

2η Εργασία

Ικανοποιησιμότητα προτάσεων στην προτασιακή λογική

ics23088

Ασκηση Α

Επιλογή παραμέτρων για το πρόβλημα SAT

$N = 10$

$M = 10, 20, \dots, 200$

$K = 4$

Ο αριθμός επαναλήψεων για κάθε M ορίσθηκε σε $T=10$

Με βάση το δοσμένο αρχείο `bcsp_generate.c` έγινε η παρακάτω αλλαγή

```
#define N 10
#define M 200
#define K 4
```

Για να παράξουμε τα input αρχεία προβλημάτων με αυτόματο τρόπο, χρησιμοποιήθηκε το παρακάτω bash script που το ονομάσαμε

`build_and_run_all_M200_10TIMES.sh`

```
#!/bin/bash

# Loop from 200 to 10, decreasing by 10 each time
for ((i=200; i>=10; i-=10)); do
    ./modify.sh bcsp_generate.c $i
    echo "Compiling bcsp_generate$i.c"
    gcc -o bcsp_generate$i bcsp_generate$i.c
    echo "Generating M$i"_1.._10".txt"
    ./bcsp_generate$i M$i 1 10

    FILE_PART1="M$i"
    for ((t=1; t<=10; t+=1)); do
        echo running hill for "$FILE_PART1""_"$t".txt"
        ./bscp hill "$FILE_PART1""_"$t".txt" "$FILE_PART1""_"$t".solution"
        echo "\n"
    done
done
```

Το script αρχικά θα παράξει 20 διαφορετικά c αρχεία με διαφορετικό $M=10, 20, \dots, 200$ για το καθένα

```
bscp_generate200.c
bscp_generate190.c
...
bscp_generate10.c
```

Μετά θα κάνει gcc compile για τα παράπανω c αρχεία και θα παράξει τα 20 εκτελέσιμα προγράμματα :

```
bscp_generate200
bscp_generate190
```

```
...
bscp_generate10
```

Μετά τρέχει και τα 20 παραπάνω προγράμματα και θα παράξει τα παρακάτω αρχεία προβλημάτων που θα γίνουν input στο πρόγραμμα

bscp

```
M200_1.txt .. M200_10.txt
M190_1.txt .. M190_10.txt
...
M10_1.txt .. M10_10.txt
```

Αμέσως μετά τρέχει το `bscp` πρόγραμμα ως εξής για το καθένα αρχείο προβλημάτων

```
./bscp hill M200_1.txt M200_1.solution
./bscp hill M200_2.txt M200_2.solution
...
./bscp hill M200_10.txt M200_10.solution
...
./bscp hill M10_1.txt M10_1.solution
./bscp hill M10_2.txt M10_2.solution
...
./bscp hill M10_10.txt M10_10.solution
```

Αν στις 10 επαναλήψεις για $M=200$ έχουμε X αρχεία `M200_*.solution` τότε αυτό σημαίνει ότι έχουμε $X/10$ λύσεις του SAT προβλήματος M/N

Για να υπολογίσουμε το ποσοστό των επιλύσιμων SAT προβλημάτων ως συνάρτηση της τιμής του λόγου M/N , χρησιμοποιούμε το παρακάτω bash script

```
#!/bin/bash

TIMES=10 # No spaces around '='

echo "M/N , P(solutions)"

# Loop from 10 to 200, increasing by 10 each time
for ((i=10; i<=200; i+=10)); do
    FILE_PART1="M$i"

    # Count the number of matching files
    FILE_COUNT=$(ls -al ${FILE_PART1}*.solution 2>/dev/null | wc -l)

    # Convert i/10 to an integer
    M_N_RATIO=$((i / 10)) # Integer division in Bash

    # Compute the percentage using awk
    PERCENTAGE=$(awk "BEGIN {printf \"%.1f\\\", $FILE_COUNT / $TIMES}")

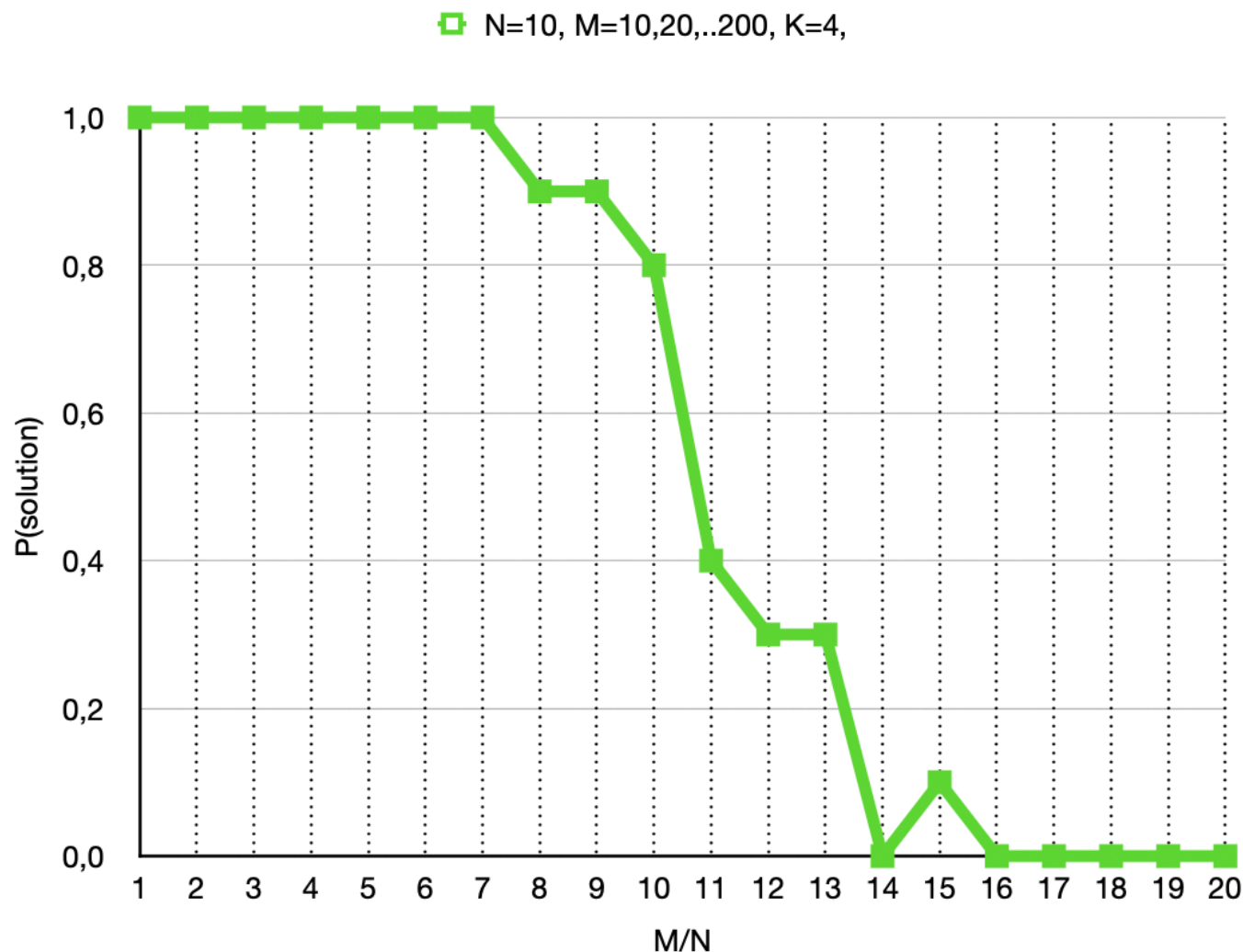
    echo "${M_N_RATIO} , ${PERCENTAGE}"
done
```

Το αποτέλεσμα δίνει το ποσοστό των επιλύσιμων προβλημάτων ως συνάρτηση της τιμής του λόγου M/N :

```
M/N , P(solutions)
1 , 1.0
2 , 1.0
3 , 1.0
```

4	, 1.0
5	, 1.0
6	, 1.0
7	, 1.0
8	, 0.9
9	, 0.9
10	, 0.8
11	, 0.4
12	, 0.3
13	, 0.3
14	, 0.0
15	, 0.1
16	, 0.0
17	, 0.0
18	, 0.0
19	, 0.0
20	, 0.0

Η γραφική παράσταση του παραπάνω table είναι αποτελεί το
διάγραμμα του ποσοστού των επιλύσιμων προβλημάτων, ως συνάρτηση της τιμής του λόγου M/N είναι το παρακάτω



Αρα ως κρίσιμη τιμή του λόγου M/N μπορεί να θεωρηθεί η τιμή 16.

Με βάση τις λύσεις από τα αρχεία `*.solution` που προέκυψαν και με το παρακάτω bash script `calc.timings.bash` υπολογίζουμε το χρόνο σε μsecs που χρειάστηκε να τρέξουμε μια φορά τον αλγόριθμο `depth` και 5 φορές τον αλγόριθμο `hill` για τον οποίο υπολογίσαμε τον μέσο όρο (average)

```
#!/bin/bash
```

```
SOLUTION_DIR="./" # Change if needed
```

```
echo "M/N, time depth ( $\mu$ s), avg time hill ( $\mu$ s)"
```

```
# Temporary files to store grouped values
```

```
DEPTH_FILE=$(mktemp)
```

```
HILL_FILE=$(mktemp)
```

```
FINAL_OUTPUT=$(mktemp)
```

```
# Find and sort files correctly using natural numerical sorting
```

```
find "$SOLUTION_DIR" -type f -name "M*.*.solution" | sort -V | while read -r SOLUTION_FILE; do
```

```
    FILE_NAME=$(basename "$SOLUTION_FILE")
```

```
    M_VALUE=$(echo "$FILE_NAME" | awk -F'[_]' '{print substr($1,2)}')
```

```
    X_VALUE=$(echo "$FILE_NAME" | awk -F'[_]' '{print $2}' | cut -d'.' -f1)
```

```
    INPUT_FILE="M${M_VALUE}_${X_VALUE}.txt"
```

```
    # Get depth time
```

```
    TIME_SPENT1=$(./bscp depth "$INPUT_FILE" "$SOLUTION_FILE" 2>/dev/null | grep "Time spent:" | awk  
'{print $3}')
```

```
    TIME_SPENT1_US=$(awk "BEGIN {printf \"%.0f\\", $TIME_SPENT1 * 1000000}")
```

```
    # Get hill time (average over 5 runs)
```

```
    SUM_TIME=0.0
```

```
    ITERATIONS=5
```

```
    for ((j=1; j<=ITERATIONS; j+=1)); do
```

```
        CURRENT_TIME=$(./bscp hill "$INPUT_FILE" "$SOLUTION_FILE" 2>/dev/null | grep "Time spent:" | awk  
'{print $3}')
```

```
        if [[ -n "$CURRENT_TIME" ]]; then
```

```
            SUM_TIME=$(awk "BEGIN {print $SUM_TIME + $CURRENT_TIME}")
```

```
        fi
```

```
    done
```

```
    TIME_SPENT2=$(awk "BEGIN {printf \"%.6f\\", $SUM_TIME / $ITERATIONS}")
```

```
    TIME_SPENT2_US=$(awk "BEGIN {printf \"%.0f\\", $TIME_SPENT2 * 1000000}")
```

```
    # Store data in temp files
```

```
    echo "$M_VALUE $TIME_SPENT1_US" >> "$DEPTH_FILE"
```

```
    echo "$M_VALUE $TIME_SPENT2_US" >> "$HILL_FILE"
```

```
done
```

```
# Compute and store averages in final output
```

```
for M in $(awk '{print $1}' "$DEPTH_FILE" | sort -nu); do
```

```
    AVG_DEPTH=$(awk -v m="$M" '$1 == m {sum+=$2; count++} END {if (count > 0) printf "%.0f", sum/count}'  
"$DEPTH_FILE")
```

```
    AVG_HILL=$(awk -v m="$M" '$1 == m {sum+=$2; count++} END {if (count > 0) printf "%.0f", sum/count}'  
"$HILL_FILE")
```

```
    echo "$M, $AVG_DEPTH, $AVG_HILL" >> "$FINAL_OUTPUT"
```

```
done
```

```
# Sort the final output numerically and print
```

```
sort -n "$FINAL_OUTPUT"
```

```
# Clean up temp files
```

```
rm "$DEPTH_FILE" "$HILL_FILE" "$FINAL_OUTPUT"
```

Το αποτέλεσμα είναι

```
./calc_timings.bash
```

M/N, avg time depth (μ s), avg time hill (μ s)

10, 14, 5
20, 18, 13
30, 26, 25
40, 51, 42
50, 80, 76
60, 88, 214
70, 139, 118
80, 178, 209
90, 212, 409
100, 373, 720
110, 276, 433
120, 507, 805
130, 520, 715
150, 695, 268

Από όπου προκύπτει το διάγραμμα του μέσου χρόνου που απαιτείται για την επίλυση των (επιλύσιμων) προβλημάτων, για κάθε έναν από τους δύο αλγορίθμους, ως συνάρτηση της τιμής του λόγου M/N.

