CellPhy - SC-Caller conversion

Alexey Kovlov, João M Alves*, Alexandros Stamatakis & David Posada

July 2020

1 Introductory Note

In the following supporting document, we will show how to easily transform a VCF file generated by the ***SC-Caller*** (a single-cell specific variant caller) into a **CellPhy-GL** readable VCF. Before we start, users should make sure that both *BCFtools* and *Tabix* from **Samtools** are installed and available.

---

2 VCF file conversion

SC-Caller VCF files can be converted by running the *sc-caller.conversion.sh* script as follows:

```
$ ./sc-caller.conversion.sh
Usage: ./sc-caller.conversion.sh inputVCF SamplePrefix
Created by: Alexey Kovlov, Joao M Alves, Alexandros Stamatakis & David Posada - June 2020

$ ./sc-caller.conversion.sh CRC_Cell-A.22.sccaller.vcf Cell-A
```

If everything went as expected, you should have generated a new VCF (**Cell-A.PL-fixed.vcf**) that is now ready for downstream analysis. Below we provide a detailed explanation of which changes are being made to the original VCF file.

2.1 Step-by-step explanation

While many callers currently being used with single-cell genomic data generate VCF files with a standard "PL" field (e.g., GATK-HaplotypeCaller, Monovar, Prosolo), it is important to highlight that the "PL" definition may differ from its standard meaning in different tools. Indeed, SC-Caller for instance uses the "PL" field to store not only the likelihood of heterozygous and alternative homozygous genotypes, but also the likelihood of sequencing noise and amplification artifacts.

Below we show an example of a VCF generated using SC-Caller (version 2.0.0).

```
$ grep -v "#" CRC_Cell-A.22.sccaller.vcf | head -n 5
#CHROM  POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  CELL001
22  16149851    .   G   A   13  .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:NA:0,6:0.6:13:91,54,13,0
22  16190307    .   A   -1C 4   .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:NA:0,1:0.6:4:15,9,2,0
22  16193737    .   A   C,G 150 multiple-genotype   NS=1    GT:SO:AD:BI:GQ:PL   0/0:True:317,1:0.993:150:91,227,64,11570
22  16195864    .   G   A   150 .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:True:136,1:0.6:150:39,93,310,3479
```

Looking at this output, you should notice that:

- SC-Caller renames the single-cell to "CELL001";
- The VCF contains calls other than single-nucleotide variants (i.e., SNVs);
- The "PL" field of **bi-allelic** sites is composed of 4 entries, as opposed to the 3 values usually observed in VCF files that were generated following the *format specifications**;
- The "PL" field will always contain 4 entries, regardless of the amount of alternative variants detected.

As a consequence, we will need to rename the sample to its proper ID and trim the VCF to exclude all indels and non-biallelic positions (as we won't be able to get the likelihood scores for all possible genotypes). This can be easily done using BCFtools.

```
# Rename sample in VCF
$ cat temp_rename
CELL001 Cell-A
$ bcftools reheader -s temp_rename CRC_Cell-A.22.sccaller.vcf -o temp.renamed.vcf

# Remove indels and non-biallelic sites
$ bcftools view --types snps -f "." temp.renamed.vcf -o temp.snvs.vcf
```

Once this is done, we should end up with a trimmed VCF that solely contains bi-allelic sites:

```
$ grep -v "##" temp.snvs.vcf | head -n 5
#CHROM  POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  Cell-A
22  16149851    .   G   A   13  .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:NA:0,6:0.6:13:91,54,13,0
22  16195864    .   G   A   150 .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:True:136,1:0.6:150:39,93,310,3479
22  16195889    .   T   C   150 .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:True:136,1:0.6:150:34,89,306,4496
22  16195900    .   A   G   150 .   NS=1    GT:SO:AD:BI:GQ:PL   0/0:True:132,1:0.6:150:34,87,298,4273
```

Afterwards, we need to transform the "PL" field at each site into the standard "PL" format. Following **SC-caller authors' suggestions**, we take the highest likelihood score of the first two values (i.e., sequencing noise, amplification artifact) as the phred-scaled genotype likelihood of the reference homozygous (0/0) genotype, and the remaining values as the likelihood for heterozygous (0/1) and alternative homozygous (1/1) genotypes, respectively.

Cautionary remark on SC-Caller developers suggestion:

It is perhaps important to mention that the SC-Caller authors suggest to take "the bigger number as the PL combined" as the 0/0 genotype likelihood. We interpreted this suggestion as to take the one with the "the highest likelihood" (which in this case would be the smallest integer value).

Our custom script will then essentially rename the previous "PL" field as "FPL" and add a standard "PL" field to our VCF to make it CellPhy readable:

```
$ grep -v "##" temp.PL-fixed.vcf | head
#CHROM  POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  Cell-A
22  16149851    .   G   A   13  .   NS=1    GT:SO:AD:BI:GQ:FPL:PL   0/0:NA:0,6:0.6:13:91,54,13,0:54,13,0
22  16195864    .   G   A   150 .   NS=1    GT:SO:AD:BI:GQ:FPL:PL   0/0:True:136,1:0.6:150:39,93,310,3479:39,310,3479
22  16195889    .   T   C   150 .   NS=1    GT:SO:AD:BI:GQ:FPL:PL   0/0:True:136,1:0.6:150:34,89,306,4496:34,306,4496
22  16195900    .   A   G   150 .   NS=1    GT:SO:AD:BI:GQ:FPL:PL   0/0:True:132,1:0.6:150:34,87,298,4273:34,298,4273
```

---

3 Merging VCF files for downstream analyses

Once all single-cell VCF files are converted, we can easily merge them into a multi-sample VCF using BCFtools.

```
$ for i in *PL-fixed.vcf
  do
  bgzip $i
  tabix -p vcf ${i}.gz
  done
$ ls *.vcf.gz > listMERGE
$ bcftools merge -l listMERGE -O vcf -o CRC-allCells.vcf
```

The VCF is now ready for CellPhy (or for any additional filtering steps).