Experiments for "Profiling the BLAST bioinformatics application for load balancing on high-performance computing clusters"

From Wikidb

Contents

Introduction

More experiments needed for the publication.

Location on Betsy: **/scratch/mikem/UserSupport/trinity.cheng/blast_surface**

Getting a new query file

Downloaded from any:

> http://www.ncbi.nlm.nih.gov/sra/SRP102422
> https://www.ncbi.nlm.nih.gov/sra/SRP102422
> https://trace.ncbi.nlm.nih.gov/Traces/index.html?view=run_browser&acc=SRR5713923&display=download

https://trace.ncbi.nlm.nih.gov/Traces/sra-reads-be/fasta?acc=SRR5713923

Convert:

> See FASTQ file to FASTA file (https://scl-wiki.fda.gov/wiki/index.php/Technical_questions#FASTQ_file_to_FASTA_file)
> https://bioinformaticsworkbook.org/dataWrangling/fastaq-manipulations/converting-fastq-format-to-fasta.html#gsc.tab=0

" Using SED

> sed can be used to selectively print the desired lines from a file, so if you print the first and 2rd line of every 4 lines, you get the sequence header and sequence needed for fasta format.
>
> time sed -n '1~4s/^@/>/p;2~4p' SRR5713923.fastq > SRR5713923_2.fasta

Experiments

batch_run_v2.sh (also found at **/scratch/mikem/UserSupport/trinity.cheng/blast_surface/batch_run_v2.sh**) is used to batch run the experiments. batch_run_v2.sh in turn runs sge_extractblast_v3.sh via *qsub* SGE command.

“threads” is added to the DESC field in the job submit line to avoid overwriting results when running the experiments in the batch mode:

```
qsub -N "blast_array_${thr}_${t}" -l h_vmem=${MEM}G -pe thread $thr -l dell01=true -l gpus=0 sge_extractblast_v3.sh type1test"$t"_"$thr"
```

So, the DESC field is combination of:

- Test type
- Test number
- Number of threads used
- Number of db sequences and
- Number of query sequences

This way BLAST will generate unique files for every qsub to avoid overwriting when run in batch mode.

Running with the new query

If we want to run with **the new query** before the batch run we need to (in extract_blast2.sh file located on **/scratch/mikem/UserSupport/trinity.cheng/blast_surface/extract_blast2.sh**):

a. Comment out lines: 14, 20 and 48
b. Uncomment lines: 13, 19 and 47.

See also Lines to modify in extract_blast2.sh.
Then you may adjust the below parameters in batch_run_v2.sh:

```
num_threads=(1 2 4 8)
NUM_REPEAT=3
```

And then run the below command from **/scratch/mikem/UserSupport/trinity.cheng/blast_surface** prompt:

```
bash  batch_run_v2.sh
```

Running with the old query

If we want to run with **the old query** before the batch run we need to (in extract_blast2.sh file located on **/scratch/mikem/UserSupport/trinity.cheng/blast_surface/extract_blast2.sh**):

a. Uncomment lines: 14, 20 and 48
b. Comment out lines: 13, 19 and 47

See also Lines to modify in extract_blast2.sh.
Then you may adjust the below parameters in batch_run_v2.sh:

```
num_threads=(1 2 4 8)
NUM_REPEAT=3
```

Then run the below command from **/scratch/mikem/UserSupport/trinity.cheng/blast_surface prompt**:

```
bash  batch_run_v2.sh
```

Lines to modify in extract_blast2.sh

Lines 13, 14:

```
13 # TIME_SUMMARY_DIR=${BASE_DIR}/time_summary                                  # new query result dir
14 TIME_SUMMARY_DIR=${BASE_DIR}/time_summary_old_query                          # old query result dir
```

Lines 19,20:

```
19 # QUERY_FILE=${BASE_DIR}/orig_query_split/SRR5713923                        # new query file
20 QUERY_FILE=${BASE_DIR}/orig_old_query_split/orig_query                      # old query file
```

Lines 47, 48:

```
47 # SPLIT_QUERY=${BASE_DIR}/orig_query_split/orig_query"$NREC_QUERY"          # new query split
48 SPLIT_QUERY=${BASE_DIR}/orig_old_query_split/orig_query"$NREC_QUERY"        # old query split
```

sge_extractblast_v3.sh

Author: Trinity Cheng, 2021 Summer Intern.

```
#$ -P CDRHID0014
#$ -cwd
#$ -l h_rt=48:00:00
#                                               $ -l h_vmem=2.5G
#$ -S /bin/sh
#$ -j y
#$ -o sge_results
#                               $ -N blast_array
#$ -t 1-90
#                               $ -pe thread 8

# This script runs all MxN combinations from m_vals and n_vals in an array job
# usage: qsub -l <nodename> sge_extractblast_v3.sh <description>
# type 1 = -q '*@@betsy_original'
# type 2 = -l bigbox
# type 3 = -l sm01
# type 4 = -l sm02
# type 5 = -l hpe01=true -l gpus=0
# type 6 = -l dell01=true -l gpus=0

# DESC is a description of the current job/experiment
DESC=$1
DBFILE=m_vals.txt
QUERYFILE=n_vals.txt
```

```
# MxN array, M is row/database, N is column/query

DBNUM=$(cat $DBFILE | wc -l)
QUERYNUM=$(cat $QUERYFILE | wc -l)

ROW=$((((((SGE_TASK_ID-1))/$QUERYNUM))+1))
COL=$((((((SGE_TASK_ID-1))%$QUERYNUM))+1))

NREC_DB=$(head -n $ROW $DBFILE | tail -n 1)
NREC_QUERY=$(head -n $COL $QUERYFILE | tail -n 1)

# run extract_blast with $NREC_DB database fragments and $NREC_QUERY query fragments
./extract_blast2.sh $NREC_DB $NREC_QUERY $DESC $NSLOTS
```

extract_blast2.sh

Author: Trinity Cheng, 2021 Summer Intern.

```
#!/bin/bash

# updated 7/21/2021

# input file: fasta format, delimiter '>'
# retrieve the first M records from 2.5 GB database, N records from query, put BLAST results in filenameMxN
# usage: extract_blast2.sh <M> <N> <filename>

# BASE_DIR=/scratch/trinity.cheng/blast_surface

BASE_DIR=/scratch/mikem/UserSupport/trinity.cheng/blast_surface

TIME_SUMMARY_DIR=${BASE_DIR}/time_summary                               # new query result dir
# TIME_SUMMARY_DIR=${BASE_DIR}/time_summary_old_query                   # old query result dir

mkdir -p ${BASE_DIR}/sge_results
mkdir -p ${TIME_SUMMARY_DIR}

QUERY_FILE=${BASE_DIR}/orig_query_split/SRR5713923                              # new query file
# QUERY_FILE=${BASE_DIR}/orig_old_query_split/orig_query                        # old query file


BASE_DB_DIR=/projects/mikem/UserSupport/ncbi/nt_2020
DB_NAME=nt
DB_FILE=${BASE_DB_DIR}/nt

# BASE_DB_DIR=/scratch/trinity.cheng/blast_surface/orig_db_split
# DB_NAME=orig_db
# DB_FILE=${BASE_DB_DIR}/${DB_NAME}

BLAST=/projects/mikem/applications/centos7/blast2.12.0_fda/ncbi-blast-2.12.0+-src/c++/ReleaseMT/bin/blastn
MAKEBLASTDB=/projects/mikem/applications/centos7/blast2.12.0_fda/ncbi-blast-2.12.0+-src/c++/ReleaseMT/bin/makeblastdb


echo $QUERY_FILE
echo $DB_FILE

NREC_DB=$1
echo $NREC_DB
# location of extracted db
SPLIT_DB=${BASE_DB_DIR}/orig_db_split/"${DB_NAME}""${NREC_DB}"
echo $SPLIT_DB

NREC_QUERY=$2
echo $NREC_QUERY
# location of extracted query
SPLIT_QUERY=${BASE_DIR}/orig_query_split/orig_query"$NREC_QUERY"                # new query split
# SPLIT_QUERY=${BASE_DIR}/orig_old_query_split/orig_query"$NREC_QUERY"          # old query split
```

```
echo $SPLIT_QUERY

DESC=$3

SLOTS=$4

# If query does not already exist, extract fragments and create split query.
if [ ! -f $SPLIT_QUERY ];
then
        awk -v N=$NREC_QUERY 'BEGIN {N_start=1; RS=">"}; {if (NR>N_start && NR<=N_start+N) {print ">" $0}}' $QUERY_FILE > $SPLIT_QUERY
fi

# If database does not already exist, extract fragments and create split database. Then, run makeblastdb to index.
if [ ! -f $SPLIT_DB ];
then
        awk -v N=$NREC_DB 'BEGIN {N_start=1; RS=">"}; {if (NR>N_start && NR<=N_start+N) {print ">" $0}}' $DB_FILE > $SPLIT_DB
        $MAKEBLASTDB -in $SPLIT_DB -dbtype nucl
fi

BASE_OUT=/scratch/mikem/UserSupport/trinity.cheng/blast_surface
# run BLAST and time, put results in sge_reuslts/"$DESC"_"$NREC_DB"X"$NREC_QUERY"
# output the time in seconds into time_summary.txt file

if [ $SLOTS == 1 ]
then
   NUM_THREADS=""
else
   NUM_THREADS="-num_threads $SLOTS"
fi

echo "CMD: time $BLAST $NUM_THREADS -dbseqnum $NREC_DB -query $SPLIT_QUERY -db $SPLIT_DB"
TIMEFORMAT="%E %U %S";
(time $BLAST $NUM_THREADS -dbseqnum $NREC_DB -query $SPLIT_QUERY -db $SPLIT_DB) &> ${BASE_OUT}/sge_results/"$DESC"_"$NREC_DB"x"$NREC_QUERY"
sleep 1
TIME=$(tail -n 1 ${BASE_OUT}/sge_results/"$DESC"_"$NREC_DB"x"$NREC_QUERY")

# output line: "M N time" for every M and N configuration into time_summary file.
## echo "$NREC_DB" "$NREC_QUERY" $TIME >> ${BASE_OUT}/time_summary/new_times_for_modeling/time_summary_"$DESC".txt
echo "$NREC_DB" "$NREC_QUERY" $TIME >> ${TIME_SUMMARY_DIR}/time_summary_"$DESC"_"$SLOTS"cpus.txt


# QUERY_FILE=${BASE_DIR}/orig_query_split/orig_query
# DB_FILE=${BASE_DIR}/orig_db_split/orig_db
# DB_NAME=SRR5713923
# DB_FILE=${BASE_DIR}/orig_db_split/${DB_NAME}
```

batch_run_v2.sh

```
#!/bin/bash

# How to run:
# Specify num_threads and NUM_REPEAT below
# And run:
# bash batch_run.sh

D=`date +"%FT%T"`
OUT="batch_run_log_"${D//:}".log"

num_threads=(1 2 4 8)
NUM_REPEAT=3

for thr in ${num_threads[@]}; do
    echo "" 2>&1 | tee >> ${OUT}
    echo "Submit jobs with threads: $thr" 2>&1 | tee >> ${OUT}
        MEM=$((22/$thr))
        for (( t=1; t<=${NUM_REPEAT}; t++ ))
        do
```

```
                echo "" 2>&1 | tee >> ${OUT}
                echo "Threads: ${thr}, Test: $t" 2>&1 | tee >> ${OUT}

                CMD="qsub -N "blast_array_${thr}_${t}" -l h_vmem=${MEM}G -pe thread $thr -l dell01=true -l gpus=0 sge_extractblast_v3.sh type1test"$t"_"$thr""
                echo "$CMD" 2>&1 | tee >> ${OUT}
                $CMD
                sleep 1

                CMD="qsub -N "blast_array_"${thr}"_"${t}"" -l h_vmem=${MEM}G -pe thread "${thr}" -q '*@@betsy_original' sge_extractblast_v3.sh type2test"$t"_"$thr""
                echo "$CMD" 2>&1 | tee >> ${OUT}
                qsub -N "blast_array_"${thr}"_"${t}"" -l h_vmem=${MEM}G -pe thread "${thr}" -q '*@@betsy_original' sge_extractblast_v3.sh type2test"$t"_"$thr"
                sleep 1

                CMD="qsub -N "blast_array_${thr}_${t}" -l h_vmem=${MEM}G -pe thread $thr -l bigbox sge_extractblast_v3.sh type3test"$t"_"$thr""
                echo "$CMD" 2>&1 | tee >> ${OUT}
                $CMD
                sleep 1

                CMD="qsub -N "blast_array_${thr}_${t}" -l h_vmem=${MEM}G -pe thread $thr -l sm01 sge_extractblast_v3.sh type4test"$t"_"$thr""
                echo "$CMD" 2>&1 | tee >> ${OUT}
                $CMD
                sleep 1

                CMD="qsub -N "blast_array_${thr}_${t}" -l h_vmem=${MEM}G -pe thread $thr -l sm02 sge_extractblast_v3.sh type5test"$t"_"$thr""
                echo "$CMD" 2>&1 | tee >> ${OUT}
                $CMD
                sleep 1

                CMD="qsub -N "blast_array_${thr}_${t}" -l h_vmem=${MEM}G -pe thread $thr -l hpe01=true -l gpus=0 sge_extractblast_v3.sh type6test"$t"_"$thr""
                echo "$CMD" 2>&1 | tee >> ${OUT}
                $CMD
                sleep 1
        done
done
```

failed_rows.sh

```
#!/bin/bash

# Specify DIR and OUTPUT in lines 5 and 6 below.
# Run as:
# bash failed_rows.sh

DIR=/scratch/mikem/UserSupport/trinity.cheng/blast_surface/time_summary
OUTPUT=failed_rows.txt
echo `date` > ${OUTPUT}
echo "Files on directory: $DIR" >> ${OUTPUT}

echo >> ${OUTPUT}
for file in ${DIR}/*; do
        RES=`awk ' NF==2 {print NR,$0} '  $file`
        if [ ! -z "$RES" ]
        then
                echo "Found defetive line(s) in ${file##*/}" >> ${OUTPUT}
                awk ' NF==2 {print NR,$0} '  $file >> ${OUTPUT}
                echo >> ${OUTPUT}
        fi
done

echo "See ${OUTPUT} for results."
```

Retrieved from "https://scl-wiki.fda.gov/wiki/index.php?title=Experiments_for_%22Profiling_the_BLAST_bioinformatics_application_for_load_balancing_on_high-performance_computing_clusters%22&oldid=54682"

- This page was last modified on 12 August 2022, at 20:57.