Title

Fast and accurate HLA typing from short-read next-generation sequence data with xHLA

Permalink

https://escholarship.org/uc/item/84r8p9gk

Journal

Proceedings of the National Academy of Sciences of the United States of America, 114(30)

ISSN

0027-8424

Authors

Xie, Chao
Yeo, Zhen Xuan
Wong, Marie
et al.

Publication Date

2017-07-25

DOI

10.1073/pnas.1707945114

Peer reviewed

Fast and accurate HLA typing from short-read next-generation sequence data with xHLA

Chao Xie[a,1], Zhen Xuan Yeo[a], Marie Wong[a], Jason Piper[a], Tao Long[b], Ewen F. Kirkness[b], William H. Biggs[b], Ken Bloom[b], Stephen Spellman[c], Cynthia Vierra-Green[c], Colleen Brady[c], Richard H. Scheuermann[d,e], Amalio Telenti[b], Sally Howard[b], Suzanne Brewerton[a], Yaron Turpaz[a,b], and J. Craig Venter[b,d,1]

[a]Human Longevity Singapore Pte Ltd., Singapore 138543; [b]Human Longevity, Inc., San Diego, CA 92121; [c]Center for International Blood and Marrow Transplant Research, Minneapolis, MN 55401; [d]J. Craig Venter Institute, La Jolla, CA 92037; and [e]Department of Pathology, University of California at San Diego, La Jolla, CA 92093

Contributed by J. Craig Venter, May 24, 2017 (sent for review December 5, 2016; reviewed by Mary N. Carrington and Michael A. Curran)

The HLA gene complex on human chromosome 6 is one of the most polymorphic regions in the human genome and contributes in large part to the diversity of the immune system. Accurate typing of HLA genes with short-read sequencing data has historically been difficult due to the sequence similarity between the polymorphic alleles. Here, we introduce an algorithm, xHLA, that iteratively refines the mapping results at the amino acid level to achieve 99–100% four-digit typing accuracy for both class I and II HLA genes, taking only ∼3 min to process a 30× whole-genome BAM file on a desktop computer.

MHC | autoimmune diseases | transplantation

Genes within the HLA complex play an integral role in the human adaptive immune system. Classical class I (HLA-A, -B, and -C) and class II (HLA-DR, -DP, and -DQ) HLA gene products function by presenting foreign antigens to T cells to trigger immune responses (1). HLA genes show incredible sequence diversity in the human population. For example, there are >4,000 known alleles for the HLA-B gene alone (2, 3). The genetic diversity in HLA genes in which different alleles have different efficiencies for presenting different antigens is believed to be a result of evolution conferring better population-level resistance against the wide range of different pathogens to which humans are exposed (4).

In addition to its role in infectious disease defense, HLA has been associated with >100 different diseases, including various autoimmune disorders (1). However, although it plays an important role in human health, people do not routinely have their HLA genes typed. With the current trend toward precision medicine, knowing their HLA types will be crucial in early diagnosis and management of many diseases. For example, autoimmune disorder patients often have chronic problems with no exact diagnosis for many years after repeated doctor visits (5, 6). Knowing patients' HLA types could lead to early diagnosis and reduce the burden on both patients and the healthcare system.

In the setting of hematopoietic stem cell transplantation (HSCT), matching of HLA alloantigens between donor and recipient to mitigate an allogeneic immune response is the single most important factor dictating the successful outcome of engraftment and survival after transplantation (7). HLA typing for HSCT donor–recipient matching in the clinical laboratory, including nucleotide sequence-based methods, focuses on characterizing sequence variations (polymorphisms) in the three classical class I alpha proteins, HLA-A, -B, and -C, and two of the classical class II beta proteins, HLA-DRB1 and -DQB1.

Finally, HLA determines reactivity and hypersensitivity reactions to a number of therapeutic agents that act as haptens. The Food and Drug Administration lists a number of drugs that require or may benefit from HLA typing before prescription to avoid severe adverse reactions (8).

HLA genes are usually typed with targeted sequencing methods: either long-read sequencing or long-insert short-read sequencing. In contrast, the most common personal genome sequencing methods use short-read shotgun technologies, which have historically been more difficult for HLA-typing algorithms (9–12). Existing algorithms are either inaccurate or slow. Given the growing popularity of precision medicine approaches and the generation of personal genomes, there is a clear need for faster and more accurate HLA-typing algorithms based on short-read shotgun data.

Results

There are three main challenges for HLA typing from short-read data. First, the high degree of sequence polymorphism means there are many potential candidate alleles for each gene, which means there is a high level of sequence "noise" when trying to find the correct allele from a large search space. Many of the existing algorithms try to reduce the noise level by filtering out less common HLA alleles from their candidate allele set. For example, OptiType ignores any allele with no reported allele frequency in AlleleFrequency.net (9, 13), whereas HLA-VBSeq only considers ∼100 HLA alleles as candidates (10). Second, despite the extensive polymorphisms within each HLA gene, alleles from different HLA genes can share regions that are extremely similar

Significance

Regulation of the human immune system is largely controlled by the HLA gene complex on chromosome 6 and is important in infectious disease immunity, graft rejection, autoimmunity, and cancer. HLA typing is traditionally performed by serotyping and/or targeted sequencing. However, the advent of precision medicine and cheaper personal genome sequencing has sprung an unmet need for a fast and accurate way of predicting HLA types from short-read sequencing data. Here, we present xHLA, an algorithm for HLA typing based on translated short reads, exhaustive multiple sequence alignment-based alignment expansion, and iterative solution set refinement that is also faster and more accurate than existing methods. Results are achievable within minutes and could greatly benefit individuals who have had their genome sequenced.

Author contributions: C.X., R.H.S., A.T., S.B., Y.T., and J.C.V. designed research; C.X., Z.X.Y., M.W., J.P., T.L., E.F.K., S.S., C.V.-G., C.B., and R.H.S. performed research; K.B., S.H., and Y.T. supervised research; C.X., J.P., W.H.B., S.S., C.V.-G., and C.B. contributed new reagents/analytic tools; C.X., Z.X.Y., M.W., J.P., T.L., E.F.K., W.H.B., K.B., R.H.S., A.T., S.H., and S.B. analyzed data; and C.X., R.H.S., A.T., and S.B. wrote the paper.

Reviewers: M.N.C., SAIC-Frederick; and M.A.C., The University of Texas MD Anderson Cancer Center.

Conflict of interest statement: C.X., Z.X.Y., M.W., J.P., T.L., E.F.K., W.H.B., K.B., A.T., S.H., S.B., Y.T., and J.C.V. are employees of Human Longevity Inc. There is a patent pending on the technique presented in this work.

Freely available online through the PNAS open access option.

Data deposition: The code of xHLA is available for noncommercial academic research use at https://github.com/humanlongevity/HLA.

[1]To whom correspondence may be addressed. Email: jcventer@humanlongevity.com or xchao@humanlongevity.com.

to each other, especially when looking at fragments resulting from short-read sequencing. Furthermore, there are many nonfunctional pseudogenes with similar sequences to functional HLA genes. Third, there are no complete reference sequences for most HLA alleles in the HLA reference database, International ImMunoGeneTics Project (IMGT)/HLA (3). Class I genes typically have exon 2 and 3 sequences available in the database, and class II genes typically have exon 2 sequences available in the IMGT/HLA database—the so-called "core exons" that comprise the antigen-recognition domain of the molecule involved in peptide presentation and interaction with the T-cell receptor. However, the availability of sequences for other exons in the database ranges from 5 to 36%. Noncore exons contribute to HLA allele polymorphism (2), and many existing algorithms, such as OptiType and HLA*PRG, which type HLA genes with core exons only, report only a group or representative allele for each gene, even when there are full-length sequences available for certain HLA alleles in the database.

xHLA Algorithm Overview. Most existing HLA-typing algorithms follow a two-stage framework: First, align potential HLA sequencing reads to a collection of HLA reference sequences (IMGT/HLA), and then find a combination of HLA alleles that can best explain the observed alignment results. Our algorithm, xHLA, follows the same general framework with differing details (Fig. 1).

Generating Alignment Matrix. The starting data for xHLA is a BAM file that contains all sequenced reads mapped to a reference genome. Potential HLA sequencing reads are extracted from the BAM file based on their mapping coordinates and aligned to the IMGT/HLA database. All existing algorithms perform this step with DNA-level alignment, often accepting a certain degree of mismatches. A key difference in our algorithm is the use of a fast and sensitive protein level aligner, DIAMOND (14), and the acceptance of only perfect matches for quality-trimmed reads. The rationale is that, in HLA typing, the most relevant resolution is four-digit typing, which describes the protein-level amino acid sequence differences encoded in the HLA genes. Therefore, four-digit typing is what most clinical and next-generation sequencing-based typing algorithms aim to achieve. The problem with DNA-level alignment is that it cannot distinguish synonymous from nonsynonymous mismatches. For example, it will rank five synonymous mismatches as more dissimilar than a single nonsynonymous one. An added advantage of using protein-level alignment is that the identity threshold is not arbitrary as in DNA alignment, because there is clear concordance between 100% protein sequence identity and four-digit HLA typing.

Another issue with alignment of short reads against the IMGT/HLA database is that the database is not a typical reference sequence database. Most aligners have been developed to find the best homologs in a database with many diverse sequences. However, in this case, the reference database contains tens of thousands of very similar sequences. For any short sequencing read, there are many equally good or perfect matches. Most off-the-shelf aligners are optimized to find a number of good matches, but are not exhaustive. HLA*PRG is the only existing algorithm that tries to solve this problem by using a graph reference sequence set with certain assumptions about the intronic sequences (11). However, the HLA*PRG solution is very slow, taking ∼11 h for one 30× coverage whole-genome sequence BAM file. xHLA uses a precomputed protein-level reference multiple sequence alignment (MSA) of known HLA alleles from the IMGT/HLA database to expand the initial alignment results. For example, if a read is aligned to 100 reference sequences in the initial alignment, xHLA will compare the read to all equivalent sequence segments in the MSA corresponding to the initial 100 matches. Consequently, xHLA is guaranteed to exhaustively find all equally good matches from the HLA database.

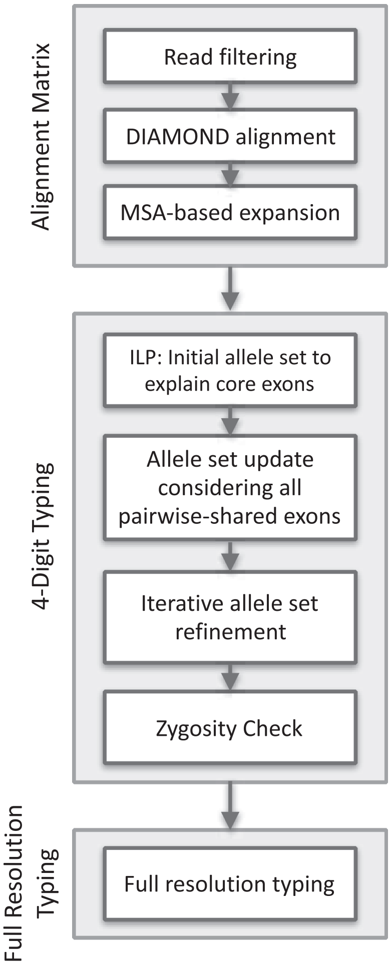

Fig. 1. Overview of the xHLA algorithm.

Four-Digit Typing. After aligning reads to the HLA reference sequences, the next task is to infer the most likely HLA allele combination that best explains the alignments. Rather than treating each gene separately, and by using alleles with the largest number of aligned reads, OptiType uses integer linear programing to find the complete allele set as a single optimization problem (9). This method is a better approach than the other methods because it explicitly considers short reads that map to multiple HLA allele candidates. However, there are two issues with this approach. First, there are often many different solution sets explaining the alignments equally or nearly equally well. Second, this method does not work when different HLA allele candidates have different numbers of reference exon sequences available. For example, allele candidates with a full set of exons available will most likely have more reads aligned to it compared with allele candidates with reference sequences from only a subset of exons available. xHLA solves these problems in three steps.

An initial set of HLA allele candidates is first selected in a similar way to the integer linear programing technique used in OptiType, by using only the core exons that are available for all HLA reference alleles in the IMGT/HLA database. Then, for each allele candidate (sol) in the initial solution set (solution_set), all alternative alleles (comp) that explain the alignments nearly as well are collected (Fig. 2). The current candidate, sol, is compared with alternative candidates, comp, based on the reads aligned to all of the exons available for both alleles

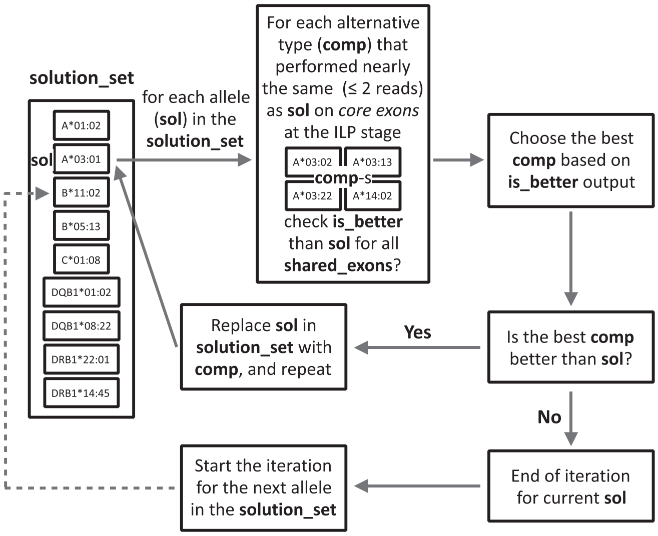
solution_set
sol
A*01:02
A*03:01
B*11:02
B*05:13
C*01:08
DQB1*01:02
DQB1*08:22
DRB1*22:01
DRB1*14:45
for each allele (sol) in the solution_set
For each alternative type (comp) that performed nearly the same (≤ 2 reads) as sol on core exons at the ILP stage
A*03:02
A*03:13
comp-s
A*03:22
A*14:02
check is_better than sol for all shared_exons?
Choose the best comp based on is_better output
Is the best comp better than sol?
Yes
Replace sol in solution_set with comp, and repeat
No
End of iteration for current sol
Start the iteration for the next allele in the solution_set

Fig. 2. Iterative allele set update considering all pairwise-shared exons.

in the IMGT/HLA database, rather than only the core exons as in the initial round (Fig. 2). When comparing the two alleles, sol and comp, only reads that are aligned to one of the two alleles, but not both, nor any other alleles in the current solution set, are considered ("is_better" in Fig. 2). This filtering step is important because reads that align to other alleles in the solution set are not informative when comparing sol and comp because it is unclear whether those reads are derived from the alleles being compared or other alleles in the solution set. If the alternative allele can explain more aligned and informative reads than the original sol, we replace the original sol with the alternative comp and repeat the iteration with a new set of alternatives (Fig. 2). When no further optimization is found, the procedure is repeated on the next allele candidate in the solution set. This procedure is a quick and crude way of pulling noncore exons into play. However, when updating an allele candidate, it is assumed that the rest of the solution set is correct, which may not be the case because the allele candidate changes through this iterative process. This limitation means that one more iterative refinement step (Fig. 3) is required.

The second iterative refinement step is similar to the procedure described above, except that all candidate alleles in the current solution set are compared with their respective set of alternative alleles in parallel, and only one candidate allele is updated after each iteration. The update is repeated until no further optimization is possible (Fig. 3).

For every HLA gene, if the above procedure produces two alleles, a check is made on whether they are true heterozygotes by comparing the informative reads aligned to the two alleles, but not to any other allele in the solution set (zygosity check). If one of the two alleles has five times more informative reads aligned to it, the heterozygous call is changed to a homozygous call.

Full-Resolution Typing. The above procedure gives a four-digit HLA allele set that best explains the alignments. We can optionally infer higher-resolution HLA types using the four-digit solution set as a starting point (Fig. 4). This process is an easier task compared with the above four-digit typing task itself, because the search space contains only higher-digit HLA alleles under each set of four-digit alleles, rather than the entire IMGT/HLA database.

Benchmarking. xHLA was tested on four public datasets benchmarked from HLA*PRG's and ATHLATES' original publications (11, 15), including two whole-genome and two exome sequencing datasets. In all cases, xHLA was more accurate than existing algorithms (Table 1). HLA*PRG performed better than other existing algorithms, but scored 83.3% (class I) and 93% (class II) accuracy for the HapMap exome data based on their own benchmark. In contrast, xHLA scored 98.3% and 100% accuracy on the same dataset. In terms of runtime, xHLA was much faster than existing algorithms, taking ∼3 min per 30× whole genome (2 × 150-bp reads) sample on a machine with 16-core CPU and 30-GB memory. In comparison, HLA*PRG was the slowest and most resource-demanding algorithm, taking ∼11 h with 80-GB memory. Other algorithms tested usually take between 15 min and 5 h for similar samples.

Additionally, xHLA was tested on two much larger, private datasets. For the whole-genome dataset, obtained through collaboration with the Center for International Blood and Marrow Transplant Research (CIBMTR; a research collaboration between the National Marrow Donor Program/Be The Match and the Medical College of Wisconsin), xHLA achieved 99.7% (class I) and 99.4% (class II) accuracy, higher than all other existing algorithms tested (Table 1). For exome data, the GeT-RM dataset (cell lines obtained from wwwn.cdc.gov/clia/Resources/GetRM/), xHLA achieved 99.5% accuracy on class I HLA alleles, but 96.1% on class II. Most of the errors (3.6% of 3.9%) for class II typing in the GeT-RM dataset were due to missing heterozygous calls (i.e., only reported one of the two heterozygous alleles), suggesting differential pull-down efficiencies of the exome enrichment probes against sequences from different HLA alleles for some class II HLA types.

Because most of the gold-standard HLA types we used for benchmarking are based on core exons of the HLA genes (15–17), the benchmarking procedure considered four-digit types with the same core exon sequences as "consistent" types. This fraction ranged from 0 to 1.9% of alleles in the benchmarking datasets. The largest number of consistent, but not identical, predictions by xHLA was the HLA-DRB1 gene in the CIBMTR data: There were 13 reported DRB1*14:01 alleles predicted as DRB1*14:54 by xHLA. The two DRB1 types shared the same core exon 2 sequence, and most of the reported DRB1*14:01 before 2006 were actually DRB1*14:54 (18). This result shows that our typing algorithm works well beyond the core exons.

Discussion

As demonstrated in our experiments, we have shown that xHLA is both faster and more accurate on whole-genome and -exome sequencing data than other methods. Specifically, seven other

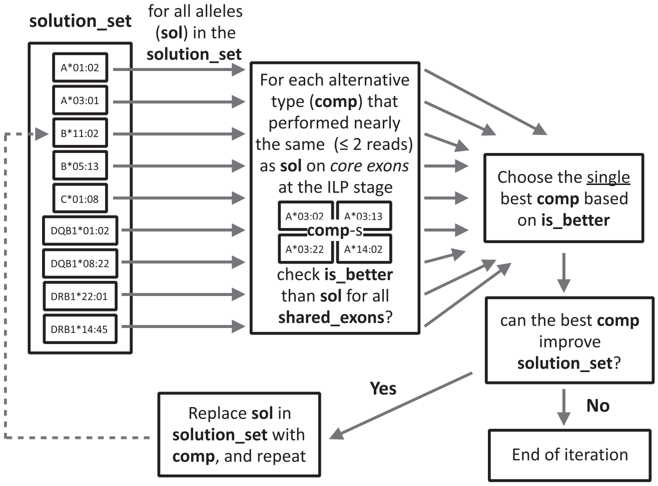
solution_set
for all alleles (sol) in the solution_set
A*01:02
A*03:01
B*11:02
B*05:13
C*01:08
DQB1*01:02
DQB1*08:22
DRB1*22:01
DRB1*14:45
For each alternative type (comp) that performed nearly the same (≤ 2 reads) as sol on core exons at the ILP stage
A*03:02
A*03:13
comp-s
A*03:22
A*14:02
check is_better than sol for all shared_exons?
Choose the single best comp based on is_better
can the best comp improve solution_set?
Yes
Replace sol in solution_set with comp, and repeat
No
End of iteration

Fig. 3. Second-round iterative allele set refinement.

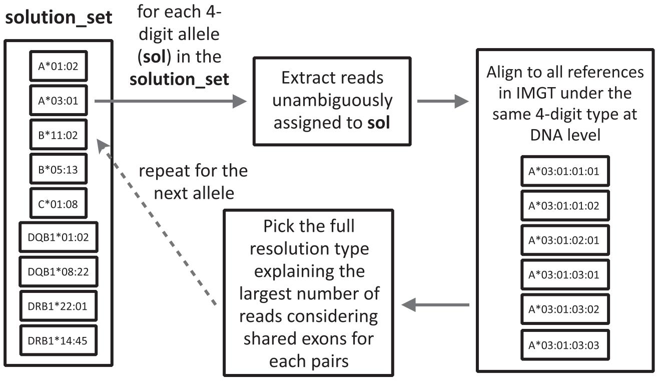
solution_set
for each 4-digit allele (sol) in the solution_set
A*01:02
A*03:01
B*11:02
B*05:13
C*01:08
DQB1*01:02
DQB1*08:22
DRB1*22:01
DRB1*14:45
Extract reads unambiguously assigned to sol
Align to all references in IMGT under the same 4-digit type at DNA level
A*03:01:01:01
A*03:01:01:02
A*03:01:02:01
A*03:01:03:01
A*03:01:03:02
A*03:01:03:03
repeat for the next allele
Pick the full resolution type explaining the largest number of reads considering shared exons for each pairs

Fig. 4. Full-resolution HLA typing as a downstream step after four-digit typing.

methods were tested against xHLA on three whole-genome and three whole-exome datasets. In all cases, xHLA performed the best.

Although all of the methods use a similar process of alignment followed by choosing the alleles that best explain the alignments, there are some differences. The first main difference of xHLA is the use of protein-level alignment and MSA-based alignment expansion, which results in a cleaner and exhaustive alignment matrix and, thus, higher accuracy compared with the other HLA-typing methods, as summarized in Table 1. This finding clearly shows the advantages of using protein-level typing because a perfect match in protein space is immutable. Therefore, there is no need to filter any of the initial data or worry about DNA sequence similarity of HLA genes and pseudogenes. There is also no bias introduced due to incomplete data in the IMGT/HLA reference sequence database. In addition, xHLA runs significantly faster than the other methods on similar hardware specifications by using DIAMOND, a double-indexing strategy that enables a 20,000× speed-up as compared with BLASTX, with a similar degree of sensitivity (14). It takes xHLA only a few minutes to process a 30× genome with 150-bp reads on a modest desktop computer. However, with shorter reads—say, 75-bp long—it will take significantly longer, because the number of matches in the exhaustive alignment matrix for shorter reads increases exponentially. Although it is focused on four-digit typing, xHLA offers an optional feature to run full-resolution HLA typing as a downstream step after four-digit typing.

Another key difference of xHLA is that it works beyond the core exons of HLA genes. One interesting observation in our benchmarking was the DRB1*14:54 and DRB1*14:01 pair: Thirteen previously reported DRB1*14:01 were predicted as DRB1*14:54, which is consistent with the fact that most previously reported DRB1*14:01 were actually DRB1*14:54 (18). We excluded DPA1 and DQA1 due to the lack of truth data for these genes.

Even though xHLA performed the best, it did drop in accuracy in one of the three exome data samples that we tested. In our private GeT-RM exome samples, the typing accuracy for class II genes was 96.1%, much lower than in other samples. The dip is likely due to differential pull-down efficiencies for different alleles by the exome enrichment kit.

By considering well-documented population-specific haplotypes (13), together with derived haplotypes from our recently published large-scale whole-genome sequencing data (19), we will be able to improve our typing accuracy further, especially in exome data, where enrichment bias occurs.

With the current movement toward precision medicine, personal genome sequencing is becoming increasingly popular and cheaper. HLA is one of the most relevant regions of the genome for personalized medicine, given its diversity and role in the immune system. xHLA is both fast and accurate in identifying HLA types from personal genome data and should be used to make HLA typing more readily available to individuals and their physicians as an added benefit to having their genomes sequenced.

Materials and Methods

The xHLA Algorithm Overview. The basic steps of the xHLA algorithm are illustrated in Fig. 1. There are three major modules: construction of alignment matrix, four-digit typing, and full-resolution typing.

Alignment Matrix.

Preprocessing. The input data of xHLA is a BAM file where sequencing reads are mapped to the hg38 human reference assembly (excluding alt contigs). Both BWA's mem mode (Version 0.7.15) (20) and Isaac (Version 0.14.02.06) (21) with default parameters work well with xHLA on diverse datasets. Because all genome sequencing projects produce a BAM file, the alignment step is not considered as part of xHLA. xHLA extracts relevant HLA reads from the BAM file (chromosome 6, position 29,886,751–33,090,696), then trims and filters them based on base quality scores. Trimming is based on BWA's trimming algorithm with Phred quality cutoff 20 from the 3′ end after first trimming Ns. Reads <70 bp or with more than five positions with Phred quality score <4 after trimming are removed.

DIAMOND alignment. The reads are then aligned to reference HLA exon protein sequences from IMGT/HLA by using DIAMOND (Version 0.8.15;

Table 1. Accuracy of xHLA compared with existing algorithms

Data type	Dataset	Class (*n*)	xHLA, %	HLA*PRG, %	PHLAT, %	HLA Reporter, %	SOAP-HLA, %	HLA-VBSeq, %	OptiType, %	ATHLATES, %
WGS	Platinum	I (18)	100	100	100	11.2				
		II (12)	100	100	100	66.3				
	1000G	I (66)	100	100	63.7	4.5				
		II (44)	100	97.7	70.9	62.7				
	CIBMTR	I (2,928)	99.7	98.5			91.0	97.5	97.0	
		II (2,872)	99.4	96.5			97.0	38.0	NA	
Exome	HapMap	I (174)	98.3	83.3	87.7	26.5				
		II (104)	100	93.0	87.3	71.3				
	1000G	I (66)	100							98.5
		II (44)	100							100
	GeT-RM	I (646)	99.5							
		II (644)	96.1							

A total of six datasets were used. Platinum, 1000G (WGS), HapMap, and 1000G (Exome) were identical to the benchmarking datasets used in refs. 11 and 15. Accuracy for HLA*PRG, PHLAT, and HLA Reporter on the public datasets were obtained from the same references. For each dataset, accuracy for HLA class I and II genes is listed separately, and the number of alleles in each dataset is shown in the table.

parameters: blastx –index-mode 1 –seg no –min-score 10 –top 20 -c 1 -C 20000). Strict filtering is performed on the alignments. The alignments need to be 100% identical and cover the full length of the query reads, unless the unaligned region of the query extends beyond the reference sequence. Because protein-level alignment is used against exons, an additional comparison of reads against incomplete codons at exon boundaries is performed. Only the alignments with equally best scores for each query read are kept.

MSA-based alignment expansion. The next step is to generate an exhaustive alignment matrix by using a precomputed MSA of known HLA alleles with MUSCLE (22). For example, if a read is aligned to 100 reference sequences by DIAMOND after the previous step, xHLA compares the read to the equivalent sequence segments in all other reference sequences in the MSA. All equally good reference matches are retained in the exhaustive alignment matrix. Although only six HLA genes (HLA-A, -B, -C, -DQB1, -DRB1, and -DPB1) are typed, all other HLA genes are used as alignment references in the above steps. Reads that map equally well to genes being typed and those that are not are considered as uninformative and excluded from further analysis.

Four-Digit Typing.

Integer linear programing based on core exons. The four-digit typing procedure is divided into four steps. The first step uses only information from "core exons" (exons 2 and 3 for class I HLA genes, and exon 2 for class II), and similar to OptiType (9), an integer linear programing approach is used to derive an initial set of HLA alleles that best explains the alignment matrix of core exons. This procedure was performed by using the lpSolve package (Version 5.6.13) in R.

Allele set update considering noncore exons. There are always many allele candidates that can perform equally or nearly equally well compared with the alleles in the initial solution set. Therefore, for each allele in the initial solution set, we replace it with one alternative at a time and keep all solution sets whose total number of explainable aligned reads reduced by only two or fewer reads. Then, we determine which performs better when considering noncore exons. Performance of each alternative allele is dependent on the other alleles in the solution set. This procedure is the second step, which crudely pulls the noncore exons into consideration (Fig. 2). When comparing original vs. alternative allele types, we consider all exons where both types have reference sequences in the database. Furthermore, only reads that align to one of the two allele types, but not to both, nor to any other alleles in the current solution set, are considered. If the alternative allele type performs better than the original type, we replace the original with the alternative and repeat the procedure for the same allele until there are no more changes. Then, we carry out the same procedure for the next allele (Fig. 2).

Iterative Allele Set Refinement. The third step is another iterative refinement procedure. In the previous step, updating an allele assumes that the other alleles in the solution set are correct. In this step, the solution set is treated as one unit, and refinements are made to best improve the performance of the solution set, rather than using only an individual allele (Fig. 3). In other words, each allele in the current solution set is compared with its alternative allele types, and potential improvements are recorded if the original is replaced with the alternative. After collecting all potential improvements, the single best refinement is chosen and committed. The procedure is repeated until there are no further changes to the solution set (Fig. 3).

Zygosity Check. The last step in four-digit typing is a zygosity check. If the previous steps produce two different alleles for an HLA gene, the number of informative reads aligned to the two alleles are compared. Informative reads are defined as reads that do not align to other HLA genes in the current solution set. If one of the two alleles has five times more informative reads aligned than the other allele, the heterozygous call is changed to a homozygous call.

Full-Resolution Typing. After producing the four-digit typing solution, xHLA can optionally perform full-resolution typing based on the four-digit solution set (Fig. 4). For each four-digit allele in the solution set, reads that are unambiguously assigned to that allele are extracted and realigned to all DNA reference sequences from IMGT/HLA under the same four-digit allele type. For each pair of full-resolution types, all exons with reference sequences in IMGT are considered, and the type explaining the largest number of unambiguous reads in the pair is taken. The final full-resolution type is identified recursively based on these pairwise comparisons.

Benchmarking. We used four public and two private datasets to benchmark xHLA. The four public datasets are identical to those used in HLA*PRG (11) and ATHLATES (15) in their original publications:

- Illumina Platinum Genomes, whole-genome sequencing (WGS), three samples, from the HLA*PRG paper (11). For this dataset, reads mapped to chromosome 6 and unmapped reads were extracted from the BAM files and mapped to hg38 (excluding alt contigs) by using BWA's mem mode with default parameters.
- The 1000 Genome Project, WGS, 11 samples, from the HLA*PRG paper (11). For this dataset, the hg19 BAM file was lifted over to hg38 with CrossMap (23).
- HapMap, exome, 29 samples, from the HLA*PRG paper (11). Reads were mapped to hg38 (excluding alt contigs) by using BWA's mem mode with default parameters.
- The 1000 Genome Project, exome, 11 samples, from the ATHLATES paper (15). Reads were mapped to hg38 (excluding alt contigs) by using BWA's mem mode with default parameters.

Accuracy measurements for existing software on the public datasets were extracted from the HLA*PRG and ATHLATES papers. Ground truth HLA allele types were obtained from each dataset's original publications (11, 15–17).

In addition, we benchmarked xHLA on two much larger private datasets:

- CIBMTR samples, WGS, 488 samples. The samples were sequenced with 2 × 150-bp reads at 30× coverage. The ground truth HLA types were obtained from CIBMTR and were generated as described (24).
- GeT-RM samples, exome, 108 samples. The samples were sequenced with 2 × 75-bp reads at 30× coverage. The ground truth HLA types were from the Centers for Disease Control and Prevention site (https://wwwn.cdc.gov/clia/Resources/GETRM/pdf/HLA_POSTER_DATA.pdf).

For the private datasets, reads were mapped to hg38 (excluding alt contigs) by using the Isaac aligner (Version 0.14.02.06) with default parameters.

ACKNOWLEDGMENTS. The CIBMTR is supported by Public Health Service Grant/Cooperative Agreement 5U24-CA076518 from the National Cancer Institute (NCI), the National Heart, Lung and Blood Institute (NHLBI), and the National Institute of Allergy and Infectious Diseases; NHLBI and NCI Grant/Cooperative Agreement 5U10HL069294; Health Resources and Services Administration (HRSA/DHHS) Contract HHSH250201200016C; and Office of Naval Research Grants N00014-14-1-0028 and N00014-15-1-0848. The views expressed in this article do not reflect the official policy or position of the National Institutes of Health, the Department of the Navy, the Department of Defense, Health Resources and Services Administration, or any other agency of the US government.

1. Shiina T, Hosomichi K, Inoko H, Kulski JK (2009) The HLA genomic loci map: Expression, interaction, diversity and disease. *J Hum Genet* 54:15–39.
2. Robinson J, et al. (2013) The IMGT/HLA database. *Nucleic Acids Res* 41:D1222–D1227.
3. Robinson J, Soormally AR, Hayhurst JD, Marsh SGE (2016) The IPD-IMGT/HLA Database—new developments in reporting HLA variation. *Hum Immunol* 77:233–237.
4. Prugnolle F, et al. (2005) Pathogen-driven selection and worldwide HLA class I diversity. *Curr Biol* 15:1022–1027.
5. Gough SCL, Simmonds MJ (2007) The HLA region and autoimmune disease: Associations and mechanisms of action. *Curr Genomics* 8:453–465.
6. Schoen C, Osborn R, How SKH, Doty MM, Peugh J (2009) In chronic condition: Experiences of patients with complex health care needs, in eight countries, 2008. *Health Aff (Millwood)* 28:w1–w16.
7. Morishima Y, et al. (2002) The clinical significance of human leukocyte antigen (HLA) allele compatibility in patients receiving a marrow transplant from serologically HLA-A, HLA-B, and HLA-DR matched unrelated donors. *Blood* 99:4200–4206.
8. Food and Drug Administration (FDA) (2016) Table of pharmacogenomic biomarkers in drug labeling. Available at www.fda.gov/Drugs/ScienceResearch/ResearchAreas/Pharmacogenetics/ucm083378.htm. Accessed June 20, 2017.
9. Szolek A, et al. (2014) OptiType: Precision HLA typing from next-generation sequencing data. *Bioinformatics* 30:3310–3316.
10. Nariai N, et al. (2015) HLA-VBSeq: Accurate HLA typing at full resolution from whole-genome sequencing data. *BMC Genomics* 16:S7.
11. Dilthey AT, et al. (2016) High-accuracy HLA type inference from whole-genome sequencing data using population reference graphs. *PLoS Comput Biol* 12:e1005151.
12. Dilthey A, Cox C, Iqbal Z, Nelson MR, McVean G (2015) Improved genome inference in the MHC using a population reference graph. *Nat Genet* 47:682–688.
13. Gonzalez-Galarza FF, et al. (2015) Allele frequency net 2015 update: New features for HLA epitopes, KIR and disease and HLA adverse drug reaction associations. *Nucleic Acids Res* 43:D784–D788.
14. Buchfink B, Xie C, Huson DH (2015) Fast and sensitive protein alignment using DIAMOND. *Nat Methods* 12:59–60.

15. Liu C, et al. (2013) ATHLATES: Accurate typing of human leukocyte antigen through exome sequencing. *Nucleic Acids Res* 41:e142.
16. Warren RL, et al. (2012) Derivation of HLA types from shotgun sequence datasets. *Genome Med* 4:95.
17. Gourraud PA, et al. (2014) HLA diversity in the 1000 genomes dataset. *PLoS One* 9:e97282.
18. Pasi A, et al. (2011) The conundrum of HLA-DRB1*14:01/*14:54 and HLA-DRB3* 02:01/*02:02 mismatches in unrelated hematopoietic SCT. *Bone Marrow Transplant* 46:916–922.
19. Telenti A, et al. (2016) Deep sequencing of 10,000 human genomes. *Proc Natl Acad Sci USA* 113:11901–11906.
20. Li H, Durbin R (2010) Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* 26:589–595.
21. Raczy C, et al. (2013) Isaac: Ultra-fast whole-genome secondary analysis on Illumina sequencing platforms. *Bioinformatics* 29:2041–2043.
22. Edgar RC (2004) MUSCLE: A multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics* 5:113.
23. Zhao H, et al. (2014) CrossMap: A versatile tool for coordinate conversion between genome assemblies. *Bioinformatics* 30:1006–1007.
24. Spellman S, et al. (2008) Advances in the selection of HLA-compatible donors: Refinements in HLA typing and matching over the first 20 years of the national marrow donor program registry. *Biol Blood Marrow Transplant* 14:37–44.