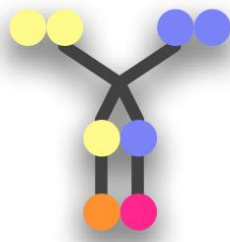


# **QUADGT: Joint Genotyping of Parental, Normal and Tumor Genomes**

## **User's Guide**



Miklós Csűrös

Department of Computer Science and Operations Research  
Université de Montréal  
Montréal, Québec, Canada

February 20, 2013

QUADGT was written by Eric Bareke and Miklós Csűrös.

The software package is distributed under the terms of the BSD license, as shown below.

Copyright © 2012, 2013 Miklós Csűrös & Eric Bareke  
All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

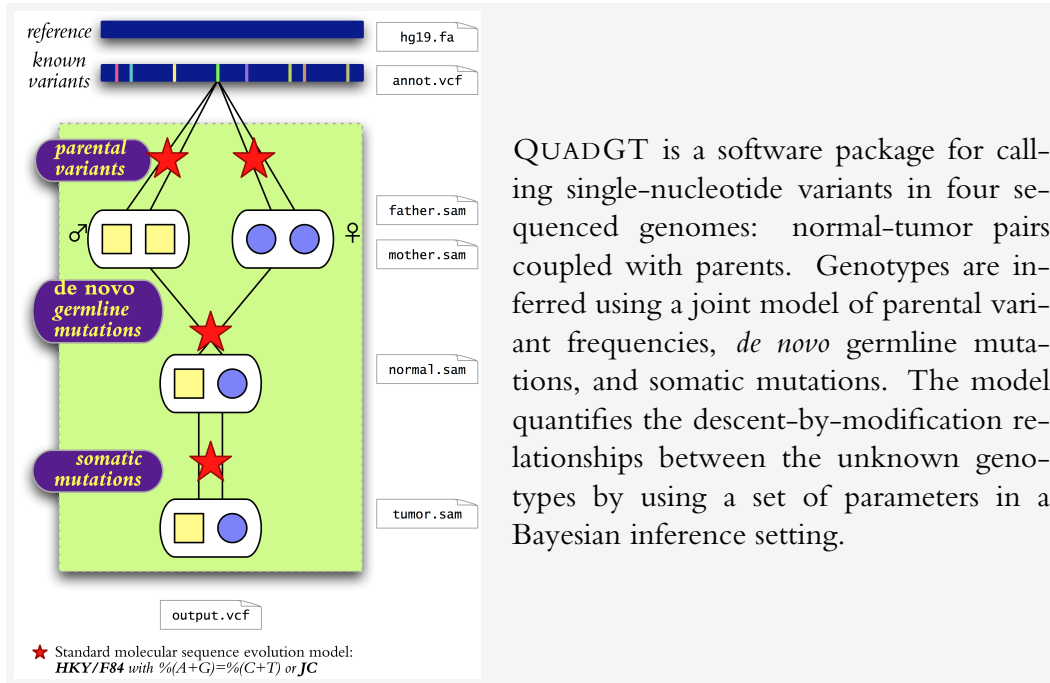
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the *Université de Montréal* nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Contents



<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Using QUADGT</b>	<b>5</b>
2.1	Installation . . . . .	5
2.2	Execution . . . . .	5
2.2.1	Quick start . . . . .	5
2.3	Work flows and best practices . . . . .	7
2.3.1	Model training . . . . .	7
2.3.2	Variant calling . . . . .	7
2.4	Command-line execution . . . . .	8
2.4.1	Java Virtual Machine options. . . . .	9
2.4.2	Multi-threaded execution . . . . .	9
2.4.3	Reference sequences . . . . .	9
2.4.4	Known reference variants . . . . .	10
2.4.5	Input alignments . . . . .	11
2.5	Variant calling . . . . .	12
2.5.1	VCF output . . . . .	12
2.5.2	INFO fields . . . . .	14
2.5.3	Genotype fields . . . . .	15
2.6	Model training . . . . .	16
2.6.1	Sequencing quality recalibration and tumor purity estimation	16
2.6.2	Model parameters . . . . .	17
2.6.3	Model parameter file. . . . .	18
<b>3</b>	<b>Model structure</b>	<b>21</b>
3.1	Molecular evolution models . . . . .	21
3.1.1	Parental genotype priors . . . . .	22
3.1.2	Parental germline mutations and inheritance . . . . .	23
3.1.3	Tumor mutations and purity . . . . .	23
3.1.4	Sequencing errors . . . . .	24

# 1 Introduction



Note that the current version of QUADGT ignores insertions and deletions in the alignments.

The software package assumes a thorough probabilistic model of single-nucleotide variants with the following notable features.

-  **Multiple alleles.** Every locus has four possible alleles — A, C, G, T. Diploid genotypes combine multi-allele frequencies with adjusted heterozygous/homozygous SNP ratios.
-  **DNA mutation models.** Point mutations between related genomes follow standard molecular evolution models. The implemented models include the basic Jukes-Cantor model and a version of the Hasegawa-Kishino-Yano (a.k.a. Felsenstein's F84) model. Our version assumes purine-pyrimidine balance<sup>1</sup> and has four parameters that adjust sequence divergence, transition/transversion ratio, and nucleotide composition (GC-content and amino/keto  $\%(A + C)/\%(T + G)$  content).

<sup>1</sup>The canonical HKY85/F84 DNA mutation model [Hasegawa M, Kishino H & Yano T. "Dating of the human-ape splitting by a molecular clock of mitochondrial DNA," *Journal of Molecular*





-  **Known variants.** QUADGT integrates prior information on minor allele frequencies from a chosen variant database such as the NHLBI Exome Sequencing Project’s Exome Variant Server [<http://evs.gs.washington.edu/>].
-  **Inheritance.** Inheritance models span autosomes, sex chromosomes, and mitochondrial DNA. *De novo* mutations follow a DNA substitution model.
-  **Basecall quality scores.** QUADGT’s model incorporates alignment quality scores, uses them in inference, and automatically recalibrates score  $\rightarrow$  probability mappings during model training.
-  **Tumor purity.** QUADGT infers tumor purity (normal-tumor sample mixture coefficient) from basecall statistics at somatic mutations, and takes it into account during variant calling.

Figure 2 shows the adopted model structure with the dependencies between genotypes, model parameters and sequencing reads.

Note that you can use QUADGT on any subset of the four related genomes, including parent-offspring trios, and normal-tumor pairs without parental samples.

## Acknowledgments

This software package was developed in conjunction with a study led by Daniel Sinnett at the Sainte-Justine UHC Research Centre (Montréal, QC) on pediatric acute lymphoblastic leukemia. The project has been supported by funds from the Terry Fox Research Institute and the Canadian Institutes for Health Research, the François-Karl-Viau Research Chair in Pediatric Oncogenomics, and the National Science and Engineering Research Council.

---

*Evolution*, **22**(2):160–174, 1985; Felsenstein J & Churchill GA. “A hidden Markov model approach to variation among sites in rate of evolution,” *Molecular Biology and Evolution*, **13**(1):93–104, 1996] has five independent parameters, but here we fix the purine-pyrimidine balance, hence four independent statistics remain. Note that Chargaff’s rules of  $\%A = \%T$  and  $\%C = \%G$  automatically imply the assumed balance condition  $\%(A + G) = \%(C + T) = \frac{1}{2}$ .

## 2 Using QUADGT

QUADGT is written entirely in 64-bit Java, and can thus be used in different operating systems, including Mac OS X, Microsoft Windows, and various Unix/Linux versions. The software is packaged in the JAR file `QuadGT.jar`, and can be executed in Java versions 1.6 and above.

### 2.1 Installation

Download the JAR file `QuadGT.jar`. If you want to work with BAM input files, then you also need the Picard/SAMTools JDK (v. 1.85). The simplest is to download `QuadGT.zip` which bundles `QuadGT.jar` with the SAMTools library under the directory `dist/`. By unpacking (`unzip QuadGT.zip`), you will have `dist/QuadGT.jar` and `dist/lib/sam-1.85.jar` (plus this very PDF document).

### 2.2 Execution

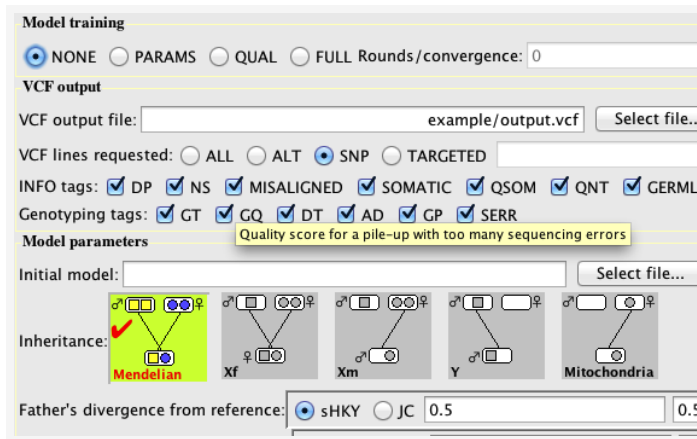
The executable is launched on the command line by

```
java <VMoptions> -jar QuadGT.jar <options> <reference> <father> <mother> <normal> <tumor>
```

(Adjust the proper path for `QuadGT.jar`.) Section 2.4 fully explains the command-line arguments.

#### 2.2.1 Quick start

By far the most convenient is to set up the command-line execution through the graphical interface of the packaged “wizard” by executing the JAR with no arguments; see Figure 1.



The wizard does not launch the variant caller, but provides an easy way to build a shell script, or to construct a single-line command. Tooltips explain what every field and button is for.

## 2.3 Work flows and best practices

### 2.3.1 Model training

In order to call variants in a new data set, first train the model separately for autosomes and sex chromosomes. It is probably sufficient to select one or few chromosomes for parameter training (use command-line options like `-chromosomes 12` or `-chromosomes 5, 12, 19`). During model training, use stringent criteria for basecall pile-up and alignment filtering (recommended: `-mapq 30`, which is the default value).

**Training parameters for autosomes.** In our experience, a “typical” chromosome (like chromosome 12) alone is sufficient for training on exome data. Select the **MENDELIAN** inheritance model. Train using 3 or 4 full training rounds (`-estimate full3` or `-estimate full3`). Save the resulting optimized model into a parameter file (`-savepars`).

**Training parameters for sex chromosomes in boy.** Train a model, just like for autosomes, separately for the X and Y reference chromosomes (use the command-line option `-chromosomes`) by selecting **Xm** and **Y** inheritance models.

**Training parameters for sex chromosomes in girl.** Train a model, just like for autosomes, for the X reference chromosomes (use the command-line option `-chromosomes`) and select the **Xf** inheritance model.

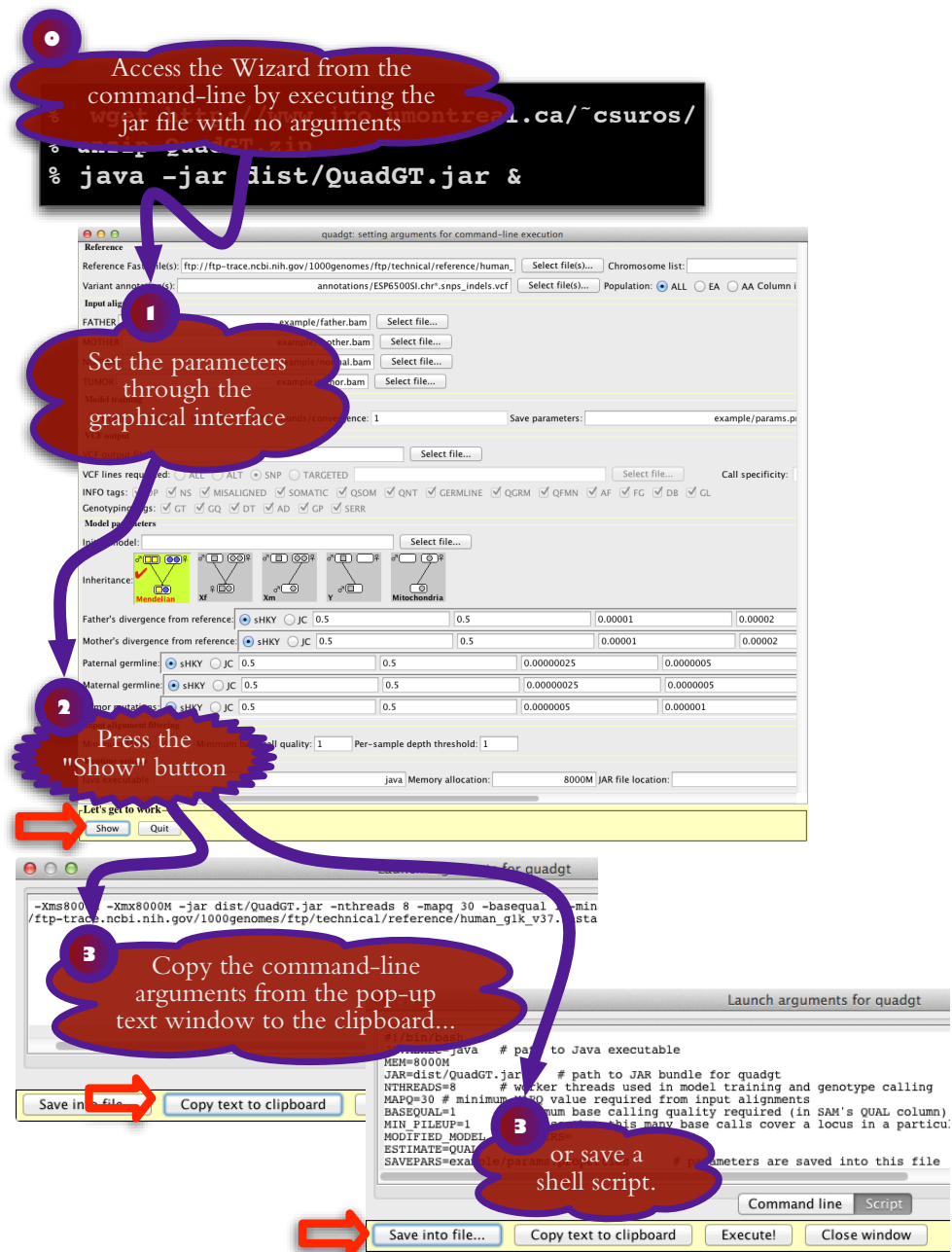


Figure 1: Plug and play. The built-in wizard assembles a shell script or a single-line command for you.



**Have the model parameters converged yet?** If you want to make sure that the model parameters have been sufficiently optimized, you can look into the saved parameter files (see §2.6.3). Look for the property `quadgt.history` which gives the successive values of log-likelihood on the Phred scale ( $-10 \log_{10} p$ ). If the log-likelihood has not changed much in the past few iterations, you are probably done.

### 2.3.2 Variant calling

Having trained the model, you can apply it to call single-nucleotide variants precisely along the sequenced sites. For a boy, you need three separate calls for autosomes, X, and Y. For a girl, you need two separate calls for autosomes and X. While producing the genotype calls, the software generates some debugging info to the standard output, and reports skipped indel-containing reads to the standard error. Specify the level of detail (SNPs, all loci with  $\geq 1$  non-reference base call, or all loci) that you need and the output file (options `-detail` and `-vcf`, see §2.5.1).

**Somatic mutations.** VCF output lines with putative somatic mutations are tagged by the `SOMATIC` info field. The `QSOM` info field gives the quality score for the somatic call. You should also inspect if the locus may have a lot of misaligned reads. Such sites are tagged by the `MISALIGNED` info field, giving a quality score for the number of implied sequencing errors. This value is the sum of individual `SERR` entries for the samples, giving the same type of quality score. The `QSOM` score is for all genotype assignments with point mutations in the alleles during clonal inheritance between normal and tumor. `SOMATIC` is called only if `QSOM > MISALIGNED`. The best bet is to rank somatic calls by the difference `QSOM - MISALIGNED`.

**De novo mutations.** VCF output lines with putative *de novo* germline mutations are tagged by the `GERMLINE` info field. The `QGRM` info field gives the quality score for the mutation call. You should also inspect if the locus may have a lot of misaligned reads, as described above for somatic calls, and rank germline mutation calls by the difference `QGRM - MISALIGNED`.

## 2.4 Command-line execution

The executable is launched on the command line by

```
java <VMoptions> -jar QuadGT.jar <options> <reference> <father> <mother> <normal> <tumor>
```

Files read by QUADGT may be compressed and either local, or remote.

**Compressed files.** Compressed files are recognized by the obligatory `.gz` extension.

**URLs.** URLs are recognized by the character sequence specifying a scheme at the beginning. If the file name contains a colon (`:`), the string before it is assumed to specify a URL scheme<sup>2</sup>.

#### 2.4.1 Java Virtual Machine options.

Useful Java virtual machine options (*VMoptions*) include `-Xms` and `-Xmx` that set the lower and upper bounds for memory usage (Java heap size). (For instance, `-Xmx8000M` allows for 8 Gigabytes of memory, which should be enough for genotyping complete human exome datasets.)

#### 2.4.2 Multi-threaded execution

Normally, QUADGT trains model parameters and calls the variants using multiple execution threads. More precisely, multiple producer threads work on different loci, and their results are combined by a single consumer thread to populate the data structures. The number of producer threads is set on the command-line by the `-nthreads` option. `-nthreads 0` completely disables multi-threaded execution. By default, as many producer threads are used as there are available CPU cores (established by calling Java's `Runtime.getRuntime().availableProcessors()` method).

#### 2.4.3 Reference sequences

The first mandatory argument (*reference*) specifies a comma-separated list of Fasta files containing the reference sequences. The list may include local files or URLs (`http` or `ftp`), and the files may be compressed by GZIP. For instance, the following argument specifies two reference files; the first one is a gzipped file at the UCSC genome browser site, and the other one is an uncompressed local file.

---

<sup>2</sup> `http` or `ftp` or whatever else is accepted by Java's `java.net.URL.openConnection` method

```
http://hgdownload.cse.ucsc.edu/goldenPath/hg18/chromosomes/chr20.fa.gz,chr21.fa
```

**Reference sequence identifiers.** Reference sequence identifiers are derived from the Fasta defines. Specifically, the identifier is the character sequence after the define’s starting character ‘>’ up to the first white space. For instance, the define

```
>chr20 chromosome 20 sequence in hg18 assembly
```

yields the identifier `chr20`. The chromosome order in the Fasta files also defines the order of mappings in the input SAM files. (So, if “chr10” comes before “chr2” in the reference Fasta, then the same order is expected in the SAM file.) Reference sequence identifiers must match the `RNAME` fields in the input SAM files exactly, except for aliases with or without the `chr` prefix. Namely, if the identifier starts with `chr`, then an alias is generated without it (e.g., `20`). Otherwise, an alias is created by prefixing the identifier with `chr`. The reference file’s `chr20` will thus match an `RNAME 20`.

**Selecting a subset of chromosomes.** You can specify that only a subset of chromosomes should be used by listing the selected ones with the `-chromosomes l` option. The list *l* gives the comma-separated (no white space) chromosome identifiers with or without the `chr` prefix, in an arbitrary order. The example below selects three chromosomes for model training, and uses a single reference Fasta file comprising all chromosomes.

```
... -estimate full13 -chromosomes 5,12,19 hg19.fa.gz ...
```

#### 2.4.4 Known reference variants

It is recommended that you use data on known variants both for the model training and variant calling. The `-snpdb l` option specifies a comma-separated list of VCF files listing known variants. The following fields are retrieved from the variant VCF: chromosome (`CHROM`), 1-based position (`POS`), identifier (`ID`), reference (major) allele (`REF`), minor allele (`ALT`), and info (`INFO`). `QUADGT` extracts the minor allele frequency, functional classification and associated gene lists from the info field given by the tags `MAF`, `FG`, `GL`. This option was developed with the University of Washington’s Exome Server Variant files in mind. `QUADGT` incorporates minor allele frequencies into the parental prior genotype probabilities and reports them (as `AF=...`) in the VCF output. You can also select a subpopulation in the file, such as African or European American (prefixes `AA_` and `EA_` in the

EVS files): use `-snpdb-pop p` with  $p = \text{EA}$  or  $p = \text{AA}$ . Finally, you can specify a different column order from the VCF ordering by `-snpdb-columns l` where  $l$  is the 6-member list of 0-based column indices for CHROM, POS, ID, REF, ALT, INFO, in this order (VCF order is `-snpdb-columns 0,1,2,3,4,7`).

### 2.4.5 Input alignments

The four mandatory arguments *father*  $\cdots$  *tumor* specify the locations of the files with the aligned sequencing reads to the reference.

**File formats.** The input files are assumed to follow *SAM Format Specification v1.4*. BAM files are parsed via the bundled Picard/SAMTools API. Gzipped files are recognized as such by the `.gz` extension, and are decompressed on the fly. The locations may be given as URLs (starting with `http:` or `ftp:`), or as local file paths.

**Missing files.** Missing or irrelevant input must be specified by a single dot (`.`) For instance, call the variants along the Y chromosome by using the arguments

```
... -D quadgt.inheritance=Y chrY.fa.gz F.sam . N.sam T.sam
```

**SAM file order.** The input SAM files must be sorted primarily by reference (RNAME field ordering the same as the order of chromosomes in the input Fasta file(s)) and secondarily by position (POS field). Alignment lines without a matching reference identifier are skipped without warning (so that the same SAM files can be used with different sets of reference chromosomes).

**Filtering low-quality input alignments.** Input alignments are filtered by minimum mapping quality (MAPQ field in the SAM file), which can be set by the `-mapq` command-line option. In SAM alignment files, a value of 255 means that the quality is unavailable (so, with a `mapq` threshold below that, those lines are included by QUADGT). Normally, QUADGT ignores alignment lines with a mapping quality below 30. With `-mapq 0`, all alignment reads are considered in the inference.

**Filtering low-quality input basecalls.** QUADGT ignores alignment positions with ambiguous matching residues (from SAM alignment's SEQ column) or low quality scores (from SAM alignment's QUAL column). By default, the minimum basecalling quality is 1, but you can set a different threshold with `-basequal q`.

For  $q = 0$ , all basecalls are included, even those with quality score 0, which is probably not a good idea because  $q = 0$  corresponds to a sure sequencing error.

**Filtering by alignment coverage.** If after alignment and basecall filtering, a locus is not covered by enough many sequencing reads, it is not included in the model training and genotype inference. Specifically, if the read depth (DT genotyping field in the VCF output) for a sample is below a threshold  $d$ , then no basecalls are assumed for the sample at that locus. By default,  $t = 1$  (i.e., this filter never triggers), but the `-min-pileup  $d$`  option sets a higher threshold to throw away rogue alignments.

## 2.5 Variant calling

QUADGT infers maximum a posteriori (MAP) genotypes at sampled loci in the four genomes. In particular, each sample's inferred genotype has maximum posterior probability given the piled-up base calls with their quality scores, and the model's inheritance and mutation parameters. The genotyping calls are reported in *Variant Call Format version 4.1*. INFO and genotyping fields for the output are requested by the `-info` and `-genotype` command-line options.

### 2.5.1 VCF output

The VCF output file is specified by the `-vcf  $f$`  option on the command line. The file  $f$  must be a local file that can be written into, or a dash ('-') for writing calls on the standard output. If the file name ends with `.gz` then QUADGT writes a compressed file (which is, alas, not compatible with tabix — you need to compress the vcf file with bgzip).

**Selecting which loci should be reported in the VCF.** You can select the level of detail (SNPs, all loci with  $\geq 1$  non-reference base call, or all loci) in the VCF output by the option `-detail  $x$` . With  $x = \text{ALL}$  all covered loci are included in the output. With  $x = \text{ALT}$ , loci with at least one non-reference basecall are reported in the output. The default setting  $x = \text{SNP}$  reports loci with single nucleotide variants in either of the four genomes. More precisely, a locus is recognized as a possible SNV-containing site if at least one sample has at least  $d = 3$  reads covering in, and there is some uncertainty about the locus being homozygous with the reference (ref-homo) through all samples. (Quantitatively, the doubt level must be at least 5%, meaning the posterior probability of having at least one

mutant alleles across the four samples.) You can change the default threshold on minimum sample coverage  $d = 1, 2, \dots$  with the option `-min-DT  $d$` . Set the minimum doubt level about **ref-homo** as `-min-SNP  $q$` , which corresponds to a probability threshold  $p = 10^{-q/10}$ , i.e., loci with less than  $p$  probability of having non-reference alleles are skipped. (`-min-SNP` speeds up SNP executions by not performing costly numerical calculations at loci that have no chance of showing variants.)

**VCF output for a set of targeted loci.** QUADGT can accept a list of genomic coordinates specifying which loci should appear in the VCF output: `-detail file`. The coordinates must be given in a VCF file (CHROM and POS are parsed from the first two columns). The output includes all the loci from *file* at maximum detail (ALL).

**Setting the specificity of genotype resolution.** QUADGT “uncalls” weakly supported genotypes, and reports ambiguous calls instead like `1 / .` (homo- or heterozygous SNP), `0 / .` (not a homozygous SNP), and `. / .` (completely ambiguous). The threshold under which resolved genotypes are replaced by ambiguous calls is set by the `-qual  $q$`  optional command-line argument. Reported genotypes in the GT field will have quality at least  $q$ .

**Model optimization and variant calling.** Model parameters can be optimized prior to genotyping in the same run by specifying both `-optimize params` and `-vcf` on the command line. Genotype calls will be made using the optimized model.

**Meta-information lines.** The VCF output begins with a number of meta-information lines that start with ‘##’. These include the mandatory file format specification (`##fileformat=VCFv4.1`), as well as the obligatory descriptions of information (`##INFO`) and genotyping (`##FORMAT`) fields. QUADGT also writes information about the samples (`##SAMPLE` lines) and their relationships (`##PEDIGREE` lines).

**QUAL column: joint call quality.** The QUAL column of the VCF file gives the quality of the *joint* genotype calls. More precisely, the posterior probability  $p$  of the joint calls is computed and the quality is reported as  $q = -10 \log_{10}(1 - p)$ . Joint quality scores for subsets (family trio and normal-tumor pair) are reported in the info sub-fields QFMN and QNT.

**Genotyping column headers.** The last four columns of the VCF output list the genotype calls for the four genomes. Genome names are standardized (FATHER, MOTHER, NORMAL and TUMOR). Refer to the ##SAMPLE meta-information lines for sample identifiers.

## 2.5.2 INFO fields

QUADGT recognizes a number of INFO subfields accessed by the `-info x` command-line option. Here, *x* is a list of subfield keys, separated by colons (‘:’). The following sub-fields can be asked for.

INFO sub-field key	Description
<b>Standard VCF 4.1 sub-fields</b>	
DP	combined depth across all samples (after filters)
NS	number of samples with at least one basecall (after filters)
SOMATIC	indicates that the record is a somatic mutation (without many suspect reads: QSOM > MISALIGNED)
<b>non-standard sub-fields</b>	
MISALIGNED	Quality score for too many misalignments. This is the sum of the sample-specific quality scores (reported in the SERR) genotyping field). A MISALIGNED= <i>q</i> entry gives the P-value $p = 10^{-q/10}$ computed as the probability of having as many sequencing errors at the locus as implied by the joint genotype distribution and the basecalling qualities. Large <i>q</i> means that the alignments are suspect at that locus. When calling somatic and <i>de novo</i> mutations, compare the reported QSOM and QGRM against MISALIGNED. (Missing MISALIGNED field means <i>q</i> = 0.)
QSOM	quality score for somatic mutation (probability of <i>no mutation</i> summed across all joint histories with identical normal-tumor genotype pairs, reported on Phred scale) You should compare it against MISALIGNED.
GERMLINE	indicates that the record is a <i>de novo</i> germline mutation from either parent (without many suspect reads QGRM > MISALIGNED)
QGRM	quality score for germline mutation (probability of <i>no mutation</i> summed across all joint histories with no germline mutations, reported on Phred scale). You should compare it against MISALIGNED.
QNT	Phred-scaled quality score for the joint normal-tumor call (from posterior probability of the joint genotype calls under GT).
QFMN	Phred-scaled quality score for the joint parents-offspring call (from posterior probability of the joint genotype calls under GT).
<b>sub-fields copied from variant annotations</b>	
AF	minor allele frequency (in the selected sub-population),
DB	flag for membership in dbSNP (set if ID is not null).
FG	Genome Variation Server class of variation function
GL	gene list



The default behavior without the `-info` option is to include all known info sub-fields.

### 2.5.3 Genotype fields

For each sample, the same set of genotyping sub-fields are reported. The FORMAT column of the VCF file lists the sub-field keys for which the values appear as colon-separated lists in the sample columns.

Genotyping sub-field key	Description
<i>Standard VCF 4.1 sub-fields</i>	
GT	genotype, encoded as allele values separated by '/' for diploid loci, indicating that they are unphased. The allele values are 0 for the reference allele (what is in the REF field), 1 for the first allele listed in ALT, 2 for the second allele list in ALT and so on. Each missing allele is specified by a dot ('.'): QUADGT uncalls genotypes below the specified minimum quality value ( <code>-qual</code> on the command line, 13 by default), so a given genotype might be called as 1/. (at least one ALT allele) or as ./ (completely unresolved).
GQ	genotype quality, encoded as a phred quality $-10 \log_{10} p(\text{genotype call is wrong})$ . The reported quality covers all joint genotypes containing the sample-specific genotype (GT).
DT	read depth for the sample (number of base calls mapped to this locus)
GP	genotype posterior probabilities comprising comma-separated floating-point log10-scaled probabilities for all possible genotypes given the set of alleles defined in the REF and ALT fields. For two alleles (REF is 0, ALT is 1), the order of genotypes is 0/0, 0/1, 1/1. For three alleles (REF is 0, ALT is 1 and 2), the order is 0/0, 0/1, 1/1, 0/2, 1/2, 2/2.
AD	number of base calls for each allele listed in REF and ALT columns, in the same order as listed
<i>non-standard sub-field</i>	
SERR	Phred-scaled P-value for the number of implied sequencing errors. Specifically, the quality score $q(k) = \lfloor -10 \log_{10} \mathbb{P}\{\# \text{errors} \geq k\} \rfloor$ is the probability that at least $k$ sequencing errors occur, where $k = \lfloor \mathbb{E} \# \text{errors} \rfloor$ is the rounded expected number of errors implied by the genotype posterior probabilities at the given locus. Large SERR values indicate confounding basecalls that distort the inference, and may be caused by misaligned reads, copy number variations, or mixtures of clonal lineages.

The default behavior without the `-genotype` option is to include all known fields. The reported genotyping fields are requested by the `-genotype x` option on the command line. The argument `x` is a colon-separated list of sub-fields keys and determines their order in the output.

## 2.6 Model training

QUADGTEstimates model parameters from input data by adapting the expectation-maximization (EM) algorithm [Dempster AP, Laird NM & Rubin DB, “Maximum likelihood from incomplete data via the *EM* algorithm, *Journal of the Royal Statistical Society. Series B (Methodological)*, **39**(1):1–38, 1977]. The EM procedure alternates between an E-step and an M-step. In an E-step (“expectation”), various statistics are collected at loci with at least one mapped base call. (At each locus, a number of posterior probabilities are computed for appropriate features (e.g., somatic mutation) based on the piled-up base calls. These are summed to obtain expected values.) In an M-step (“maximization”), model parameters are set from the collected posterior statistics. (DNA substitution model parameters are numerically optimized, others are set by analytical formulas.) The algorithm maximizes the model likelihood by alternating the two steps until convergence.

Model training is selected by using the option `-estimate mode` where *mode* is one of PARAMS, QUAL or FULL (case-insensitive keywords). In QUAL mode, input alignments are scanned once, and one EM optimization round is performed for basecalling quality recalibration and tumor purity estimation; see §2.6.1. In PARAMS mode, input alignments are scanned once and multiple EM optimization rounds are done to estimate all other model parameters including parental genotype priors, germline and tumor line mutation probabilities; see §2.6.2. In FULL mode, multiple tours of QUAL-PARAMS optimizations are done, with increasing stringency on convergence. In our experience  $d = 3$  tours are sufficient (default setting). More or less pedantic settings are specified by `-estimate fulld` where  $d$  is the requested number of QUAL-PARAMS tours: e.g., `-estimate full5` requests 5 tours.

### 2.6.1 Sequencing quality recalibration and tumor purity estimation

**Basecalling qualities.** Normally, the basecall qualities  $q$  reported in the input SAM files (QUAL field) map to sequencing error probabilities  $p$  by the formula  $p = 10^{-q/10}$  (i.e., qualities are expressed as *decibans*). In order to offset a

possible bias, QUADGT uses a custom mapping (parameter `quadgt.qual`) from sequencing quality to error probability, which applies to all four samples. The `-estimate qual` command-line option requests the EM algorithm for computing the maximum-likelihood quality mapping. Only one iteration is performed (`-optimize` is ignored). The following command-line arguments call for a quality recalibration step by using reads mapped to chromosome 12 in data set `xyz`.

```
... -loadpars ini.p -estimate qual -savepars opt.p chr12.fa xyz_[FMNT].sam
```

In the above example, QUADGT adjusts the quality map of the initial model specified by the file `ini.p` and saves the optimized model to `opt.p`.

**Tumor purity.** The QUAL step estimates also the tumor purity by collecting statistics on allele sampling frequencies at somatic mutation sites. (More precisely, QUADGT relies on each basecall’s posterior probability for originating from the tumor sample given the somatic mutation and its error probability, weighed by the posterior distribution over normal-tumor genotype pairs.) Starting from an experimentally determined tumor purity measure (see the `quadgt.purity` parameter in §2.6.3), the tumor-sampling frequency should be optimized through multiple rounds for accurate somatic mutation inference.

## 2.6.2 Model parameters

The command-line option `-estimate params` invokes the EM algorithm with one or more iterations for optimizing other model parameters than the sequencing quality map and the tumor purity. The number of iterations is set by the `-optimize` option; by default, only one EM-pair is executed. The optimized parameters are

- ★ parental genotype priors: divergence from reference (`quadgt.divergence.*`) and heterozygosity coefficients (`quadgt.inb.*`) for father and mother separately `quadgt.divergence.*`  
`quadgt.inb.*`
- ★ germline mutation model (`quadgt.germline.*`) for mother and father separately `quadgt.germline.*`
- ★ tumor mutation model (`quadgt.tumor`) `quadgt.tumor`

Note that tumor sampling purity is not changed during PARAMS iterations. The following commands optimize model parameters through 5 iterations and save the result.

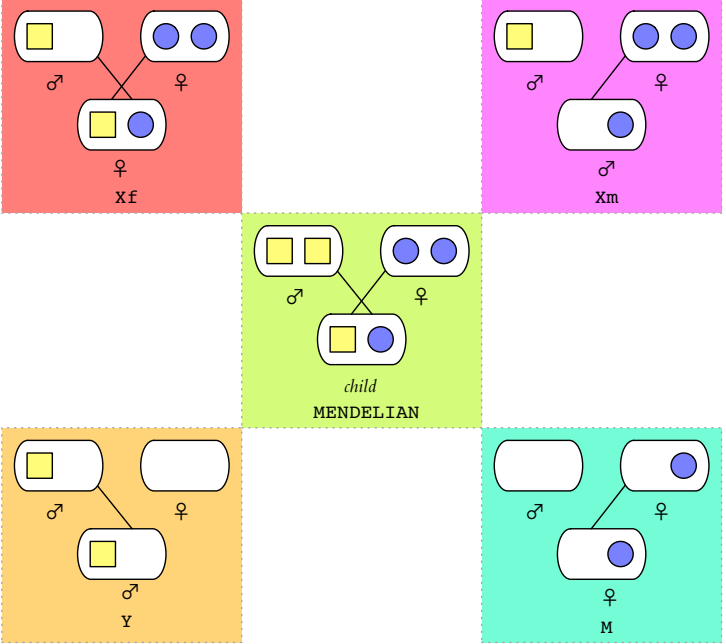
```
... -estimate params -optimize 5 -savepars opt.p chr12.fa xyz_[FMNT].sam
```

Alternatively, the optimization can be stopped based on a convergence threshold as `-optimize  $x$` . In order to distinguish between the ways of using `-optimize`,  $x$  must contain a decimal point (write 10.0 instead of 10). QUADGT iterates until the log-likelihood (on the Phred scale) drops by less than  $x$ .

### 2.6.3 Model parameter file.

The model parameter file is a standard Java Properties file (usually having the file extension `.properties`). The file is a text file containing comment lines preceded by `#` or `!` and parameter-value pairs in the format  $p=x$  (parameter  $p$  takes value  $x$ ). The following entries are used by QUADGT.

Parameter	Description	Typical values
quadgt.divergence.father	divergence between father's alleles and reference genome (frequency of mutant alleles, see Eq. (3))	$10^{-5}..10^{-4}$
quadgt.divergence.mother	divergence between mother's alleles and reference genome (frequency of mutant alleles, see Eq. (3))	$10^{-5}..10^{-4}$
quadgt.inb.father	heterozygosity coefficient in father's genome: $\gamma = 1 - H/H_{HWE}$ where $H$ is the frequency of heterozygote loci and $H_{HWE}$ is the theoretical frequency in Hardy-Weinberg equilibrium, see Eq. (4)	$\approx 0.5$
quadgt.inb.mother	heterozygosity coefficient in mother's genome, see Eq. (4)	$\approx 0.5$
quadgt.germline.father	<i>de novo</i> mutation model in paternal germline	$10^{-6}..10^{-5}$
quadgt.germline.mother	<i>de novo</i> mutation model in maternal germline	$10^{-6}..10^{-5}$

Parameter	Description	Typical values
quadgt.inheritance	<p>inheritance model determined by ploidy of father, mother, and child: MENDELIAN, Xf, Xm, Y, M, are allowed values (upper/lower case must be respected in the constants). Note that the father's mitochondrial genotype is not inferred with M setting.</p> 	
quadgt.tumor	mutation model in somatic lineage	$10^{-5}..10^{-4}$
quadgt.purity	purity of tumor sample (fraction of reads coming from tumor genome)	between 0 and 1
quadgt.quals	comma-separated list of quality score mappings to error probabilities for $q = 0, 1, 2, \dots$	Phred scale
quadgt.history	list of optimization steps and log-likelihoods that resulted in this set of parameters, written by QUADGT.	list of QUAL and PARAMS entries with log-likelihood in brackets

**Loading and saving parameters.** Model parameters are read from and written to a `.properties` file (of course, you can keep the same text file format but choose another extension). The initial parameter set may be specified by the `-loadpars f` option on the command line. The file *f* may be a local path or a URL. The optimized parameter set is saved by the option `-savepars f`: here, *f* must be a local file that can be written into.

**Manual modification of parameters.** Model parameters can be modified by editing the parameter file (there are dedicated editors, but you can use any text editor: make sure that there is no white space next to the equal signs), or by using the `-D  $p=x$`  option on the command line that sets parameter  $p$  to value  $x$ . Parameter values are initialized in the order of the `-loadpars` and `-D` options. So, if `-D` comes after `-loadpars` on the command-line, then the value specified by `-D` prevails over the parameter file's value for the same parameter. The following command-line arguments apply an optimized model to chromosome X (for a daughter).

```
... -loadpars opt.p -D quadgt.inheritance=Xf chrX.fa xyz_[FMNT].sam
```

The following command-line arguments set the purity. The saved parameter file will contain the provided purity value.

```
... -D quadgt.purity=0.82 -savepars ini.p ...
```

**Encoding mutation models.** QUADGT implements the Jukes-Cantor model and a 4-parameter variant of the Hasegawa-Kishino-Yano model to quantify mutation probabilities and parental divergence. Substitution models are encoded in the parameter file by a letter-code of the model (JC or sHKY) followed by a separator (|) and a comma-separated list of parameters.

```
quadgt.divergence.father=JC|6.021271489463273E-4
quadgt.tumor=sHKY|0.5381772904768927,0.5114960711782521,\
2.3026876803737186E-5,6.957774894927987E-5
```

If only a letter code (JC or sHKY) is given, default model parameters are assumed: the example below selects the Jukes-Cantor model for tumor mutations with default parameters.

```
... -D quadgt.tumor=JC
```

If you specify only a floating-point number, then Jukes-Cantor is assumed:

```
... -D quadgt.divergence.father=1e-5
```

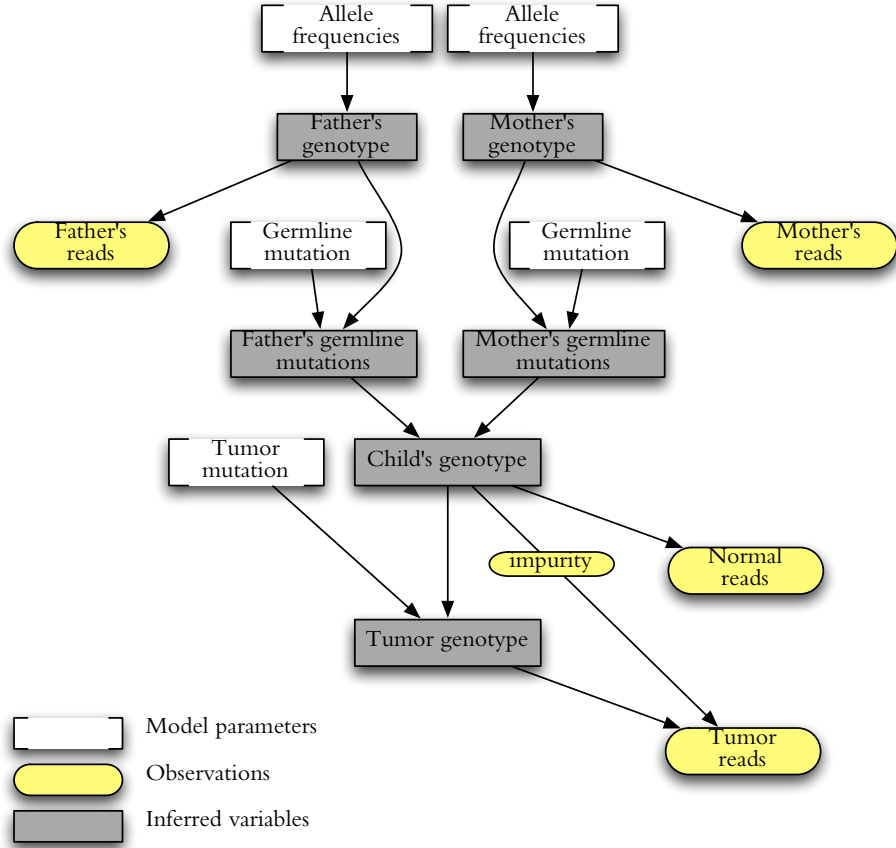


Figure 2: Model structure showing dependencies for genotype inference

### 3 Model structure

Figure 2 illustrates the assumed probabilistic graphical model.

#### 3.1 Molecular evolution models

QUADGT implements the Jukes-Cantor model and a version of the Hasegawa-Kishino-Yano model.

**JC model.** Substitution probabilities for each ancestral-descendant allele pair  $X \rightarrow Y$  are defined by a single parameter  $t$  as

$$\mu_{x \rightarrow y} = \mathbb{P}\{Y = y \mid X = x\} = \begin{cases} \frac{1}{4} + \frac{3}{4}e^{-4t/3} & \{x = y\} \\ \frac{1}{4}(1 - e^{-4t/3}) & \{x \neq y\} \end{cases} \quad (1)$$

**HKY/F84 model.** The model assumes purine-pyrimidine balance  $\%(\text{A} + \text{G}) = \%(\text{C} + \text{T}) = \frac{1}{2}$ , which is automatically satisfied under Chargaff's rules of  $\%\text{A} = \%\text{T}$  and  $\%\text{C} = \%\text{G}$ . Stationary allele frequencies are determined by two parameters  $a, c \in [0, 1]$ :

$$\begin{aligned} \pi[\text{A}] &= \frac{a}{2} & \pi[\text{G}] &= \frac{1-a}{2} \\ \pi[\text{C}] &= \frac{c}{2} & \pi[\text{T}] &= \frac{1-c}{2} \end{aligned} \quad (2a)$$

Accordingly, GC-content is  $\frac{1-a+c}{2}$ , and amino (A + C) content is  $\frac{a+c}{2}$ . Two additional parameters  $q, r$  determine transition and transversion probabilities. For each ancestral-descendant allele pair  $X \rightarrow Y$ , the transition probability is

$$\begin{aligned} \mu_{x \rightarrow y} &= \mathbb{P}\{Y = y \mid X = x\} \\ &= \begin{cases} r \cdot \pi[y] & \text{if } x \rightarrow y \text{ is a transition } \text{A} \leftrightarrow \text{G} \text{ or } \text{C} \leftrightarrow \text{T} \\ q \cdot \pi[y] & \text{if } x \rightarrow y \text{ is a transversion } \{\text{A}, \text{G}\} \leftrightarrow \{\text{C}, \text{T}\} \\ 1 - \sum_{y \neq x} \mu_{x \rightarrow y} & \text{if } x = y \end{cases} \end{aligned} \quad (2b)$$

### 3.1.1 Parental genotype priors

Prior diploid genotype frequencies  $\phi$  at the parents are determined by two parameters: *divergence*  $\nu$  (quadgt.divergence.\*) and *heterozygosity coefficient*  $\gamma$  (quadgt.inb.\*), for mother and father separately. The divergence defines the prior haploid allele frequency for the parent: at each locus, base  $y$  appears with probability

$$\pi[y] = \nu_{x \rightarrow y} \quad (3a)$$

where  $x$  is the reference nucleotide at the locus, and  $\nu_{x \rightarrow y}$  is the substitution probability by the assumed molecular evolution model. At loci with annotations about the frequency of minor alleles, the allele priors are set instead as

$$\pi[y] = \sum_x \pi_{\text{ref}}[x] \cdot \nu_{x \rightarrow y} \quad (3b)$$

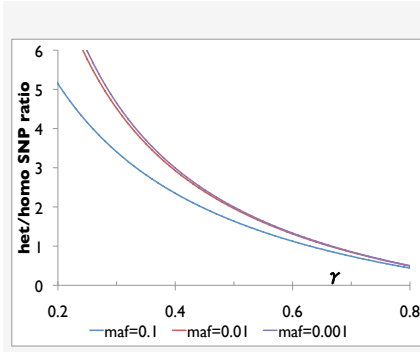
quadgt.divergence.\*  
quadgt.inb.\*



For diploid loci, the coefficient  $\gamma \leq 1$  adjusts the homozygous/heterozygous ratio:

$$\begin{aligned}\phi(xx) &= \left(1 + \gamma \frac{1 - \pi[x]}{\pi[x]}\right) \pi^2[x] \quad (\text{homozygous } xx) \\ \phi(xy) &= 2(1 - \gamma) \pi[x] \pi[y] \quad (\text{heterozygous } xy)\end{aligned}\tag{4}$$

(In other words,  $\gamma$  is numerically analogous to the so-called inbreeding coefficient, or F-statistic.)



At a bi-allelic locus with minor allele frequency  $\text{maf}$ ,

$$\begin{aligned}\frac{\phi(xy)}{\phi(yy)} &= \frac{2(1 - \gamma) \times (1 - \text{maf}) \times \text{maf}}{\text{maf}^2 + \gamma(1 - \text{maf}) \times \text{maf}} \\ &= \frac{2(1 - \gamma)}{\frac{\text{maf}}{1 - \text{maf}} + \gamma} \\ &\approx 2 \frac{1 - \gamma}{\gamma}.\end{aligned}$$

For the human genome,  $\gamma \approx 1/2$ , meaning that heterozygous SNPs are about twice as frequent as homozygous SNPs. At haploid loci (e.g., chromosome Y),  $\gamma$  is not used because genotype frequencies equal allele frequencies:  $\phi(x) = \pi[x]$ .

### 3.1.2 Parental germline mutations and inheritance

Mutations within the parental germlines are determined by a standard molecular evolution model for substitutions in DNA. Let  $x$  denote the original status of the inherited allele from father or mother, and  $x'$  denote the same allele at the end of the germline before gametogenesis. The molecular evolution model specifies the probability of germline mutations  $\mu_{x \rightarrow x'}$  (`quadgt.germline.*`). Inheritance is determined by the parents' and offspring's ploidy. Implemented inheritance models (`quadgt.inheritance`) include diploid autosomes (Mendelian inheritance), as well as those applicable to sex chromosomes (XY system) and mitochondrial genomes.

`quadgt.germline.*`

`quadgt.inheritance`

### 3.1.3 Tumor mutations and purity

Mutations in the tumor genome are determined by a standard molecular evolution model for substitutions in DNA (from normal to tumor genome). As for the germline mutation model, substitutions  $\mu$  at a haploid locus are determined by

a standard molecular evolution model (`quadgt.tumor`). Tumor sample sequencing reads mix the tumor and normal genomes. The expected fraction of tumor-specific reads is set by a *purity* parameter (`quadgt.purity`).

#### 3.1.4 Sequencing errors

Sequencing errors are inferred by relying on the supplied quality values (QUAL field in the SAM files). The corresponding error probabilities are initially determined by the canonical Phred quality scale (error probability =  $10^{-q/10}$ ), but may be replaced with a custom mapping from quality to probability (`quadgtquals`).