution followed by sampling and notification)} \\
\quad + p_{m(pb,-,n)}^{(i)}(t_j,t_r)\psi\rho\upsilon \ \textit{i.e., branch i contains a hidden partner notified by r at } t_r > t_i\textit{,} \\
\qquad\qquad\qquad\qquad \textit{i notified another hidden partner} \\
\quad \textit{(mixed partner-standard-notifier branch evolution then sampling and notification)} \\
\\
p_{pb}^{(r)}(t_j)p_{pa}^{(i)}(t_r)\phi(1-\upsilon) \ \textit{if i got notified at time } t_j \le t_r \le t_i \textit{ and did not notify} \\
\textit{(partner-before-notification branch evolution followed by} \\
\textit{notified partner branch evolution, notification-provoked sampling and no notification)} \\
\quad + p_{m(p,-)}^{(i)}(t_j,t_r)\psi\rho(1-\upsilon) \ \textit{i.e., branch i contains a hidden partner notified by r} \\
\qquad\qquad\qquad\qquad \textit{at } t_j \le t_r \le t_i\textit{, i did not notify} \\
\quad \textit{(mixed partner-standard branch evolution followed by sampling and no notification)} \\
\quad + p_{m(p,n)}^{(i)}(t_j,t_r)\psi\rho\upsilon \ \textit{i.e., branch i contains a hidden partner notified both by r and i} \\
\quad \textit{(mixed partner-notifier branch evolution followed by sampling and notification)} \\
\quad + p_{m(p,-,n)}^{(i)}(t_j,t_r)\psi\rho\upsilon \ \textit{i.e., branch i contains a hidden partner notified by r} \\
\qquad\qquad\qquad\qquad \textit{at } t_j \le t_r \le t_i\textit{, i notified another hidden partner} \\
\quad \textit{(mixed partner-standard-notifier branch evolution then sampling and notification)}
\end{cases} \tag{21}
$$

$$
l^{(n,n)}(i,r) \quad = \begin{cases}
p_{b}^{(i)}(t_j)\phi\upsilon \ \textit{if i got notified at time } t_r < t_j \textit{ and notified their most recent patner} \\
\textit{(notified notifier branch evolution then notification-provoked sampling and notification)} \\
\\
p_{n}^{(i)}(t_j)\psi\rho\upsilon \ \textit{if i got notified at time } t_r > t_i \textit{ and notified their most recent patner} \\
\textit{(notifier branch evolution followed by standard sampling and notification)} \\
\\
p_{n}^{(r)}(t_j)p_{b}^{(i)}(t_r)\phi\upsilon \ \textit{if i got notified at time } t_j \le t_r \le t_i \textit{ and notified} \\
\textit{(notifier branch evolution followed by notified notifier branch evolution,} \\
\textit{notification-provoked sampling and notification)}
\end{cases} \tag{22}
$$

Case 2: i is an internal node

Let us now consider an internal node i with two child nodes, i_0 and i_1.

Case 2.1: i is an unnotified internal node

We will start with the configuration where the i's branch corresponds to an unnotified individual. Hence the i's branch evolution is represented by $p^{(i)}(t_j)$ with a transmission event at the end of it, where each of the child branches can correspond to the donor (probability density of 2λ). We will consider three possibilities: (2.1.1) when both i_0 and i_1 are internal nodes; (2.1.2) when both i_0 and i_1 are tips; and (2.1.3) when one of them is an internal node and the other one is a tip.

Case 2.1.1: i is an unnotified internal node with internal-node children $i0$ and $i1$

First, let's assume that i_0 and i_1 are internal. Then none of them can be a notifier (since notifiers correspond to tips), and hence none of them is notified (as i is also unnotified):

$$l^{(\not{n})}(i) = p^{(i)}(t_j)2\lambda l^{(\not{n})}(i_0)l^{(\not{n})}(i_1) \tag{23}$$

Case 2.1.2: i is an unnotified internal node with tip children $i0$ and $i1$

Secondly, let's assume that i_0 and i_1 are tips. Then each of them could have notified the other one:

$$\begin{aligned} l^{(\not{n})}(i) = p^{(i)}(t_j)2\lambda \cdot \Big(& l^{(\not{n},\not{n})}(i_0)l^{(\not{n},\not{n})}(i_1) \leftarrow \textit{neigher } i_0 \textit{ nor } i_1 \textit{ notified} \\ & + l^{(\not{n},n)}(i_0)l^{(n,\not{n})}(i_1,i_0) \leftarrow i_0 \textit{ notified } i_1 \textit{ and } i_1 \textit{ did not notify} \\ & + l^{(n,\not{n})}(i_0,i_1)l^{(\not{n},n)}(i_1) \leftarrow i_0 \textit{ did not notify, but } i_1 \textit{ notified } i_0 \\ & + l^{(n,n)}(i_0,i1)l^{(n,n)}(i_1,i_0) \Big) \leftarrow i_1 \textit{ and } i_0 \textit{ notified each other} \end{aligned} \tag{24}$$

Case 2.1.3: i is an unnotified internal node whose children are a tip and an internal node

Lastly, let's assume that i_0 is a tip and i_1 is internal (the scenario where i_1 is a tip and i_0 is internal can be obtained by swapping the labels). Then i_0 could have notified i_1:

$$\begin{aligned} l^{(\not{n})}(i) = p^{(i)}(t_j)2\lambda \cdot \Big(& l^{(\not{n},\not{n})}(i_0)l^{(\not{n})}(i_1) \leftarrow i_0 \textit{ did not notify } i_1 \\ & + l^{(\not{n},n)}(i_0)l^{(n)}(i_1,i_0) \Big) \leftarrow i_0 \textit{ notified } i_1 \end{aligned} \tag{25}$$

Case 2.2: i's branch is internal and (at least initially) corresponds to an (eventually notified) partner

In the other case the internal node i's branch starts as a partner who will eventually get notified by a tip r. There are two possibilities: either the node i corresponds to the partner or the partner is hidden along the i's branch and i corresponds to someone within i's recipient subtree (e.g., if $i = C$ in Fig. 2(2a)). In the first case, the i's branch evolution is represented by the partner branch evolution $p_p^{(i)}(t_j, t_r)$:

$$p_p^{(i)}(t_j,t_r) = \begin{cases} p_{pa}^{(i)}(t_j) & \textit{if } t_r < t_j, \\ p_{pb}^{(i)}(t_j) & \textit{if } t_r > t_i, \\ p_{pb}^{(r)}(t_j)p_{pa}^{(i)}(t_r) & \textit{if } t_j \le t_r \le t_i. \end{cases} \tag{26}$$

This branch has a transmission event at the end of it, where the donor is known (probability density of λ): The donor corresponds to the subtree containing the tip representing the sampling of the individual i.

In the second case, the i's branch evolution is represented by $p_{m(p,-)}^{(i)}(t_j, t_r)$ with a transmission event at the end of it, where the donor is unknown (probability density of 2λ): any of the subtrees can be the donor.

Again, we will consider three possibilities for i's child nodes: (2.2.1) when both i_0 and i_1 are internal nodes; (2.2.2) when both i_0 and i_1 are tips; and (2.2.3) when one of them is an internal node and the other one is a tip.

Case 2.2.1: i's branch is internal and (at least initially) corresponds to a partner with internal-node children $i0$ and $i1$

First, let's assume that i_0 and i_1 are internal. Then none of them can be a notifier (since notifiers correspond to tips), and hence either exactly one of them is notified by r, or the partner is hidden:

$$\begin{aligned} l^{(n)}(i,r) =& p_p^{(i)}(t_j,t_r)\lambda \cdot \Big(l^{(n)}(i_0,r)l^{(\not{n})}(i_1) \leftarrow i_0 \textit{ is notified by } r \\ &\qquad + l^{(\not{n})}(i_0)l^{(n)}(i_1,r) \Big) \leftarrow i_1 \textit{ is notified by } r \\ &+ p_{m(p,-)}^{(i)}(t_j,t_r)2\lambda l^{(\not{n})}(i_0)l^{(\not{n})}(i_1) \leftarrow \textit{hidden partner} \end{aligned} \tag{27}$$

Case 2.2.2: i's branch is internal and (at least initially) corresponds to a partner with tip children $i0$ and $i1$

Secondly, let's assume that i_0 and i_1 are tips. Then each of them could have notified the other one (in addition to one of them potentially being notified by r):

$$\begin{aligned} l^{(n)}(i,r) =& p_p^{(i)}(t_j,t_r)\lambda \cdot \Big(l^{(n,\not{n})}(i_0, first(r,i_1))l^{(\not{n},n)}(i_1) \leftarrow r \textit{ and } i_1 \textit{ notified } i_0,\ i_0 \textit{ did not notify} \\ &+ l^{(n,n)}(i_0, first(r,i_1))l^{(n,n)}(i_1,i_0) \leftarrow r \textit{ and } i_1 \textit{ notified } i_0,\ i_0 \textit{ notified } i_1 \\ &+ l^{(\not{n},n)}(i_0)l^{(n,\not{n})}(i_1, first(r,i_0)) \leftarrow r \textit{ and } i_0 \textit{ notified } i_1,\ i_1 \textit{ did not notify} \\ &+ l^{(n,n)}(i_0,i_1)l^{(n,n)}(i_1, first(r,i_0)) \leftarrow r \textit{ and } i_0 \textit{ notified } i_1,\ i_1 \textit{ notified } i_0 \\ &+ l^{(n,n)}(i_0,r)l^{(n,\not{n})}(i_1,i_0) \leftarrow r \textit{ notified } i_0,\ i_0 \textit{ notified } i_1,\ i_1 \textit{ did not notify} \\ &+ l^{(n,\not{n})}(i_0,i_1)l^{(n,n)}(i_1,r) \leftarrow r \textit{ notified } i_1,\ i_1 \textit{ notified } i_0,\ i_0 \textit{ did not notify} \\ &+ l^{(n,\not{n})}(i_0,r)l^{(\not{n},\not{n})}(i_1) \leftarrow r \textit{ notified } i_0,\ i_0 \textit{ and } i_1 \textit{ did not notify} \\ &+ l^{(\not{n},\not{n})}(i_0)l^{(n,\not{n})}(i_1,r) \Big) \leftarrow r \textit{ notified } i_1,\ i_0 \textit{ and } i_1 \textit{ did not notify} \\ &+ p_{m(p,-)}^{(i)}(t_j,t_r)2\lambda \cdot \Big(l^{(n,\not{n})}(i_0,i_1)l^{(\not{n},n)}(i_1) \leftarrow r \textit{ notified a hidden partner,} \\ &\qquad\qquad i_1 \textit{ notified } i_0,\ i_0 \textit{ did not notify} \\ &\quad + l^{(n,n)}(i_0,i_1)l^{(n,n)}(i_1,i_0) \leftarrow r \textit{ notified a hidden partner,} \\ &\qquad\qquad i_1 \textit{ notified } i_0,\ i_0 \textit{ notified } i_1 \\ &\quad + l^{(\not{n},n)}(i_0)l^{(n,\not{n})}(i_1,i_0) \leftarrow r \textit{ notified a hidden partner,} \\ &\qquad\qquad i_0 \textit{ notified } i_1,\ i_1 \textit{ did not notify} \\ &\quad + l^{(\not{n},\not{n})}(i_0)l^{(\not{n},\not{n})}(i_1) \Big) \leftarrow r \textit{ notified a hidden partner,} \\ &\qquad\qquad i_0 \textit{ and } i_1 \textit{ did not notify} \end{aligned}$$

$$\textit{where } first(a,b) = \begin{cases} a \textit{ if } t_a \leq t_b \\ b \textit{ if } t_a > t_b. \end{cases} \tag{28}$$

Case 2.2.3: i's branch is internal and (at least initially) corresponds to a partner, i's children are a tip and an internal node

Lastly, let's assume that i_0 is a tip and i_1 is internal (the scenario where i_1 is a tip and i_0 is internal can be obtained by swapping the labels). Then i_0 could have notified i_1 (in addition to one of them being potentially notified by r):

$$
\begin{aligned}
l^{(n)}(i,r) = & p_p^{(i)}(t_j,t_r)\lambda \cdot \Big(l^{(n,n)}(i_0,r) l^{(n)}(i_1,i_0) \leftarrow r \textit{ notified } i_0,\ i_0 \textit{ notified } i_1 \\
& + l^{(n,\not{n})}(i_0,r) l^{(\not{n})}(i_1) \leftarrow r \textit{ notified } i_0,\ i_0 \textit{ did not notify} \\
& + l^{(\not{n},n)}(i_0) l^{(n)}(i_1, first(r,i_0)) \leftarrow \textit{both } r \textit{ and } i_0 \textit{ notified } i_1 \\
& + l^{(\not{n},\not{n})}(i_0) l^{(n)}(i_1,r) \Big) \leftarrow r \textit{ notified } i_1,\ i_0 \textit{ did not notify} \\
& + p_{m(p,-)}^{(i)}(t_j,t_r) 2\lambda \cdot \Big(l^{(\not{n},n)}(i_0) l^{(n)}(i_1,i_0) \leftarrow r \textit{ notified a hidden partner,} \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i_0 \textit{ notified } i_1 \\
& \qquad\qquad + l^{(\not{n},\not{n})}(i_0) l^{(\not{n})}(i_1) \Big) \leftarrow r \textit{ notified a hidden partner,} \\
& \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i_0 \textit{ did not notify}
\end{aligned}
\tag{29}
$$

$$
\textit{where } first(a,b) = \begin{cases} a \textit{ if } t_a \leq t_b \\ b \textit{ if } t_a > t_b. \end{cases}
$$

Tree likelihood calculation with a pruning algorithm and its time complexity

Note that $l^{(n)}(i,r)$, $l^{(n,n)}(i,r)$ and $l^{(n,\not{n})}(i,r)$ depend on the notifier r and the notification time t_r. Note that for a node i any tip r that is sister to any internal node on the path between i and the tree root could be its potential notifier. During the tree likelihood calculation we first annotate each node i with a set of potential notifiers N_i. For that we first add tips' to their parent nodes' potential notifier sets. We then perform a preorder tree traversal, where for every visited internal node i we add its potential notifiers to its child nodes $i0$ and $i1$: $N_{i0} := N_{i0} \cup N_i - \{i0\}$, $N_{i1} := N_{i1} \cup N_i - \{i1\}$ (removing the child itself, as if it is a tip it will be found among its parent's potential notifiers). This preprocessing is only performed once, takes time O(N) where N is the number of tips in the tree and is negligible with respect to the rest of the parameter inference procedure.

Then we perform a postorder tree traversal, where for each node i we calculate $l^{(\not{n},\not{n})}(i)$, $l^{(\not{n},n)}(i)$, $l^{(n,\not{n})}(i,r)$ and $l^{(n,n)}(i,r)$ $\forall r \in N_i$ if it is a tip, or $l^{(\not{n})}(i)$ and $l^{(n)}(i,r)$ $\forall r \in N_i$ if it is internal. This part takes O(R + N) time, where O(N) accounts for calculation of $l^{(\not{n})}(i)$ for all the $N-1$ internal tree nodes and of $l^{(\not{n},\not{n})}(i)$ and $l^{(\not{n},n)}(i)$ for N tips, while R is the sum of numbers of potential notifiers over all tree nodes and O(R) accounts for calculation of $l^{(n)}(i,r)$, $l^{(n,\not{n})}(i,r)$ and $l^{(n,n)}(i,r)$ $\forall i$ $\forall r$.

For the best case (a perfectly balanced tree, where each tip is in a cherry), the number of potential notifiers is one for the tips (i.e., the sister tip) and zero for the internal nodes. Hence $R_{balanced} = N$ and the likelihood calculation time is linear: O(N). The worst case is represented by a caterpillar tree (where each internal node has a tip child). Each internal node has k potential notifiers (tips in its supertree), where k is the node's depth in the tree, the root's depth being 0 and the deepest internal node's depth being $N-2$. Moreover each tip has as many notifiers as its parent node, apart from the two deepest tips who have $N-1$ notifiers each instead of $N-2$ as they are in a cherry and hence could have notified each other. This hence sums up to $R_{caterpillar} = 2\big(1+2+\ldots+(N-3)\big) + 2(N-1) = N^2 - 2N + 2$ and the time is hence quadratic: $O(N^2)$. On our BD-PN simulated dataset of 100 trees with N=500-1 000 tips each, $R \in [2N, 38N]$ with a median at $7N$, suggesting a likelihood calculation time close to the linear one.

Acknowledgments

The authors would like to thank Dr Mathieu Moslonka-Lefebvre who initiated this project by proposing to study partner notification. He worked on a different version of the PN model (which is not part of this manuscript) and on the initial version of the

PN test. We would like to thank Dr Jakub Voznica for fruitful discussions on partner notification mechanisms. We would also like to thank the HPC cluster team in Institut Pasteur for their support with computer simulations. O.G. was supported by the Paris Artificial Intelligence Research Institute (PRAIRIE, ANR-19-P3IA-0001).

References

1. Grenfell BT, Pybus OG, Gog JR, Wood JLN, Daly JM, Mumford JA, et al. Unifying the Epidemiological and Evolutionary Dynamics of Pathogens. Science. 2004;303(5656).

2. Volz EM, Koelle K, Bedford T, Bhattacharya T, Delaporte E. Viral Phylodynamics. PLoS Computational Biology. 2013;9(3):e1002947. doi:10.1371/journal.pcbi.1002947.

3. Vaughan TG, Scire J, Nadeau SA, Stadler T. Estimates of early outbreak-specific SARS-CoV-2 epidemiological parameters from genomic data. Proceedings of the National Academy of Sciences. 2024;121(2):e2308125121. doi:10.1073/pnas.2308125121.

4. Volz EM, Kosakovsky Pond SL, Ward MJ, Leigh Brown AJ, Frost SDW. Phylodynamics of infectious disease epidemics. Genetics. 2009;183(4):1421–1430. doi:10.1534/genetics.109.106021.

5. Drummond AJ, Rambaut A, Shapiro B, Pybus OG. Bayesian coalescent inference of past population dynamics from molecular sequences. Molecular Biology and Evolution. 2005;22(5):1185–1192. doi:10.1093/molbev/msi103.

6. Pybus OG, Rambaut A, Harvey PH. An integrated framework for the inference of viral population history from reconstructed genealogies. Genetics. 2000;155(3):1429–1437. doi:10.1093/genetics/155.3.1429.

7. Kendall DG. On the generalized “birth-and-death” process. Ann Math. 1948;19(1):1–15.

8. Maddison W, Midford P, Otto S. Estimating a Binary Character’s Effect on Speciation and Extinction. Systematic Biology. 2007;56(5):701–710. doi:10.1080/10635150701607033.

9. Stadler T. On incomplete sampling under birth-death models and connections to the sampling-based coalescent. Journal of Theoretical Biology. 2009;261(1):58–66. doi:10.1016/j.jtbi.2009.07.018.

10. Stadler T. Sampling-through-time in birth-death trees. J Theor Biol. 2010;267:396–404.

11. Macpherson A, Louca S, Mclaughlin A, Joy JB, Pennell MW. Unifying Phylogenetic Birth-Death Models in Epidemiology and Macroevolution. Systematic Biology. 2021;71(1). doi:10.1093/sysbio/syab049.

12. Stadler T, Bonhoeffer S. Uncovering epidemiological dynamics in heterogeneous host populations using phylogenetic methods. Philosophical Transactions of the Royal Society B: Biological Sciences. 2013;368(1614):20120198–20120198. doi:10.1098/rstb.2012.0198.

13. Stadler T, Kühnert D, Rasmussen DA, du Plessis L. Insights into the Early Epidemic Spread of Ebola in Sierra Leone Provided by Viral Sequence Data. PLoS Currents. 2014;6. doi:10.1371/currents.outbreaks.02bc6d927ecee7bbd33532ec8ba6a25f.

14. Stadler T, Kühnert D, Bonhoeffer S, Drummond AJ. Birth-death skyline plot reveals temporal changes of epidemic spread in HIV and hepatitis C virus (HCV). Proceedings of the National Academy of Sciences of the United States of America. 2013;110(1):228–33. doi:10.1073/pnas.1207965110.

15. Kühnert D, Stadler T, Vaughan TG, Drummond AJ. Phylodynamics with Migration: A Computational Framework to Quantify Population Structure from Genomic Data. Molecular biology and evolution. 2016;33(8):2102–16. doi:10.1093/molbev/msw064.

16. El-Sadr WM, Platt J, Bernitz M, Reyes M. Contact Tracing: Barriers and Facilitators. Am J Public Health. 2022;112(7):1025–1033. doi:10.2105/AJPH.2022.306842.

17. World Health Organization. Country policy review on HIV partner notification services. In: Guidelines on HIV Self-Testing and Partner Notification: Supplement to Consolidated Guidelines on HIV Testing Services. World Health Organization; 2016.Available from: `https://www.ncbi.nlm.nih.gov/books/NBK401681/`.

18. Ward H, Bell G. Partner notification. Medicine (Abingdon). 2014;42(6):314–317. doi:10.1016/j.mpmed.2014.03.013.

19. Zucchetti A, Brown A, Hughes G, Nardone A, Gill O, Delpech V, et al. HIV in the United Kingdom 2014 Report: data to end 2013. London, UK: Public Health England; 2014. Available from: `https://www.bhiva.org/file/sqJBbVwZimSFv/HIV_Partner_Notification_Standards_2015.pdf`.

20. Voznica J, Zhukova A, Boskova V, Saulnier E, Lemoine F, Moslonka-Lefebvre M, et al. Deep learning from phylogenies to uncover the transmission dynamics of epidemics. Nature Communications. 2022;13:3896. doi:10.1038/s41467-022-31511-0.

21. Mooers A, Gascuel O, Stadler T, Li H, Steel M. Branch Lengths on Birth–Death Trees and the Expected Loss of Phylogenetic Diversity. Systematic Biology. 2012;61(2):195–203. doi:10.1093/sysbio/syr090.

22. Zhukova A, Hecht F, Maday Y, Gascuel O. Fast and Accurate Maximum-Likelihood Estimation of Multi-Type Birth–Death Epidemiological Models from Phylogenetic Trees. Systematic Biology. 2023; p. syad059. doi:10.1093/sysbio/syad059.

23. Felsenstein J. Maximum Likelihood and Minimum-Steps Methods for Estimating Evolutionary Trees from Data on Discrete Characters. Systematic Biology. 1973;22(3):240–249. doi:10.1093/sysbio/22.3.240.

24. Wilks SS. The Large-Sample Distribution of the Likelihood Ratio for Testing Composite Hypotheses. The Annals of Mathematical Statistics. 1938;9(1):60–62. doi:10.1214/aoms/1177732360.

25. Huerta-Cepas J, Serra F, Bork P. ETE 3: Reconstruction, Analysis, and Visualization of Phylogenomic Data. Molecular Biology and Evolution. 2016;33(6):1635–1638. doi:10.1093/molbev/msw046.

26. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. Array programming with NumPy. Nature. 2020;585(7825):357–362. doi:10.1038/s41586-020-2649-2.

27. Rasmussen DA, Kouyos R, Günthard HF, Stadler T. Phylodynamics on local sexual contact networks. PLOS Computational Biology. 2017;13(3):1–23. doi:10.1371/journal.pcbi.1005448.

28. Swiss HIV Cohort Study, Schoeni-Affolter F, Ledergerber B, Rickenbach M, Rudin C, Günthard HF, et al. Cohort profile: the Swiss HIV Cohort study. Int J Epidemiol. 2010;39(5):1179–1189. doi:10.1093/ije/dyp321.

29. Zhukova A, Dunn D, Gascuel O. Modeling Drug Resistance Emergence and Transmission in HIV-1 in the UK. Viruses. 2023;15(6):1244. doi:10.3390/v15061244.

30. Dunn D, Pillay D. UK HIV drug resistance database: background and recent outputs. Journal of HIV therapy. 2007;12(4):97–8.

31. To TH, Jung M, Lycett S, Gascuel O. Fast Dating Using Least-Squares Criteria and Algorithms. Systematic biology. 2016;65(1):82–97. doi:10.1093/sysbio/syv068.

32. Williams I, Churchill D, Anderson J, Boffito M, Bower M, Cairns G, et al. British HIV Association guidelines for the treatment of HIV-1-positive adults with antiretroviral therapy 2012. HIV Med. 2012;13 Suppl 2:1–85. doi:10.1111/j.1468-1293.2012.01029.x.

33. Churchill D, Waters L, Ahmed N, Angus B, Boffito M, Bower M, et al. British HIV Association guidelines for the treatment of HIV-1-positive adults with antiretroviral therapy 2015. HIV Medicine. 2016;17(S4):s2–s104. doi:10.1111/hiv.12426.

34. Kirwan P, Chau C, Brown A, Gill O, Delpech V, contributors. HIV in the UK - 2016 report. London, UK: Public Health England; 2016. Available from: `https://assets.publishing.service.gov.uk/media/5a74fbc540f0b6399b2afbd6/HIV_in_the_UK_report.pdf`.

35. FitzJohn RG, Maddison WP, Otto SP. Estimating Trait-Dependent Speciation and Extinction Rates from Incompletely Resolved Phylogenies. Systematic Biology. 2009;58(6):595–611. doi:10.1093/sysbio/syp067.

36. Goldberg EE, Igić B. Tempo and Mode in Plant Breeding System Evolution. Evolution. 2012;66(12):3701–3709. doi:10.1111/j.1558-5646.2012.01730.x.

37. MacPherson A, Louca S, McLaughlin A, Joy JB, Pennell MW. Unifying Phylogenetic Birth–Death Models in Epidemiology and Macroevolution. Systematic Biology. 2022;71(1):172–189. doi:10.1093/sysbio/syab049.

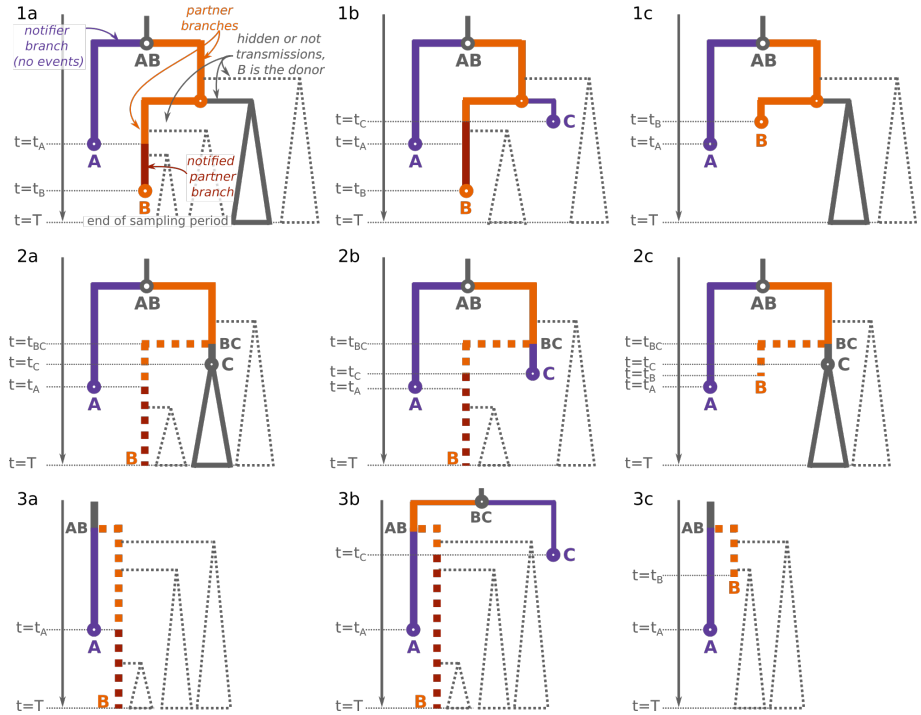

Fig 2. Examples of partner notification in a transmission tree. Let us assume that an individual A got sampled at time t_A (violet tip) and notified their most recent partner B (orange), who is either observed (**1**) or hidden (**2-3**). The (observed as in **1-2** or hidden as in **3**) internal node AB corresponds to the transmission between A and B. The fact that A is the notifier of B implies that there is no hidden event along its branch AB-A (violet), and that in any transmission on the path AB-B (orange or dark-red) the donor is B. Note, that since the most-recent-partner relationship is not always symmetric, B might be the most recent partner of more than one individual and hence might have got notified by someone else too (e.g., by C in **b**, who in **1b,2b** was infected by B, and in **3b** could have been infected by or have infected B, who have then infected A). Moreover, between the moment of B's first notification (t_A for **a**, t_C for **b**) and the moment of sampling of B (t_B) the removal rate of B becomes ϕ instead of ψ (dark red). Finally, B might have been removed via a standard procedure before getting notified by their partner(s) (with rate ψ and a probability ρ, **c**). In the panels **3** B's subtree is fully hidden, and hence the transmission node AB is also unobserved. Therefore the A's branch is composed of two parts, where the bottom part (violet) corresponds to A and contains no hidden transmissions, while the top part could correspond to A or not (e.g in **3b** it corresponds to B) and might contain hidden transmissions. In the panels **2a-b,3a-b** B is hidden as by the end of the sampling period ($t = T$) B is not yet sampled, while in **2c,3c** B is hidden as B got removed via the standard procedure without sampling before any notification.
A's branch is modeled with a notifier branch probability (4) in panels **1-2**, and with a mixed branch probability (**??**) where the bottom part corresponds to the notifier (A) in panels **3**. If B did not notify their most recent partner upon sampling in panel **1**, branches on the path AB-B are modeled with a combination of partner probability before notification (orange, (5)) and partner probability after notification (dark red, (6)). Otherwise, the part after the transmission from B to their last partner is modeled with either a both notified and notifier branch probability (7) (as in **1a-b**) or a notifier branch probability (4) (as in **1c**), depending on the B's notified status at that time. In the panels **2**, the AB-C branch is modeled with a mixed branch probability (**??**), where the top part corresponds to the partner state (B).

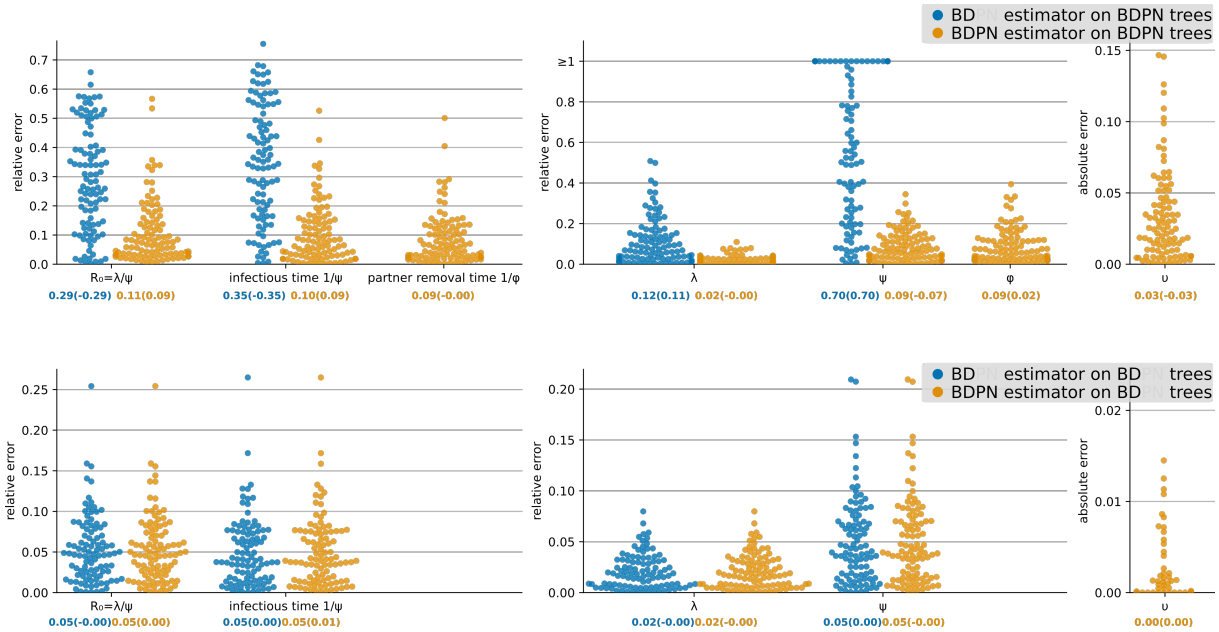

Fig 3. Comparison of inference accuracy of bdpn (yellow) and bd (blue) estimators on a data set of 100 trees of 500–1 000 tips generated under BD-PN model (top) and a data set of 100 trees of 500–1 000 tips generated under BD model (bottom), with ρ parameter being fixed to its true value. We show the swarmplots (coloured by estimator) of errors for each test tree and parameter (epidemiological on the left, rates in the middle and probability on the right). For the epidemiological and rate parameters relative errors are shown (normalized distance between the estimate and the true value), for the partner notification probability υ the absolute errors are shown (distance between the estimate and the true value, where for the BD model the true value is zero). Average error and in parentheses average bias (absolute for probability, relative otherwise) are displayed for each parameter and method below their swarmplot.

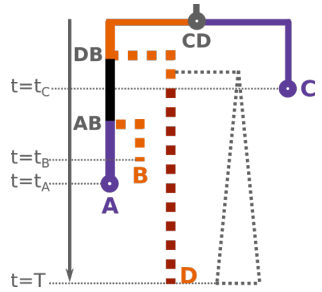

Fig 4. An example of mixed tip branch with two hidden partner subtrees: an individual C got sampled at time t_C and notified their most recent partner D. D however, stayed unobserved as D did not get sampled by the end of the sampling period (time T). Before notification (at time t_C) D transmitted two times, the latter transmission leading to an unobserved recipient subtree (gray dotted triangle) and the former transmission (DB) leading to an observed recipient subtree (which includes a sampled individual A and a hidden individual B). The D's recipient A transmitted further (transmission AB). Once A got sampled at time t_A, they notified their most recent partner B, who however was already removed by then via the standard procedure without sampling (at time $t_B < t_A$). Hence the branch CD-A represents a mixed branch with states partner(CD-DB)-standard(DB-AB)-notifier(AB-A).