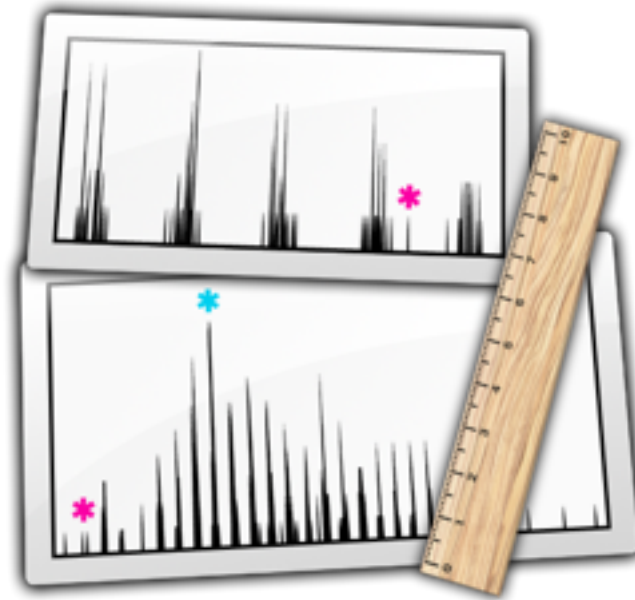


tandem



Summary

All microsatellite analysis software expects allele sizes given in integer numbers, while allele scoring produces allele sizes with two decimals that are dependent not only on fragment length, but also on fluorescent dye, and GC content. Therefore, allele binning is not a trivial task. tandem fills a gap of the microsatellite workflow by rounding allele sizes to valid integers, depending on the microsatellite repeat units. Publish-ready vector graphics output shows allele size distribution and visualizes the rounding method. The average rounding error is given and indicates the overall quality of microsatellite data.

tandem runs natively on Macintosh computers. Source code is written in Ruby and works on Mac, Windows, and Linux computers. tandem is easy to use. All you need is an input file.

Problem

When analyzing microsatellites, one typically uses software like Genemapper (Applied Biosystems) to score the sizes of fluorescently labelled PCR products. These products are a combination of forward primer, flanking region, microsatellite, flanking region, and reverse primer. Of these, both primer sizes are known, and flanking region sizes are assumed to be constant among all individuals. Thus, variation in PCR product sizes among individuals should directly reflect different numbers of microsatellite repeats. In the case of dinucleotide repeats, alleles are expected to be either only even or only odd integers. However, Genemapper calculates allele sizes from comparison with labeled size standards of known size that are added to all samples before running them on a capillary sequencer. Errors are introduced by minor differences between runs and capillaries, the precision limit of sequencers, and imperfect linear regressions between size standard and PCR product run lengths. As a result, calculated allele sizes hardly ever are integers. Instead Genemapper measures allele sizes to two decimals. Its built-in automated binning method requires reference data, which is often not available in population genetic studies and has a number of problems associated with it (Amos *et al.* 2007). Thus, unbinned allele size data is commonly exported in tables like the one shown in fig. 1.

	A	B	C	D	E
1		Locus 1		Locus 2	
2	sample 1	182.02	187.73	189.6	189.6
3	sample 2	183.93	183.93	181.33	189.59
4	sample 3	180.07	187.67	189.66	191.64
5	sample 4	183.88	197.04	191.68	210.18
6	sample 5	183.94	185.82	189.66	189.66
7	sample 6	183.93	185.83	189.69	191.77
8	sample 7	182.04	183.93	193.77	199.9
9	sample 8	183.84	197.07	191.67	210.08
10	sample 9	183.88	185.78	189.56	191.64
11	sample 10	183.89	183.89	189.63	189.63
12	sample 11	183.88	191.44	189.66	199.9
13	sample 12	185.74	193.3	191.71	199.9

Fig. 1: Allele size table, as given by Genemapper.

However, all microsatellite analysis software, including Arlequin (Excoffier *et al.* 2005), Structure (Falush *et al.* 2003), Migrate (Beerli 2006), IM (Hey & Nielsen 2004), and others, expect integer allele sizes. Of course, you could use MS Excel to either cut off all decimals, or to round all values to the nearest integer. You could even program Excel to round only to even or odd numbers in every case. And you could easily find out whether rounding errors are larger to the even, or to the odd numbers. But this still has some problems associated with it. One of them is demonstrated in fig. 2. This figure is part of the tandem output and shows the relative densities of allele sizes for a single microsatellite locus (don't confuse this with stutter bands!). As you can see, allele sizes do not peak at integer numbers. Instead, a consistent negative shift of, say 0.3 bp can be found throughout all values. In this case, rounding to the nearest even number would be fine for most allele sizes, but not for some outliers. The allele marked with an asterisk would be rounded down to 324, and not to the de facto closer peak at ~325.7. It's easy to imagine more serious situations, where a larger fraction of allele sizes is rounded the wrong way.

Worse than this, peaks may not only be shifted, but the average distance between peaks can be slightly less or more than the microsatellite repeat size. Thus you may find dinucleotide loci with average peak distances of, say 1.9, or 2.1, a phenomenon called 'allelic drift' (Idury & Cardon 1997). This is because Genemapper expects collinearity between size standard and PCR product mobility, which is not necessarily the case. Migration rates of alleles depend not only on its length, but also on GC content (Amos *et al.* 2007) and fluorescent labels (Wenz *et al.* 1998). Fig. 3 shows an example where peak distance is less than 2.0. The two peaks marked with asterisks would be rounded to 368 and 370 bp, while the peak in between would be split in two, and merged with the 368 and 370 bp peaks. This will change peak patterns and underestimate allelic variation. Similar problems arise if peak distances are larger than the original repeat size.

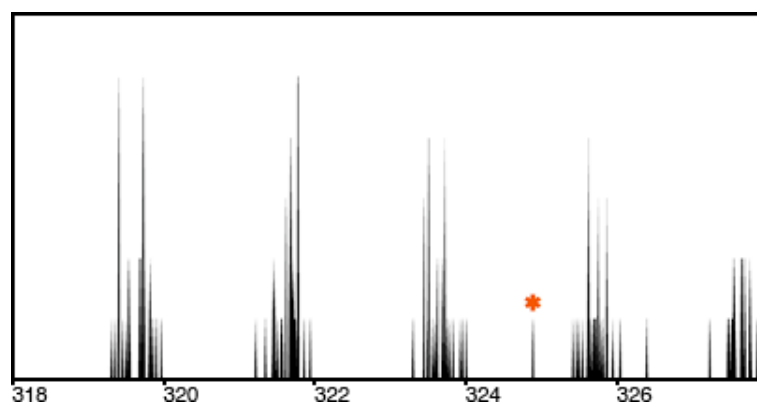


Fig. 2: Allele size peaks with a small shift to the left.

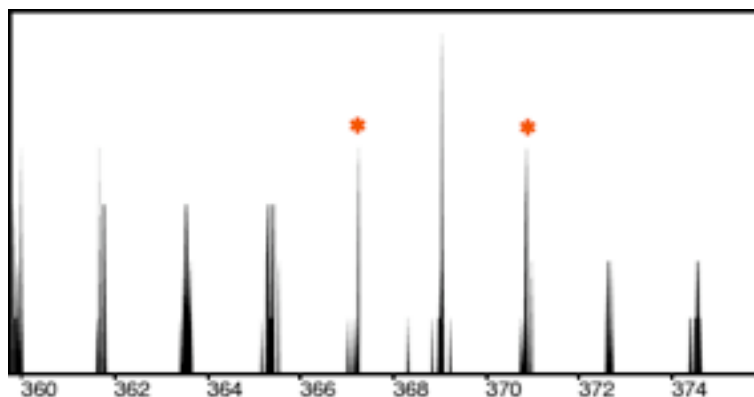


Fig. 3: Allele size peaks with distances of less than 2.0 bp.

Solution

This is where tandem comes in. tandem goes through tab delimited versions of Excel sheets like the one shown in fig. 1, and rounds all allele sizes to integer numbers. But instead of simply rounding to the nearest even or odd number (or other numbers following tri-, tetra-, etc-, nucleotide repeat patterns) tandem finds the most consistent way of rounding. It transforms all observed allele sizes using the power function

$$(\text{transformed allele size}) = a + b \times (\text{observed allele size})^c$$

and exhaustively optimizes parameters a, b, and c, so that rounding errors are minimal when rounding transformed allele sizes to integers that fit the expected nucleotide repeat patterns (e.g. when rounding transformed allele sizes only to even, or only to odd integers).

More precisely, the following values of a, b, and c are tested:

a: $-0.3 \times \text{repeat size}$ — $+0.3 \times \text{repeat size}$; steps of $0.05 \times \text{repeat size}$ (i.e. -0.6, -0.5, -0.4, ..., +0.4, +0.5, +0.6 for dinucleotide repeats)

b: 0.85 — 1.15; steps of 0.01 (i.e. 0.85, 0.86, 0.87, ... 1.13, 1.14, 1.15)

c: 0.985 — 1.015; steps of 0.005 (i.e. 0.985, 0.990, 0.995, 1.000, 1.005, 1.010, 1.015)

Boundaries were chosen based on empirical tests. Also see Amos *et al.* (2007) for boundaries of b. Despite not covering a whole repeat size, boundaries of a are sufficient as rounding of transformed allele sizes is always tested both to odd, and to even integers (for dinucleotide repeats, and in a similar manner for tri-, tetra-, etc. repeats.)

tandem's way of rounding is made transparent by HTML output including publish-ready SVG vector graphic plots that show, per locus, the original allele size distributions, as well as the fitting of the data to the according repeat size pattern (see fig. 4). Two plots are given per locus, one for the full range of allele sizes, and one focussing on the part with the highest density. Per locus, the optimized parameters a, b, c, and the average rounding error are reported alongside other relevant information (see below). Rounding error outliers are highlighted and indicate individuals that should be removed from the data set, or problems with the specified repeat size. The consistency of tandem's rounding scheme immediately becomes obvious from these plots. Grey vertical lines indicate bin centers after allele transformation. If these match the peaks of the allele size distribution (shown in black), as they do in fig. 4, tandem successfully optimized all parameters of the power function, so that rounding errors are minimized. This means that tandem was able to bin alleles in the most consistent way *relative to each other*. If you've got your own microsatellite data set that you're not going to combine with data sets from other laboratories, and if your next step would be a population genetic analysis with software like Arlequin, Structure, Migrate, or IM, you're fine with that, as these programs only use relative distances, and never the absolute values. Some applications, like IM, even expect only the number of repeats (not the allele size in bp), whereby it is commonly assumed that the shortest allele in the data set has a repeat number of 0. In these cases, relative consistency is all you need to worry about, and you don't need to read the rest of this chapter.

However, there are cases when you need to know the true *absolute* fragment length, for example when combining data sets scored in different laboratories, or with different fluorescent dyes that differently affect capillary migration rates. In these cases, you'd like to know, for example, whether the peak on the far left in fig. 4 really is produced by alleles of 362 bp, or rather of 360, or 364 bp. As allelic drift commonly produces between-peak distances that deviate from the actual repeat size (as shown in fig. 3), one could assume that the observed fragment lengths are far from the actual fragment lengths. For example, if you find between-peak distances of roughly 1.8 bp instead of 2.0 bp, as in fig. 3, you could assume that allelic migration is slower than size standard migration by a factor of $1.8/2.0 = 0.9$. Thus, you would expect that a fragment of 400 bp migrates as fast as the size standard of 360 bp ($400 \text{ bp} \times 0.9$), and therefore, that the peak observed at 360 bp actually corresponds to fragments of 400 bp. In order to test this assumption, we've sequenced a microsatellite locus that showed between-peak distances of roughly 1.8 bp. We did this for six specimens. Surprisingly, we did not find the expected large differences between observed and actual fragment lengths. Instead, observed fragment lengths reflected the actual fragment lengths rather precisely, with minor differences of 2-4 bp. This leads us to conclude that the relation between fragment lengths and migration rates is nonlinear, and may for example be affected by changing GC content with length when the microsatellite motif is GC or AT. Taken together, the absolute fragment length is difficult to predict, and tandem cannot guarantee to find it. Given our sequencing results, the above power function was extended to be conservative about absolute fragment lengths. tandem by default fixes the shortest observed allele size per locus to equal the transformed allele size:

$$(\text{transformed allele size}) = a + b \times (\text{observed allele size} - \text{shortest observed allele size})^c,$$

where a is at least the shortest observed allele size ($\pm 0.3 \times \text{repeat size}$; see above). This should result in estimates of fragment lengths that are at least very close to the actual fragment lengths. However, if you want to be sure about absolute fragment lengths, you will need to sequence the locus in one of the specimens (choose a homozygote specimen). tandem provides a way to include your sequencing result in its calculations, and adjusts its estimates accordingly (see chapter 'Advanced settings').

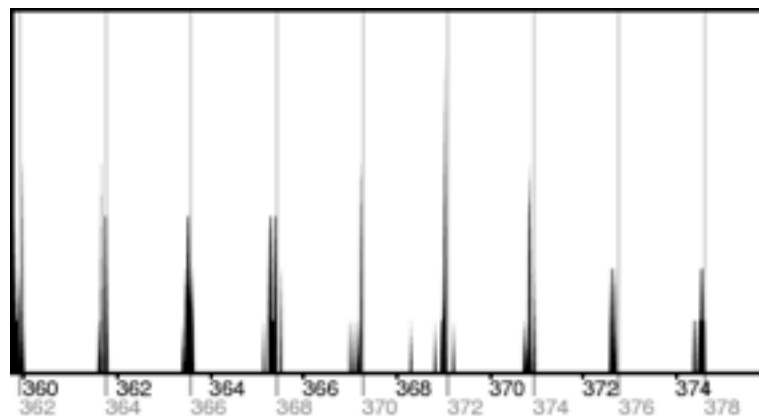


Fig. 4: Values in grey show allele sizes after transformation.

Input file

Since no standard file format exists for the microsatellite workflow, and most downstream applications use their own specific format, there's a number of programs that mainly do conversion from spreadsheet format to specific formats and thus provide excellent starting points for microsatellite analysis. These conversion programs include MSA (Dieringer & Schlötterer 2003), Convert (Glaubitz 2004), and Create (Coombs et al.

2007). While the format of Create is very flexible, those of MSA and Convert are more strict, but still very easy to build. tandem is designed for minimal configuration effort, and thus supports both MSA and Convert format. This means you can create your MSA or Convert input file as usual, but without the need of rounding allele sizes. tandem does that for you.

If using Convert format, there's one minor detail to add: Since Convert format does not include information about the repeat size, but this information is important for tandem, please add the repeat size (i.e. whether its a di-, tri-, etc- nucleotide repeat) in the cells directly above the cells containing the names of the loci. While tandem nevertheless tries to estimate the repeat size from the data, and alerts you if there's a conflict between its estimate and the repeat size you specified, this estimation doesn't always work. You're on the safe side if you specify a repeat size.

MSA and Convert input files are shown in fig. 5 and fig. 6. Examples for both MSA and Convert format are included in the tandem distribution package as Excel (xls) files. These files include comments on how to enter information according to the respective formats. **In order to run these example files with tandem, you need to open them in Excel and save them again as tab-delimited files!** (Click 'Save as' and choose 'Tab Delimited Text (.txt)' format). **This applies not only to the example files, but to all input files.**

	A	B	C	D	E
1	2			2	
2					
3			Locus 1		
4	pop 1	d	1	182.02	187.73
5	pop 1	d	1	183.93	183.93
6	pop 1	d	1	180.07	187.67
7	pop 1	d	1	183.88	197.04
8	pop 1	d	1	183.94	185.82
9	pop 1	d	1	183.93	185.83
10	pop 2	d	1	182.04	183.93
11	pop 2	d	1	183.84	197.07
12	pop 2	d	1	183.88	185.78
13	pop 2	d	1	183.89	183.89

Fig. 5: MSA input format, unbinned.

	A	B	C	D	E
1	Comment				
2	npops = 2				
3	nloci = 7	2		2	
4		Locus_1		Locus_2	
5	pop = pop_1				
6	sample_1	182.02	187.73	189.6	189.6
7	sample_2	183.93	183.93	181.33	189.59
8	sample_3	180.07	187.67	189.66	191.64
9	sample_4	183.88	197.04	191.68	210.18
10	sample_5	183.94	185.82	189.66	189.66
11	sample_6	183.93	185.83	189.69	191.77
12	pop = pop_2				
13	sample_7	182.04	183.93	193.77	199.9

Fig. 6: Convert input format, unbinned.

Running tandem

Mac version (GUI)

Running the Mac version of tandem could hardly be easier. Simply double-click tandem, and you'll be prompted to choose an input file. Do so, and tandem immediately starts running. Runs take around twenty seconds per locus, for data sets containing a few hundred individuals, on a 2 GHz iMac. At the end of the run, tandem gives you a short message, telling you which file format it recognized (MSA or Convert), and where it wrote the output files (which is in the folder of the input file).

Source code

If using the source code version, I recommend to copy 'tandem.rb' as well as the folder 'tandem_resources' into the directory of the input file. And if you're using Windows, I also recommend to choose, as a directory for all these files something as simple as the Desktop, because longer paths are difficult to navigate to with the Windows command line.

While the programming language Ruby is installed by default on Macintosh computers, you'll first need to get it if you're on Windows. There's a number of packages available, but I recommend the one-click, self-contained installer found at <http://rubyinstaller.rubyforge.org/wiki/wiki.pl>. Once you've got Ruby, open the Terminal (on a Mac) or the command prompt (on Windows; click Start > Run, then type 'command'). Test whether Ruby is installed by typing `ruby -v`, and you should get a version number as reply. Navigate into the directory of the input file by typing `cd` (on Mac) or `chdir` (on Windows), followed by a space and the path to this directory (e.g. `chdir DESKTOP`). Once you're there, type

`ruby tandem.rb -i <input_file_name>`

to start the run, where `<input_file_name>` needs to be replaced with the actual name of your input file. The output will be written to the same directory.

Output files

tandem outputs a file which looks exactly like your input file, but with rounded allele sizes. It is called '`<input_file_name>_tandem.txt`'. tandem also writes an HTML file with embedded SVG vector graphics, called '`<input_file_name>_tandem.htm`'. The SVG images plot the frequency of allele sizes, first over the full allele size range, then over the shortest range containing 50% of all allele sizes. You can edit these SVG images for publication using software such as Adobe Illustrator or a free alternative, such as [Inkscape](#). You'll find the SVG plots in folder '`<input_file_name>_tandem_resources`'.

In addition, tandem writes a bin file and a panel file that can be imported into Genemapper:

'`<input_file_name>_tandem_bins.txt`' and '`<input_file_name>_tandem_panel.txt`'. These files enable you to use tandem results as reference data for future analyses with Genemapper. This makes sense if you have a large data set that probably includes all the existing alleles for a particular locus, and if the average rounding error is small. If you intend to use tandem results as Genemapper reference data, you should also consider sequencing each locus in one individual, and using a fixpoint file (see 'Advanced settings') to improve reliability in absolute allele size estimates. The bin file contains, per locus, all allele sizes after transformation and rounding (*i.e.* integer numbers) as Genemapper bin names, and the corresponding untransformed alleles as Genemapper bin centers. This reference data allows you to use Genemappers automated binning function, and to export binned alleles directly from Genemapper. This may save some time in your microsatellite workflow, however, if your reference data wasn't exhaustive, you risk missing new alleles that fall outside of established bins. Also note that per locus, you should use the same fluorescent dye

in future analyses as you did when you created the reference data with tandem. This is important because fluorescent dyes may affect migration rates of alleles.

Furthermore, the HTML file contains the following information for every locus:

- **Specified repeat size:** this is what you specified in the cells above loci names. If your specification is readable for tandem (i.e. if you entered an integer number), it will be used for rounding. The maximum repeat size is currently set to 8. If you seriously think your repeat size is larger, contact me, and I can change this setting. It is recommended that you specify the repeat size, if you know it, as tandem cannot guarantee to estimate the repeat size correctly.
- **Estimated repeat size:** tandem's best guess for the actual repeat size, given the data. If this estimate is not what you specified, it will be written in red to alert you. You may want to check the respective SVG plots in this case. If you did not specify a repeat size, tandem uses its own estimate for rounding. If tandem is not able to estimate the repeat size, a question mark will be given. If no repeat size is specified, and tandem cannot estimate it, the default repeat size of 2 will be used for rounding.
- **Individuals:** The number of individuals for which allele size information is found. This is calculated as the total number of allele sizes divided by two, as two alleles are expected per individual.
- **Allele range:** The smallest and the largest allele size of this locus are given.
- **a, b, c:** Parameters of the power function
(transformed allele size) = $a + b \times (\text{observed allele size} - \text{smallest observed allele size})^c$
that is applied to all observed allele sizes in order to minimize rounding error.
- **Fixed:** This information is given if you've specified a fixpoint for a particular locus (see 'Advanced settings'). It is not included in the summary table, but shows up in the text given for every locus, just above the SVG plots. This also tells you how far all values had to be shifted in order to agree with your specified fixpoint.
- **Rounded to:** In case of dinucleotide repeats, this tells you whether tandem rounded to even or odd numbers. Similar for tri-, tetra-, etc- repeats.
- **Error:** The average rounding error over all allele sizes of this locus. The error is always between 0 and the repeat size divided by two, and is highlighted in red, if it is above 0.1 times the repeat size. Thus, for dinucleotide repeats it is between 0 and 1 and highlighted when above 0.2. High error values indicate either problems with your data, or problems with the rounding method (which, again, is more likely when your data is bad). Check this value in any case, and consider it for publication!
- **Error outliers:** If you mark this checkbox, the 30 worst alleles are highlighted by asterisks in both the full range and detail plots, and listed in a table below both plots. Only alleles with rounding errors greater than 0.5 are given. Move the mouse cursor to a particular asterisk, and the name and row of the respective individual will appear in a tooltip. The table further includes information on original and transformed allele size, bin center, and rounding error. JavaScript must be enabled for this feature.

Advanced settings

This section explains possible arguments when using the source code version on the command line.

- **ruby tandem.rb -v**
tandem tells you its version number and exits.

■ **ruby tandem.rb -verbose -i <input_file_name>**

tandem runs as normal, but gives you detailed information about the progress of its exhaustive search.

■ **ruby tandem.rb -linear -i <input_file_name>**

tandem uses a linear function instead of the power function described above. This means that parameter c is fixed to 1.000. With this setting, the exhaustive search is roughly seven times faster than with the power function, but results may not be as good. If you have a very large data set, it may be a good idea to use this function. You will be able to tell from the SVG plots whether tandem found a good solution or not. If it hasn't, you should run tandem again with standard settings.

This argument can be combined with `-verbose`.

■ **ruby tandem.rb -neldermead -i <input_file_name>**

tandem runs a heuristic search using the Nelder-Mead Downhill Simplex method to optimize parameters a , b , and c of the above power function. This search is much faster than the exhaustive search, however, the method is prone to getting trapped in local optima. If you have a very large data set and a small number of alleles, you may consider using this setting. However, you should carefully inspect the SVG plots and rerun tandem with standard settings in case it couldn't find a good solution.

This argument can be combined with `-verbose`.

■ **ruby tandem.rb -i <input_file_name> -f <fixpoint_file_name>**

If you know the actual absolute allele size of a specimen (*e.g.* because you've sequenced the locus for this specimen), you can specify this allele size in a separate file, and tandem will adjust its transformation of allele sizes so that the specimen's observed allele size is rounded to the specified actual allele size. The file should be in the same folder as the input file, `tandem.rb`, and the folder `tandem_resources`, and it should look like shown in fig. 7. Per line, tandem expects (i) the name of the locus, exactly as stated in the input file, (ii) the observed allele size of the respective specimen, and (iii) the actual allele size as determined by sequencing. As in the input file, values should be separated by tabs (if you prepare the `fixpoint_file` in MS Excel, save as tab-delimited file). You can specify fixpoints for several loci in the same file, but you can only specify one fixpoint per locus.

	A	B	C	D	E
1	Locus_1	147.56	150		
2	Locus_4	283.12	286		
3	Locus_5	187.74	183		
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					

Fig. 7: Fixpoint file with locus names in column A, observed allele sizes in column B, and actual allele size in column C. The file should be saved in tab-delimited format, in the folder of the input file.

How to cite tandem

Matschiner M, Salzburger W (2009) TANDEM: integrating automated allele binning into genetics and genomics workflows. *Bioinformatics*, **25**(15), 1982-1983.

References

Amos W, Hoffman JI, Frodsham A *et al.* (2007) Automated binning of microsatellite alleles: problems and solutions. *Molecular Ecology Resources*, **7**, 10-14.

Berli P (2006) Comparison of Bayesian and maximum-likelihood inference of population genetic parameters. *Bioinformatics*, **22**, 341-345.

Coombs JA, Letcher BH, Nislow KH (2007) Create 1.0 - Software to create and convert codominant molecular data. Available at <http://lsc.usgs.gov/CAFL/Ecology/Ecology/html>.

Dieringer D, Schlötterer, C (2003) Microsatellite Analyser (MSA): a platform independent analysis tool for large microsatellite data sets. *Molecular Ecology Notes*, **3**, 167-169.

Excoffier L, Laval G, Schneider S (2005) Arlequin ver. 3.0: an integrated software for population genetics data analysis. *Evolutionary Bioinformatics Online*, **1**, 47-50.

Falush D, Stephens M, Pritchard J (2003) Inference of population genetic structure using multilocus genotype data: linked loci and correlated allele frequencies. *Genetics*, **164**, 1567-1587.

Glaubitz, JC (2004) Convert: A user-friendly program to reformat diploid genotypic data for commonly used population genetic software packages. *Molecular Ecology Notes*, **4**, 309-310.

Hey J, Nielsen R (2004) Multilocus methods for estimating population sizes, migration rates and divergence time, with applications to the divergence of *Drosophila pseudoobscura* and *D. persimilis*. *Genetics*, **167**, 747-760.

Idury RM, Cardon LR (1997) A simple method for automated allele binning in microsatellite markers. *Genome Research*, **7**, 1104-1109.

Wenz HM, Robertson JM, Menchen S *et al.* (1998) High-precision genotyping by denaturing capillary electrophoresis. *Genome Research*, **8**, 69-80.