Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: XXX XXX XXXX
XXXXXXXXXXXX

Subject Section XXXX

HaploHTML5: A Comprehensive Pedigree Drawing and Haplotype Visualisation Web Application

Mehmet Tekman [1]*, Alan Medlar [2], Monika Mozere [1], Robert Kleta [1], and Horia Stanescu [1]

[1] Division of Medicine, University College London, London, NW3 2PF, UK and
[2] Institute of Biotechnology, University of Helsinki, Helsinki, 00014, Finland.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Haplotype reconstruction is an important tool for understanding the aetiology of human disease. Haplotyping infers the most likely phase of observed genotypes conditional on constraints imposed by the genotypes of other pedigree members. The results of haplotype reconstruction, when visualised appropriately, show which alleles are identical by descent despite the presence of untyped individuals. When used in concert with linkage analysis, haplotyping can help delineate a locus of interest and provide a succinct explanation for the transmission of the trait locus. Unfortunately, the design choices made by existing haplotype visualisation programs do not scale to large numbers of markers. Indeed, following haplotypes from generation to generation requires excessive scrolling back and forth. In addition, the most widely-used program for haplotype visualisation produces inconsistent recombination artefacts for the X chromosome.

Results: To resolve these issues, we developed HaploHTML5, a novel web application for haplotype visualisation and pedigree drawing. HaploHTML5 takes advantage of HTML5 to be fast, portable and avoid the need for local installation. It can accurately visualise autosomal and X-linked haplotypes from both outbred and consanguineous pedigrees. Haplotypes are coloured based on identity by descent using a novel A* search algorithm and we provide a flexible viewing mode to aid visual inspection. HaploHTML5 can currently process haplotype reconstruction output from Allegro, Genehunter, Merlin and Simwalk.

Availability: HaploHTML5 is licensed under GPLv3 and is hosted and maintained via Bitbucket.

Web Application: `http://mtekman.bitbucket.io/haplo_html5`

Source Code: `https://www.bitbucket.io/mtekman/haplo_html5`

Supplementary information: Supplementary data is available from *Bioinformatics* online.

Contact: m.tekman@ucl.ac.uk or h.stanescu@ucl.ac.uk

1 Introduction

Family studies remain an important approach to investigate the aetiology of monogenic disorders. Linkage analysis, together with haplotype reconstruction, is used to identify putative locations of disease traits. Linkage analysis tests whether a given gene region co-segregates with the trait locus, whereas haplotype reconstruction infers the phase information lost during genotyping, i.e. the parental origin of each allele. In doing so, regions of interest can be found using linkage analysis and those regions delineated with inferred recombinations from haplotype reconstruction. Once a region has been identified, candidate genes can be selected for sequencing based on information from sequence databases (tissue-specific expression, homology, etc) or, if no candidate presents itself, all genes from the identified region can be screened for mutations using, for example, exome sequencing (Bockenhauer *et al.*, 2012).

Many parametric linkage analysis programs also perform haplotype reconstruction based on maximum likelihood. However, to integrate these analyses together requires advanced visualisation methods to intuitively display haplotypes together with the pedigree structure and to colour haplotypes based on identity by descent (IBD).

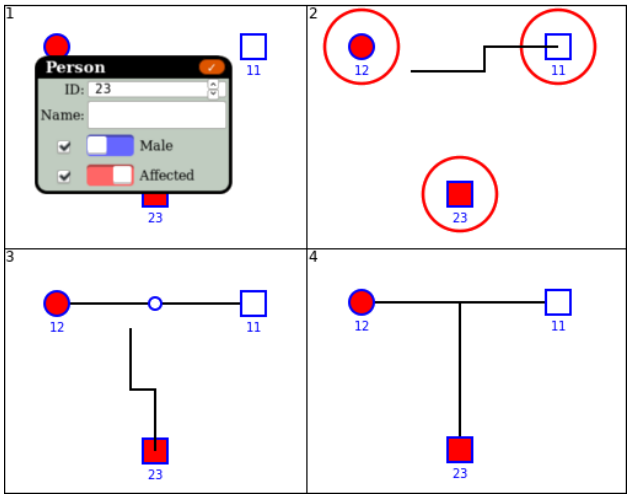

Fig. 1. Pedigree Drawing View show the four stages of creating a pedigree: (1) Adding individuals and modifying their properties, (2) Joining mates with a Mateline with anchor points made visible with red circles, (3) Joining offspring to their parents through a Childline with anchor points made visible with white circles, (4) Completing a trio.

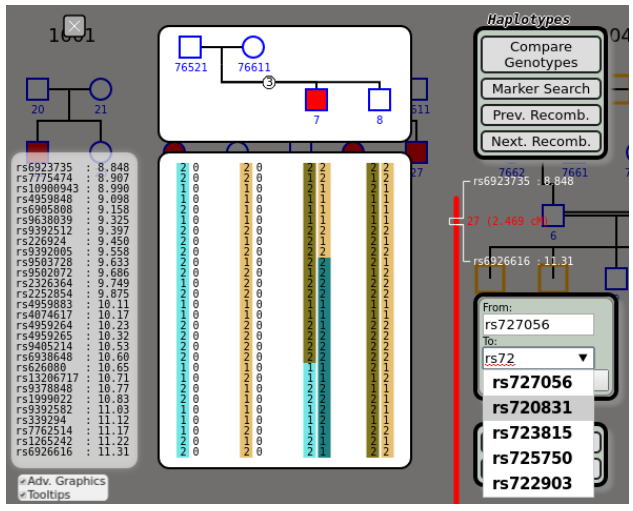

Fig. 2. Haplotype Comparison View, presenting: (Left) marker names and genetic positions; (Top-Centre) four individuals selected from a larger pedigree with a degree of separation of 3 generations between individuals 76521 and 76611 and individuals 7 and 8; (Centre) haplotypes coloured by identity by descent for the selected individuals, with two recombinations being shown in individual 7 at flanking markers rs9392005-rs626080 on separate alleles; (Top-Right) context-dependent buttons; (Centre-Right) floating region indicator with the top handle at rs6923735 and the bottom handle at rs6926616 depicting a view that spans 27 markers and 2.469 cM; (Bottom-Right) marker search window with drop-down autocompletion in progress.

There are many programs available for visualization, with Haplopainter being the most popular by number of citations (Thiele and Nürnberg, 2005). However, our experience with Haplopainter has shown that viewing haplotypes in-line with the pedigree does not scale to large numbers of markers. Indeed, to compare haplotypes between generations requires excessive scrolling and the user has to re-identify the same region of interest over and over again in each successive generation. In addition, Haplopainter does not always correctly display which alleles are IBD, creating inconsistent recombination artefacts for the X chromosome.

To resolve these issues we present HaploHTML5, a novel web application for haplotype visualization and pedigree drawing. HaploHTML5 is designed specifically for navigating high numbers of markers, providing a more intuitive viewing mode compared to other programs. We identified the cause of Haplopainter's IBD colouring issues on the X chromosome (speculative, it would be perfect if it was a flaw in their method and not just a bug, needs to be reworded if just a bug) and present a novel A* search-based method to ensure haplotypes are displayed correctly. Finally, HaploHTML5 is web-based and therefore runs in any HTML5-compatible browser and does not require local installation. Despite being web-based, it is fast and the user experience is similar to a native application, utilizing menus and a drag and drop interface.

2 Approach

HaploHTML5 is a comprehensive web application for haplotype visualisation and pedigree drawing. The interface is shown in Figure 2 (it would be better to have a comprehensive overview with annotations pointing out features). Here we will enumerate and expand upon the core features.

2.1 Pedigree Drawing

Pedigrees are drawn with a simple drag and drop interface (I am guessing, I have not used the program recently, change if wrong) and are compliant with the Pedigree Standardization Work Group (PSWG) specification (Bennett *et al.*, 1995, 2008). The standard is already familiar to clinicians and allows individuals in the pedigree to be annotated with patient metadata.

Individuals are added to the pedigree using drag and drop(?) and the properties of that individual (sex, affection status, etc) edited from a dialog box. Relationships between individuals are added by drawing *matelines* and *childlines*. Matelines indicate marriages and childlines connect children to their parents' mateline. Lines snap to context-dependent anchor points that become visible when adding relationships (Figure 1). Both members of each mateline are vertically aligned with one another and move together as a single unit. Siblings bound to the same mateline are similarly aligned automatically.

Projects can contain multiple families, and complex consanguineous relationships are automatically detected and represented via double-lines. Pedigrees can be loaded and saved from local browser storage. Pedigrees can be imported and exported in standard LINKAGE (pre-makeped) format.

2.2 Haplotype Visualisation

Haplopainter, visualises haplotypes in-line with the pedigree (is this true of any other program?). However, for large numbers of markers, comparing the same region of interest between several generations is inconvenient, requiring the user to scroll down and find the same locus over and over again. In our experience, this makes the procedure unnecessarily time-consuming and can lead to errors.

In HaploHTML5, haplotype comparison is performed in a separate viewing mode. Selected individuals are aligned vertically and grouped by family. Haplotypes are displayed underneath each individual, allowing for side-by-side comparison, irrespective of generation. The relatedness of individuals can be optionally displayed, with relationship lines stating their degree of separation (see Figure 2).

Haplotypes can be inspected by scrolling the page or dragging the region indicator displayed on the chromosome overview. The chromosome

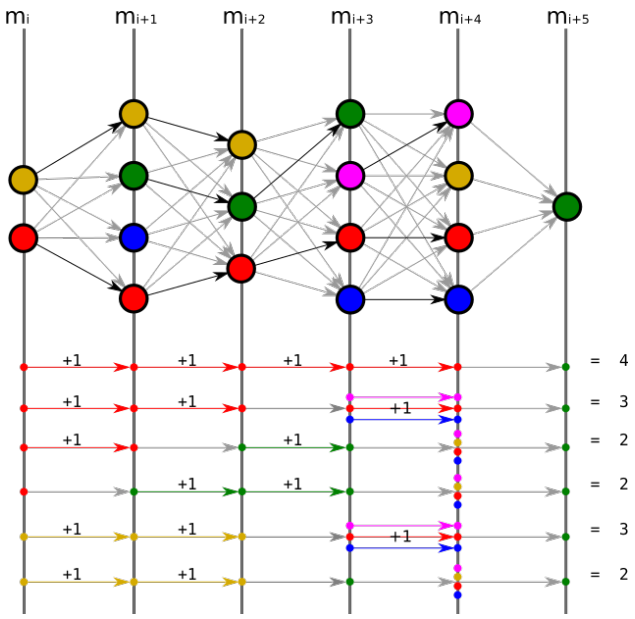

Fig. 3. (Top) A multi-layer network graph depicting five founder alleles as uniquely coloured nodes within a marker locus stretching from m_i to m_{i+5}. Black arrows depict desired contiguous founder group stretches, and grey arrows indicate recombinations from one founder group to another. (Bottom) Six possible routes explored by the search algorithm, with contiguous stretches being rewarded $+1$ to the total path sum. The first route has the largest path sum of 4 and is the most optimal path in the region under consideration.

overview identifies the location of the currently displayed region, shows flanking marker names and the length of the region in centiMorgans. The region of interest can be expanded or contracted by dragging the handles on either side of the region indicator in the chromosome overview. Alternatively, the region of interest can be defined by specific markers. Flanking markers can selected from either a drop-down autocompletion box (shown in Figure 2) or by clicking a button to select the next/previous recombination point.

2.3 IBD Colouring

In HaploHTML5, haplotypes are coloured by IBD by converting the task of resolving ambiguous parentage into a path-finding problem and performing A* search to determine the path with least recombinations. A* search is an efficient path-finding algorithm used in real-time mapping applications (Stout, 1996; Seet *et al.*, 2004). (these citations look a bit random, is there a nice review article instead?) In a connected graph with weighted edges, an optimal path between two nodes is found by minimising the total edge-cost.

A* search is a best-first search algorithm that, in the process of finding the optimal path, maintains a "frontier" of nodes from which the node deemed most likely to be the next intermediate node on the path to the target node is selected. The search procedure is admissible on condition that the estimated cost to the target node is not greater than the true cost from the next intermediate node to the target node (Hart *et al.*, 1968), under the following heuristic: $f(n) = g(n) + h(n)$, where n is an intermediate node on the path, $g(n)$ is the cumulative cost of the path (from start node to n), and $h(n)$ is the heuristic that estimates the lowest cost from n to the target node.

In a genomic context this can be conceptualised as a multi-layer network graph, where edges only exist between consecutive layers. Each layer represents an individual-marker locus, and each node a distinct

Table 1. Trio allele configurations, with applicable Mendelian child genotypes describing four levels of inheritance ambiguity. An allele is Allele-Specific (AS) if it can unambiguously match a specific allele from a given parent, and Parent-Specific (PS) if a specific parent can be matched.

Parental Alleles		*Child Alleles*			
Case	*GTs*	*Mend.*	*AS*	*PS*	*Group*
Homozygous	AA | AA	AA	××	××	Orphaned
	BB | BB	BB	××	××	Orphaned
	AA | BB	AB	××	✓✓	Bound-Parent
Heterozygous	AB | AB	AA	✓✓	××	Bound-Allele
		BB	✓✓	××	Bound-Allele
		AB	✓✓	××	Bound-Allele
Homozygous with Heterozygous	AA | AB	AA	××	××	Orphaned
		AB	×✓	×✓	Locked
	BB | AB	BB	××	××	Orphaned
		AB	✓×	✓×	Locked

founder allele (Figure 3). The algorithm traverses from one end of a chromosome to the other under the heuristic of minimizing the number of recombinations. A maximum of $2f$ nodes are possible in each layer, where f is the number of founders. This number is often far smaller due to the manner in which the graph is initialised (see Section 3.2.1).

2.4 File Format Support

HaploHTML5 was designed to primarily accept phased genotypes in XXX format (do programs output phased genotypes in the same file format or is every program different?), however, it can additionally utilise supplementary gene flow information output by specific programs. Some applications incorporate this information within the main haplotypes output file, whereas others provide it in a supplementary file. Merlin (Abecasis *et al.*, 2002), for example, outputs founder alleles in a file called `gene.flow`. Allegro (Gudbjartsson *et al.*, 2005) and Simwalk (Sobel *et al.*, 2002) both output the optimal descent graph, stating the gene flow between generations. Haplotypes from Genehunter (Kruglyak *et al.*, 1996) and Merlin do not include sex data, requiring it be inferred through parentage for all but the last generation whose sex is declared unknown. This information can be provided separately. Further marker data (e.g. SNP ID, genetic distance, etc) is displayed upon discovery and preserved across successive sessions in local storage.

3 Methods

3.1 Web Technologies

HaploHTML5 is implemented using HTML5 and JavaScript. Unlike other haplotype visualisation programs that require local installation (including installation of dependencies), HaploHTML5 runs in any compliant web browser. Features like A* search IBD colouring (described in detail in Section 3.2) make use of JavaScript's typed arrays to eliminate the redundancy of the default numeric float type, compacting large numeric sets into small 8 bit decimal arrays. Fast 2D graphics rendering is performed using the HTML5 canvas-based KineticJS library (`http://kineticjs.com`). KineticJS renders graphics to layers which are implemented as separate canvas elements. HaploHTML5 uses two layers for passive and active drawing (what is passive and active drawing?), with graphics being transferred between the two as necessary to limit the number of redraw operations. Animation is used to transition between the pedigree drawing and haplotype comparison modes. Visual effects can be disabled if, for example, the browser does not support hardware acceleration

and performance is degraded. (something about compatibility? minimum versions of web browsers?)

3.2 A* search IBD Colouring

The procedure to ensure that haplotypes are coloured appropriately to reflect identity by descent is split into three distinct phases: initialising the network graph, determining the optimal path and final cleanup operations.

3.2.1 Graph Initialisation

We initialise the graph with a top-down pass of each pedigree to create founder allele nodes and, for each non-founder, define the set of alleles that can be inherited from each parent at each locus. Founder alleles are inherited in a non-parental specific fashion where any valid genotype configuration would contribute to the set under a pre-set disease model. This is to ensure that consanguineous pedigrees are permitted by making no assumptions upon whether a parental allele is maternal or paternal. (what does the blue text mean? how is the disease model involved in haplotyping?) The complexity of resolving parental alleles with child alleles can be summarised by four tiers of allele-pair specificity (in ascending order of complexity): Locked, Bound-Parent, Bound-Allele, and Orphaned. Locked alleles can unambiguously assign at least one allele to a specific parent and to a specific allele within that parent. Bound-Parent alleles can match each allele to a parent, but not to a specific allele within that parent. Bound-Allele is the opposite; where each allele matches one in either parent, but neither allele is parent-specific (I thought the input was phased genotypes, would this not make everything Bound-Parent? can this text therefore be simplified?). Orphaned provides no specificity. A summary of valid genotype configurations and their specificity is outlined in Table 1. In X-linked analysis, the process is adapted to reflect the fact males are hemizygous and therefore have only a single allele.

3.2.2 Finding the Optimal Path

A parental exclusion group is determined for non-founders based upon the set of previously-derived parental founder groups. (what does the blue text mean? what is a parental exclusion group? this is never defined!) The A* search algorithm then attempts to find the optimal path through each individual-marker layer of networked nodes, aiming to minimize the total number of recombinations. The procedure is outlined in Algorithm 1. The set of possible paths through the graph are restricted by which founder alleles can legally be assigned and by the parental exclusion set. ("parental exclusion set" needs to be defined)

A working set of eight examined paths are expanded upon with paths added/removed as determined by their respective running totals upon each iteration of the working set. Paths with the same running total of recombinations are excluded from the working set in order to encourage more diversity. (how does this relate to the frontier of nodes being expanded by A*? Also, does this make sense? Two paths can surely have the same number of recombinations but be very different, right?)

The path with the minimum number of recombinations from the working set is output, and a set of the founder groups within is used for parental exclusion upon evaluation of offspring chromosomes (what does the blue text mean?).

3.2.3 Cleanup Operations

It may be required for the search algorithm to be run multiple times for the same chromosome under an alternative set of parental exclusion groups (what does this mean?), should a valid path not be found in the first attempt (I would not call this cleanup). Upon full evaluation of a pedigree, the network graphs associated with each marker locus are discarded and only the optimal paths are stored.

```
begin
    prefetch relevant chromosome
    maxPath ← global max. num. of paths to explore
    numMark ← total num. markers in chromosome
    P ← parental exclusion set of illegal colours
    complete ← initialise empty list of completed paths.
    frontier ← array of working paths, initialised with array of
    colours at first marker-layer in chromosome as starting points
    while frontier > 0 do
        sort frontier by desc. length and select first maxPath
        a ← shift frontier to select first active path
        C ← colours in path a at marker-layer length a - 1
        for c ∈ C do
            s ← perform lookahead and count contiguous stretch of c
            if s < 1 or c ∈ P then skip c
            r ← clone path a with c appended s times
            if length r > numMark - 1 then
                push r to complete
            else
                push r to frontier
            end
        end
    end
    sort complete by desc. length
    return shift complete
end
```

Algorithm 1: A* search upon a chromosome of pre-initialised multi-layer network graph, with founder alleles represented as colours. Sort operations apply in-place, and shift operations truncate from the head of an array.

3.3 Comparative Analysis

The purpose of haplotype visualisation is to help distinguish between loci whose segregation is concordant with the disease trait from those that are not. In such regions there will be a clear distinction in IBD information, defined by the disease model, between affected and unaffected individuals. To give researchers an overview of their data, HaploHTML5 defines a score based on the disease model that is plotted on the chromosome overview to identify such regions. To give an example of how this works, assuming a dominant disease trait, if at a given locus all affected individuals have either $\{red, red\}$ or $\{red, green\}$ founder alleles and the unaffected individuals have $\{green, green\}$, then this region will receive a high score. If, on the other hand, the model were recessive, it would receive a lower score. The score is additive across pedigrees under the assumption of genetic homogeneity. Scores can be exported to a text file.

(you need to provide an expression defining this score concretely, otherwise it is too vague)

4 Discussion

HaploHTML5 provides a unified environment to create, analyse, and visualize pedigrees together with their associated haplotypes. Pedigree creation allows for large families to be drawn using the mouse and exported or saved to local storage between sessions. Processing for IBD colouring is highly efficient and the results viewable across multiple families simultaneously. We provide several methods to inspect the displayed haplotype data including: displaying haplotypes between user-specified flanking markers, skipping between recombination points and scoring by haplotype consistency with the disease model.

4.1 Path-finding approach

The A* search is not restricted to SNPs, but can accept polymorphic markers (e.g. VNTRs and STRs) (can the current implementation handle polymorphic markers or is this future work?). The path-finding approach processes each chromosome separately with the only interaction between homologs being subject to the parental exclusion groups that are passed on from the last chromosome processed. (I don't understand blue text, of course chromosomes are processed separately, they are independent, why is something passed from the last chromosome processed?) In the future this could be adapted to explore monosomy, trisomy, and tetrasomy cases. (I changed this to future because the input is phased genotypes, which would exclude doing this)

The parental exclusion parameters carried across homologous chromosomes may appear to hinder the analysis of consanguineous pedigrees where the same founder allele group may appear in multiple sets due to the non-singular path of descent that the group may take to reach an individual. This is trivially resolved however, as consanguinity is bound to the parental Mateline and a cursory step will slacken the exclusion constrains by intersecting all homologous sets in order to find and remove any overlapping groups. (this is very confusing, it needs to be simplified)

Another conceivably useful feature of the approach is the flexibility in which it infers parental alleles whilst never explicitly assuming a maternal or paternal allele based on ordering, but testing for compatibility with the child alleles across all orientations and leaving the resolution to the path-finding process. (confusing) Though the current HaploHTML5 release (v1.51) requires phased genotypes as input, the potential to resolve unphased genotypes exists as a possibility. (this means it would be doing haplotype reconstruction, maybe don't say this, a reviewer is going to pick on it)

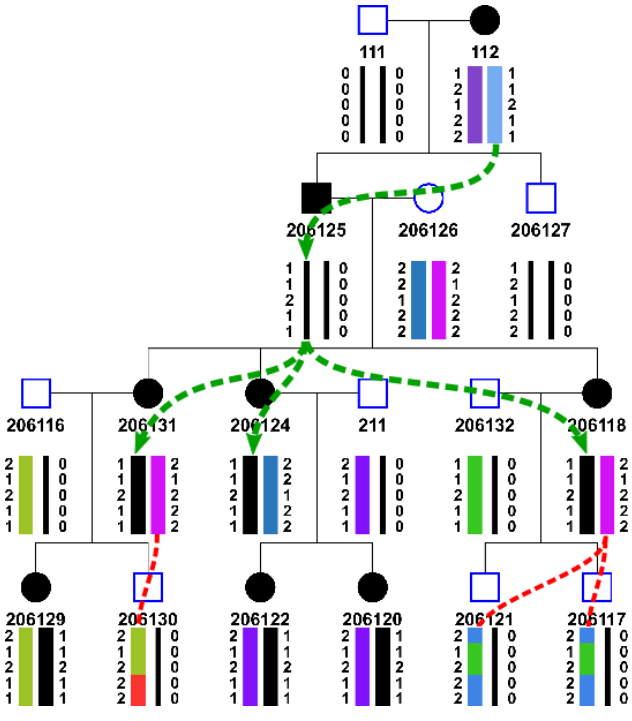

Fig. 4. HaploPainter interpretation of a five marker X-linked analysis. Colours indicate identity by descent. Arrows are overlayed to show the true flow of genetic data based on genotypes, with green indicating inconsistent colouring between successive meioses and red depicting erroneous inheritance.

4.2 Visualisation Accuracy

The haplotype visualisation performed by HaploHTML5 was compared with HaploPainter. For all autosomal pedigrees analysed, the same points of recombination were identified from Allegro (ihaplo.out), Genehunter (haplo.chr), Merlin (merlin.chr), and Simwalk (HEF.ALL) output files. Beyond the simple pedigrees, we have used HaploHTML5with four non-trivial families: autosomal dominant (27 members, 23-bit); a highly consanguineous autosomal recessive (24 members, 29-bit); and an X-linked dominant (17 members, 15-bit).

For the X-linked pedigree, HaploPainter produced dubious recombination artefacts (Figure 4) where the last generation of individuals (particularly 206130, 206121 and 206117) appear to have undergone multiple recombinations within a short genetic distance (< 1 cM). These results are in contrast to HaploHTML5, that correctly shows the correct IBD colouring (see side-by-side comparison Figure 5). (I think it would be a mistake to put this in the discussion, but should be in the intro or approach)

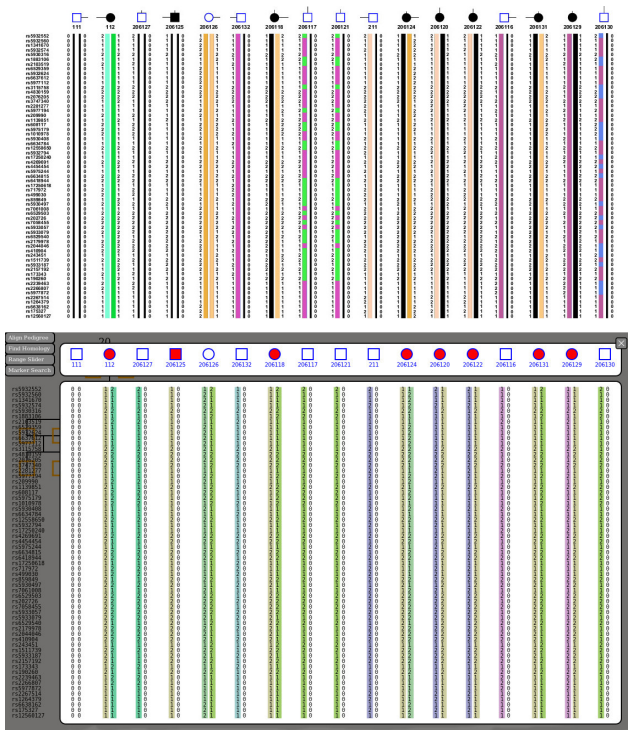

Fig. 5. A comparison of the X-linked dominant pedigree showing a mid-region of chrX spanning 72 markers. (Top) HaploPainterwith output modified only for horizontally alignment, and (Bottom) HaploHTML5showing all members via default Comparison View.

4.3 Privacy

HaploHTML5 currently operates in-browser via either local or web deployment, and analyses are restricted to a single user. In the interests of scientific collaboration, it is likely that the end-user would want to share their analysis with other researchers working on the same project. Due to the sensitivities of patient data, however, as well as the possibility of identifying individuals based on pedigree structure alone, HaploHTML5 was designed with the intention of not requiring any client-server communication after the web application has loaded. The discretion

of patient data is ultimately left to the user, and we provide the option to strip patient names and other annotations on export.

4.4 Future Work

HaploHTML5 was built on top of KineticJS because of its stability; active development being frozen since 2014. However, in order for HaploHTML5 to benefit from performance improvements it will need to migrate to one of the primary alternatives, either ConcreteJS (`http://concretejs.com/`), by the author of KineticJS or KonvaJS (`https://github.com/konvajs/`) that both other distinct features and advantages that will need to be evaluated.

Future versions of HaploHTML5 will aim to integrate the visualization and creation modes to provide more flexibility, for example, to allow for modifying an existing pedigree after haplotype data is loaded (this does not make sense, why would you do this?). Additional features could include SVG export and selective visualisation of multiple regions to help produce publication quality figures.

Acknowledgements

R.K. is supported by St. Peter's Trust for Kidney, Bladder and Prostate Research, the David and Elaine Potter Charitable Foundation, Kids Kidney Research, Garfield Weston Foundation, Kidney Research UK, the Lowe Syndrome Trust, the Mitchell Charitable Trust, and the European Union, FP7 (grant agreement 2012-305608 "European Consortium for High-Throughput Research in Rare Kidney Diseases (EURenOmics)".

Conflict of interest: None declared.

References

Abecasis, G. R., Cherny, S. S., Cookson, W. O., and Cardon, L. R. (2002). Merlin-rapid analysis of dense genetic maps using sparse gene flow trees. *Nature Genetics*, **30**(1), 97–101.

Bennett, R. L., Steinhaus, K. A., Uhrich, S. B., O'Sullivan, C. K., Resta, R. G., Lochner-Doyle, D., Markel, D. S., Vincent, V., and Hamanishi, J. (1995). Recommendations for standardized human pedigree nomenclature. *Journal of Genetic Counseling*, **4**(4), 267–279.

Bennett, R. L., French, K. S., Resta, R. G., and Doyle, D. L. (2008). Standardized human pedigree nomenclature: update and assessment of the recommendations of the National Society of Genetic Counselors. *Journal of genetic counseling*, **17**(5), 424–433.

Bockenhauer, D., Medlar, A. J., Ashton, E., Kleta, R., and Lench, N. (2012). Genetic testing in renal disease. *Pediatric Nephrology*, **27**(6), 873–883.

Gudbjartsson, D. F., Thorvaldsson, T., Kong, A., Gunnarsson, G., and Ingolfsdottir, A. (2005). Allegro version 2. *Nat Genet*, **37**(10), 1015–1016.

Hart, P. E., Nilsson, N. J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, **4**(2), 100–107.

Kruglyak, L., Daly, M. J., Reeve-Daly, M. P., and Lander, E. S. (1996). Parametric and nonparametric linkage analysis: a unified multipoint approach. *American Journal of Human Genetics*, **58**(6), 1347–1363.

Seet, B.-C., Liu, G., Lee, B.-S., Foh, C.-H., Wong, K.-J., and Lee, K.-K. (2004). A-star: A mobile ad hoc routing strategy for metropolis vehicular communications. In *International Conference on Research in Networking*, pages 989–999. Springer.

Sobel, E., Papp, J. C., and Lange, K. (2002). Detection and Integration of Genotyping Errors in Statistical Genetics. *American Journal of Human Genetics*, **70**(2), 496–508.

Stout, B. (1996). Smart moves: Intelligent pathfinding. *Game developer magazine*, **10**, 28–35.

Thiele, H. and Nürnberg, P. (2005). HaploPainter: a tool for drawing pedigrees with complex haplotypes. *Bioinformatics*, **21**(8), 1730–1732.