# 1 Introduction

This document supplements the paper, '**Improving the gene-structure annotation of the Apicomplexan parasite *Neospora caninum* fulfils a vital requirement towards an *in silico* derived vaccine'**. It provides the LINUX command line syntax for all programs and scripts executed in the study. **A program's default parameter values are used unless otherwise stated.** Note that some commands are line wrapped due to the restricted page width.

# 2 Transcriptome assembly

NC-Liverpool tachyzoites were collected at three different passages for RNA extraction. Consequently, there were three biological replicates with corresponding RNA-Seq datasets, referred henceforth as NCLIV1, NCLIV2, and NCLIV3. Each RNA-Seq dataset was processed separately using the same methodology. Raw reads comprised paired-ends in a FASTQ format with 100 bp read lengths.

# count the number of reads (R1 and R2 is used to distinguish the two read files)
wc -l NCLIV1_R1.fastq
= 86209384 / 4 (a fastq file contains 4 lines per read) = 21552346

# number of reads in each file
NCLIV1_R1.fastq = 21552346
NCLIV1_R2.fastq = 21552346
NCLIV2_R1.fastq = 18503094
NCLIV2_R2.fastq = 18503094
NCLIV3_R1.fastq = 20328775
NCLIV3_R2.fastq = 20328775

## *2.1 Step 1 (part of quality control)*

Adaptor sequences still remaining in the raw reads were removed using cutadapt (version 1.2.1). For R1 files, adaptor sequence mainly occurred at the beginning of read with a poly-A tail and for R2 files the adaptor sequence mainly occurred within or at the end of read with a poly-A tail. Only one type of adaptor sequence was found in each read file:

NCLIV1 R1 = TruSeq Adapter, Index 3
GATCGGAAGAGCACACGTCTGAACTCCAGTCACTTAGGCATCTCGTATGCCGTCTTCTGCTTG
NCLIV2 R1 = TruSeq Adapter, Index 4
GATCGGAAGAGCACACGTCTGAACTCCAGTCACTGACCAATCTCGTATGCCGTCTTCTGCTTG
NCLIV3 R1 = TruSeq Adapter, Index 5

GATCGGAAGAGCACACGTCTGAACTCCAGTCACACAGTGATCTCGTATGCCGTCTTCTGCTTG

NCLIV1. NCLIV2, and NCLIV3 R2 = Reverse complement TruSeq Universal Adapter
AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT

Example syntax:
cutadapt --help
cutadapt --overlap=10 -g
AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT
NCLIV3_R2.fastq > adaptor_removed_NCLIV3_R2.fastq

Note: overlap=10 – only finds a match with more than 10 characters of the adaptor sequence (default = 3)

## *2.2 Step 2 (part of quality control)*

FASTX-Toolkit (http://hannonlab.cshl.edu/fastx_toolkit/index.html) version 0.0.14 provided the short-reads pre-processing tools for FASTQ files .

fastx_trimmer -f 15 -Q33 -i adaptor_removed_NCLIV1_R1.fastq -o trimmed_NCLIV1_R1.fastq

Note: f 15 – removes the first 15 bases i.e. read length reduced from 100 to 85 bp; Q33– indicates the file uses Illumina encoded quality scores rather than Sanger encoding.

fastq_quality_filter -q 20 -p 80 -Q33 -i trimmed_NCLIV1_R1.fastq -o Q_trimmed_NCLIV1_R1.fastq

Note: q 20 and p 80 retains only reads for which 80% of their bases are above a Phred quality score of 20.

fastx_clipper -l 15 -Q33 -i Q_trimmed_NCLIV1_R1.fastq -o size_Q_trimmed_NCLIV1_R1.fastq

Note: l – removes reads of less than 15 bp in length

## *2.3 Step 3 (part of quality control)*

Given the use of VERO cells as a host cell line for the growth of *N. caninum*, the raw reads were contaminated with reads derived from *Chlorocebus sabaeus* (Green monkey) mRNA. All reads were mapped to the *C. sabaeus* genome using Tophat2 (Version 2.0.9). Reads not mapped were assumed to be *N. caninum* reads.
# downloaded Green Monkey chromosomes from:
ftp://ftp.ncbi.nlm.nih.gov/genbank/genomes/Eukaryotes/vertebrates_mammals/Chlorocebus_sabaeus/Chlorocebus_sabeus_1.0/Primary_Assembly/assembled_chromosomes/FASTA/

# used wget to download each chromosome

```
wget
ftp://ftp.ncbi.nlm.nih.gov/genbank/genomes/Eukaryotes/vertebrates_mammals/Chlorocebus_sabaeus/
Chlorocebus_sabeus_1.0/Primary_Assembly/assembled_chromosomes/FASTA/chr2.fa.gz

# decompressed gz files
gunzip *.gz

# joined all monkey chromosome files into 1 file
cat * >> vero.fa

# built index for monkey genome using bowtie2 (version 2.2.0)

bowtie2-build vero.fa vero

# tested that the index was properly installed
bowtie2 -c vero GCGTGAGCTATGAGAAAGCGCCACGCTTCC

# aligned paired-end reads to monkey genome with tophat2
tophat2 vero size_Q_trimmed_NCLIV1_R1.fastq size_Q_trimmed_NCLIV1_R2.fastq
```

Note: reads not mapped, assumed to be *Neospora* reads. By default unmapped reads are saved in a file called unmapped.bam

## *2.4 Step 4 (transcriptome assembly – 'mapping-first' approach)*

```
# converted BAM to SAM (Samtools version 0.1.19)
samtools view -h -o unmapped.sam unmapped.bam

# converted SAM to fastq using picard-tools-1.99
java -Xmx8g -jar SamToFastq.jar INPUT=unmapped.sam FASTQ=l_NCLIV1_R1.fastq
SECOND_END_FASTQ=r_NCLIV1_R2.fastq VALIDATION_STRINGENCY=SILENT
```

Note: SECOND_END_FASTQ is required for second end of the pair fastq; and setting validation stringency to SILENT prevented program from an unexpected termination (default value is STRICT).

```
# joined all Neospora chromosome files into 1 file
cat *.fasta >> neospora.fa

# built index for Neospora genome
bowtie2-build neospora.fa neospora
```

# aligned paired-end reads to Neospora genome

```
tophat2 neospora l_NCLIV1_R1.fastq r_NCLIV3_R2.fastq -p 8 -i 20 -I 8000 --coverage-search --min-coverage-intron 20 --max-coverage-intron 8000
```

Note: p – the number of threads to align reads (default = 1); i – the minimum intron length (default = 70); I – the maximum intron length (default is 500000); coverage-search – enables coverage search; min-coverage-intron – the minimum intron length that may be found during coverage search (default = 50); max-coverage-intron –the maximum intron length that may be found during coverage search (default = 20000). The minimum and maximum intron values were determined from *Toxoplasma gondii* gene annotation (see Table 3 in paper). The mapped reads are saved in a file called accepted_hits.bam by default.

# assembled transcripts with Cufflinks (version 2.1.1)

```
cufflinks accepted_hits.bam -p 8 -I 80000 --min-intron-length 20
```

Note: p – the number of threads to align reads (default = 1); max-intron-length – maximum intron length (default is 300000); min-intron-length – minimum intron size allowed in genome (default = 50 bp). The minimum and maximum intron values were determined from *Toxoplasma gondii* gene annotation (see Table 3 in paper). The output format for transcripts was in a General Transfer Format (GTF). An in-house Perl script was used to extract exon base pair (bp) locations and mRNA sequences.

## *2.5 Step 5 (transcriptome assembly – 'assembly-first' (de novo) approach)*

This approach used Trinity to assemble reads directly into transcripts (contigs). The contigs were splice-aligned to the *N. caninum* genome to delineate intron and exon structures using PASA with BLAT and GMAP as the aligners.

# downloaded trinity from
http://trinityrnaseq.sourceforge.net/

# command to determine version of Trinity

```
./Trinity.pl --version
```

Trinity version = trinityrnaseq_r2013_08_14

# command to run Trinity

```
./Trinity.pl --seqType fq --JM 10G --left l_NCLIV1_R1.fastq  --right r_NCLIV1_R2.fastq --CPU 8
```

Note: seqType fq – indicates the read file is in a fastq format; CPU  number of CPUs to use (default = 2). The output is a FASTA file containing the contigs (default name = Trinity.fasta)

# ran PASA for each chromosome

```
Launch_PASA_pipeline.pl -c alignAssembly.config -C -R -g chrIa.fasta -t Trinity.fasta --ALIGNERS
blat,gmap --CPU 8
```

Note: The above command creates a "sample_mydb_pasa" directory in mysql_data directory. This name is determined by a setting in the configuration file alignAssembly.config
e.g.
```
 # MySQL settings
MYSQLDB=sample_mydb_pasa
```

The output format for transcripts is in a General Transfer Format (GTF). An in-house Perl script was used to extract exon base pair (bp) locations and mRNA sequences.

# 3 Assembly of Wellcome Trust Sanger Institute RNA-Seq reads

The Wellcome Trust Sanger Institute determined the transcriptome of the invasive stage (tachyzoite) of *N. caninum* Liverpool using mRNA sequencing on an Illumina GAIIx machine. Samples of RNA were extracted at days three, four and six from parasites grown asynchronously in cell culture. The RNA-Seq data derived from day three and four were reported in the study by Reid and colleagues to be similar and were pooled for most analyses.

Transcriptome sequencing data (i.e. RNA-Seq reads) were submitted to ArrayExpress (https://www.ebi.ac.uk/arrayexpress/) with accession # E-MTAB-549 for N. caninum sequences. This data is also available from NCBI Sequence Read Archive (SRA) at http://www.ncbi.nlm.nih.gov/sra/.

```
# download ERX010661 DAY 3 paired-end reads
wget ftp://ftp.sra.ebi.ac.uk/vol1/fastq/ERR029/ERR029935/ERR029935_1.fastq.gz
wget ftp://ftp.sra.ebi.ac.uk/vol1/fastq/ERR029/ERR029935/ERR029935_2.fastq.gz

# download ERX010660 DAY 4 paired-end reads
wget ftp://ftp.sra.ebi.ac.uk/vol1/fastq/ERR029/ERR029939/ERR029939_1.fastq.gz
wget ftp://ftp.sra.ebi.ac.uk/vol1/fastq/ERR029/ERR029939/ERR029939_2.fastq.gz

# decompress files
gzip -d ERR029935_1.fastq.gz
gzip -d ERR029935_2.fastq.gz
gzip -d ERR029939_1.fastq.gz
gzip -d ERR029939_2.fastq.gz

#Number of reads for each file
ERR029935_1.fastq = 15,711,963
```

ERR029935_2.fastq = 15,711,963
ERR029939_1.fastq = 16,904,928
ERR029939_2.fastq = 16,904,928

# build index for neospora genome
bowtie2-build neospora.fa neospora

# aligned DAY 3 and DAY 4 paired-end reads to Neospora genome
tophat2 neospora ERR029935_1.fastq,ERR029939_1.fastq ERR029935_2.fastq,ERR029939_2.fastq -p 8 -i 20 -I 8000 --coverage-search --min-coverage-intron 20 --max-coverage-intron 8000

Note: p – the number of threads to align reads (default = 1); i – the minimum intron length (default = 70); I – the maximum intron length (default is 500000); coverage-search – enables coverage search; min-coverage-intron – the minimum intron length that may be found during coverage search (default = 50); max-coverage-intron –the maximum intron length that may be found during coverage search (default = 20000). The minimum and maximum intron values were determined from *Toxoplasma gondii* gene annotation (see Table 3 in paper). The mapped reads are saved in a file called accepted_hits.bam by default.

# Tophat output information
Left reads:
Input: 32616891
Mapped: 22881391 (70.2% of input)
of these: 1125690 ( 4.9%) have multiple alignments (2305 have >20)
Right reads:
Input: 32616891
Mapped: 23591966 (72.3% of input)
of these: 1149306 ( 4.9%) have multiple alignments (2362 have >20)
71.2% overall read alignment rate.

Aligned pairs: 21736706

# assembled transcripts with Cufflinks (version 2.1.1)
cufflinks accepted_hits.bam -p 8 -I 80000 --min-intron-length 20

Note: p – the number of threads to align reads (default = 1); max-intron-length – maximum intron length (default is 300000); min-intron-length – minimum intron size allowed in genome (default = 50 bp). The minimum and maximum intron values were determined from *Toxoplasma gondii* gene annotation (see Table 3 in paper). The output format for transcripts was in a General Transfer Format (GTF). An in-house Perl script was used to extract exon base pair (bp) locations and mRNA sequences.

# 4 BLAST

The NCBI Basic Local Alignment Search Tool (BLAST) finds regions of local similarity between sequences.

```
# download BLAST + version 2.2.28 for LINUX
lwp-download ftp://ftp.ncbi.nlm.nih.gov/blast/db/ncbi-blast-2.2.28+-ia32-linux.tar.gz

# decompress tar file
tar -xvpf ncbi-blast-2.2.28+-ia32-linux.tar.gz

# append to the path so that BLAST programs will run from any directory
PATH=$PATH:$HOME/blast/ncbi-blast-2.2.25+/bin
export PATH

# show all available pre-formatted BLAST databases
perl update_blastdb.pl --showall

# use Perl script update_blastdb.pl to download pre-formatted BLAST databases from NCBI
perl update_blastdb.pl nr
perl update_blastdb.pl nt
```

Note: nr – non-redundant protein sequence database with entries from GenPept, Swissprot, PIR, PDF, PDB, and NCBI RefSeq; nt – nucleotide sequence database with entries from all traditional divisions of GenBank, EMBL, and DDBJ (there are redundant entries).

## 4.1 Compare NC-Liverpool mRNAs and proteins with Toxoplasma gondii proteins

```
# blastn between existing NC-Liverpool mRNA (7111) and Toxoplasma gondii ME49 (8315) mRNA
# (repeated for each chromosome)
blastn -query neospora/chrIa_mrna.seq -subject toxoplasma/chrIa_mrna.seq  -outfmt "10 evalue qseqid qlen sseqid slen bitscore score pident nident mismatches positive"  -max_target_seqs 5 -out blast_output/chrIa_blastn.txt

# blastp between existing NC-Liverpool proteins (7111) and Toxoplasma gondii ME49 (8315) proteins
blastp -query nc_proteins.fasta -subject tg_proteins.fasta  -outfmt "10 evalue qseqid qlen sseqid slen bitscore score pident nident mismatches positive"  -max_target_seqs 5  -out blastp.txt
```

Note: evalue – Expect value; qseqid – Query sequence ID; qlen – Query sequence length; sseqid – Subject sequence ID; slen – Subject sequence length; bitscore – Bit score, score – Raw score; pident – Percentage of

identical matches; nident – Number of identical matches; mismatch – Number of mismatches; positive – Number of positive-scoring matches, max_target_seqs – Maximum number of aligned sequences to keep

### *4.2 Compare predicted genes and deduced proteins with current annotation*

# blastn between predicted sequences and existing NC-Liverpool gene sequences
# (repeated for each chromosome)
blastn -query aug_chrIa.seq -subject genes_chrIa.seq -outfmt "10 evalue qseqid qlen sseqid slen bitscore score pident nident mismatch positive" -max_target_seqs 5 -out blastn_aug_chrIa.txt

Note: aug– AUGUSTUS gene predictions. The above was repeated for genes predicted by GlimmerHMM, N-SCAN, BLAT, and GMAP.

# blastp between deduced protein sequences from predicted genes and existing NC-Liverpool protein sequences
blastp -query aug_proteins.fasta -subject existing_proteins.fasta -outfmt "10 evalue qseqid qlen sseqid slen bitscore score pident nident mismatch positive" -max_target_seqs 5 -out aug_blastp.txt

Note: aug– proteins from AUGUSTUS gene predictions. The above was repeated for proteins predicted by GlimmerHMM and N-SCAN.

# blastp between existing *Neospora* and existing *Toxoplasma* proteins
blastp -query nc_proteins.fasta -subject tg_proteins.fasta -outfmt "10 evalue qseqid qlen sseqid slen bitscore score pident nident mismatch positive" -max_target_seqs 5 -out all_blastp.txt

# blastp on the NCBI nr database
blastp -query nc_proteins.fasta -db nr -outfmt "10 evalue qseqid qlen sseqid slen bitscore score pident nident mismatches positive" -max_target_seqs 100 -num_threads 8 -out nc_homologs.txt

Note: num_threads – Number of threads (CPUs) to use in the BLAST search

## 5 Gene prediction

### *5.1 AUGUSTUS 2.5.5*

How to train AUGUSTUS can be found at:
http://molecularevolution.org/molevolfiles/exercises/augustus/training.html

# setup at folder for the model parameters in ../augustus.*2.5.5*/config/species
perl new_species.pl --species=toxoplasma

# train the model using *Toxoplasma* validated genes
etraining --species=toxoplasma aug_train_data.gb

# execute AUGUSTUS for each *Neospora* chromosome to make gene predictions
augustus --species=toxoplasma chrIa.txt > chrIa.gff

# Information on the GFF format obtained from:
http://www.sanger.ac.uk/resources/software/gff/

## 5.2 GlimmerHMM 3.0.1

# download Glimmer
lwp-download ftp://ftp.cbcb.umd.edu/pub/software/glimmerhmm/GlimmerHMM-3.0.1.tar.gz

# decompress tar.gz file
gunzip GlimmerHMM-3.0.1.tar.gz
tar -xvf GlimmerHMM-3.0.1.tar

# to train GlimmerHMM: first run make in the train directory. Read readme.train from GlimmerHMM/train. The train directory is specified with the -d option when using GlimmerHMM.
Training syntax: GlimmerHMM <fasta_file> <exon_file> [optional_parameters]

# run trainGlimmerHMM for each *Toxoplasma* chromosome
perl trainGlimmerHMM seq_genes_chr_Ia.fasta glim_exons.txt

# execute glimmerhmm_linux for each *Neospora* chromosome to make gene predictions
glimmerhmm_linux chrIa.fasta -d toxoplasma -o glimmer_results.gff –g

Note: o – output file; g – print output in gff format

## 5.3 BLAT (v. 34x10)

# download executable from:
http://hgdownload.cse.ucsc.edu/admin/exe/linux.x86_64/blat/

# documentation from:
http://genome.ucsc.edu/goldenPath/help/blatSpec.html

# run blat with no arguments to get help

# information on Perl scripts obtained from
http://www.molecularevolution.org/molevolfiles/exercises/augustus/prediction.html

# download Perl scripts from: http://augustus.gobics.de/binaries/scripts/

# run blat on each *Neospora* chromosome
./blat -noHead chrIa.fasta ests_all_strains.fasta est_chrIa.psl

Note: 'ests_all_strains.fasta' contains all Neospora expressed sequence tags (ESTs) downloaded from dbEST, and 'est_chrIa.psl' is the output file

# filter the alignments
cat est_chrIa.psl | perl filterPSL.pl --best --minCover=80 > filtered_est_chrvIa.psl

Note: --best and --minCover ensures that for each query the best alignment(s) is selected and only if it covers at least 80% of the query length. An in-house Perl script is used to convert the PSL format to a FASTA format.

## *5.4 GMAP (version 2013-07-20)*

# download executable from:
http://research-pub.gene.com/gmap/

# decompress file
gunzip gmap-gsnap-2013-07-20.tar.gz
tar -xvf gmap-gsnap-2013-07-20.tar

# for gmap help
gmap --help

# build the genome database
gmap_setup -d neospora chrIa.fasta chrIb.fasta <*etc. for each Neospora chromosome*>
# run gmap
gmap --db=neospora --format=1 ests_all_strains.fasta > gmap.psl

Note: db – genome database; format = 1 – for PSL format; 'ests_all_strains.fasta' contains all *Neospora* ESTs downloaded from dbEST. An in-house Perl script is used to convert the PSL format to a FASTA format.

## *5.5 N-SCAN (version 4.0)*

# download N-SCAN from:
http://mblab.wustl.edu/software/download/

```
# extract files
gunzip Nscan_4.0_linux.tar.gz
tar -xvf Nscan_4.0_linux.tar

# see README file in installation directory for other dependencies that need to be installed and N_SCAN setup.

# run N-SCAN for each Neospora chromosome
perl Nscan_driver.pl --nomask -d nscanOutput chrIa.fa nscandriver.config
```

Note: Nscan_driver.pl runs a pipeline of programs. The parameters for the pipeline are contained in nscandriver.config. N-SCAN's output is in GTF2 format (see http://mblab.wustl.edu/GTF2.html). Parameters: d – output directory; nomask – do not mask target sequence with RepeatMasker.