HiC pairs

After mapping the Hi-C sequencing reads to the reference genomes, the resulting BAM files still contain many alignments that do not represent true chromatin contacts. To address this, the alignments were processed using the tool pairtools to reconstruct valid Hi-C contact pairs. This step identifies which two genomic regions were physically ligated together during the Hi-C experiment and removes common technical artefacts, such as self-ligated fragments, dangling ends, and duplicated read pairs. Only uniquely mapped read pairs were retained. The output forms the basis for building contact matrices and the 3D genome structure.

```
#!/bin/bash -l
#SBATCH --job-name=hic_pairs
#SBATCH --output=/scratch/grp/msc_appbio/Group2_ABCC/HiC/logs/%x_%j.out
#SBATCH --error=/scratch/grp/msc_appbio/Group2_ABCC/HiC/logs/%x_%j.err
#SBATCH --time=06:00:00
#SBATCH --mem=32G
#SBATCH --cpus-per-task=8
#SBATCH -p msc_appbio

# Convert sorted BAM alignments into filtered Hi-C contact pairs

module load python
module load samtools
source activate hic-env   # must contain pairtools

BASE=/scratch/grp/msc_appbio/Group2_ABCC/HiC
ALIGN=$BASE/results/alignments
PAIRS=$BASE/results/pairs
REF=$BASE/references
CHROMS=$BASE/results/chromsizes

mkdir -p $PAIRS $CHROMS

# Build chrom.sizes files from the exact FASTA references you are using.
# This gives pairtools a definitive list/order/length of contigs.
if [ ! -f $CHROMS/BY4742_SRR6823436.chrom.sizes ]; then
  samtools faidx $REF/BY4742_SRR6823436.contigs.fasta
  cut -f1,2 $REF/BY4742_SRR6823436.contigs.fasta.fai > $CHROMS/BY4742_SRR6823436.chrom.sizes
fi

if [ ! -f $CHROMS/SY14.chrom.sizes ]; then
  samtools faidx $REF/SY14.fa
  cut -f1,2 $REF/SY14.fa.fai > $CHROMS/SY14.chrom.sizes
fi

# Define BAM -> chrom.sizes mapping
declare -a JOBS=(
  "BY4742_R1_BY4742_SRR6823436.sorted.bam BY4742_SRR6823436.chrom.sizes"
  "BY4742_R2_BY4742_SRR6823436.sorted.bam BY4742_SRR6823436.chrom.sizes"
  "SY14_R1_SY14.sorted.bam SY14.chrom.sizes"
  "SY14_R2_SY14.sorted.bam SY14.chrom.sizes"
)

cd $ALIGN

for ENTRY in "${JOBS[@]}"; do
  set -- $ENTRY
  BAM=$1
  CSIZES=$2
  SAMPLE=$(basename $BAM .sorted.bam)

  echo "=== Making pairs for $SAMPLE ==="

#   keep only uniquely-mapped pairs (pair_type == "UU")
#   remove common Hi-C artefacts: self, dangling ends, corners
#   deduplicate pairs

  pairtools parse \
      --nproc-in 4 \
      --nproc-out 4 \
      --chroms-path $CHROMS/$CSIZES \
      $BAM \
    | pairtools sort --nproc 4 \
    | pairtools select '(pair_type == "UU") and (not is_self) and (not is_dangling) and (not is_corner)' \
    | pairtools dedup --nproc 4 \
    > $PAIRS/${SAMPLE}.pairs.gz

  pairtools stats $PAIRS/${SAMPLE}.pairs.gz > $PAIRS/${SAMPLE}.pairs.stats.txt

  echo "Wrote: $PAIRS/${SAMPLE}.pairs.gz"
done

echo "All pairs generated."
```