Smooth Descent: a ploidy-aware algorithm to improve linkage mapping in the presence of genotyping errors

Alejandro Thérèse Navarro

Wageningen University & Research <https://orcid.org/0000-0003-0281-8423>

Peter M Bourke

Wageningen University & Research

Eric van de Weg

Wageningen University & Research

Corentin Clot

Wageningen University & Research

Paul Arens

Wageningen University & Research

Richard Finkers

Wageningen University & Research

Chris Maliepaard (✉ chris.maliepaard@wur.nl)

Wageningen University & Research <https://orcid.org/0000-0002-7319-5270>

Research Article

Keywords: Linkage mapping, genotyping error, identity-by-descent, imputation

Posted Date: June 27th, 2022

DOI: <https://doi.org/10.21203/rs.3.rs-1165750/v2>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Abstract

Linkage mapping is an approach to order markers based on recombination events. Mapping algorithms cannot easily handle genotyping errors, which are common in high-throughput genotyping data. To solve this issue, strategies have been developed, aimed mostly at identifying and eliminating these errors. One such strategy is SMOOTH (van Os et al. 2005), an iterative algorithm to detect genotyping errors. Unlike other approaches, SMOOTH can also be used to impute the most probable alternative genotypes, but its application is limited to diploid species and to markers heterozygous in only one of the parents. In this study we adapted SMOOTH to expand its use to any marker type and to autopolyploids with the use of identity-by-descent probabilities, naming the updated algorithm Smooth Descent (SD). We applied SD to real and simulated data, showing that in the presence of genotyping errors this method produces better genetic maps in terms of marker order and map length. SD is particularly useful for error rates between 5% and 20% and when error rates are not homogeneous among markers or individuals. With a starting error rate of 10%, SD reduced it to $\sim 5\%$ in diploids, $\sim 7\%$ in tetraploids and $\sim 8.5\%$ in hexaploids. Conversely, the correlation between true and estimated genetic maps increased by 0.03 in tetraploids and by 0.2 in hexaploids, while worsening slightly in diploids (~ 0.0015). We also show that the combination of genotype curation and map re-estimation allowed us to obtain better genetic maps while correcting wrong genotypes. We have implemented this algorithm in the R package SmoothDescent.

Key Message

Smooth Descent detects genotyping errors and imputes their true values. Used iteratively during linkage mapping, it improves map order and genotype correctness.

Introduction

Linkage mapping is the process by which a set of markers segregating in a population are grouped and ordered. Each marker is placed within a linkage group, oftentimes corresponding to a chromosome, and given a genetic position within that group. The usefulness of genetic mapping has made it a consistent tool during the past century: starting with the study of trait co-segregation in *Drosophila* (Sturtevant 1913), continuing to the proof of the linear structure of genes and chromosomes (Benzer 1959), and the first QTL analyses (Lander and Botstein 1989). Its relevance has not diminished nowadays, as it enables the study of genomic patterns of recombination, thereby highlighting the functional and structural properties of a genome. Linkage maps are also an essential tool for studies in organisms without a reference genome (e.g. Hu et al. 2021), in plant and animal QTL studies and in the assembly and improvement of genome sequences (Mascher and Stein 2014; Fierst 2015).

Genetic mapping algorithms have been greatly influenced by the progress of genotyping. As newer technologies provided larger marker sets, novel mapping algorithms had to be developed to handle growing numbers of markers (Cheema and Dicks 2009). The most recent genotyping techniques, sequencing-based methods such as genotyping by sequencing (Elshire et al. 2011) or whole genome

sequencing (Varshney et al. 2014), are able to identify and genotype millions of variants in a single analysis but suffer from a common drawback: an increased proportion of genotyping errors. That is particularly problematic for the purpose of genetic mapping, since the ordering algorithms on which many mapping approaches rely are notoriously sensitive to errors (Hackett and Broadfoot 2003; van Os et al. 2005; Cartwright et al. 2007). The main consequences of genotyping errors on genetic map construction are sub-optimal map orders and inflated map lengths (i.e. >100 cM). The general strategy to deal with this problem has been to detect and eliminate highly spurious markers (Lincoln and Lander 1992; van Os et al. 2005; Cartwright et al. 2007; Wu et al. 2008; Cheema and Dicks 2009; Liu et al. 2014; Rastas et al. 2016), although the errors can also be explicitly modelled, increasing the number of retained markers (Bilton et al. 2018).

Polyploidy, the presence of more than two chromosome sets in an organism, is a relatively common condition in crop species (e.g. rose, potato, strawberry, sugarcane, wheat) that poses special challenges in linkage mapping. In autopolyploids, which usually originate from genome duplication within a single species, polysomic segregation and double reduction require specialized methods of linkage estimation (Bourke et al. 2018a). In allopolyploids, arising from interspecific hybrids, segregation usually follows a diploid pattern, but genotyping can be more inaccurate due to the difficulty of distinguishing between homoeologous sequences (Kaur et al. 2012). Although these issues have been addressed with specialized tools and approaches (Glover et al. 2016; Bourke et al. 2018b), these tools were not designed with consideration of the high error proportion in sequencing-based genotype data, and due to the unique challenges of polyploids, diploid-oriented tools cannot be used.

In this study, we aimed to develop a ploidy-aware approach that would help in using high-throughput genotyping information for genetic mapping, without discarding vast amounts of data due to an increased error rate. Therefore, we adapted SMOOTH (van Os et al. 2005), a simple and efficient method for error detection and correction based on the identification of unlikely genotype scores. The original algorithm was only applicable to diploids and to markers heterozygous in only one of the parents. By using identity-by-descent (IBD) probabilities, we extended this model to any ploidy and marker segregation type. Additionally, we changed the k-nearest neighbours approach used in SMOOTH to an interval-based approach, which improves identification and correction of errors in maps with a heterogeneous marker distribution. We term this updated method Smooth Descent, the IBD-based descendent of SMOOTH. Similar to the original algorithm, Smooth Descent requires a preliminary map to be applied, thus it should be thought of as part of an iterative mapping approach, so that with each round of mapping and smoothing a better map is obtained.

This algorithm has been implemented as an R package called ‘SmoothDescent’. The package also generates so-called “graphical genotypes” that can be used as a quality assessment tool by researchers, along with other visualizations.

Materials And Methods

Smooth Descent approach

SMOOTH and Smooth Descent are both based on the same principle: comparing an observation (error sensitive) and expectation (error tolerant) matrix of genotypes and identifying as errors the inconsistencies between both matrices. The difference lies in the way genotypes are expressed in both approaches: as raw genotype scores in SMOOTH, and as Identity-by-Descent (IBD) probabilities in Smooth Descent.

Naive IBD probabilities

The algorithm begins with parental phasing and a preliminary map that indicates the order and distances of markers. A number of methods can be used, experimental and computational, to obtain parental phasing (Browning and Browning 2011; He et al. 2018; Al Bkhetan et al. 2021) and a preliminary map (Rastas 2017; Bilton et al. 2018). In our software, mapping is performed by polypmapR (Bourke et al. 2018a) and parental phasing is expected to be obtained by the researcher.

Phased parental genotypes are expressed using the homologue matrix H , in which columns represent parental homologues and rows are markers, ordered according to the preliminary map. The number of columns, H_1 to H_p , will be the sum of parental ploidies: in a diploid cross, there would be 4 columns, two for each parent; in a tetraploid cross, 8 and in a cross between a diploid and a tetraploid, 6 columns would be specified. The first set of columns correspond to the homologues of the first parent, and the rest to the homologues of the second parent. Each cell of the H matrix contains a 0 when that homologue holds the reference allele A at that marker, and 1 if it holds the alternative allele B. Because of this, only biallelic markers can be used in Smooth Descent. The choice of reference allele will not influence IBD calculations, and thus it can be done at random. For a diploid cross, an example of H would be:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In a tetraploid:

$$H = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

First, we will calculate the error-sensitive, observed IBD probabilities or naïve IBD probabilities. For that we need to obtain all possible homologue combinations that can be inherited, which we denote as *configurations* with the symbol c_i . This will depend on the number of homologues that parent 1 and parent 2 pass on to the offspring, which in turn depends on their ploidy.

In the case of a diploid, parent 1 can provide two homologues: H_1 or H_2 ; while parent 2 can provide another two: H_3 or H_4 . Thus, there are four configurations, $c_1 = \{H_1, H_3\}$, $c_2 = \{H_1, H_4\}$, $c_3 = \{H_2, H_3\}$, $c_4 = \{H_2, H_4\}$. On the other hand, in a tetraploid example, each parent will provide two homologues. Thus, a single parent can provide any of six pairs of homologues: (H_1, H_2) , (H_1, H_3) , (H_1, H_4) , (H_2, H_3) , (H_2, H_4) or (H_3, H_4) . Moreover, due to multivalent formation, double reduction scenarios are possible, meaning that parent 1 could also contribute (H_1, H_1) , (H_2, H_2) , (H_3, H_3) or (H_4, H_4) . If both parents are tetraploid, this amounts to 100 possible configurations. However, since double reduction is relatively rare, and for the sake of simplicity, it has not been considered in this implementation of Smooth Descent. Thus, we will only consider the 36 configurations possible, *i.e.* we assume that no double recombination occurs.

The next step is to determine the marker *dosage*, d_j (of the alternative allele) of each configuration. This must be calculated independently for each marker. For one marker, matrix H assigns either 0 or 1 to each parental homologue. The inherited dosage of that configuration is simply the sum of the associated parental homologues. For instance, for the first marker (row) in the diploid example, $c_1 = \{H_1, H_3\}$ thus $d_1 = 1 + 0 = 1$ while $c_3 = \{H_2, H_3\}$ thus $d_3 = 0 + 0 = 0$. For the first marker of the tetraploid example, $c_1 = \{H_1, H_2, H_5, H_6\}$ thus $d_1 = 0 + 1 + 0 + 0 = 1$ etc.

To obtain IBD probabilities for one individual, one must consider the observed genotype of that individual. Since an individual must hold one of the described configurations, only those configurations whose dosage matches the observed genotype are *possible configurations*. For each genotype g , we denote the set of possible configurations as C_g where k_g the number of possible configurations. When no double reduction is considered, all configurations are equally probable, thus the IBD probability of H_i is:

$$p(H_i|g) = \frac{\sum_{j \in C_g} f(c_j, H_i)}{k_g}$$

Where $f(c_j, H_i)$ is an indicator function that takes the value 1 if H_i belongs to c_j and 0 otherwise.

$$f(c_j, H_i) : \begin{cases} \text{if } H_i \in c_j \text{ then } 1 \\ \text{if } H_i \notin c_j \text{ then } 0 \end{cases}$$

Thus, for each row of H we can calculate associated homologue probabilities depending on the genotypes. Applying this formula for all markers and all homologues for a single individual yields an IBD probability matrix which we will denote as I_0 . We call the obtained IBD probabilities naïve because each marker is estimated independently of adjacent markers.

Weighted average IBD

One of the methods implemented for IBD prediction in Smooth Descent is based on a local weighted average of observed IBD around a marker, inspired by SMOOTH's proposal and similar to the procedure suggested by Wu et al. (2008). This requires two steps: first, defining the set of local markers and second, the weights to be applied to each marker.

Let's start with marker m_i . The set of local markers, L_i are those markers closer than l from m_i , where l is a chosen distance threshold (we chose $l = 10$ cM, but a different threshold can be provided). Additionally, low-informative markers will be excluded from the local set. We defined these as markers for which the observed IBD probability is within the 0.3–0.7 range (see Error Prediction section for more information). Since the predicted IBD is calculated per homologue, this means that L_i will differ slightly per homologue.

The weight for the observed IBD probability at marker m_j will be proportional to the chance that there is no recombination between m_i and m_j . This no-recombination probability can be obtained from the distance estimates:

$$1 - \rho_{ij} = 1 - f(d_{ij})$$

Where $1 - \rho_{ij}$ is the probability of no recombination and $f(d_{ij})$ is a reversed mapping function of the distance between m_i and m_j . Three functions have been implemented: Morgan's, Haldane's and Kosambi's. We can define the weights as:

$$w_j = \frac{1 - \rho_{ij}}{\sum_{k \in L_i} (1 - \rho_{ik})}$$

For each individual, the predicted IBD probability for marker m_i will then be the weighted average of all the markers in L_i for which $d_{ij} < l$ and the observed IBD probability is informative. Applying this along the I_0 matrix will allow us to calculate the predicted IBD matrix $\hat{I}_0$.

Error prediction

The principle of error prediction is to identify markers for which their observed and predicted IBD probabilities (i_0 and $\hat{i}_0$) disagree strongly, meaning that the observed genotype clearly indicates a homologue inheritance that does not match with most surrounding markers. More formally, an error can be identified if $\left| i_0 - \hat{i}_0 \right| > \delta$, where δ is an error threshold preferably above 0.7.

Due to this definition, low-informative markers (with observed probabilities between 0.3 and 0.7) must be excluded from the IBD prediction step. The contrast $\left| i_0 - \hat{i}_0 \right|$ will not reach a high value if either i_0 or $\hat{i}_0$ are close to 0.5. The observed IBD i_0 will be close to 0.5 if the observed inheritance is uncertain, which means we do not have enough information to discern whether that genotype is an error. The predicted IBD $\hat{i}_0$, should be close to 0.5 if the set of local markers have both high and low IBD probabilities, indicating that there is a local disagreement on inheritance. If low-informative markers are kept, even if many informative markers exist that clearly indicate homologue inheritance, the presence of low-informative markers will centralize the local weighted average and prevent identification of putative errors. Thus, low-informative markers should be removed from IBD prediction.

Hidden Markov model IBD

As previously stated, SD depends on a comparison between an error-sensitive, observed IBD matrix and an error-tolerant, predicted IBD matrix. We have presented weighted average approach suggested by the developers of SMOOTH, but hidden Markov models (HMMs) are also a common approach to obtain error-tolerant IBD estimates (Zheng et al. 2016, 2021; Mollinari and Garcia 2019). One such model has been included in SD, in particular that implemented within polyqtlR (Bourke et al. 2021), an expanded version of the TetraOrigin model (Zheng et al. 2016). Thus, error estimation can be performed by comparing naïve, error-sensitive, IBD probabilities to the weighted-average or HMM-based IBD probabilities, both error-tolerant.

Genotype correction and iteration

When a marker is detected as erroneous, a new genotype can be imputed by computing the most likely marker genotype according to the predicted IBD. The new set of genotypes can be used to calculate an improved map, and a corrected IBD matrix, I_1 . The previous steps can then be repeated to obtain a new error matrix E_1 and further improved genotypes. Thus, an iterative approach emerges, where in each iteration the genotypes are further corrected. As iterations progress the genetic map is expected to change less, and thus we are more certain of the achieved order. In view of caution regarding the introduction of artefacts, the error threshold was set at $\delta = 0.9$ during the first iteration, and then slowly decreased to 0.7 as iterations progress.

Best iteration selection

When using Smooth Descent, we must choose the best iteration according to some criterion. We offer the R^2 estimate of the second-order polynomial relationship between inter-marker distance d_{ij} and the recombination frequency r_{ij} (not to be confused with distance-based recombination frequency ρ used for IBD prediction). Unlike ρ , r is calculated during the mapping process through a likelihood or Bayesian method and is the basis of the final map order. In a good map, the relationship between r_{ij} and d_{ij} should

be mostly linear, where high recombination frequencies lead to high distances. Thus, the iteration with the highest R^2 can be considered the best.

Real data

Data from strawberry (*Fragaria x ananassa*) data was obtained from whole genome sequencing of 48 individuals from an F1 population. Variant discovery was performed using bcftools and genotyping with the R package “updog” (Gerard et al. 2018), allowing to genotype ~ 2400 unique markers across all chromosomes. Due to a skim sequencing strategy, many genotyping errors were expected and observed, which proved this dataset useful for testing our approach. Since strawberry is an allopolyploid with strict chromosomal pairing behaviour, the data could be treated as that of a diploid.

Data from sweet potato (*Ipomoea batatas*) was taken from Mollinari et al. (2020). Sequencing was performed using the polyploidy-optimized method described in GBSpoly (Wadl et al. 2018). The obtained read counts were passed to SuperMASSA (Serang et al. 2012) and genotypes were filtered for quality. For chromosome 15, a final count of 1513 genotypes were obtained for 287 individuals. These genotypes were used with SD, creating a preliminary map *de novo* and performing genotype correction on the genotypes. A single iteration of SD was used since no more improvements could be made subsequently.

Data from diploid potato was taken from Clot et al. (in preparation). The dataset consisted of 1536 full-sibs from a cross between two heterozygous clones C (USW5337.3) and E (77.2102.37). This population was skim sequenced to an average coverage of ~ 1.5x. Parent specific SNPs were called using bcftools v.1.13 and used to impute haplotypes in bins of 0.1Mbp resulting in 4893 female and 4735 male segregating markers. Smooth Decent was used based on physical position with five rounds at prediction interval of 1 Mbp and two final rounds with a prediction intervals of 5 and 10 Mbp respectively.

Simulated data

PedigreeSim (Voorrips and Maliepaard 2012), a program that simulates meiotic pairing and recombination for a range of pedigrees and ploidies, was used to simulate genotype data. Ten F1 populations of 100 diploid, tetraploid and hexaploid full-sib individuals were simulated, where every individual had one single chromosome containing 200 segregating markers distributed at variable densities along the chromosome. Error rates were applied randomly by changing the genotypes of 1%, 5%, 10%, 20% of the markers.

Additionally, two special cases were designed to test the effect of variable error rates across individuals (special case A) and across markers (special case B). Special case A contained 80 individuals with an error rate of 0.02 and 20 individuals with an error rate of 0.3. Special case B had the same error rate for all individuals, but variable across markers, ranging in a continuous curve along the chromosomes. The curve was defined as a smooth spline passing through the error rates 0.02, 0.1, 0.3, 0.02 and 0.1 at approximately 25 cM intervals along the chromosome. Thus, high error rate markers were located close to one another and at the centre of the chromosome.

Each genotype dataset was mapped using Smooth Descent with 10 iterations and tested using the weighted average or HMM method for computing error-tolerant IBD probabilities. To evaluate the effectiveness of SD, as well as the additional tools tested, three parameters were used: genotyping error, the percentage of genotypes different from the true genotypes; position correlation, the correlation between the true map positions and estimated map positions; and map length, the size of the estimated genetic map.

Software comparison

SD is a unique tool since it is the only available tool that aims at building polyploid (and diploid) linkage maps while simultaneously correcting genotyping errors. However, other tools exist that can perform one of the two functions. We have compared SD to polymapR (Bourke et al. 2018a), a polyploid linkage mapping approach that does not perform genotype correction; and to MAPpoly (Mollinari and Garcia 2019), a HMM approach that is able to correct genotypes and re-estimate marker positions but that does not re-compute linkage map orders.

Ten F1 populations equivalent to those described in the Simulated Data section were used. Genotyping errors were added at a rate of 1%, 5%, 10%, 15%, 20%, 25% and 30%. For each population and error rate four approaches were tested: polymapR, MAPpoly, SD using weighted average IBD prediction and SD with HMM IBD prediction. For both MAPpoly and SD the same preliminary map was provided. Additionally, the error prior provided to MAPpoly was the actual simulated error rate. Lastly, SD results were obtained with 5 iterations.

After running each approach, position correlation (correlation between true and estimated map positions), map length and computational run-time were obtained. Genotyping error was only calculated for SD and MAPpoly methods, since polymapR does not perform genotype correction.

Results

Simulated Data

A total of 10 populations per ploidy were tested with 6 different levels of genotyping error and two IBD prediction methods, showing the usefulness of Smooth descent (SD) in correcting genotypes, improving map orders and shortening map lengths (Fig. 1). It can be observed how the most impactful changes occur in the first few iterations: the biggest change in genotype correctness (Fig. 1 top), the largest improvement in genetic map correctness (Fig. 1 middle) and the biggest reduction in map length (Fig. 1 bottom). Note that map length was particularly short in polyploids (~ 60 cM in tetraploids and ~ 45 cM in hexaploids), an issue that seems to stem from preliminary map calculation.

Ploidy is an important factor in the behaviour of SD, moving from a genotype corrector at lower ploidies to a map corrector in higher ploidies. In diploid cases (Fig. 1 left column) SD is able to halve genotyping error (e.g. ~5% reduction in the 0.1 error rate scenario, Fig. 1 left top, Table 1) and to shorten map lengths,

especially in the highest error rate cases (e.g. ~30 cM shortening, Fig. 1 left bottom, Table 1). Nevertheless, in diploids, SD does not impact the correlation between true and estimated map positions, since the preliminary map is already highly correlated to the true map, although longer. In contrast, in polyploid scenarios reduction of genotyping error is smaller (Table 1, Fig. 1 middle and right columns), but the correlation between true and estimated maps improves substantially, especially in the hexaploid case. Map size reduction is of the same order, about 30cM. Importantly, for lower error rate cases, there was a slight increase in genotyping errors, although this did not affect the correlation with the true map or map size.

The two IBD prediction methods tested (weighted average and HMM) performed similarly in diploids but had some differences as ploidy increased. Genotyping error correction was better for the HMM as ploidy and initial error rate increased. Consequently, estimated map positions and map sizes were also better for the HMM in high ploidy and high error rate cases. However, at lower error rates the HMM method produced a larger increase in genotyping errors (Table 1, error rate 0.01).

Table 1

Average change between preliminary map and last iteration of Smooth Descent. Two error rate cases (0.01 and 0.1) are shown to illustrate the difference between the last iteration of SD and the preliminary error rate (Δ Error), correlation between the true map positions and estimated map positions (Δ Correlation) and map size (Δ Size). All values were calculated as last iteration – preliminary value (positive means increase, negative means decrease). Values are shown for all ploidies and IBD estimation methods (hmm is hidden Markov model and weight is weighted average method).

Error rate (%)	Ploidy	IBD method	Δ Error (%)	Δ Correlation	Δ Size (cM)
1	2	<i>hmm</i>	-0.27	-0.0008	-3.10
1	2	<i>weight</i>	0.16	-0.0032	-8.27
1	4	<i>hmm</i>	1.58	-0.0034	-6.05
1	4	<i>weight</i>	0.69	-0.0013	-7.94
1	6	<i>hmm</i>	1.76	-0.0031	-7.12
1	6	<i>weight</i>	0.53	0.0014	-4.70
10	2	<i>hmm</i>	-4.98	-0.0011	-29.41
10	2	<i>weight</i>	-5.15	-0.0020	-35.01
10	4	<i>hmm</i>	-2.94	0.0302	-29.00
10	4	<i>weight</i>	-2.33	0.0298	-22.44
10	6	<i>hmm</i>	-1.51	0.2354	-30.88
10	6	<i>weight</i>	-0.95	0.2083	-24.64

Real data

Two real datasets were tested using Smooth Descent, a low-depth dataset of garden strawberry (*Fragaria x ananassa*) (Fig. 2A), chromosome 15 of *Ipomoea batatas* (Fig. 2B) and a low-depth dataset of a diploid potato (Fig. 2C). Each strawberry chromosome was mapped using a relatively small population genotyped at low depth. Smooth Descent corrected up to 13% of genotypes, largely correlating with depth so that samples sequenced at lower depth had more genotype corrections. About 3.5% of studied chromosomes had a depth above 10x and had more than 2% of genotypes corrected, an unexpected result probably caused by errors during mapping leading to overcorrection of some samples.

The dataset of autohexaploid *I. batatas* was used to test SD in a scenario with better genotype accuracy. SD corrected 7.38% of genotypes while maintaining an equivalent relationship between the physical and genetic maps (Fig. 2B). This highlights the ability of SD to improve genotype accuracy even in situations where there have not been major issues in defining linkage map.

Lastly, a diploid dataset of potato was genotyped using very sequencing coverage of $\sim 1.5x$, which suggested a low-quality genotypic dataset (Clot et al.). Separate parental maps were generated and each group of markers was corrected using SD with physical order as an input, since a high quality potato genome sequence was available. The results show a drastic improvement in the correlation between the physical and genetic maps before and after applying Smooth Descent.

Software comparison

The performance of Smooth Descent (SD) was compared to two similar software tools: polymapR (Bourke et al. 2018a) and MAPpoly (Mollinari and Garcia 2019). The former performs linkage mapping in polyploids without considering genotyping errors. The latter uses a pre-determined order and a HMM method to obtain new map distances and new genotypes.

In Fig. 3 we can see the improvements that SD brings. The reconstructed maps have better position correlation and shorter lengths with SD, particularly when the error rates increase. Importantly, only SD changes the order as genotyping errors are corrected, a feature that is clearly useful especially as the error rate and ploidy increases (Fig. 3 top left). As expected, higher error rates lead to longer maps when using polymapR, but surprisingly, in MAPpoly that is the case with both very low or very high error rates. Note that polyploid map lengths are much shorter than expected, an issue that is common to polymapR and SD. In terms of genotyping error correction, MAPpoly is better than SD in diploids, but both perform equivalently well in polyploids, except in higher error rates where the HMM of SD is somewhat better. Lastly, the computation time needed for 5 iterations of SD is around 400s in diploids and tetraploids, and around 1000s or 2500s in hexaploids for the weighted average or HMM methods. In comparison, polymapR was always faster, which is to be expected since SD is iteratively running polymapR. MAPpoly time consumption was much higher as ploidy and error rate increased, with very long waiting times in hexaploids.

Overall, SD is better at recovering the correct order and shortening maps regardless of the situation. MAPpoly was better in the diploid scenario in terms of genotype correction and time consumption but became equivalent or worse than SD in tetraploids and hexaploids.

Discussion

In this study we have shown that Smooth Descent is able to substantially reduce genotyping errors, particularly in diploids, and to greatly improve marker order in polyploid linkage mapping, especially using the HMM approach. Moreover, when compared to related tools, SD computes better linkage maps with an equivalent or better level of genotype correction. Our findings are supported by analysis of real data: there was a clear correlation between sequencing depth and estimated genotyping errors in a low-depth strawberry dataset, and an accurate genetic map was obtained after correcting around 7.4% of genotyping errors in hexaploid sweet potato. Thus, we have shown that genotype correction is a useful method to improve linkage mapping in the presence of genotyping errors.

In contrast, the most popular strategy of error management in current genetic mapping software is marker or genotype removal. In JoinMap this is achieved through a Bayesian parameter (Liu et al. 2014), while Lep-Map2 does so through a Hidden Markov Model (HMM) (Broman et al. 2003; Rastas et al. 2016). GUSMap, on the other hand, does not remove errors but compensates their impact in map length, also within an HMM model (Bilton et al. 2018). Finally, HighMap uses SD's predecessor SMOOTH (van Os et al. 2005), and thus could benefit from the developments presented here (Liu et al. 2014).

The genotype correction approach presented in this article depends on transmitting confident parental information to uncertain offspring genotypes. Essentially, if most local markers indicate that one chromosomal region of a parent has been inherited, the offspring genotypes should match parental haplotypes. This rationale, and therefore the accuracy of SD, depends on two important factors: marker order and parental phasing.

Marker order

The set of local markers used to identify wrong genotypes is clearly defined by marker order. It is not crucial that marker order is exact, but the overall preliminary order should be correlated to the true order. In instances where the provided preliminary order is very far off from the true order, SD will not be able to impute genotypes correctly and any map improvement will be spurious.

Marker order can be determined by a linkage mapping procedure where a measure of linkage and an ordering algorithm is used to obtain a genetic map. In our implementation of SD these correspond to polypmapR (Bourke et al. 2018a) and MDSMap (Preedy and Hackett 2016) respectively. Both processes are sensitive to genotyping errors, meaning that as errors increase, the accuracy of the estimated linkage map will decrease. Consequently, there is a natural upper limit to the level of genotyping error that SD can tolerate: once the error rate impedes the calculation of a relatively good preliminary genetic map, SD

stops being useful. This also means that if different methods were designed that could compute marker orders independently of genotyping errors, SD applicability would be expanded.

Linkage mapping is not the only way to determine marker order. As reference genomes are built, it is increasingly common to obtain physical positions for markers. If such information is available, one could apply SD using physical, instead of genetic positions. This opens the possibility of using SD to datasets that are too large to be mapped using linkage techniques, but that could benefit from an error-cleaning algorithm. Moreover, since the order would not need to be re-calculated after genotype correction, only a single iteration of the algorithm would be necessary. Nevertheless, particularly for the weighted-average IBD estimation procedure, the usage of physical positions rather than genetic positions could be problematic since physical distances do not represent the same recombination probabilities along the genome. In centromeres a distance of 100,000 bp will include less recombinations than 100,000 bp in the chromosome arms. This should not be a major problem in the application of SD though, since the recombination frequencies are used relative to each other within small local intervals. Furthermore, if the locations of the pericentromeric regions are known (which they often are), then it would be possible to generate pseudo-cM positions of markers to circumvent this issue.

Parental phasing

To calculate identity-by-descent (IBD) probabilities, the backbone of genotype error detection and correction in SD, accurate parental phases or parental haplotypes are required. In this study we have not aimed at characterizing the effects of parental phasing in SD, as there has been much research dedicated to this complex issue (Browning and Browning 2011; He et al. 2018; Al Bkhetan et al. 2021), both in diploids and in polyploids. Currently, there are two types of approaches that can be used to establish parental phasing: based on marker scores or on sequence reads.

Marker scores have been used within several Hidden Markov Models (HMM) to obtain accurate phases. Recent studies in diploid data showed that consensus haplotyping approaches are the most accurate (Al Bkhetan et al. 2021), although individually tools like SHAPEIT4 (Delaneau et al. 2019) and BEAGLE5 (Browning et al. 2018) have the best performances in terms of time efficiency and accuracy. Several HMM have also been developed focused on polyploid data which can estimate phases: MAPpoly (Mollinari and Garcia 2019), polyOrigin (Zheng et al. 2021), and polyqtlR (Bourke et al. 2021). Although many of these methods consider genotyping error in their estimations, since phasing depends on marker segregation, an increased genotyping error rate in the target population can decrease phasing accuracy.

Alternatively, reads can be used to perform haplotype assembly: by observing multiple polymorphisms in a single read one can infer the most likely haplotype phases. Multiple tools have been developed to produce long-range haplotypes using short reads, long reads or a combination of both (Garg 2021). In diploids, WhatsHap (Patterson et al. 2015) and HapCut2 (Edge et al. 2017) are the most popular methods, being able to produce chromosome-level haplotypes when combining short and long read data (Garg 2021). In polyploids, the assembly problem is more complex, which has required the development of specific tools such as HapCompass (Aguiar and Istrail 2012), HapTree (Berger et al. 2014) and SDhaP

(Das and Vikalo 2015). Although useful, the accuracy of these tools is quite variable depending on depth and ploidy (Motazediz et al. 2017), never reaching the performance of their diploid counterparts. More recent developments like WhatsHap polyphase (Schrinner et al. 2020), based on long-read sequencing or Hap10 (Majidian et al. 2020), oriented to link-read data, are promising in closing the gap between diploid and polyploid haplotype assembly.

Application of Smooth Descent

The original idea behind the development of SD was to create a tool that would be able to utilize low-depth, inaccurate genotypes to obtain accurate linkage maps. Intuitively, we expected that confident parental phasing would be enough to create such an approach. We have shown that indeed, if parental information is accurate and marker order is well established, genotype correction can be performed, and accurate linkage maps obtained. Thus, we can imagine the following genotyping setup for an F1 population. First, the two parents are sequenced at high depth using long-read sequencing, in order to compute parental haplotype phases. Secondly, the F1 population is genotyped using low-depth short reads. If a marker order is not established yet, SD can be used iteratively to improve genotypes and obtain an accurate linkage map. Otherwise, a single iteration of SD is used to eliminate as many genotyping errors as possible. If the marker number is relatively small, the HMM method of SD is applied, if the dataset is larger the more efficient, although less accurate, weighted average method is used. Finally, a set of corrected genotypes is obtained. In this manner, SD would reduce genotyping costs by allowing a lower depth of sequencing in the F1 offspring.

Overall, SD is a simple and informative software tool. It estimates IBDs, calculates error rates per marker and individual and can impute corrected genotypes. Our implementation, together with MDSmap (Preedy and Hackett 2016) and polymapR (Bourke et al. 2018a) allows SD to work in multiple ploidies and with large datasets. We also provide many visualization tools which will help uncover the hidden information within genotyping data and turn Smooth Descent into SMOOTH's descendent.

Declarations

Funding

This work was supported by the TKI projects LWV20.112 Application of sequence-based multi-allelic markers in genetics and breeding of polyploids (BO-68-001-042-WPR) LWV20.085 Genetics Assisted Assembly of Complex Genomes (BO-68-001-033-WPR), and Novel genetic and genomic tools for polyploid crops (KV 1605-020/TU-16006; BO-68-001-001).

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Data availability

R package SmoothDescent can be found on github: github.com/Alethere/SmoothDescent

The data used on this article can be found on FigShare: doi.org/10.6084/m9.figshare.20038589

Author Contribution

ATN: designed software, performed research, drafted and reviewed manuscript. PB, EvdW: proposed research, supervised research, reviewed manuscript. CC: contributed to manuscript and results. PA: obtained funding, supervised research, reviewed manuscript. RF: supervised research, reviewed manuscript. CM: proposed research, obtained funding, supervised research, reviewed manuscript.

References

1. Aguiar D, Istrail S (2012) HapCompass: A Fast Cycle Basis Algorithm for Accurate Haplotype Assembly of Sequence Data. *J Comput Biol* 19:577–590. <https://doi.org/10.1089/cmb.2012.0084>
2. Al Bkhetan Z, Chana G, Ramamohanarao K et al (2021) Evaluation of consensus strategies for haplotype phasing. *Brief Bioinform* 22:1–12. <https://doi.org/10.1093/BIB/BBAA280>
3. Benzer S, ON THE TOPOLOGY OF THE GENETIC FINE STRUCTURE (1959) *Proc Natl Acad Sci* 45:1607–1620. <https://doi.org/10.1073/pnas.45.11.1607>
4. Berger E, Yorukoglu D, Peng J, Berger B (2014) HapTree: A Novel Bayesian Framework for Single Individual Polyplotyping Using NGS Data. *PLoS Comput Biol* 10:e1003502. <https://doi.org/10.1371/journal.pcbi.1003502>
5. Bilton TP, Schofield MR, Black MA et al (2018) Accounting for errors in low coverage high-throughput sequencing data when constructing genetic maps using Biparental outcrossed populations. *Genetics* 209:65–76. <https://doi.org/10.1534/genetics.117.300627>
6. Bourke PM, van Geest G, Voorrips RE et al (2018a) polymapR—linkage analysis and genetic map construction from F1 populations of outcrossing polyploids. *Bioinformatics* 34:3496–3502. <https://doi.org/10.1093/bioinformatics/bty371>
7. Bourke PM, Voorrips RE, Hackett CA et al (2021) Detecting quantitative trait loci and exploring chromosomal pairing in autopolyploids using polyqtlR. *Bioinformatics* 37:3822–3829. <https://doi.org/10.1093/bioinformatics/btab574>
8. Bourke PM, Voorrips RE, Visser RGF, Maliepaard C (2018b) Tools for Genetic Studies in Experimental Populations of Polyploids. *Front Plant Sci* 9:513. <https://doi.org/10.3389/fpls.2018.00513>
9. Broman KW, Wu H, Sen S, Churchill GA (2003) R/qtl: QTL mapping in experimental crosses. *Bioinformatics* 19:889–890. <https://doi.org/10.1093/bioinformatics/btg112>
10. Browning BL, Zhou Y, Browning SR (2018) A One-Penny Imputed Genome from Next-Generation Reference Panels. *Am J Hum Genet* 103:338. <https://doi.org/10.1016/J.AJHG.2018.07.015>

11. Browning SR, Browning BL (2011) Haplotype phasing: existing methods and new developments. *Nat Rev Genet* 12:703–714. <https://doi.org/10.1038/nrg3054>
12. Cartwright DA, Troggio M, Velasco R, Gutin A (2007) Genetic Mapping in the Presence of Genotyping Errors. *Genetics* 176:2521–2527. <https://doi.org/10.1534/genetics.106.063982>
13. Cheema J, Dicks J (2009) Computational approaches and software tools for genetic linkage map estimation in plants. *Brief Bioinform* 10:595–608. <https://doi.org/10.1093/bib/bbp045>
14. Clot C, Wang X, Thérèse Navarro A et al Marker imputation and linkage map construction using a skim sequenced diploid potato population
15. Das S, Vikalo H (2015) SDhaP: haplotype assembly for diploids and polyploids via semi-definite programming. *BMC Genomics* 16:260. <https://doi.org/10.1186/s12864-015-1408-5>
16. Delaneau O, Zagury JF, Robinson MR et al (2019) Accurate, scalable and integrative haplotype estimation. *Nat Commun* 2019 10:1–10. <https://doi.org/10.1038/s41467-019-13225-y>
17. Edge P, Bafna V, Bansal V (2017) HapCUT2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Res* 27:801–812. <https://doi.org/10.1101/GR.213462.116>
18. Elshire RJ, Glaubitz JC, Sun Q et al (2011) A Robust, Simple Genotyping-by-Sequencing (GBS) Approach for High Diversity Species. *PLoS ONE* 6:e19379. <https://doi.org/10.1371/journal.pone.0019379>
19. Fierst JL (2015) Using linkage maps to correct and scaffold de novo genome assemblies: methods, challenges, and computational tools. *Front Genet* 6:220. <https://doi.org/10.3389/fgene.2015.00220>
20. Garg S (2021) Computational methods for chromosome-scale haplotype reconstruction. *Genome Biol* 2021 22:1–24. <https://doi.org/10.1186/S13059-021-02328-9>
21. Gerard D, Ferrão LFV, Garcia AAF, Stephens M (2018) Genotyping Polyploids from Messy Sequencing Data. *Genetics* 210:789–807. <https://doi.org/10.1534/genetics.118.301468>
22. Glover NM, Redestig H, Dessimoz C (2016) Homoeologs: What Are They and How Do We Infer Them? *Trends Plant Sci* 21:609–621. <https://doi.org/10.1016/j.tplants.2016.02.005>
23. Hackett CA, Broadfoot LB (2003) Effects of genotyping errors, missing values and segregation distortion in molecular marker data on the construction of linkage maps. *Heredity (Edinb)* 90:33–38. <https://doi.org/10.1038/sj.hdy.6800173>
24. He D, Saha S, Finkers R, Parida L (2018) Efficient algorithms for polyploid haplotype phasing. *BMC Genomics* 19:110. <https://doi.org/10.1186/s12864-018-4464-9>
25. Hu Q, Liu Y, Liao X et al (2021) A high-density genetic map construction and sex-related loci identification in Chinese Giant salamander. *BMC Genomics* 22:230. <https://doi.org/10.1186/s12864-021-07550-0>
26. Kaur S, Francki MG, Forster JW (2012) Identification, characterization and interpretation of single-nucleotide sequence variation in allopolyploid crop species. *Plant Biotechnol J* 10:125–138. <https://doi.org/10.1111/J.1467-7652.2011.00644.X>

27. Lander ES, Botstein D (1989) Mapping mendelian factors underlying quantitative traits using RFLP linkage maps. *Genetics* 121:185–199. <https://doi.org/10.1093/genetics/121.1.185>
28. Lincoln SE, Lander ES (1992) Systematic detection of errors in genetic linkage data. *Genomics* 14:604–610. [https://doi.org/10.1016/S0888-7543\(05\)80158-2](https://doi.org/10.1016/S0888-7543(05)80158-2)
29. Liu D, Ma C, Hong W et al (2014) Construction and analysis of high-density linkage map using high-throughput sequencing data. *PLoS ONE* 9. <https://doi.org/10.1371/journal.pone.0098855>
30. Majidian S, Kahaei MH, De Ridder D (2020) Hap10: Reconstructing accurate and long polyploid haplotypes using linked reads. *BMC Bioinformatics* 21:1–18. <https://doi.org/10.1186/S12859-020-03584-5/FIGURES/6>
31. Mascher M, Stein N (2014) Genetic anchoring of whole-genome shotgun assemblies. *Front Genet.* <https://doi.org/10.3389/fgene.2014.00208>. 5:
32. Mollinari M, Garcia AAF (2019) Linkage Analysis and Haplotype Phasing in Experimental Autopolyploid Populations with High Ploidy Level Using Hidden Markov Models. *G3 Genes|Genomes|Genetics* 9:3297–3314. <https://doi.org/10.1534/G3.119.400378>
33. Mollinari M, Olukolu BA, Da Pereira GS et al (2020) Unraveling the Hexaploid Sweetpotato Inheritance Using Ultra-Dense Multilocus Mapping. *G3 Genes|Genomes|Genetics*. 10:281–292. <https://doi.org/10.1534/G3.119.400620>
34. Motazed E, Finkers R, Maliepaard C, de Ridder D (2017) Exploiting next-generation sequencing to solve the haplotyping puzzle in polyploids: a simulation study. *Brief Bioinform* 19:bbw126. <https://doi.org/10.1093/bib/bbw126>
35. Patterson M, Marschall T, Pisanti N et al (2015) WhatsHap: Weighted Haplotype Assembly for Future-Generation Sequencing Reads. *J Comput Biol* 22:498–509. <https://doi.org/10.1089/cmb.2014.0157>
36. Preedy KF, Hackett CA (2016) A rapid marker ordering approach for high-density genetic linkage maps in experimental autotetraploid populations using multidimensional scaling. *Theor Appl Genet* 129:2117–2132. <https://doi.org/10.1007/s00122-016-2761-8>
37. Rastas P (2017) Lep-MAP3: robust linkage mapping even for low-coverage whole genome sequencing data. *Bioinformatics* 33:3726–3732. <https://doi.org/10.1093/BIOINFORMATICS/BTX494>
38. Rastas P, Calboli FCF, Guo B et al (2016) Construction of ultradense linkage maps with Lep-MAP2: Stickleback F2 recombinant crosses as an example. *Genome Biol Evol* 8:78–93. <https://doi.org/10.1093/gbe/evv250>
39. Schrunner SD, Mari RS, Ebler J et al (2020) Haplotype threading: accurate polyploid phasing from long reads. *Genome Biol* 21:252. <https://doi.org/10.1186/s13059-020-02158-1>
40. Serang O, Mollinari M, Garcia AAF (2012) Efficient Exact Maximum a Posteriori Computation for Bayesian SNP Genotyping in Polyploids. *PLoS ONE* 7:e30906. <https://doi.org/10.1371/journal.pone.0030906>
41. Sturtevant A (1913) The linear arrangement of six sex-linked factors in *Drosophila* as shown by their mode of association. *Z Indukt Abstamm Vererbungsl* 10:293–294. <https://doi.org/10.1007/bf01943452>

42. van Os H, Stam P, Visser RGF, van Eck HJ (2005) SMOOTH: a statistical method for successful removal of genotyping errors from high-density genetic linkage data. *Theor Appl Genet* 112:187–194. <https://doi.org/10.1007/s00122-005-0124-y>
43. Varshney RK, Terauchi R, McCouch SR (2014) Harvesting the Promising Fruits of Genomics: Applying Genome Sequencing Technologies to Crop Breeding. *PLOS Biol* 12:e1001883. <https://doi.org/10.1371/JOURNAL.PBIO.1001883>
44. Voorrips RE, Maliepaard CA (2012) The simulation of meiosis in diploid and tetraploid organisms using various genetic models. *BMC Bioinformatics* 13:248. <https://doi.org/10.1186/1471-2105-13-248>
45. Wadl PA, Olukolu BA, Branham SE et al (2018) Genetic diversity and population structure of the usda sweetpotato (*ipomoea batatas*) germplasm collections using gbspoly. *Front Plant Sci* 9:1166. <https://doi.org/10.3389/FPLS.2018.01166/BIBTEX>
46. Wu Y, Bhat PR, Close TJ, Lonardi S (2008) Efficient and Accurate Construction of Genetic Linkage Maps from the Minimum Spanning Tree of a Graph. *PLoS Genet* 4:e1000212. <https://doi.org/10.1371/journal.pgen.1000212>
47. Zheng C, Amadeu RR, Munoz PR, Endelman JB (2021) Haplotype reconstruction in connected tetraploid F1 populations. *Genetics* 219:2020.12.18.423519. <https://doi.org/10.1093/genetics/iyab106>
48. Zheng C, Voorrips RE, Jansen J et al (2016) Probabilistic multilocus haplotype reconstruction in outcrossing tetraploids. *Genetics* 203:119–131. <https://doi.org/10.1534/genetics.115.185579>

Figures

Figure 1

Results of 10 simulated populations across error rates and ploidies. Within each panel, each column represents a ploidy and panel the top row shows the results for the IBD estimation with the weighted average procedure (IBD = weight) and the bottom row for the IBD estimation with the HMM (IBD = hmm). Top panel, genotyping error, the rate of genotypes that are different from the true genotypes. Middle panel, position correlation, the correlation between true genetic positions and estimated positions in a

genetic map. Bottom panel, map length, the size in cM of the estimated maps. In each plot, points represent individual observations and lines are the average. Each colour represents one simulated error type, with special_A being heterogeneous rate across individuals and special_B being heterogeneous rate along the map.

Figure 2

Error detection and marker order in two real datasets after applying Smooth Descent. A) Relationship between sequencing depth and the rate of markers corrected by Smooth Descent for each chromosome of 52 individuals of strawberry (*Fragaria x ananassa*). B) Relationship between physical and genetic positions of 1513 markers in chromosome 15 of *Ipomoea batatas*, before and after correcting 7.38% of genotype calls using Smooth Descent. C) Relationship between physical and genetic positions of 1716

markers in chromosome 12 of *Solanum tuberosum*, before and after using Smooth Descent to correct low depth genotypes based on a physical order.

Figure 3

Software comparison with other tools. Average observations of 10 populations per ploidy with simulated genotyping errors. Top left, correlation between true map position and estimated map position. Top right, map length in cM. Bottom left, time spent in seconds, note logarithmic y axis. Bottom right, genotyping error of corrected genotypes. Grey dashed line indicates the starting error rate. Note that polymapR does not produce corrected genotypes and thus is not included in this panel. Each color and shape corresponds to a different approach: red circle, polymapR, green triangle MAPpoly, blue square (sd_hmm) SD with HMM approach, purple cross (sd_weight) SD with weighted average approach.