SCIENTIFIC REPORTS 6 (2016) 33987 doi:10.1038/srep33987

Epigraph: A Vaccine Design Tool Applied to an HIV Therapeutic Vaccine and a Pan-Filovirus Vaccine

James Theiler[1,2], Hyejin Yoon[1], Karina Yusim[1,2], Louis J. Picker[3], Klaus Früh[3], and Bette Korber[1,2,*]

[1]Los Alamos National Laboratory, Los Alamos, NM 87545, USA
[2]New Mexico Consortium, Los Alamos, NM 87544, USA
[3]Oregon Health and Science University, Portland, OR 97239, USA
[*]Correspondence to btk@lanl.gov

ABSTRACT

Epigraph is an efficient graph-based algorithm for designing vaccine antigens to optimize potential T-cell epitope (PTE) coverage. Epigraph vaccine antigens are functionally similar to Mosaic vaccines, which have demonstrated effectiveness in preliminary HIV non-human primate studies. In contrast to the Mosaic algorithm, Epigraph is substantially faster, and in restricted cases, provides a mathematically optimal solution. Epigraph furthermore has new features that enable enhanced vaccine design flexibility. These features include the ability to exclude rare epitopes from a design, to optimize population coverage based on inexact epitope matches, and to apply the code to both aligned and unaligned input sequences. Epigraph was developed to provide practical design solutions for two outstanding vaccine problems. The first of these is a personalized approach to a *therapeutic* T-cell HIV vaccine that would provide antigens with an excellent match to an individual's infecting strain, intended to contain or clear a chronic infection. The second is a pan-filovirus vaccine, with the potential to protect against all known viruses in the *Filoviradae* family, including ebolaviruses. A web-based interface to run the Epigraph tool suite is available (http://www.hiv.lanl.gov/content/sequence/EPIGRAPH/epigraph.html).

Introduction

HIV is highly variable, largely as a consequence of immune selection acting on this highly mutable virus during chronic infection[1–4]; even the most conserved regions of HIV are variable at the epitope level[5,6]. T-cell epitopes are short contiguous stretches of protein, peptides generally between 9-12 amino acids long, which are presented on the surface of infected cells to enable recognition, and to trigger a T-cell based immune-response. Epitope variability limits the cross-reactive potential of single antigen vaccines, such as a natural protein or a consensus sequence[7,8]. Mosaic vaccines were originally designed to contend with HIV diversity by including a small set of (typically two to four) complementary antigens, rather than a single antigen. They include several artificial proteins that resemble natural proteins, but are collectively designed to maximally cover diverse epitopes in a targeted population[9], offering highly improved epitope coverage over combinations of natural sequences.

Mosaics and Epigraphs solve essentially the same optimization problem (PTE coverage), and are thus expected to behave the same way experimentally. Mosaic antigens have already been designed, synthesized, and tested, and have shown promise on a variety of fronts. When expressed, Mosaic antigens have folded well in terms of binding discontinuous antibodies, and are highly immunogenic, eliciting both T-cell and antibody responses[8,10,11]. T-cell responses induced by Mosaics effectively target HIV infected cells[12], and are more cross-reactive than those induced by natural proteins[8,10,13–15]. Mosaic vaccines have shown promise against HIV-1[8,10,11,13,16], as well as other variable pathogens, including the viruses that cause Hepatitis C[17], Ebola[18], and Influenza[19].

Despite the similarities in the overall optimization criteria, Epigraphs provide substantial advantages over our original Mosaic strategy. Mosaics use a genetic algorithm[9], while Epigraphs use a much faster graph-based approach (see Formulation). This speed, as well as the structure of the mathematical framework, facilitated the addition of new features to the Epigraph tool suite[20]. More importantly, while Mosaics provided a near optimal solution for antigen design to maximize PTE coverage by a vaccine across a simple population, the code was not readily adapted to more complex problems. We developed Epigraph to enable computational solutions to two pressing T cell vaccine design problems that were intractable using the computationally

slower Mosaic algorithm: a pan-filovirus T-cell vaccine and a strategy for matching vaccines to infecting strains in a therapeutic setting.

Recently there has been resurgence of interest in T cell-based vaccines. SIV vaccine antigens presented in rhesus macaque cytomegalovirus (RhCMV) vectors generate prolific T-cell responses that enable stringent control and progressive clearance of pathogenic SIV upon infection in over 50% of vaccinated monkeys. These responses violate traditional paradigms of T cell-mediated immunity, and provide new impetus for exploring T-cell vaccine approaches[21,22]. There is also keen interest in focusing vaccine-stimulated T-cell responses on conserved regions, to shift immunodominance to epitopes with a limited capacity to escape because they are under fitness constraints[16,23–25]. Such T-cell vaccination strategies may be beneficial in either a preventive or therapeutic setting. It was to pursue these innovative vaccine strategies that we developed Epigraph, a flexible and computationally efficient strategy for optimizing epitope coverage in a variety of scenarios. The coupling of Epigraph antigen design with contemporary vaccine delivery approaches offers a promising strategy with the potential to advance vaccine efforts against the challenge of highly variable pathogens.

Here we applied Epigraph to provide potential solutions to two outstanding vaccine design problems. First, we describe a Tailored Therapeutic Vaccine (TTV) approach. In contrast to a vaccine that prevents infection, in therapeutic setting it is possible to obtain sequences from the infected person who will be treated, and attempt to match their infecting strain as closely as possible to the vaccine. It is not currently feasible to make a new matched vaccine for every person you treat. The TTV approach enables the design of a small set (a half a dozen or so) vaccine antigens, a practical number for manufacture. Then, for each individual, the best two or three antigen subset of these six can be selected to provide a "tailored" match to viral sequences sampled from the patient to be treated. The Epigraph-based TTV code optimizes the set of vaccine antigens for manufacture, such that the set will sample the diversity of the target population, and enable the best vaccine matches overall for infected individuals in the target population. Here we apply TTV approach to HIV, though the strategy is general. A single TTV design run can loop over more than a thousand basic Epigraph runs, so the computational efficiency is essential to complete the Epigraph design.

We next explored the problem of how to design a vaccine that could cover the viral diversity found in the entire *Filoviradae* family, which includes ebolavirus and marburgvirus viruses, as well as other related viruses that can cause fatal hemorrhagic fevers in human and non-human primates. First we used Epigraph to define the most conserved regions of the filovirus proteome, then we used it to design antigens that would best cover the diversity that was found in those regions. We explored dozens of design strategies to finally identify a vaccine solution that met three criteria. First, we felt it was critical that the potential T-cell epitope coverage of the ebola virus species, that has historically seeded most outbreaks, not be compromised. Second, we wanted the design to have the potential to elicit responses against of the full range of known diversity of viruses in the *Filoviradae* family. Third, we made sure that the conserved regions that were included spanned relatively large stretches of protein, so that epitopes representing a broad spectrum of human leukocyte antigens (HLAs) would be included. Exploring the combinatorics of the many design options we considered to meet these criteria would have been prohibitive using the slower Mosaic code, but through systematic use of Epigraph, we were able to identify a promising design strategy that met all three criteria.

Epigraph Formulation

Central to both Epigraphs and Mosaics is the concept of potential epitope coverage. Because *known* T-cell epitopes are very densely packed in HIV[5], we consider every contiguous epitope-length fragment (*i.e.,* every k-mer) to be a *potential* epitope. We usually set $k = 9$ as the length of potential T-cell epitopes (PTEs), as this is the optimal length of most cytotoxic T-cell Class I presented epitopes[5,9], but solutions optimized on $k = 9$ are still very good for other common epitope lengths of 8-12 amino acids (Supplementary Table 1). If using a PTE length of 9, the first PTE in a sequence will be the peptide from position 1 to 9 in the protein, the second PTE from 2 to 10, *etc.*

Here we will briefly describe the steps taken in the Epigraph algorithm, to impart an intuition for what the algorithm is doing; a more detailed and precise mathematical description is provided in the Methods. Fig. 1 provides an illustration of the Epigraph strategy.

The first step in Epigraph design is to assemble a representative sample of N protein sequences that embodies the viral diversity in a population that will be targeted for vaccine use (*e.g.,* a phylogenetic clade, a country, or the world). The input proteins do not have to be aligned, but they can be, as will be discussed below. Each sequence in the set is decomposed into all possible 9-mers, and the number, n, of recurrences of a particular 9-mer in the sample population is tallied. Each unique 9-mer found at least once in the sample population will be associated with its frequency in the population. For instance, if there are 1000 sequences in the sample population ($N = 1000$), and a particular 9-mer was found exactly matched in 200 of those sequences ($n = 200$), then the frequency of that 9-mer is $n/N = 0.2$. We characterize the potential cross-reactivity of an antigen by the sum of the frequencies of all the 9-mers in the antigen sequence; if we divide this quantity by the sum of frequencies of all the distinct 9-mers in the population, that provides the coverage score.

Next, as illustrated in Fig. 1, a graph is created. Formally, a graph is a collection of nodes and edges (edges connect pairs of nodes). In our graph, each node corresponds to a unique 9-mer, and two nodes are connected by an edge whenever the 9-mers in those two nodes share an overlap of 8 amino acids. A path through this graph is a sequential assembly of connected nodes, with the last 8 amino acids of each node matching the first 8 amino acids of the subsequent node. These overlaps allow such a path to be associated with a single sequence of amino acids. Epigraph (implicitly) considers all the paths – there are exponentially many of them – in the graph and identifies an optimal path. The criterion for optimality is the coverage score, which is proportional to the sum of frequencies associated with the nodes in the path. And from these nodes, we can construct an intact full-length protein sequence. This sequence is the first antigen in our vaccine, and it will contain the most common 9-mers in the target population, to the extent possible given the constraint that those 9-mers have to overlap so that they can be expressed with a single complete protein sequence. For a monovalent vaccine, this antigen is all we need. But once the first antigen has been generated, we can produce a second complementary sequence by finding a second path through the graph that again optimizes the sum of frequencies, but this time without including the frequencies of 9-mers that were already included in the first antigen. This second step is achieved by setting the frequencies of those initial 9-mers to zero during this second optimization. In this way, if a particular 9-mer is an essential block for building a complete protein, it can be incorporated into both antigens, but as it does not increase the coverage score, it will not be favored. This process is repeated until the desired number of antigens is generated for a polyvalent vaccine.

Comparison of Epigraph with Mosaics

The Mosaic approach optimizes coverage with a genetic algorithm in a loop that alternately recombines regions of natural proteins at random breakpoints and creates pools of these *in silico* generated recombinants, and then selects those candidates with the best coverage for the next generation from among those pools[9]. In contrast, the Epigraph algorithm optimizes that criterion by finding a path through the k-mer overlap graph (Fig. 1). Epigraph solutions generally have a slightly improved PTE population coverage relative to Mosaics when applied to HIV proteins (Supplementary Tables 2 and 3). While this coverage advantage is small, the computational advantage in terms of run-time is substantial. Epigraph can complete a basic vaccine design in seconds on a laptop (Supplementary Table 8), while Mosaic designs can take hours to days to approach optimization[26]. When Mosaic antigens are used to initialize an Epigraph run, coverage scores can often be increased, albeit very slightly (Supplementary Table 4). Thus high quality (and in certain cases, mathematically optimal) antigens can be very rapidly determined with Epigraph, and this leads to new opportunities for innovative vaccine design that would otherwise be computationally onerous to pursue.

Two caveats are in order here. One is that an Epigraph solution is mathematically optimal only if the directed graph is acyclic – that is to say, the graph generated in Fig. 1 contains no cycles. One simple example of how a cycle can arise in a graph is when a 9-mer is precisely repeated in two different places in a protein. In practice, most graphs we have used are not acyclic, and that usually means we need to do some pre-processing (see Methods: De-cycling). A second caveat is that optimality only applies to the single antigen (monovalent) case. For polyvalent vaccines, we employ heuristics to bootstrap the monovalent optimality (see Methods: Polyvalent vaccines).

Excluding Rare Epitopes

Natural but rare PTEs are undesirable in a vaccine because they can elicit type-specific responses. Natural HIV proteins carry a surprising number of such PTEs, and when these rare forms are immunogenic or immunodominant, they may curtail the cross-reactive potential of a vaccine. One of the analysis tools in the Epigraph tool suite[20] provides the frequencies of every distinct PTE in a population, and the output provides a sobering lens with which to view natural HIV diversity. For example, the Los Alamos HIV database M group Env alignment[27] of 4,250 sequences contains over 650,000 distinct 9-mer peptides; of these, over 500,000 are unique, each appearing only once in the population. This is an average of 120 unique PTEs per natural Env sequence, and responses to such PTEs would likely be strain-specific. Even among Gag sequences, one of the most conserved HIV proteins, there are just under 129,000 distinct naturally found 9-mers found in 4,596 M group sequences, over 60% of which appear only once, an average 18 completely unique 9-mers per natural sequence.

While Epigraphs generally disfavor rare epitopes, we can constrain solutions to strictly avoid them. The Epigraph tool suite[20] directly enables users to investigate the relationship between PTE coverage and rare epitope exclusion. To illustrate this, Epigraph solutions based on global HIV database protein alignments were obtained for HIV proteins Env, Gag, Pol and Nef (Fig. 2). Because Env includes hypervariable regions, inclusion of some rare epitopes is required for Epigraph antigens to create a complete protein: its largest cutoff is $n_o = 2$, so for a complete Env to be generated, some PTEs must be included that are only found three times in the full database. By contrast, Gag, Nef, and Pol antigens can readily be constructed for $n_o > 100$ (that is, the rarest epitope in the vaccine appears in over 100 target population sequences). As Fig. 2 illustrates, this can be accomplished with minimal PTE coverage loss, and thus merits consideration in future vaccine designs. In practice, the vaccine designer can create this graph of coverage versus n_o and based on this trade-off, select a threshold n_o to use for a final Epigraph run.

For practical use, particularly with a large number of input sequences in the target population, we recommend using a cutoff of at least $n_o = 1$, so that each PTE included in the Epigraph sequences appears more than just once in the sample target population. But users are encouraged to use a larger cutoff, as long as there is negligible cost in terms of coverage. Excluding rare variants also speeds up the computation time (Supplemental Table 8). We remark that for sample target populations with only a few target sequences – such as the Ebola set with only 32 distinct protein sequences – we required that epitopes only be present once in the set to be considered for inclusion in the Epigraph antigens, to improve coverage of all variants.

Aligned and unaligned input sequences

The default variant of the Epigraph algorithm does not require that the input sequences in the sample target population be aligned. The k-mer overlap graph depends only on the PTEs that are in the sequences, not on their positions in those sequences.

A variant of Epigraph was developed, however, that uses *aligned* target population sequences for input, and produces antigens on output that are aligned with these input sequences. The extra structure that is imposed on the aligned solution often, though not always, leads to slightly lower PTE coverage scores (see Supplementary Table 9). But an important consequence of this structure is that the aligned variant produces a graph that is, by construction, acyclic; this eliminates the need for a heuristic de-cycling step.

Optimizing on inexact matches

A further advantage of the extra structure imposed by alignment is that it permits other variants of the antigen design algorithms that would be impractical with the open unaligned graph. One such extension is the optimization of coverage by inexact matches. The motivation here is that an antigen epitope may still be cross-reactive with a target epitope, even though they don't exactly match. Instead of maximizing a coverage that gives credit to an antigen PTE only if it exactly matches a corresponding PTE in the sample target population, we modify our criterion to give credit for approximately matching PTEs in the target population. For example, if the antigen includes the 9-mer `VTSSNMNNA`, then it gets credit not only for every appearance of `VTSSNMNNA` in the target population, but also for appearances of `VTSSNMNNC`, `VTSSNMNDA`, *etc.*, which agree with the antigen 9-mer `VTSSNMNNA` in 8/9 of the positions. As we describe in the Supplement, we can optimize on inexact coverage and still give "bonus" credit to exact matches. We employ inexact-match coverage in our design of an Ebola vaccine, described below.

RESULTS

Tailored Therapeutic Vaccines

A tailored therapeutic vaccine (TTV) is intended for a treatment situation in which the patient is already infected (hence, *therapeutic* instead of preventative), and a sample of a patient's infecting viral quasispecies sequence is available (allowing the vaccine to be *tailored* to the patient's specific infection). Given current technology and costs, it is not feasible to create a *de novo* vectored vaccine for every subject. It is feasible, however, to sequence a sample of each patient's virus. Thus the more modest goal addressed here is to manufacture an affordable pool of m distinct vaccine antigens, from which only a subset ($n < m$) is delivered to each patient, with the subset chosen to maximize PTE coverage of the patient's viral sequences by the selected Epigraphs

Given the pool of m sequences, it is straightforward to select the best subset of n for each patient. Since n and m are small, one can quickly consider all possible subsets, and choose the one with maximum coverage of that patient's sequences. The challenge is to construct the pool of m artificial antigens so that these n-out-of-m subsets are optimally effective.

A plausible but suboptimal approach is to create an m-valent Epigraph vaccine to optimize global coverage. A problem with this approach, especially for sequentially derived antigens, is that with each additional sequence, increasingly rare k-mers are included in later sequences, so the first n Epigraphs are typically the best choice for most individuals, offering no increase in coverage using a tailored approach over a simple Epigraph n-valent approach.

We explored several alternative strategies to achieve better coverage of sequences from the population of interest. The best of these, which is detailed in Algorithm 4, employs a clustering strategy. We start with a single antigen sequence $\mathbf{q}_o$ that Epigraph produces by optimizing coverage over the whole sample population set. This will be used as the first sequence in the manufactured set. We then partition the sample population sequences into $m-1$ clusters, with the grouping based on PTE similarity scores (excluding the PTEs that were found in the initial sequence $\mathbf{q}_o$). Epigraph is separately applied to each cluster to obtain a centroid sequence for that cluster; *i.e.,* $\mathbf{q}_i$ is obtained by maximizing the coverage over the sequences in i'th cluster provided by the PTEs in the antigen set $\{\mathbf{q}_o, \mathbf{q}_i\}$. The process is iterative: the population sequence set is re-clustered by re-assigning each sequence to the $\mathbf{q}_i$ that maximally covers the PTEs in that sequence, and the new assignments lead to updated $\mathbf{q}_i$'s, and so on until convergence. Finally, $\{\mathbf{q}_o, \mathbf{q}_1, \ldots, \mathbf{q}_{m-1}\}$ is the set of m antigens that are manufactured.

The outcome can be sensitive to the initial conditions (clustering begins with initial random centroids), so we perform 100 complete runs using different starting sequences, and retain the best one as our solution. Given the number of iteration cycles in

a single run, and the number of repeat runs with different initial conditions, the computational speed of the Epigraph algorithm is critical.

We applied TTV to three potential HIV target sequence populations for treatment studies: 189 contemporary B-clade sequences sampled in the United States, 199 contemporary C clade sequences sampled in Southern Africa, and 2015 Los Alamos HIV database 4596 global M-group Gag sequences (Supplementary Table 5). Gag was used for this pilot study because it is richly populated with beneficial epitopes in natural HIV infection[28], and because SIV Gag responses are well-characterized in the context of RhCMV vector delivery[21,22], and so it is a natural choice for inclusion in a CMV Tailored HIV vaccine. We evaluated p24 separately, because it is the most conserved region within Gag[5].

Fig. 3 illustrates PTE coverage of contemporary B clade US Gag and p24 sequences by bivalent vaccines. B clade Epigraphs are markedly better than any combination of 2 natural B clade strains, while the best coverage was achieved by 2 TTV antigens selected from a pool of 6 (Fig. 3).

Extra epitopes in a vaccine that are not matched in the individual's infection may trigger irrelevant responses, potentially diminishing vaccine-induced beneficial responses[16]. So we also monitored the extra PTEs in the TTVs. For given n, increasing the size of the manufactured pool from $m = 2$ to $m = 6$ increases coverage (by over ten percent for the M-group) without increasing the number of extra PTEs. Compared to the full-length Gag, the conserved-region p24 achieves improved coverage and dramatically reduced extras. (Fig. 3; also see Supplementary Table 5.) However, Gag is ∼500 amino acids in length, while p24 has only ∼230 amino acids, so the increased coverage of p24 comes at the cost of encompassing fewer PTEs. Increasing the number vaccine antigens increases the epitope coverage with diminishing returns, and at the cost of including many more mismatched epitopes. For example, in Gag, increasing the number of tailored antigens from 1 to 2 to 3 increases PTE coverage from 61% to 75% to 79%, but also increases the average number of extra epitopes (in the vaccine but not in the target protein) from 196 to 537 to 773 (Supplementary Table 5).

Filovirus/Ebola

Here we propose an Epigraph solution to a conserved-region pan-filovirus vaccine. The Epigraph tool was used for two purposes: first to define conserved regions within the proteome, and then to design the best combination of antigens within those regions for maximizing PTE coverage. A T-cell based Epigraph design may be particularly useful for viruses in the family *Filoviridae*, because vaccine-elicited T-cell responses to ebolaviruses are protective in non-human primates (NHPs)[29,30], and filoviruses are highly diverse[31–33]. We assembled, annotated and aligned all available filovirus proteomes as we worked the vaccine project, and made the alignments available as part of our new HFV database[31–33].

Viruses in the *Filoviridae* family have caused nearly 50 outbreaks in humans since their discovery in 1967, the most recent of which was the devastating 2014 West African epidemic[34,35]. There are five distinct species in the *Ebolavirus* genus virus: Ebola virus (EBOV), Sudan virus (SUDV), Reston virus (RESTV), Taï Forest virus (TAFV), and Bundibugyo virus (BDBV)[36]. There are two types of virus in the *Marburgvirus* genus: Marburg virus (MARV) and Ravn virus (RAVV)[37]. Lloviu virus (LLOV) is the only known species of the third genus *Cuevavirus*, discovered in bats in the Iberian Peninsula[36].

Most vaccine efforts (reasonably) focus on EBOV, SUDV, and MARV, as these viruses are historically the most frequent causes of these outbreaks[30,38]. There is a high degree of conservation within a species (Fig. 4B), so, for example, a response to any EBOV vaccine would likely be cross-reactive with other EBOV outbreaks. Future outbreaks, however, may result from the re-emergence of a virus from a rare species, or a virus from a new species not yet encountered. There is historical precedent for this: Bundibugyo virus was first identified in a 2007 outbreak and re-emerged in 2012, and Taï Forest virus infected an individual studying a chimpanzee outbreak in 1994. Reston virus has recurrently emerged in primate facilities, is lethal in monkeys, infects pigs, and sometimes causes exposed people to develop antibodies, although to date all people who developed antibodies have been asymptomatic[34]. Thus it may be prudent to develop a vaccine that is potentially effective across all 8 distinctive *Filoviridae* species/variants[39], in parallel with the currently prioritized development of effective vaccines against common outbreak species[40].

We first used PTE coverage by full proteome Epigraphs as a means to define the four most conserved regions across all members of the *Filoviridae* family. This was based on a comprehensive set of full genome sequences[32]. To identify conserved regions, we created an aligned two-antigen Epigraph vaccine solution for the full filovirus proteome, and used local PTE coverage provided by the Epigraph solution as our measure of conservation. We used an 8/9 matched minimum coverage of 80% as the criterion to define conserved regions. We required a minimum contiguous stretch of at least 100 amino acids for inclusion as a conserved region, tolerating short dips in coverage due to diverse PTEs caused by isolated variable positions. Based these criteria, we identified the four most conserved regions in the proteome (Fig. 4A, Supplementary Table 6). The conserved regions collectively span 825 amino acids, which is a reasonable insert size for many vectors.

Having identified conserved regions, our next step was to design the vaccine. Because a future outbreak is most likely to be the result of a virus from one of the common species, we did not want to compromise the cross-reactive potential for those viruses. But, as discussed above, it is also possible that a future outbreak may more closely resemble rare-in-human, or even

as-yet-undiscovered, members of the *Filoviridae* family. More specifically, our criteria are to:

i) preserve PTE coverage of EBOV, the most common species in human outbreaks
ii) maintain excellent PTE coverage of SUDV and MARV, which also have caused recurrent outbreaks; and
iii) given the constraints imposed by the first two criteria, provide extensive PTE coverage of all other Filovirus species.

There are many possible paths to achieve good *Filoviridae* PTE coverage, and we systematically explored the outcomes of different design strategies, including optimization of Epigraph vaccine antigens using the 34 outbreak sequences simultaneously, as well as combinations that used serial optimization, either starting with a natural sequence, or starting with an Epigraph solution based on the 5 representative sequences selected from members of the *Ebolavirus* genus, or on a set of 8 representative sequences that sampled filovirus diversity. We compared PTE coverage based on full proteins that have been commonly used in vaccines, to the coverage based on conserved regions. We also explored the impact of optimization on imperfect matches, and exclusion of rare epitopes. A priori, we didn't know which of these strategies would provide the best solutions, and in these series of comparisons, dozens of Epigraphs runs were conducted and compared; the speed of the Epigraph code enabled a thorough and systematic exploration of design options. In Supplementary Table 7, we show we show coverage results for the subset solutions that we considered of greatest interest. We show the 'B' and 'E' solutions from those tables, with coverage breakdowns for each species, in Fig. 4, as they provide the best 2 and 3 antigen solution given the specific criteria i-iii discussed above. The B solution started with a 5 species *Ebolavirus* Epigraph solution; this was fixed (we call it sequence "a") to enforce good coverage of this historically important genus, the cause of many highly lethal outbreaks. Then a second Epigraph sequence was designed that offered maximum complementary coverage of the full 34 sequence outbreak set relative to sequence "a", for use as a bivalent vaccine. The E solution again started with "a", and two Epigraphs were simultaneously solved that again offered maximum complementary coverage of the full 34 sequence outbreak set relative to "a", this time intended for use as a trivalent vaccine.

We conclude that Epigraph vaccines based on conserved regions in the Ebola proteome suggest a pan-Filoviridae vaccine may be feasible, with the potential to maintain reactivity to the recurrent outbreak strains, while extending cross-reactivity across the known diversity of filoviruses (Fig. 4), and perhaps beyond, to viruses related to the *Filoviradae* family that have not yet been encountered. In particular, a trivalent conserved-region Epigraph vaccine achieves (>90%) PTE coverage of viruses across the *Marburgvirus* and *Ebolavirus* genera, with significant cross-reactive potential against the very distinctive *Cuevavirus* (Fig. 4). In contrast, single natural EBOV Glycoprotein or Nucleoprotein vaccines[30,40] have poor cross-reactive potential with viruses of other species; the multi-modality in the plots in Fig. 4 is due to the fact that within species, even between outbreaks viruses are highly related, but between species and genera sequence distances are much greater. Even combinations of natural antigens have limited potential for cross-reactivity (Fig. 4).

DISCUSSION

Building on the principles used for Mosaic vaccines – namely, the collective design of multiple antigens to maximize PTE coverage – Epigraphs employ a graph-based dynamic programming strategy that is computationally much more efficient and, under restricted conditions, mathematically optimal (as we show in the Methods section). This high performance at low cost expands the design space for novel vaccine approaches. Our Epigraph vaccine design tool suite[20] includes the ability to define and exclude rare epitopes, to use aligned or unaligned input sequences, and to use inexact matches as an optimization criterion.

By maximizing PTE coverage, polyvalent Epigraphs are markedly more efficient than natural sequences in making use of the sequence space available in antigen inserts. The most common forms of epitopes are favored, while rare type-specific epitopes, which are found in virtually every natural HIV strain, are disfavored, and can be explicitly excluded. If a variant of an epitope is already present in one antigen, other antigens in the set will tend to pick up other common variants. Such epitope complementarity in Mosaic antigen sets has been shown experimentally to extend both the breadth and the depth of the vaccine response in animal models[10,13,15]. Also, Epigraphs provide a logical framework for reagent as well as vaccine design: for example, Epigraph sequences could be used as a foundation to design an optimal set of peptides to explore immune responses in a population infected by a variable pathogen, and would have an advantage over a consensus sequence as amino acid positions in an alignment are not considered in isolation, rather the frequency of local combinations of amino acids are considered.

We incorporated Epigraph code into our TTV design algorithm, to use PTE similarity as a foundation for defining clusters of viruses, an immunological perspective, rather than less relevant Hamming or phylogenetic distances. TTVs are therapeutic vaccines that could best cover the population diversity for the purpose of tailoring a vaccine to individual patient's infecting virus within the context of manufacturing a limited and feasible number of antigens. We also explored the idea that in designing TTVs (and more generally, in choosing how many antigens to deliver in any polyvalent vaccine), the benefits of increased PTE coverage should be balanced with the cost of including mismatched (*i.e.,* extra) epitopes in a vaccine. The impact of mismatched epitopes on T-cell responses is not understood, but can now be experimentally evaluated in the context of antigen presentation in RhCMV vectors in NHPs, where the effect of varying these parameters on protective efficacy be tested experimentally.

Previous experiments in NHPs using Mosaic HIV vaccines show that use of polyvalent proteins, computationally designed to maximize PTE coverage, results in vaccine-elicited T-cell responses that have increased cross-reactivity and potency[8,10,11,14,15]. Further, blending conserved region approaches and PTE coverage design may be advantageous[6,16,25]. By analogy, coupling Epigraphs with a conserved region strategy enables vaccine designs with the potential to extend cross-reactivity across *Filoviridae*, which may be important in an uncertain future when the next outbreak virus is not predictable. For the pan-filovirus conserved region design, we used Epigraph to create a local epitope coverage map that provided an immunologically relevant, epitope-centric identification of the most conserved regions in the diverse viral family. The exploration of many possible design options enabled us to identify a vaccine design that has the potential to provide cross-reactive coverage across all of the *Filoviridae* family without compromising the coverage of ebolaviruses and marburgviruses. Of course the vaccine antigen designs we present here are based on predictions of cross-reactive potential, and have yet to be experimentally validated. But given earlier success with Mosaic vaccines in NHPs, and the extent of PTE coverage we observe, these Epigraph designs have promise, and experimental evaluation of these designs is underway. Our new Epigraph algorithm allowed us to solve design problems that were intractable using our first generation Mosaic strategy. We believe the Epigraph code has the potential to aid in discovery of optimal design strategies for other highly variable viruses.

METHODS

Epigraph algorithm

A sample set $\mathsf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_N\}$ of N protein sequences is taken to characterize the variability of a virus over a population that will be targeted for vaccine use. Each PTE is assigned a frequency corresponding to the fraction of sequences in S in which the PTE appears. For example, if $\mathbf{e} = \texttt{VTSSNMNNA}$, and if n is the number of sequences in the sample set S in which the 9-mer VTSSNMNNA appears, then its frequency is given by $f(\mathbf{e}) = n/N$.

We will write $\mathscr{E}(\mathbf{s})$ as the set of PTEs that appear in the sequence $\mathbf{s}$. For example, if $\mathbf{s} = \texttt{VTSSNMNNADSVWLR}\ldots$, then $\mathscr{E}(\mathbf{s}) = \{\texttt{VTSSNMNNA}, \texttt{TSSNMNNAD}, \texttt{SSNMNNADS}, \texttt{SNMNNADSV}, \texttt{NMNNADSVW}, \texttt{MNNADSVWL}, \texttt{NNADSVWLR}, \ldots\}$. For a *set* S of several sequences, $\mathscr{E}(\mathsf{S})$ is the union of the sets $\mathscr{E}(\mathbf{s})$ over all $\mathbf{s} \in \mathsf{S}$; in other words, $\mathbf{e} \in \mathscr{E}(\mathsf{S})$ if and only if $\mathbf{e} \in \mathscr{E}(\mathbf{s})$ for some $\mathbf{s} \in \mathsf{S}$. Even if an epitope $\mathbf{e}$ appears in multiple sequences, it still appears only once in $\mathscr{E}(\mathsf{S})$.

An artificial antigen $\mathbf{q}$ is a sequence that resembles a natural protein but contains PTEs that correspond to the most frequently appearing PTEs in the population sample S. Writing $\mathscr{E}(\mathbf{q})$ as the set of PTEs that appear in $\mathbf{q}$, we say that an antigen $\mathbf{q}$ exhibits good coverage if the $f(\mathbf{e})$'s are large for the $\mathbf{e}$'s in $\mathscr{E}(\mathbf{q})$. More formally, we define

$$\text{Coverage}(\mathbf{q}) = \sum_{\mathbf{e} \in \mathscr{E}(\mathbf{q})} f(\mathbf{e}) \Big/ \sum_{\mathbf{e} \in \mathscr{E}(\mathsf{S})} f(\mathbf{e}). \tag{1}$$

The numerator is the sum of the frequencies of PTEs that appear in $\mathbf{q}$, and the denominator is the sum over *all* PTEs. This coverage corresponds to the total cross-reactive potential of all the epitopes in the vaccine antigens. We don't have a detailed model for how reactive each epitope is, or even for which k-mers are true epitopes; in the face of this uncertainty, we treat all k-mers equally. For a highly immunogenic protein like HIV-1 Gag, T-cell epitopes have been identified in the literature (and summarized in the Los Alamos HIV database[41]) that tile across the entire Gag protein, providing a rationale for this assumption.

A polyvalent vaccine consists of several artificial antigens: $\mathsf{Q} = \{\mathbf{q}_1, \ldots, \mathbf{q}_m\}$. And Coverage($\mathsf{Q}$) is given by Eq. (1) with the sum over $\mathbf{e} \in \mathscr{E}(\mathsf{Q})$.

The main idea in Epigraph is that we can express this formulation as a directed graph (Fig. 1). Each node in the graph corresponds to a distinct k-mer, and a directed edge connects two k-mers $(\mathbf{e}_a, \mathbf{e}_b)$ if they overlap by $k-1$ characters, as illustrated in the Fig. 1(b) inset. We remark that this k-mer overlap graph, which is closely related to a de Bruijn graph[42], is widely used in genome assembly[43,44].

A *path* through the graph is a connected sequence of nodes $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_L$: there is a directed edge from $\mathbf{e}_1$ to $\mathbf{e}_2$, from $\mathbf{e}_2$ to $\mathbf{e}_3$, and so on until the last edge connects $\mathbf{e}_{L-1}$ to $\mathbf{e}_L$. Such a path corresponds to a sequence of $L+k-1$ characters, which defines the artificial antigen $\mathbf{q}$. The coverage associated with that antigen is directly proportional to the sum of the frequencies associated to the nodes in the path: $f(\mathbf{e}_1) + f(\mathbf{e}_2) + \cdots + f(\mathbf{e}_L)$.

For computational convenience, we add Begin and End nodes to the graph, connected respectively to the first and last k characters in each sequence. Epigraph (see Algorithm 1) finds a path P from Begin to End that optimizes the total frequency $\sum_{\mathbf{e} \in P} f(\mathbf{e})$ of epitopes in that path. The algorithm for finding the optimal path is straightforwardly equivalent to well-known algorithms in graph theory[45], and uses dynamic programming, a strategy often used in bioinformatic applications[46,47]. It consists of a forward loop, followed by a backward loop. The forward loop computes $F(\mathbf{e})$ for all the nodes, where $F(\mathbf{e})$ is the maximum total frequency over all paths that end in $\mathbf{e}$. The backward loop builds the path that achieves the maximal score.

Let $\mathsf{P}(\mathbf{e})$ be the set of predecessors of node $\mathbf{e}$: that is, the set of nodes $\mathbf{e}'$ for which there exists a directed edge that connects

from $\mathbf{e}'$ to $\mathbf{e}$. Then we have

$$F(\mathbf{e}) = f(\mathbf{e}) + \max_{\mathbf{e}' \in \mathsf{P}(\mathbf{e})} F(\mathbf{e}') \tag{2}$$

If the set of predecessors $\mathsf{P}(\mathbf{e})$ is empty, then we define $F(\mathbf{e}) = f(\mathbf{e})$. In particular, $F(\texttt{Begin}) = f(\texttt{Begin}) = 0$. If all of the sequences in S contain only amino acid characters, then the `Begin` node will be the only node with no predecessors. If there is a non-amino-acid character (*e.g.,* an '`X`' indicating an ambiguous base call in the DNA sequence, or a '`#`' indicating a frame shift) in any of the sequences, then the PTE immediately after that character might also lack a predecessor. For a directed acyclic graph, there exists a "topological ordering" of the epitopes[45], $\mathbf{e}_1, \mathbf{e}_2, \ldots$, with the property that if $(\mathbf{e}_i, \mathbf{e}_j)$ is a directed edge, then $i < j$. By proceeding in this topological order, we can straightforwardly evaluate Eq. (2) for all the nodes.

Having evaluated $F(\mathbf{e})$ for all the nodes $\mathbf{e}$, we will start at the node $\mathbf{e}^*_0 = \texttt{END}$, and iterate backwards:

$$\mathbf{e}^*_{p+1} = \text{argmax}_{\mathbf{e} \in \mathsf{P}(\mathbf{e}^*_p)} F(\mathbf{e}) \tag{3}$$

If the set $\mathsf{P}(\mathbf{e}^*_p)$ is empty, then $\mathbf{e}^*_p$ has no predecessors, and we are finished: usually, $\mathbf{e}^*_p = \texttt{Begin}$. The sequence of epitopes $\mathbf{e}^*_p, \mathbf{e}^*_{p-1}, \ldots, \mathbf{e}^*_0$ corresponds to a reconstructed sequence $\mathbf{q}$ of $p+k-2$ characters that optimizes the epitope coverage by an intact artificial protein that resembles a natural protein. If the argmax operator does not have a unique value, there are multiple solutions, all equivalent and optimal in the sense of coverage.

If this directed graph has no cycles, then Epigraph finds a path that maximizes Eq. (1), providing a rigorously optimal solution. This optimization is done with computational effort that scales only linearly with the size (as measured in nodes and edges) of the graph. In practice the directed graph created from S may not be acyclic, though it is often very nearly so, especially for larger values of k. For this case, we developed a heuristic scheme to "de-cycle" the graph, by iteratively identifying cycles and then removing low-value edges until no cycles remain (see Algorithm 2 and Methods: De-cycling).

As an aside, we further remark that the logic that defines $F(\mathbf{e})$ in Eq. (2) can be employed to define $x(\mathbf{e})$ for all epitopes in the graph:

$$x(\mathbf{e}) = 1 + \max_{\mathbf{e}' \in \mathsf{P}(\mathbf{e})} x(\mathbf{e}') \tag{4}$$

If $(\mathbf{e}_a, \mathbf{e}_b)$ is a directed edge in the graph, then Eq. (4) guarantees that $x(\mathbf{e}_a) < x(\mathbf{e}_b)$. In particular, if we use $x(\mathbf{e})$ as a horizontal position associated with node $\mathbf{e}$ (which we do in Fig. 1), then we will have the property that all directed edges point from left to right. As with Eq. (2), the definition in Eq. (4) requires that the graph be acyclic.

If a user wishes to exclude rare epitopes, they an do this by selecting a cutoff frequency for exclusion for an Epigraph run. The Epigraph tool suite enables a user to explore of the cost in terms of overall coverage as the cutoff frequency increases (Fig. 2), to make an informed decision regarding the selection of the cutoff value. Epigraph will then eliminating nodes in the graph for which $f(\mathbf{e}) \leq f_o = n_o/N$, where f_o is a cutoff frequency (and n_o is a cutoff count), and N is the number of sequences in the population.

De-cycling

The population of sequences gives rise to a directed graph, but this graph may contain cycles, and the Epigraph algorithm requires that the graph be acyclic. One way cycles can arise is when an identical k-mer is found directly repeated in the same sequence; this is not common, but it does happen. In the Los Alamos HIV database 2014 alignment, out of $N = 4250$ HIV Env sequences, 91 of them (2.1%) have at least one 9-mer directly repeated in a sequence. Similarly 0.3% of Nef, 0.8% of Pol, and 1.3% of Gag sequences, carry such repeats. A more common way for cycles to arise, however, comes from effective repeats of an epitope across multiple sequences.

On the other hand, we have found that, particularly for larger values of k (and larger values of f_o), the graph is often very nearly acyclic, and can be made acyclic with only a few perturbations to the graph. The optimal solution to this perturbed graph is then taken as a nearly-optimal solution to the original graph. The problem of removing the least number of edges to produce an acyclic graph is equivalent to the NP-hard "minimum feedback arc set" problem[48,49]. Thus we use a heuristic approach for making these perturbations; we keep the same nodes from the original graph, but successively cut edges until an acyclic graph remains.

To eliminate cycles, we first have to find cycles, and to help with this task, we decompose the graph into "strongly connected components" (for an acyclic graph, every node is its own strongly connected component) – a task that can be performed in linear time[50]. Within a single strongly connected component, there exists a path from every node to every other node, and this makes cycles easy to find: if $\mathbf{e}_a$ and $\mathbf{e}_b$ are two nodes in the same component, then the directed path from $\mathbf{e}_a$ to $\mathbf{e}_b$ can be merged with the directed path from $\mathbf{e}_b$ to $\mathbf{e}_a$ to form a cycle that includes both $\mathbf{e}_a$ and $\mathbf{e}_b$.

Each time a cycle is located in the graph, we choose one of the edges in the cycle to remove from the graph. This choice is heuristic, but since cutting edges has the effect of isolating nodes, we seek cuts that isolate low-value nodes. For each edge $(\mathbf{e}_a, \mathbf{e}_b)$ we define a value, based on $f(\mathbf{e}_a)$ and $f(\mathbf{e}_b)$; then we choose the edge with the smallest value and remove it from the graph. A very simple and (empirically) effective heuristic is to take the value to be the sum $f(\mathbf{e}_a) + f(\mathbf{e}_b)$. In our experiments, we employed a slight modification of this heuristic. If $\mathbf{e}_a$ is the *sole* predecessor of $\mathbf{e}_b$, then cutting edge $(\mathbf{e}_a, \mathbf{e}_b)$ will isolate node $\mathbf{e}_b$, so we add a further cost of $f(\mathbf{e}_b)$. Similarly, if $\mathbf{e}_b$ is the sole successor to $\mathbf{e}_a$, then we add $f(\mathbf{e}_a)$. See Algorithm 2 for details.

Polyvalent vaccines

In the polyvalent, or "cocktail", version of the problem, we seek $m > 1$ artificial sequences $\mathsf{Q} = \{\mathbf{q}_1, \ldots, \mathbf{q}_m\}$ that collectively maximize the coverage of PTEs in the sample target population S. Typically m is small, only 2 or 3, due to both implementation costs and the biological "cost" of including more rare epitopes as m increases, which may divert the immune response from more useful conserved epitopes. We write $\mathscr{E}(\mathsf{Q}) = \mathscr{E}(\mathbf{q}_1) \cup \cdots \cup \mathscr{E}(\mathbf{q}_m)$ as the set of epitopes that appear in at least one of the sequences in Q, and seek to optimize the sum $\sum_{\mathbf{e} \in \mathscr{E}(\mathsf{Q})} f(\mathbf{e})$ of the frequencies for all the epitopes that appear in Q.

To find a cocktail of $m > 1$ antigens, the Epigraph algorithm is applied in a sequential manner, as shown in Algorithm 3. To see how this works, suppose we have a solution to the m'-sequence problem; to extend this to the $(m'+1)$-sequence problem, we try to optimize the *complementary* coverage, and pick up to the extent possible high-frequency PTEs that were not sampled in the first m' sequences. This is done by setting the f value of the already-covered PTEs to zero. The graph structure is the same, but the update of the cumulative scores is based on these new f values. Although revisiting epitopes from the $\{\mathbf{q}_1, \ldots, \mathbf{q}_{m'}\}$ sequences is allowed if essential to complete the path, it is discouraged because there is no gain in the coverage score. Specifically, define the modified frequency

$$f^*(\mathbf{e}) = \begin{cases} 0 & \text{if } \mathbf{e} \in \mathscr{E}(\mathbf{q}_1) \cup \cdots \cup \mathscr{E}(\mathbf{q}_{m'}) \\ f(\mathbf{e}) & \text{otherwise} \end{cases} \tag{5}$$

and, as in Eq. (2), let $F^*(\mathbf{e}) = f^*(\mathbf{e}) + \max_{\mathbf{e}' \in \mathsf{P}(\mathbf{e})} F^*(\mathbf{e}')$.

Thus, for instance, we can find an $m' = 1$ solution using Epigraph on the original frequencies $f(\mathbf{e})$; then extend to $m' = 2, 3, \ldots, m$ by optimizing complementary coverage at each stage, using the modified $f^*(\mathbf{e})$. The $m = 1+1$ column in Supplementary Table 2 corresponds to this sequential approach.

Once an initial polyvalent solution has been determined, iterative refinement of sequential solutions can improve the final coverage. Given initial sequences $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_m$, we can go back and recompute a new solution for $\mathbf{q}_1$. This is done by starting with the original frequency values for each of the epitopes, but setting to zero those epitopes that are covered by $\mathbf{q}_2, \ldots, \mathbf{q}_m$. The optimization of this complementary coverage problem leads to a new $\mathbf{q}_1$. One can loop through all of the initial solutions this way, each time optimizing the appropriate complementary coverage.

The iterative refinement scheme can also be applied to other initial conditions; *e.g.,* one can use a consensus, natural or Mosaic solution as an initial sequence. Supplementary Table 4 shows how a Mosaic solution can be improved by using the Mosaic solution as a starting place for an iterative Epigraph refinement. If Mosaic antigens are allowed to evolve for many generations, they may eventually evolve to a solution that is better in terms of PTE coverage than a first-pass Epigraph solution. But even these solutions might be improved with iterative Epigraph refinement.

Multiple trials can also be used to improve coverage. Here, instead of using Epigraph to obtain an $m' = 1$ solution, use a random sequence for $\mathbf{q}_1$. With this as a starting point, sequentially add new sequences $\mathbf{q}_2, \ldots, \mathbf{q}_m$, followed by iterative refinement until convergence is achieved. We can do this for many random initial sequences, and keep the solution that gives the best coverage. Iterative refinements with multiple trials were used for the $m = 2$ column in Supplementary Table 2.

References

1. Fischer, W. *et al.* Transmission of single HIV-1 genomes and dynamics of early immune escape revealed by ultra-deep sequencing. *PLoS ONE* **5**, e12303 (2010).
2. Liu, M. K. *et al.* Vertical T cell immunodominance and epitope entropy determine HIV-1 escape. *J. Clinical Investigation* **123**, 380–393 (2013).
3. Bar, K. J. *et al.* Early low-titer neutralizing antibodies impede HIV-1 replication and select for virus escape. *PLoS Pathogens* **8**, e1002721 (2012).
4. Liao, H. X. *et al.* Co-evolution of a broadly neutralizing HIV-1 antibody and founder virus. *Nature* **496**, 469–476 (2013).
5. Korber, B., Letvin, N. L. & Haynes, B. F. T cell vaccine strategies for human immunodeficiency virus, the virus with a thousand faces. *J. Virology* **83**, 8300–14 (2009).

6. Stephenson, K. E. *et al.* Full-length HIV-1 immunogens induce greater magnitude and comparable breadth of T lymphocyte responses to conserved HIV-1 regions compared with conserved-region-only HIV-1 immunogens in rhesus monkeys. *J. Virology* **86**, 11434–11440 (2012).
7. Gaschen, B. *et al.* Diversity considerations in HIV-1 vaccine selection. *Science* **296**, 2354–60 (2002).
8. Hulot, S. L. *et al.* Comparison of Immunogenicity in Rhesus Macaques of Transmitted-Founder, HIV-1 Group M Consensus, and Trivalent Mosaic Envelope Vaccines Formulated as a DNA Prime, NYVAC, and Envelope Protein Boost. *J. Virology* **89**, 6462–6480 (2015).
9. Fischer, W. *et al.* Polyvalent vaccines for optimal coverage of potential T cell epitopes in global HIV-1 variants. *Nature Medicine* **13**, 100–106 (2007).
10. Barouch, D. *et al.* Mosaic HIV-1 vaccines expand the breadth and depth of cellular immune responses in rhesus monkeys. *Nature Medicine* **16**, 319–323 (2010).
11. Barouch, D. *et al.* Protective efficacy of a global HIV-1 Mosaic vaccine against heterologous SHIV challenges in rhesus monkeys. *Cell* **155**, 531–539 (2013).
12. Ndhlovu, Z. M. *et al.* Mosaic HIV-1 Gag antigens can be processed and presented to human HIV-specific CD8+ T cells. *J. Immunology* **186**, 6914–6924 (2011).
13. Santra, S. *et al.* Mosaic vaccines elicit CD8+ T lymphocyte responses that confer enhanced immune coverage of diverse HIV strains in monkeys. *Nature Medicine* **16**, 324–8 (2010).
14. Santra, S. *et al.* Breadth of cellular and humoral immune responses elicited in rhesus monkeys by multi-valent Mosaic and consensus immunogens. *Virology* **428**, 121–127 (2012).
15. Abdul-Jawad, S. *et al.* Increased valency of conserved-mosaic vaccines enhances the breadth and depth of epitope recognition. *Molecular Therapy* (2015). doi:10.1038/mt.2015.210.
16. Yang, O. O. *et al.* Short Conserved Sequences of HIV-1 Are Highly Immunogenic and Shift Immunodominance. *J. Virology* **89**, 1195–1204 (2015).
17. Yusim, K. *et al.* Hepatitis C genotype 1 Mosaic vaccines are immunogenic in mice and induce stronger T-cell responses than natural strains. *Clinical and Vaccine Immunology* **20**, 302–305 (2013).
18. Fenimore, P. W. *et al.* Designing and testing broadly-protective filoviral vaccines optimized for cytotoxic T-lymphocyte epitope coverage. *PLoS ONE* **7**, e44769 (2012).
19. Kamlangdee, A., Kingstad-Bakke, B., Anderson, T. K., Goldberg, T. L. & Osorio, J. E. Broad protection against avian influenza virus by using a modified vaccinia Ankara virus expressing a Mosaic hemagglutinin gene. *J. Virology* **88**, 13300–13309 (2014).
20. Epigraph Tool Suite. http://www.hiv.lanl.gov/content/sequence/EPIGRAPH/epigraph.html (2016) (Date of access: 20 June 2016).
21. Hansen, S. G. *et al.* Cytomegalovirus vectors violate CD8+ T cell epitope recognition paradigms. *Science* **340**, 1237874 (2013).
22. Hansen, S. G. *et al.* Immune clearance of highly pathogenic SIV infection. *Nature* **502**, 100–104 (2013).
23. Létourneau, S. *et al.* Design and pre-clinical evaluation of a universal HIV-1 vaccine. *PLoS ONE* **2**, e984 (2007).
24. Kulkarni, V. *et al.* Altered response hierarchy and increased T-cell breadth upon HIV-1 conserved element DNA vaccination in macaques. *PLoS ONE* **9**, e86254 (2014).
25. Ondondo, B. *et al.* Novel Conserved-region T-cell Mosaic Vaccine With High Global HIV-1 Coverage Is Recognized by Protective Responses in Untreated Infection. *Molecular Therapy* **24**, 832–842 (2016).
26. Mosaic explanation. http://www.hiv.lanl.gov/content/sequence/MOSAIC/mosaic_explanation.html (2012) (Date of access: 20 June 2016).
27. HIV alignments. http://www.hiv.lanl.gov/content/sequence/NEWALIGN/align.html (2016) (Date of access: 20 June 2016).
28. Mothe, B. *et al.* Definition of the viral targets of protective HIV-1-specific T cell responses. *J. Translational Medicine* **9**, 208 (2011).
29. Sullivan, N. J. *et al.* CD8+ cellular immunity mediates rAd5 vaccine protection against Ebola virus infection of nonhuman primates. *Nature Medicine* **17**, 1128–1131 (2011).

30. Zhou, Y. & Sullivan, N. J. Immunology and evolvement of the adenovirus prime, MVA boost Ebola virus vaccine. *Current Opinion in Immunology* **35**, 131–136 (2015).

31. Carroll, S. A. *et al.* Molecular evolution of viruses of the family Filoviridae based on 97 whole-genome sequences. *J. Virology* **87**, 2608–2616 (2013).

32. Yusim, K. *et al.* Integrated sequence and immunology filovirus database at Los Alamos. *Database (Oxford)* **2016** (2016).

33. Filovirus and HFV Sequence Alignments. http://hfv.lanl.gov/content/sequence/NEWALIGN/align.html (2015) (Date of access: 20 June 2016).

34. Outbreaks Chronology: Ebola Virus Disease. http://www.cdc.gov/vhf/ebola/outbreaks/history/chronology.html (2016) (Date of access: 20 June 2016).

35. Marburg Hemorrhagic Fever Distribution Map. http://www.cdc.gov/vhf/marburg/resources/distribution-map.html (2014) (Date of access: 20 June 2016).

36. Negredo, A. *et al.* Discovery of an ebolavirus-like filovirus in Europe. *PLoS Pathogens* **7**, e1002304 (2011).

37. Kuhn, J. H. *et al.* Virus nomenclature below the species level: a standardized nomenclature for filovirus strains and variants rescued from cDNA. *Arch. Virology* **159**, 1229–1237 (2014).

38. Kuhn, J. H. *et al.* Evaluation of perceived threat differences posed by filovirus variants. *Biosecurity and Bioterrorism* **9**, 361–371 (2011).

39. Holtsberg, F. W. *et al.* Pan-ebolavirus and Pan-filovirus Mouse Monoclonal Antibodies: Protection against Ebola and Sudan Viruses. *J. Virology* **90**, 266–278 (2015).

40. Henao-Restrepo, A. M. *et al.* Efficacy and effectiveness of an rVSV-vectored vaccine expressing Ebola surface glycoprotein: interim results from the Guinea ring vaccination cluster-randomised trial. *Lancet* (2015).

41. Gag CTL/CD8+ Epitope Map. http://www.hiv.lanl.gov/content/immunology/maps/ctl/Gag.html (2016) (Date of access: 11 August 2016).

42. de Bruijn, N. G. A combinatorial problem. *Proc. Koninklijke Nederlandse Akademie van Wetenschappen. Series A* **49**, 758–764 (1946).

43. Pevzner, P. A., Tang, H. & Waterman, M. S. An Eulerian path approach to DNA fragment assembly. *Proc. National Academy of Science* **98**, 9748–9753 (2001).

44. Compeau, P. E. C., Pevzner, P. A. & Tesler, G. How to apply de Bruijn graphs to genome assembly. *Nature Biotechnology* **29**, 987–991 (2011).

45. Gross, J. L., Yellen, J. & Zhang, P. *Handbook of Graph Theory* (CRC Press, Taylor and Francis Group, Boca Raton, FL, 2014), 2nd edn.

46. Smith, T. F. & Waterman, M. S. Identification of common molecular subsequences. *J. Molecular Biology* **147**, 195–197 (1981).

47. Giegerich, R. A systematic approach to dynamic programming in bioinformatics. *Bioinformatics* **16**, 665–677 (2000).

48. Garey, M. & Johnson, D. *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W. H. Freeman, New York, 1979).

49. Chen, J., Liu, Y., Lu, S., O'Sullivan, B. & Razgon, I. A fixed-parameter algorithm for the directed feedback vertex set problem. *J. ACM* **55**, 21:1–21:19 (2008).

50. Tarjan, R. E. Depth-first search and linear graph algorithms. *SIAM J. Computing* **1**, 146–160 (1972).

51. Hagberg, A. A., Schult, D. A. & Swart, P. J. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, 11–15 (Pasadena, CA USA, 2008).

Acknowledgments

This work was funded through: National Institute of Allergy and Infectious Diseases awards NIAID 2 R44 AI100343-02, and CHAVI-ID; UM1-AI100645, the Center for HIV/AIDS Vaccine Immunology and Immunogen Discovery; The Bill and Melinda Gates Foundation Global Health Proposal OPP1108533; and Royalty Funds from Los Alamos National Laboratory.

We thank Goutam Gupta, Jose Olivares, and Jurgen Schmidt (LANL) for their efforts to make the Ebola work possible through internal Los Alamos funding. We thank Tomas Hanke and Andrew McMichael (Oxford), and Barton Haynes and Larry Liao (Duke), for thoughtful advice regarding the design of the pan-Filovirus T-cell vaccine. We thank Aric Hagberg, Misha Chertkov, and Diane Oyen for useful discussions about graph-theoretic algorithms, and the authors of the NetworkX library for making the source code freely available.

Author Contributions

JT and BK developed the concepts and designed the vaccines. JT wrote the Epigraph software, HJ and BK helped with software quality control, and HJ effected migration to the website. KY assembled the Ebola sequence metadata. KF and LJP introduced the tailored therapeutic vaccine idea (which was the original inspiration for a more cost-effective way to optimize epitope coverage), and contributed to consideration of biological issues. JT and BK primarily wrote the paper, with input from all authors.

Competing Financial Interest

Authors KF, LJP, BK, and JT are co-inventors on a patent application (PCT/US15/54067) on "HIV Vaccines Comprising One or More Population Episensus Antigens." Authors LJP and KF have a significant financial interest in TomegaVax Inc., a company that may have a commercial interest in the results of this research and technology. This potential individual and institutional conflict of interest has been reviewed and managed by OHSU.

ALGORITHMS AND FIGURES

Algorithm 1 Epigraph: FIND OPTIMAL PATH THROUGH A GRAPH OF EPITOPES

Require: Directed Acyclic Graph G, including
two nodes labeled `Begin` and `End`, and at least one path connecting them
a function $\mathsf{P}(\mathbf{e})$ that specifies the predecessors to node $\mathbf{e}$
a function $f(\mathbf{e})$ that specifies frequency of epitope $\mathbf{e}$ in the population
a topological ordering of nodes: $\mathbf{e}_0, \mathbf{e}_1, \ldots, \mathbf{e}_{N+1}$; with $\mathbf{e}_0 = \texttt{Begin}$ and $\mathbf{e}_{N+1} = \texttt{End}$
▷ *Topological ordering implies: if* $(\mathbf{e}_j, \mathbf{e}_i)$ *is a directed edge, then* $j < i$
▷ *Equivalently: if* $\mathbf{e}_j \in \mathsf{P}(\mathbf{e}_i)$*, then* $j < i$

$F(\mathbf{e}_0) \leftarrow 0$ ▷ *Initialize*
for $i = 1 \ldots N$ **do** ▷ *Forward loop*
  $F(\mathbf{e}_i) \leftarrow f(\mathbf{e}_i) + \max_{\mathbf{e}' \in \mathsf{P}(\mathbf{e}_i)} F(\mathbf{e}')$ ▷ $F(\mathbf{e})$ *is sum of* $f(\mathbf{e}')$
  ▷ *for* $\mathbf{e}'$ *in best path that ends at node* $\mathbf{e}$
$\mathbf{e}^*_0 \leftarrow \texttt{End}$ ▷ *Start at* `End` *and work backwards*
for $p = 0, 1, 2, \ldots$ **do** ▷ *Backward loop*
  $\mathbf{e}^*_{p+1} \leftarrow \operatorname{argmax}_{\mathbf{e} \in \mathsf{P}(\mathbf{e}^*_p)} F(\mathbf{e})$ ▷ $\mathbf{e}^*_{p+1}$ *is best predecessor of node* $\mathbf{e}^*_p$
  if $\mathbf{e}^*_{p+1} = \texttt{Begin}$ **then**
    return $[\mathbf{e}^*_p, \mathbf{e}^*_{p-1}, \ldots, \mathbf{e}^*_1]$ ▷ *Return optimal path*

Algorithm 2 Decycle: REMOVE ALL CYCLES FROM A GRAPH

Require: A directed graph G, including:
a function $\mathsf{S}(\mathbf{e})$ that specifies the successors to node $\mathbf{e}$, and
a function $\mathsf{P}(\mathbf{e})$ that specifies the predecessors to node $\mathbf{e}$
a function $f(\mathbf{e})$ that specifies frequency of epitope $\mathbf{e}$ in the population
Require: Functions STRONGLY_CONNECTED_COMPONENTS and SHORTEST_PATH,
both are provided by `NetworkX` software package[51]

repeat
  $J \leftarrow$ STRONGLY_CONNECTED_COMPONENTS(G)
    ▷ *J is a list of all components; each component is a list of nodes in G*
  $J \leftarrow J - \{j \in J, \text{such that } |j| = 1\}$
    ▷ *Discard all single-node components – no cycles there!*
  for all $j \in J$ **do** ▷ *For each component*
    repeat
      Choose $(a,b) \in j$ ▷ *Randomly choose two nodes from the selected component*
      $C \leftarrow$ CYCLEFROMTWONODES(G, a, b)
      if $C \neq \emptyset$ **then**
        $(\mathbf{e}_a, \mathbf{e}_b) \leftarrow$ WEAKEDGEINCYCLE(G, C)
        Remove edge $(\mathbf{e}_a, \mathbf{e}_b)$ from G
    until $C = \emptyset$
until $J = \emptyset$ ▷ *G is acyclic; we are done.*

procedure CYCLEFROMTWONODES(G, a, b)
  $P_{ab} \leftarrow$ SHORTEST_PATH(G, a, b)
  $P_{ba} \leftarrow$ SHORTEST_PATH(G, b, a)
  if either call to SHORTEST_PATH fails **then**
    $C \leftarrow \emptyset$
  else
    $C \leftarrow P_{ab} + P_{ba}$ ▷ *Merge two paths into a cycle*
  return C

procedure WEAKEDGEINCYCLE(G, C)
  Write C as a list of nodes $[\mathbf{e}_1, \ldots, \mathbf{e}_k]$ and edges $[(\mathbf{e}_1, \mathbf{e}_2), (\mathbf{e}_2, \mathbf{e}_3), \ldots, (\mathbf{e}_k, \mathbf{e}_1)]$
  for all $(\mathbf{e}_i, \mathbf{e}_j)$ in $[(\mathbf{e}_1, \mathbf{e}_2), (\mathbf{e}_2, \mathbf{e}_3), \ldots, (\mathbf{e}_k, \mathbf{e}_1)]$ **do**
    $v_{ij} \leftarrow f(\mathbf{e}_i) + f(\mathbf{e}_j)$ ▷ *v is heuristic "value" of edge*
    $v_{ij} \leftarrow v_{ij} + f(\mathbf{e}_i)$ **if** $|\mathsf{S}(\mathbf{e}_i)| = 1$ ▷ *Add value if cutting edge would isolate $\mathbf{e}_i$*
    $v_{ij} \leftarrow v_{ij} + f(\mathbf{e}_j)$ **if** $|\mathsf{P}(\mathbf{e}_j)| = 1$ ▷ *Add value if cutting edge would isolate $\mathbf{e}_j$*
  Let $i_o, j_o \leftarrow \text{argmin } v_{ij}$
  return $(\mathbf{e}_{i_o}, \mathbf{e}_{j_o})$ ▷ *return lowest-value edge in cycle*

Algorithm 3 Cocktail: FIND (AND ITERATIVELY REFINE) A SET OF m ANTIGENS

Require: Directed Acyclic Graph G, including
a function $f(\mathbf{e})$ that specifies frequency of epitope $\mathbf{e}$ in the population
Require: Function EPIGRAPH(G, f) that returns a sequence $\mathbf{q}$, corresponding to a path through the graph G that maximizes $\sum_{\mathbf{e}\in\mathscr{E}(\mathbf{q})} f(\mathbf{e})$ ▷ *See Algorithm 1*

$\mathsf{Q} \leftarrow \emptyset$
$f^*(\mathbf{e}) \leftarrow f(\mathbf{e})$ for all epitopes $\mathbf{e} \in \mathsf{G}$ ▷ *Initialize*
for $n = 1 \ldots m$ **do** ▷ *Sequential solution*
  $\mathbf{q}_n \leftarrow$ EPIGRAPH(G, f^*) ▷ *Compute next antigen sequence* $\mathbf{q}_n$
  $\mathsf{Q} \leftarrow \mathsf{Q} \cup \{\mathbf{q}_n\}$ ▷ *Add* $\mathbf{q}_n$ *to vaccine*
  for $\mathbf{e} \in \mathscr{E}(\mathbf{q}_n)$ **do** ▷ *Now that* $\mathbf{e}$ *is in the vaccine*
    $f^*(\mathbf{e}) \leftarrow 0$ ▷ *No credit for including* $\mathbf{e}$ *in subsequent antigens*

▷ *At this point,* $f^*(\mathbf{e}) = 0$ *for all* $\mathbf{e} \in \mathscr{E}(\mathsf{Q})$

repeat ▷ *Iterative Refinement (optional)*
  for $n = 1 \ldots m$ **do**
    $\mathsf{Q} \leftarrow \mathsf{Q} - \{\mathbf{q}_n\}$ ▷ *Remove sequence* $\mathbf{q}_n$ *from vaccine*
    for $\mathbf{e} \in \mathscr{E}(\mathsf{Q}) - \mathscr{E}(\mathbf{q}_n)$ **do** ▷ *With* $\mathbf{e}$ *not in the vaccine anymore,*
      $f^*(\mathbf{e}) \leftarrow f(\mathbf{e})$ ▷ $f^*(\mathbf{e})$ *gives credit for including* $\mathbf{e}$ *in subsequent antigens*
    $\mathbf{q}_n \leftarrow$ EPIGRAPH(G, f^*) ▷ *Compute replacement for old sequence* $\mathbf{q}_n$
    $\mathsf{Q} \leftarrow \mathsf{Q} \cup \{\mathbf{q}_n\}$ ▷ *Add* $\mathbf{q}_n$ *to vaccine*
    $f^*(\mathbf{e}) \leftarrow 0$ for all $\mathbf{e} \in \mathscr{E}(\mathbf{q}_n)$
until no change in $\mathbf{q}_1, \ldots, \mathbf{q}_m$

return cocktail of m sequences: $\mathsf{Q} = \{\mathbf{q}_1, \ldots, \mathbf{q}_m\}$

Algorithm 4 TTV: TAILORED THERAPEUTIC VACCINE

Require: Sequence set $\mathsf{S} = \{\mathbf{s}_1, \ldots, \mathbf{s}_N\}$,
Require: Directed Acyclic Graph, G
Require: Function EPIGRAPH(G, f) that returns a sequence $\mathbf{q}$, corresponding to a path through the graph G that maximizes $\sum_{\mathbf{e}\in\mathscr{E}(\mathbf{q})} f(\mathbf{e})$ ▷ *See Algorithm 1*

Define $u(\mathbf{s}, \mathbf{e}) = 1$ if epitope $\mathbf{e}$ appears in sequence $\mathbf{s}$ ▷ *i.e., if* $\mathbf{e} \in \mathscr{E}(\mathbf{s})$
$f(\mathbf{e}) \leftarrow (1/N) \sum_{\mathbf{s}\in\mathsf{S}} u(\mathbf{s}, \mathbf{e})$ ▷ *Frequency of epitope* $\mathbf{e}$ *in sequence set* S
$\mathbf{q}_o \leftarrow$ EPIGRAPH(G, f) ▷ $\mathbf{q}_o$ *is centroid of the full data set* S

Randomly select $m-1$ sequences from S; call them $\mathbf{q}_1, \ldots, \mathbf{q}_{m-1}$.
repeat
  Initialize: $\mathsf{S}_1 = \cdots = \mathsf{S}_{m-1} = \emptyset$
  for $i = 1 \ldots N$ **do** ▷ *Loop over sequences*
    for $n = 1 \ldots m-1$ **do** ▷ *Loop over centroids*
      $c_{in} \leftarrow \sum_{\mathbf{e}\in\mathscr{E}(\mathbf{s}_i)} u(\{\mathbf{q}_o, \mathbf{q}_n\}, \mathbf{e})$ ▷ *Coverage of sequence* $\mathbf{s}_i$ *by antigens* $\{\mathbf{q}_o, \mathbf{q}_n\}$
      ▷ *Here,* $u(\{\mathbf{q}_o, \mathbf{q}_n\}, \mathbf{e}) = \max[u(\mathbf{q}_o, , \mathbf{e}), u(\mathbf{q}_n, \mathbf{e})]$
    $n \leftarrow \text{argmax}_{n'} c_{in'}$
    $\mathsf{S}_n \leftarrow \mathsf{S}_n \cup \{\mathbf{s}_i\}$ ▷ *Put* $\mathbf{s}_i$ *into cluster* S_n
  for $n = 1 \ldots m-1$ **do** ▷ *Loop over clusters*
    $f^{(n)}(\mathbf{e})(1/N) \leftarrow \sum_{\mathbf{s}\in\mathsf{S}_n} u(\mathbf{s}, \mathbf{e})$ ▷ *Frequency of* $\mathbf{e}$ *in sequence cluster* S_n
    $f^{(n)}(\mathbf{e}) \leftarrow 0$ for all $\mathbf{e} \in \mathscr{E}(\mathbf{q}_o)$ ▷ *No credit for epitopes already covered by* $\mathbf{q}_o$
    $\mathbf{q}_n \leftarrow$ EPIGRAPH$(\mathsf{G}, f^{(n)})$ ▷ *New centroid for* S_n
until convergence
return tailored therapeutic vaccine: $\mathbf{q}_o, \mathbf{q}_1, \ldots, \mathbf{q}_{m-1}$

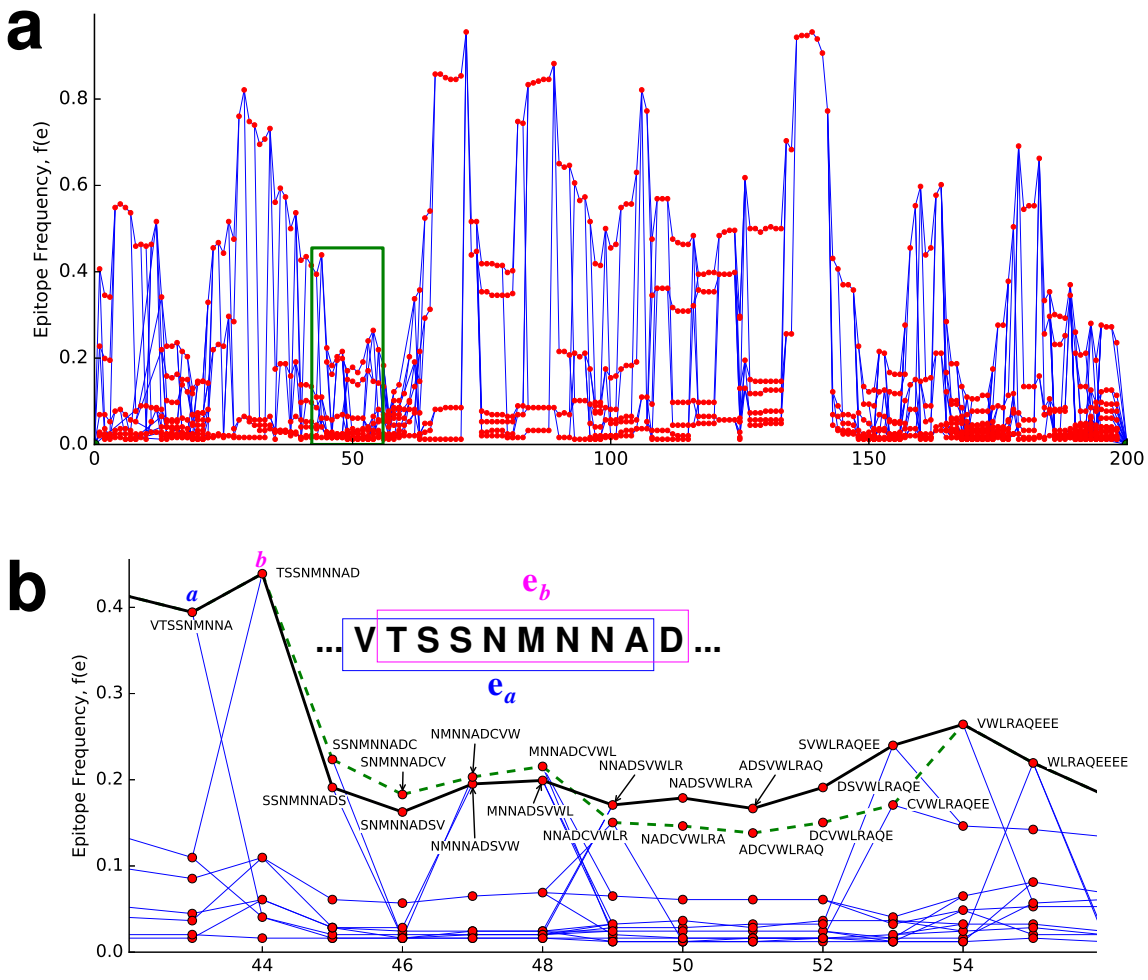

Figure 1. **(a)** Full graph for the CRF01-AE clade of the Nef protein. The green rectangle is an inset shown in (B). Nodes are red dots, and represent each k-mer variant, with $k = 9$. The edges are thin blue lines that connect epitopes whose sequences overlap by $k-1$ amino acids, as shown for the first two epitopes ($\mathbf{e}_a$ = VTSSNMNNA, $\mathbf{e}_b$ = TSSNMNNAD) in the upper left of (B). Although the topological properties of the graph do not depend on the node positions, this plot uses the vertical axis to indicate epitope frequency in the target sequence set, $y = f(\mathbf{e})$, for each node. The horizontal position of the nodes is chosen so that all directed edges connect from left to right. The ideal path through this graph keeps as much as possible to the largest y-values; this path defines a protein sequence that maximizes epitope coverage of the population. **(b)** The inset shows two paths through the nodes. The solid black line is the optimal path, and corresponds to the sequence VTSSNMNNAD S VWLRAQEEEE while the dashed green corresponds to VTSSNMNNAD C VWLRAQEEEE. The dashed line achieves higher $f(\mathbf{e})$ values on 4 nodes, but the solid line has higher $f(\mathbf{e})$ for 5 nodes, and $\sum f(\mathbf{e})$ is higher. Note there is no path that includes the highest-valued nodes for all horizontal positions.

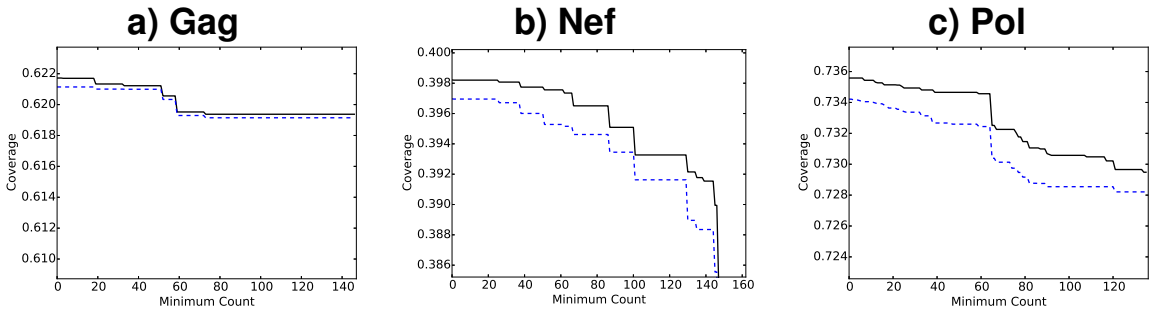

Figure 2. Excluding rare epitopes. We see that excluding rare variants decreases the coverage, but only slightly. Coverage of polyvalent ($m = 2$) solutions is shown as a function of minimum count n_o. These graphs are created by sequentially increasing n_o and eliminating all nodes $\mathbf{e}$ from the graph for which $f(\mathbf{e}) \leq f_o = n_o/N$, where N is the number of sequences in the sample population set. This continues until the maximum n_o is achieved for which a path still exists from `Begin` to `End`. Note that this maximum value can be computed directly from the graph, before this sequential process is employed. Blue dashed lines correspond to coverage given by the direct sequential algorithm; the black solid lines are based on the best solutions after 100 random restarts. To facilitate comparison, the vertical axis, in all three plots, is restricted to a range of 0.015.

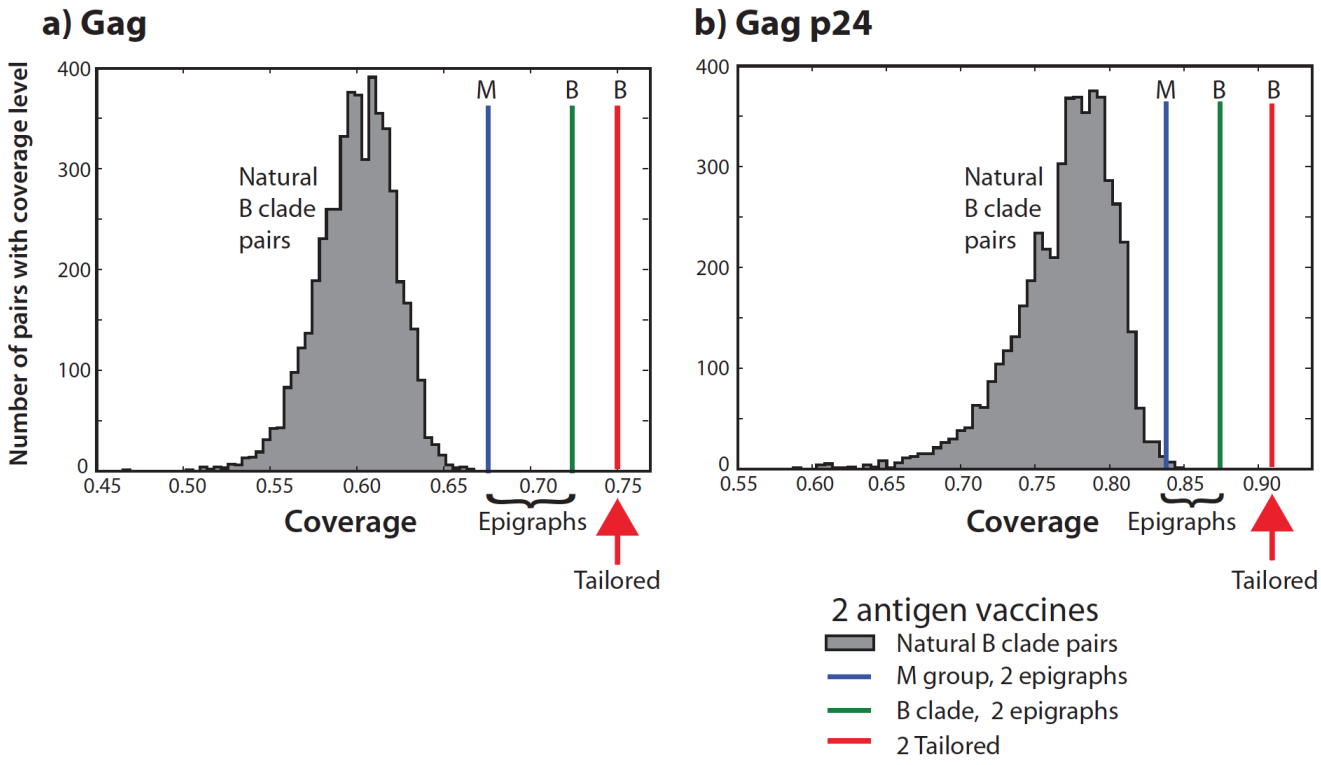

Figure 3. Two-antigen vaccine coverage. Comparisons illustrating the average epitope coverage per sequence of 189 B clade sequences isolated in the United States within the last decade, considered as a hypothetical target population for a tailored therapeutic vaccine (TTV). To illustrate PTE coverage using a pair of natural within-B clade sequences as vaccine antigens, 5000 randomly selected pairs of natural B clade sequences (gray) were evaluated as potential vaccines, and the distribution of average coverage of the sequences by natural pairs of antigens is shown in the gray histogram. This is compared to the average coverage provided by a two-antigen set of M group Epigraphs (M database, blue), a two-antigen set of global B clade Epigraphs (B database, green), and a US B clade TTV where the $n = 2$ best matches from a set of $m = 6$ representative Epigraphs for manufacture were chosen as a "tailored" match for each of the 189 natural B clade US sequences. The TTV antigens provide the best matches. Of note, the global M group two-antigen Epigraph solutions perform better than two natural B clade Gag proteins even in a within-clade setting, and the M group Epigraphs have the potential for a global response at or near this level of PTE coverage across all clades. **(A)** The comparisons for the full Gag protein, **(B)** The comparisons for only the conserved p24 region.

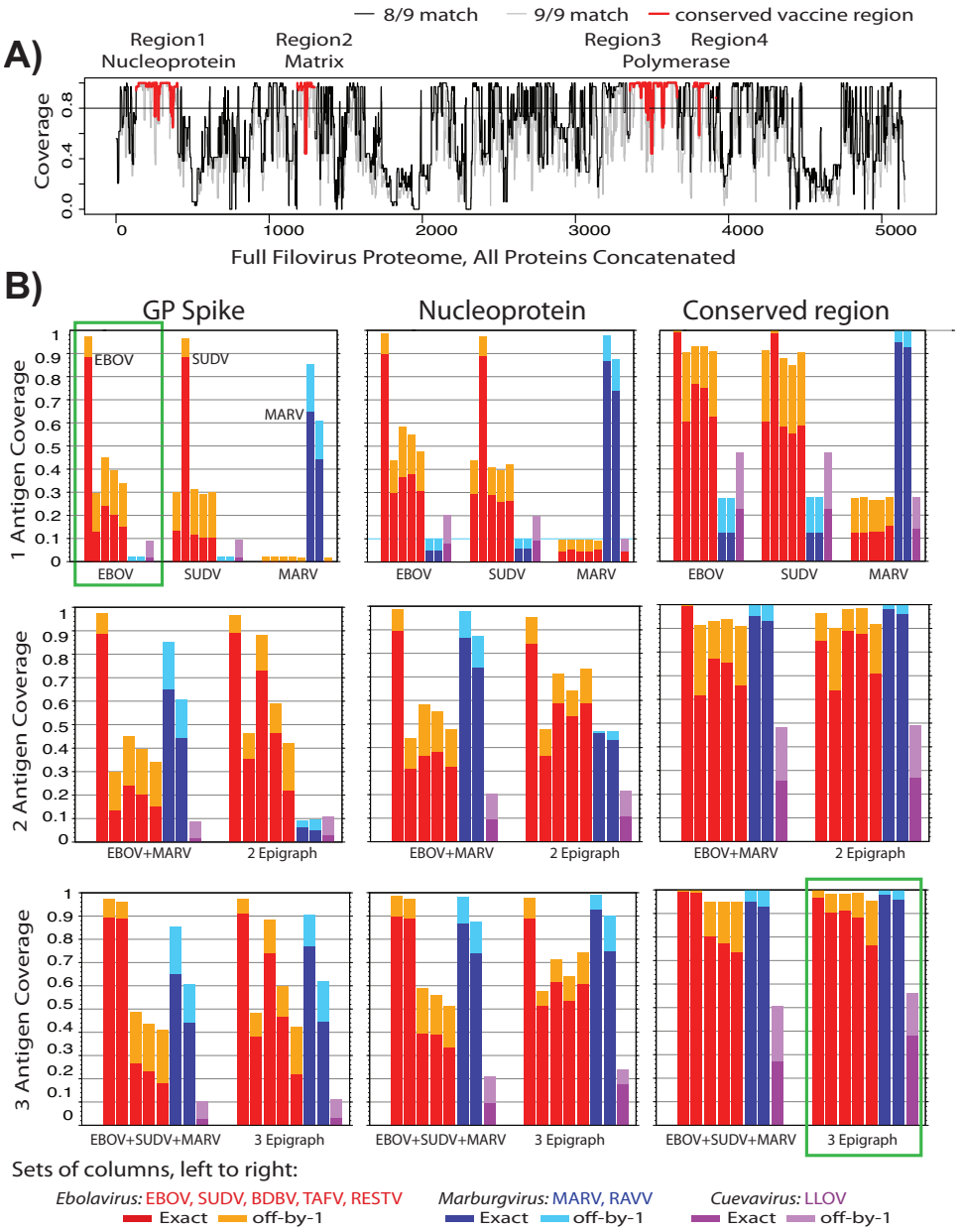

Figure 4: **Ebola Epigraphs.** caption on next page

Figure 4. Ebola Epigraphs. **(A)** PTE Epigraph coverage of Ebola relative to a full proteome alignment, including one representative sequence per human outbreak. All 7 proteins in the Filovirus proteome (excluding soluble GPs) were concatenated, 2 Epigraph sequences were generated spanning the full proteome, and these were used to identify the most conserved regions in the proteome based on PTE coverage, highlighted in red. The black line shows 8/9 coverage, the gray line the 9/9, of the population by the 2 Epigraphs, for each consecutive 9-mer in the alignment. The four highly conserved regions together span 825 amino acids. **(B)** PTE coverage of Filovirus species by different vaccine options. The natural vaccine candidates used were the reference strains EBOV Yambuku-Mayinga, NC_002549; SUDV Gulu, NC_006432; and MARV Mt. Elgon-Musoke, NC_001608. (The four-letter uppercase species names the use standard nomenclature, described in the text.) Columns represent the average PTE coverage for a given species, ordered left-to-right according to the legend, for different vaccine options. Deeper colors show 9/9 PTE matches, lighter colors the added coverage by 8/9 matches. *Ebolavirus* genus species are red, *Marburgvirus* blue, and *Cuevavirus* purple. There is a high level of PTE coverage within-species. Vaccines being evaluated in West Africa use a natural EBOV GP antigen[30,40], and PTE coverage would be excellent for other EBOV strains, but poor for other species (green box, top left). In contrast, a three-antigen conserved-region Epigraph has excellent coverage across all known sequences sampled from *Filoviridae* (green box, bottom right).

Supplementary Material

Epigraph: A Vaccine Design Tool Applied to an HIV Therapeutic Vaccine and a Pan-Filovirus Vaccine

James Theiler[1,2], Hyejin Yoon[1], Karina Yusim[1,2], Louis J. Picker[3], Klaus Früh[3], and Bette Korber[1,2]

[1]Los Alamos National Laboratory, Los Alamos, NM 87545, USA
[2]New Mexico Consortium, Los Alamos, NM 87544, USA
[3]Oregon Health and Science University, Portland, OR 97239, USA

Aligned Sequences

For aligned sequences, the positions $t = 1 \ldots T$ are well-defined for each amino acid character, and the artificial sequence $\mathbf{q}$ that we wish to construct will also be of length T. We write $\mathbf{s}_n[t]$ as the t'th character in the n'th sequence. It will also be useful to introduce the notation $\mathbf{s}[t:u]$ for the subsequence of $\mathbf{s}$ that begins at position t and ends at position $u-1$. We define a new frequency function that depends on position: $f(t,\mathbf{e})$ is the fraction of sequences in S for which $\mathbf{e} = \mathbf{s}[t:t+k]$. The "coverage" of a sequence $\mathbf{q}$ is given by

$$\frac{1}{T-k}\sum_{t=1}^{T-k} f(t,\mathbf{q}[t:t+k]). \tag{S-1}$$

For the aligned-sequence problem, the nodes of our graph will be associated with distinct $(t,\mathbf{e})$ values.

In order to align sequences, one has to deal with insertions and deletions, and this introduces gaps into the aligned sequences. For example, the sequences `ACDEGHI` and `ADEFGHI` are better aligned as `ACDE-GHI` and `A-DEFGHI`. The gap character is treated differently from an amino acid character:

1. For a given sequence $\mathbf{s}$, if $\mathbf{s}[t]$ is not a gap character, then we associate the epitope at position t as the first k non-gap characters, beginning with the character $\mathbf{s}[t]$. For example if $k = 9$ and $\mathbf{s}$ =`GNF--RNQRK-IVKCFNCGK...`, then the PTE associated with $t = 2$ is `NFRNQRKIV`.

2. If $\mathbf{s}[t]$ is the gap character, then we make a "placeholder epitope" whose first character is the gap character, and whose subsequent characters are the next $k-1$ non-gap characters. For these epitopes, we set $f(\mathbf{e}) = 0$. In the example above, the PTE associated with $t = 4$ is the placeholder epitope `-RNQRKIVK`.

The rules for connecting edges to consistent epitope pairs are also modified. The first rule is that edges are only supplied for adjacent positions; a directed edge can only connect a node at position t with another at position $t+1$. As with ungapped sequences, two adjacent epitopes are considered consistent if the last $k-1$ characters of the first epitope agree with the first $k-1$ characters of the second epitope. But we also consider the pair consistent if the second epitope begins with a gap character, and the remaining $k-1$ characters match the last $k-1$ characters of the first epitope. For example: `ACDEFGHIK` and `-CDEFGHIK` are consistent, `-CDEFGHIK` is consistent with itself; and `-CDEFGHIK` and `CDEFGHIKL` are consistent.

Inexact matches

To evaluate the inexact-match coverage, we introduce a set $\mathsf{H}_d(\mathbf{e})$ that includes all epitopes within Hamming distance d of the epitope $\mathbf{e}$. If $\mathscr{E}$ is a set of epitopes, then we write $\mathsf{H}_d(\mathscr{E})$ as the set of all epitopes that are within distance d of some epitope in $\mathscr{E}$. That is, $\mathsf{H}_d(\mathscr{E}) = \bigcup_{\mathbf{e}\in\mathscr{E}} \mathsf{H}_d(\mathbf{e})$. Similar to Eq. (1), we can define the inexact-match coverage in terms of the frequencies $f(\mathbf{e})$ of all the epitopes that approximately match the epitopes in the vaccine:

$$\text{Coverage}_d(\mathbf{q}) = \sum_{\mathbf{e}\in\mathsf{H}_d(\mathscr{E}(\mathbf{q}))} f(\mathbf{e}) \Big/ \sum_{\mathbf{e}\in\mathscr{E}(\mathsf{S})} f(\mathbf{e}) \tag{S-2}$$

Note that *if* the $\mathsf{H}_d(\mathbf{e})$ were disjoint for all $\mathbf{e}\in\mathscr{E}$, then we would be able to write

$$\sum_{\mathbf{e}\in\mathsf{H}_d(\mathscr{E})} f(\mathbf{e}) = \sum_{\mathbf{e}\in\mathscr{E}}\sum_{\mathbf{e}'\in\mathsf{H}_d(\mathbf{e})} f(\mathbf{e}') = \sum_{\mathbf{e}\in\mathscr{E}} \widetilde{f}(\mathbf{e}) \tag{S-3}$$

where

$$\widetilde{f}(\mathbf{e}) = \sum_{\mathbf{e}'\in\mathsf{H}_d(\mathbf{e})} f(\mathbf{e}'), \tag{S-4}$$

which suggests that optimization based on $\widetilde{f}(\mathbf{e})$, in place of $f(\mathbf{e})$, would optimize off-by-d coverage. Unfortunately, however, the $\mathsf{H}_d(\mathbf{e})$ are *not* in general disjoint, so $\widetilde{f}(\mathbf{e})$ in general overestimates the "value" of $\mathbf{e}$ (*i.e.,* the contribution of $\mathbf{e}$ to the coverage). This is what prevents us from using this scheme to optimize the inexact-match coverage for unaligned sequences.

The same problem holds, in principle, for aligned sequences, but the overlap of the Hamming sets is much smaller, and we find that we *can* use this scheme for aligned sequences. We follow the idea in Eq. (S-4), and define $\widetilde{f}(t,\mathbf{e}) = \sum_{\mathbf{e}' \in \mathsf{H}_d(\mathbf{e})} f(t,\mathbf{e}')$. In our experiments, we actually used a slight variant of this expression

$$\widetilde{f}(t,\mathbf{e}) = \lambda f(t,\mathbf{e}) + \sum_{\mathbf{e}' \in \mathsf{H}_d(\mathbf{e})} f(t,\mathbf{e}') \tag{S-5}$$

where $\lambda = 0.1$. This mostly optimizes the number of inexact matches, but the $\lambda f(t,\mathbf{e})$ term provides a small bonus for exact matches.

The notion extends straightforwardly for polyvalent vaccines, though the bookkeeping is a little trickier. If $\mathscr{E}_o$ is the set of epitopes exactly matched by the first m' antigens in a vaccine, then $\mathsf{H}_d(\mathscr{E}_o)$ is the set of epitopes that are covered in an off-by-d sense. To account for the fact that these epitopes are already covered, they need to be excluded from the sum that defines $\widetilde{f}(t,\mathbf{e})$. In particular, write $\mathsf{H}'(\mathbf{e})$ as the set of epitopes that are in $\mathsf{H}_d(\mathbf{e})$ but not in $\mathsf{H}_d(\mathscr{E}_o)$. Then

$$\widetilde{f}(t,\mathbf{e}) = \begin{cases} 0 & \text{if } \mathbf{e} \in \mathscr{E}_o \\ \lambda f(t,\mathbf{e}) + \sum_{\mathbf{e}' \in \mathsf{H}'(\mathbf{e})} f(t,\mathbf{e}') & \text{otherwise,} \end{cases} \tag{S-6}$$

and the next antigen $(m'+1)$ is obtained by finding the path through the graph that optimizes $\sum_t \widetilde{f}(t,\mathbf{e}_t)$.

Filovirus alignment

We created a master input sequence alignment, using the Los Alamos Filovirus database[33]. This alignment includes 34 sequences – a single representative sequence for every human outbreak that has at least one full length genomic sequence available, as well as a representative of RESTV and LLOV, which have not been isolated from humans – to capture the extent of known *Filoviridae* diversity, while weighting the sampling towards recurrent outbreak strains. Hence the alignment includes 10 EBOV sequences, 7 SUDV, 2 BDBV, 1 TAFV, 1 RESTV, 1 LLOV, 3 RAVV, and 9 MARV. Within human outbreaks, sequences are highly similar, and so each outbreak is represented only once. To select a single representative that approximated the index case of each outbreak, we chose a sequence from the earliest sample in outbreak, when temporal data was available. If multiple isolates were sequenced from that sample, we picked a natural sequence that was either identical or closest to the consensus from the first time point. We then translated each gene, including the full-length Glycoprotein GP, but not the secreted forms, sGP and ssGP, and we concatenated the proteins into a full proteome alignment of all 7 Filovirus proteins. This served as a baseline for vaccine design. We then created two subsets of this data for staged vaccine design exploration. The first included only 8 sequences, one representative each from EBOV, SUDV, BDBV, TAFV, RESTV, LLOV, RAVV, MARV, so we could explore the vaccine design outcome if all species were weighted equally. The second one contained only a single representative virus from each of the 5 species in the *Ebolavirus* genus.

Table 1. Variation in coverage with length of epitope. Five separate Epigraph $m = 2$ solutions (with no iterative refinement or multiple starts) are computed for the B-clade Gag protein sequences. Each solution optimizes a different length epitope, with k_o varying from 8 to 12. The coverage for each solution is evaluated five different ways, based on epitope lengths varying from 8 to 12. Numbers within a column are meant to be compared, since they are all evaluated by the same criterion. We see in each case that the optimal coverage, as evaluated for k-mer epitopes, is achieved by the solution for which $k_o = k$. But we also see that the differences are not substantial.

Optimized with epitope length k_o	Evaluated with epitope length k				
	$k=8$	$k=9$	$k=10$	$k=11$	$k=12$
$k_o=8$	0.760252	0.725588	0.687727	0.650896	0.617354
$k_o=9$	0.756754	0.726877	0.692048	0.657789	0.626116
$k_o=10$	0.750047	0.723911	0.697076	0.668338	0.641497
$k_o=11$	0.749303	0.723031	0.696266	0.670036	0.644357
$k_o=12$	0.747089	0.721145	0.694929	0.669224	0.644787

Table 2. Compare 9-mer coverage for Mosaic and Epigraph. Epigraph results are based on five runs with different random number seeds; the best and the median coverage fractions are reported. (Epigraph is much faster to run than Mosaic, so it is feasible to run it five times and keep the best solution.) The main thing to notice is that the coverage fractions are so similar, with the difference typically in the fourth decimal place. Mosaics were generated using 10 hour run times on a 48 Core AMD Opteron cluster via the HIV database portal, and used population sizes of 400 (http://www.hiv.lanl.gov/content/sequence/MOSAIC/). Epigraph usually has the slight advantage; in only 7 of the 48 cases did the Mosaic solution have the best coverage. Even the median Epigraph solution outperformed Mosaic most of the time. Here, $m = 1+1$ corresponds to the sequential cocktail solution (optimize for $m = 1$, fix that solution, and find the optimal complementary sequence), and $m = 2$ refers to iterative refinement and multiple random starts.

	$m = 1$		$m = 1+1$		$m = 2$	
Protein and Clade	Mosaic	Epigraph (best/median)	Mosaic	Epigraph (best/median)	Mosaic	Epigraph (best/median)
Gag E	0.687065	0.687068 0.687068	0.769795	0.770351 0.770351	0.770029	0.770971 0.770971
Gag B	0.613677	0.613664 0.613491	0.677744	0.727352 0.727084	0.726545	0.727352 0.727091
Gag C	0.603501	0.605896 0.603367	0.712061	0.711849 0.711297	0.711460	0.711904 0.711813
Gag M	0.464367	0.464368 0.464157	0.621413	0.621209 0.620641	0.620995	0.621718 0.621405
Nef E	0.456775	0.456775 0.456775	0.597314	0.596507 0.596507	0.595638	0.596507 0.596507
Nef B	0.363338	0.363365 0.363338	0.480749	0.480515 0.480489	0.482010	0.482288 0.482001
Nef C	0.391395	0.390911 0.390911	0.527219	0.527535 0.527535	0.527125	0.527750 0.527750
Nef M	0.285417	0.285613 0.285584	0.396963	0.397462 0.396834	0.398129	0.398710 0.398410
Pol E	0.791086	0.791086 0.791086	0.866294	0.866522 0.866522	0.865880	0.866639 0.866639
Pol B	0.703716	0.703722 0.703722	0.798170	0.798906 0.798906	0.798047	0.798945 0.798906
Pol C	0.732098	0.731738 0.731726	0.817086	0.817760 0.817636	0.817081	0.817802 0.817636
Pol M	0.612655	0.612701 0.612558	0.734066	0.734081 0.734062	0.735310	0.735433 0.735260
Env B	0.378914	0.379337 0.379334	0.467372	0.468104 0.468086	0.467647	0.468269 0.468251
Env C	0.397118	0.397204 0.397204	0.489801	0.490135 0.490135	0.489693	0.490498 0.490454
Env E	0.493367	0.493401 0.493401	0.577307	0.577352 0.577352	0.576875	0.577570 0.577467
Env M	0.282000	0.282412 0.282244	0.379683	0.379903 0.379740	0.381089	0.381442 0.380971

Table 3. Aligned-sequences Epigraph performance. Performance of Epigraph applied to aligned sequences vs Mosaic for 9-mers. Because the algorithm is more nearly deterministic for the aligned case, the Epigraph results are based on a single run. As in Supplementary Table 2, note that the Epigraph and Mosaic coverage fractions very similar. But while the unaligned Epigraph usually outperformed the Mosaic solutions, the aligned Epigraph is outperformed by Mosaic just over half of the time (27 out of 48). Asterisks indicate the higher score of Mosaic vs Epigraph.

Protein	$m = 1$		$m = 1+1$		$m = 2$	
and Clade	Mosaic	Epigraph	Mosaic	Epigraph	Mosaic	Epigraph
Gag E	0.687065	0.687068*	0.769795	0.769894*	0.770029	0.770523*
Gag B	0.613677*	0.613473	0.677744	0.726896*	0.726545	0.726903*
Gag C	0.603501*	0.602685	0.712061*	0.711228	0.711460*	0.711259
Gag M	0.464367	0.465539*	0.621413*	0.619661	0.620995*	0.620100
Nef E	0.456775*	0.450319	0.597314*	0.586679	0.595638*	0.586741
Nef B	0.363338*	0.362313	0.480749*	0.477803	0.482010*	0.479306
Nef C	0.391395*	0.384473	0.527219*	0.517049	0.527125*	0.517263
Nef M	0.285417*	0.279744	0.396963*	0.387946	0.398129*	0.388959
Pol E	0.791086	0.791109*	0.866294	0.866443*	0.865880	0.866493*
Pol B	0.703716	0.703968*	0.798170	0.798931*	0.798047	0.798931*
Pol C	0.732098*	0.731857	0.817086	0.817685*	0.817081	0.817685*
Pol M	0.612655*	0.612590	0.734066	0.734103*	0.735310	0.735691*
Env B	0.378914	0.379268*	0.467372	0.467930*	0.467647	0.467930*
Env C	0.397118	0.397142*	0.489801*	0.489724	0.489693	0.489988*
Env E	0.493367*	0.493353	0.577307*	0.576248	0.576875*	0.576248
Env M	0.282000*	0.281127	0.379683*	0.377815	0.381089*	0.379208

Table 4. Improving Mosaic solutions. Epigraph can be used to "tweak" the Mosaic solution provided by the genetic algorithm; this provides a small, but always positive, improvement.

	$m = 2$		
Protein		Epigraph	
and Clade	Mosaic	Tweak	Improvement
Gag E	0.770029	0.770501	0.000472
Gag B	0.726545	0.726733	0.000188
Gag C	0.711460	0.711603	0.000143
Gag M	0.620995	0.621332	0.000337
Nef E	0.595638	0.595845	0.000207
Nef B	0.482010	0.482053	0.000043
Nef C	0.527125	0.527776	0.000651
Nef M	0.398129	0.398636	0.000507
Pol E	0.865880	0.866402	0.000522
Pol B	0.798047	0.798393	0.000346
Pol C	0.817081	0.817775	0.000694
Pol M	0.735310	0.735427	0.000117
Env B	0.467647	0.467842	0.000195
Env C	0.489693	0.490032	0.000339
Env E	0.576875	0.577013	0.000138
Env M	0.381089	0.381510	0.000421

Table 5. Summary statistics for Tailored Therapeutic Vaccines. Shown are PTE Coverage (fraction of 9-mers in a natural strain that were perfectly matched by a 9-mers in the vaccine), and cost in terms of Extras (9 mers in the vaccine that are not found in the natural strains). In (a), these values calculated for each of 189 sequences included in the post-2005 US B clade alignment; in (b) for a set of 199 post-2005 C clade sequence from southern Africa; and in (c) a larger set of 4596 sequences was used, spanning the full M group. In these vaccine designs, n is the number of antigens delivered and m is the number of antigens manufactured (and from which the best n of m were chosen for each sequence, individually). For the straight Epigraph vaccines, $n = m$. For the Tailored vaccines, we propose manufacturing $m = 6$ antigens, and delivering $n = 2$ or $n = 3$. Coverage increases with increasing n or m, but the number of Extras depends mostly only on n. We also observed that using the conserved p24 instead of the full Gag protein dramatically increases the coverage and the reduces the number of "Extras", but p24 spans only 231 amino acids long out of Gags full 500 amino acids (based on the HIV reference strain HXB2), reducing the number of potential epitopes that could be targeted by over half.

(a) Evaluated against B-US clade

			Gag		p24	
n	m	Vaccine	Coverage	Extras	Coverage	Extras
1	1	M Epigraph	0.55488	237.815	0.73131	59.979
1	1	B Epigraph	0.61174	195.720	0.76421	52.646
2	2	M Epigraph	0.67323	629.333	0.84204	214.296
2	2	B Epigraph	0.72471	553.899	0.87373	201.233
3	3	B Epigraph	0.75734	786.772	0.90810	297.571
2	6	B Tailored	0.75226	536.619	0.90859	187.889
3	6	B Tailored	0.78669	773.000	0.93556	280.704

(b) Evaluated against C-SA group

			Gag		p24	
n	m	Vaccine	Coverage	Extras	Coverage	Extras
1	1	M Epigraph	0.39886	318.437	0.48548	114.739
1	1	C Epigraph	0.59591	200.809	0.73525	59.040
2	2	M Epigraph	0.57552	682.704	0.76864	230.593
2	2	C Epigraph	0.70760	558.608	0.84494	204.578
3	3	C Epigraph	0.73852	778.603	0.88154	338.417
2	6	C Tailored	0.73440	527.623	0.87996	191.538
3	6	C Tailored	0.76513	767.759	0.90373	290.985

(c) Evaluated against M-group

			Gag		p24	
n	m	Vaccine	Coverage	Extras	Coverage	Extras
1	1	M Epigraph	0.46416	289.843	0.59210	93.267
2	2	M Epigraph	0.62121	664.677	0.78726	229.504
3	3	M Epigraph	0.67648	931.225	0.83539	299.959
2	6	M Tailored	0.68690	621.867	0.84368	214.370
3	6	M Tailored	0.71526	909.538	0.87381	313.950

Table 6. Conserved regions selected for vaccine design. Numbering is relative to proteins in the reference strain Zaire 1976 EBOV virus, NC_002549. See Fig. 4A in the main text for more detail.

Protein	start	stop	length
Nucleoprotein	132	410	279
Matrix	71	193	123
Polymerase.1	540	854	314
Polymerase.3	952	1060	109
Total: 825 amino acids			

Table 7. Ebola coverage: A summary of coverage statistics for a subset of Ebola Epigraph sequence options we explored. Solution A is the best single Epigraph using the 5 Ebola species set (Epigraph a). B is based on fixing "a", and complementing it with an Epigraph that will give the best coverage of the 34 outbreak sequence set (Epigraphs a+b). C is the 2 Epigraph solution, simultaneously solved, for the 34 outbreak set (Epigraphs c1+c2). D fixes the two Epigraphs in B, and adds a third complementary sequence to improve coverage of the 34 outbreak sequences (Epigraphs a+b+d). E fixes epigraph A, and then simultaneously solves for 2 more complementary sequences that maximize coverage of the 34 outbreak sequences, for a total of 3 antigens (Epigraphs a+e1+e2). F is the simultaneously solved 3 Epigraph set that best covers the 34 outbreak sequences (Epigraphs f1+f2+f3). E is the strategy that we preferred, because it improves coverage of diverse sequences in rare species while maintaining excellent coverage of recurrent outbreak forms. Note that E will not perfectly match the optimal solution for the 34 outbreak sequences, solution F, but we preferred solution E as it provided better coverage of rarer species of *Ebolavirus*, with negligible loss of EBOV or SUDV coverage.

	Optimization:		exact	exact	$\lambda = 0.1$	$\lambda = 0.1$		
	Evaluation:		9/9	8/9	9/9	8/9		
Protein	vaccine	m					price	gain
Nucleoprotein	A	1	0.4953	0.6196	0.4858	0.6270	-0.0096	0.0074
Nucleoprotein	B	2	0.5571	0.6365	0.5545	0.6419	-0.0026	0.0053
Nucleoprotein	C	2	0.6919	0.7920	0.6815	0.7953	-0.0104	0.0033
Nucleoprotein	D	3	0.6830	0.7410	0.6765	0.7498	-0.0065	0.0088
Nucleoprotein	E	3	0.7490	0.8334	0.7369	0.8389	-0.0121	0.0054
Nucleoprotein	F	3	0.8206	0.9047	0.8106	0.9077	-0.0100	0.0030
Region1	A	1	0.8052	0.9461	0.7897	0.9535	-0.0155	0.0074
Region1	B	2	0.8619	0.9491	0.8781	0.9651	0.0162	0.0160
Region1	C	2	0.8850	0.9651	0.8781	0.9651	-0.0068	0.0000
Region1	D	3	0.9361	0.9808	0.9362	0.9875	0.0001	0.0067
Region1	E	3	0.9361	0.9808	0.9362	0.9875	0.0001	0.0067
Region1	F	3	0.9503	0.9851	0.9350	0.9889	-0.0153	0.0038
Region2	A	1	0.7635	0.9496	0.7461	0.9530	-0.0174	0.0035
Region2	B	2	0.8061	0.9345	0.7954	0.9350	-0.0107	0.0005
Region2	C	2	0.8430	0.9678	0.8430	0.9678	0.0000	0.0000
Region2	D	3	0.9223	0.9775	0.8721	0.9957	-0.0501	0.0182
Region2	E	3	0.9223	0.9775	0.8721	0.9957	-0.0501	0.0182
Region2	F	3	0.9384	0.9693	0.8721	0.9957	-0.0662	0.0263
Region3	A	1	0.7674	0.9114	0.7550	0.9231	-0.0124	0.0117
Region3	B	2	0.8357	0.9434	0.8328	0.9517	-0.0029	0.0083
Region3	C	2	0.8693	0.9505	0.8547	0.9593	-0.0147	0.0088
Region3	D	3	0.9280	0.9831	0.9202	0.9838	-0.0078	0.0007
Region3	E	3	0.9280	0.9831	0.9202	0.9838	-0.0078	0.0007
Region3	F	3	0.9449	0.9854	0.9228	0.9875	-0.0221	0.0021
Region4	A	1	0.7980	0.9762	0.7505	0.9822	-0.0475	0.0059
Region4	B	2	0.8235	0.9732	0.7991	0.9703	-0.0245	-0.0029
Region4	C	2	0.8416	0.9505	0.8235	0.9732	-0.0181	0.0227
Region4	D	3	0.9275	0.9819	0.8905	0.9881	-0.0370	0.0061
Region4	E	3	0.9275	0.9819	0.8905	0.9881	-0.0370	0.0061
Region4	F	3	0.9464	0.9840	0.8905	0.9881	-0.0559	0.0041
Spike	A	1	0.3883	0.5400	0.3659	0.5531	-0.0224	0.0132
Spike	B	2	0.4214	0.4969	0.3993	0.5357	-0.0221	0.0388
Spike	C	2	0.5644	0.6939	0.5506	0.7006	-0.0138	0.0068
Spike	D	3	0.5877	0.6715	0.5792	0.6830	-0.0085	0.0115
Spike	E	3	0.6504	0.7633	0.6281	0.7666	-0.0223	0.0033
Spike	F	3	0.7285	0.8376	0.7214	0.8397	-0.0071	0.0022

Table 8. Computation. Run times, in seconds, for the Epigraph algorithm on a modern laptop computer. The basic run ($m = 1$) creates a graph from the sequence data, removes cycles, and then uses the dynamical programming scheme in Eq. (2) and Eq. (3) to find the best path. Because the dynamical programming step is so fast, there is very little marginal cost in finding a second path, using the sequential approach, without ($m = 1 + 1$) or with ($m = 2$) iterative refinement. The last column shows the cost of computing new pairs of antigens using iterative refinement with $T = 100$ random initializations for the first path. (Note the Env E data sample does not permit a solution with rare epitopes excluded.)

Protein and Clade	Number of Sequences	Number of Distinct PTEs	$m = 1$ (basic)	$m = 1+1$ (sequential)	$m = 2$ (iterative)	$m = 2$ $T = 100$
Exclude rare epitopes: $n_o = 1$						
Gag E	673	8416	1.29	1.33	1.34	13.66
Gag B	1729	17273	3.04	3.15	3.20	25.75
Gag C	940	12823	2.18	2.17	2.21	20.93
Gag M	4596	45404	8.27	8.48	8.91	73.26
Nef E	246	2894	0.55	0.57	0.58	3.83
Nef B	1780	14522	1.81	1.86	1.92	17.37
Nef C	749	7523	1.01	1.04	1.09	8.85
Nef M	4040	30879	4.09	4.33	4.55	48.23
Pol E	348	7315	1.26	1.29	1.33	15.03
Pol B	1072	19732	3.26	3.37	3.40	38.20
Pol C	414	10882	1.75	1.84	1.86	19.77
Pol M	2780	42048	7.89	8.17	8.51	90.40
Env B	1433	58320	7.77	8.07	8.07	89.32
Env C	1124	45221	5.67	5.86	6.11	65.05
Env E	420	17825	–	–	–	–
Env M	4250	152343	25.79	26.92	29.09	381.47
Include all epitopes: $n_o = 0$						
Gag E	673	22107	3.98	3.62	3.69	43.11
Gag B	1729	48297	15.38	15.42	15.83	102.67
Gag C	940	35166	6.82	6.63	7.40	67.80
Gag M	4596	128940	52.01	56.44	50.23	366.43
Nef E	246	8248	1.53	1.59	1.61	13.09
Nef B	1780	44497	6.93	6.83	7.60	85.47
Nef C	749	21639	3.03	3.15	3.35	38.77
Nef M	4040	90850	16.95	16.03	17.01	223.88
Pol E	348	17749	2.48	2.70	2.90	38.44
Pol B	1072	45381	10.28	10.59	11.17	98.29
Pol C	414	26632	4.44	4.49	5.00	53.01
Pol M	2780	103471	38.19	38.18	40.33	395.15
Env B	1433	254412	123.29	137.06	210.90	997.47
Env C	1124	200964	64.72	82.08	72.15	618.25
Env E	420	71841	11.87	11.30	12.53	194.43
Env M	4250	666137	1063.41	1161.57	1260.84	4473.97

Table 9. Compare unaligned and aligned epigraphs. This table contains no new information, but combines results from Table 2 and Table 3 to display side-by-side the performance of unaligned and aligned epigraphs. In general the differences are small, with the largest difference, for the $m = 2$ Nef C case, just over 0.01. For Gag, Nef, and Env, we see that the unaligned performance is almost always better. For Pol, the aligned performance is often better though it bears remarking that for Pol, the differences between the two are particularly small, always less than 0.0005. Asterisks indicate larger values.

Protein	$m = 1$		$m = 1+1$		$m = 2$	
and Clade	Unaligned	Aligned	Unaligned	Aligned	Unaligned	Aligned
Gag E	0.687068	0.687068	0.770351*	0.769894	0.770971*	0.770523
Gag B	0.613491*	0.613473	0.727084*	0.726896	0.727091*	0.726903
Gag C	0.603367*	0.602685	0.711297*	0.711228	0.711813*	0.711259
Gag M	0.464157	0.465539*	0.620641*	0.619661	0.621405*	0.620100
Nef E	0.456775*	0.450319	0.596507*	0.586679	0.596507*	0.586741
Nef B	0.363338*	0.362313	0.480489*	0.477803	0.482001*	0.479306
Nef C	0.390911*	0.384473	0.527535*	0.517049	0.527750*	0.517263
Nef M	0.285584*	0.279744	0.396834*	0.387946	0.398410*	0.388959
Pol E	0.791086	0.791109*	0.866522*	0.866443	0.866639*	0.866493
Pol B	0.703722	0.703968*	0.798906	0.798931*	0.798906	0.798931*
Pol C	0.731726	0.731857*	0.817636	0.817685*	0.817636	0.817685*
Pol M	0.612558	0.612590*	0.734062	0.734103*	0.735260	0.735691*
Env B	0.379334*	0.379268	0.468086*	0.467930	0.468251*	0.467930
Env C	0.397204*	0.397142	0.490135*	0.489724	0.490454*	0.489988
Env E	0.493401*	0.493353	0.577352*	0.576248	0.577467*	0.576248
Env M	0.282244*	0.281127	0.379740*	0.377815	0.380971*	0.379208