

Team Project Proposal Assignment

Miguel Aguilar

Jose Chavez

Andrew Gilbert

Chris Oppewall

Project title

Duel of the Data Structures

Team information

Team member's name	Discipline	STAT 312 or STAT 542 student?
Project Contact: Andrew Gilbert	Comp. Sci.	312
Miguel Aguilar	Comp. Sci.	312
Jose Chavez	Comp. Sci.	312
Chris Oppewall	Comp. Sci.	312

Progress

Completion	Activity	Completion Date	Comments
100%	Design test framework	???	No official design phase
100%	Build test framework	2 Nov 2013	Written in Java, shown to Dr Smith.
100%	Pilot study	7 Nov 2013	Date is that of data generation, according to Git repository.
100%	Analyse pilot study	7 Nov 2013	
	Write team project update	7 Nov 2013	This document.
100%	Find implementations of all data structures	8 Nov 2013	Done ahead of time.
0%	Run tests	15 Nov 2013	
0%	Analyse full study	22 Nov 2013	
0%	Create presentation	6 Dec 2013	

Reproduced above is our team's table of activities, with updated completion dates, comments, and a percent completion list. In general, the code has been working well. Our testing framework has already produced

data spanning all our desired factors, and is capable of producing an arbitrary number of replicates with full randomization. Writing the code took slightly longer than expected, with completion being approximately the 2nd of November, rather than the 1st. The data for the pilot study has been gathered and analyzed: the results will be reported later in this paper.

Pilot study and power analysis

We have completed a pilot study, consisting of two of our experimental factors—the data structure type and the operation on the structure—varied over two levels each in a full-factorial experiment. The resulting data was gathered by our program and put into a CSV with minimal human interaction.

All the factor levels involved in our experiment can be controlled exactly, as the computer is running all the tests. In the case of our pilot study in particular, the only factors tested were categorical and were fully under the control of our software (assuming the software wasn't doing anything really strange we don't know about!) Each of the results in our raw data is the time at nanosecond precision to execute the operation on a group of random items with a clean instance of the data structure. Each test run uses a fresh instance of the data structure, and a fresh block of random data which is generated on the spot before timing begins.

From the pilot study data, we calculated a standard deviation of 8073696, which was used in our power analysis below. The dotplot (figure 1) showed fairly tight clustering of data points with a few outliers. The ANOVA results (see page 5) indicate that there is a definite interaction between operation and data structure type. That's something worth exploring: is that expected according to computational complexity? If not, what's causing it?

Our pilot study basically sustained our current expectations, with our system working perfectly. We see no reason not to simply run the next step of our project and begin analysis.

Our power analysis (see figure 2 and the Minitab output in figure) indicates that we should need 224 repetitions to achieve a resolution of 1ms with a target power of 0.95. A run of our program to achieve that many repetitions with all the factors and levels required 67.19994490468333 minutes,¹ and produced a mere 229 kB of data. Given the automated nature of the testing, this is not unreasonable, although it may be expedient to run the 140 repetitions required for 80% power instead if we decide there is no reason for the higher power level.

The effect sizes were chosen based on a quick summary of mean deltas from the Tukey output (page 5). We looked at each of the deltas in millions of nanoseconds (milliseconds), and noticed that the lowest delta was roughly 5 million. Realizing that that represents a delta of 5 milliseconds, and that Java may or may not be measuring to nanoseconds anyway (in fact, it probably isn't), we decided to generate power analyses for effect sizes of 1 ms, 5 ms, and 10 ms. 1 ms is 1/5th of the smallest observed delta, so it should be more than good enough to find most changes. 5 ms is right near the smallest observed delta, which was between the less-meaningful means of BST overall and Trie overall. Since there's a known interaction, those means could probably be ignored.

¹Note: This statistic, like 68.738247% of other statistics, was carried to a meaningless degree of accuracy

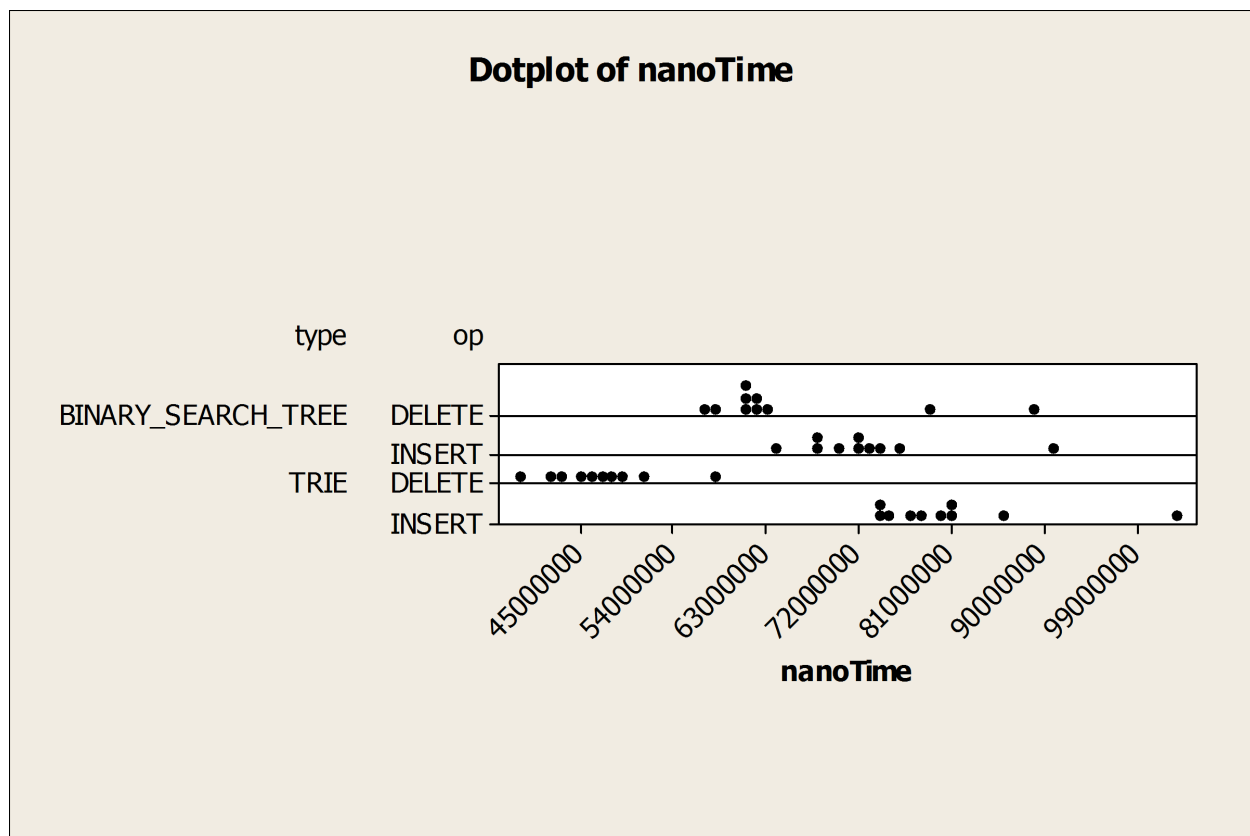


Figure 1: Dotplot of time taken (in nanoseconds) vs. data structure type and operation under test

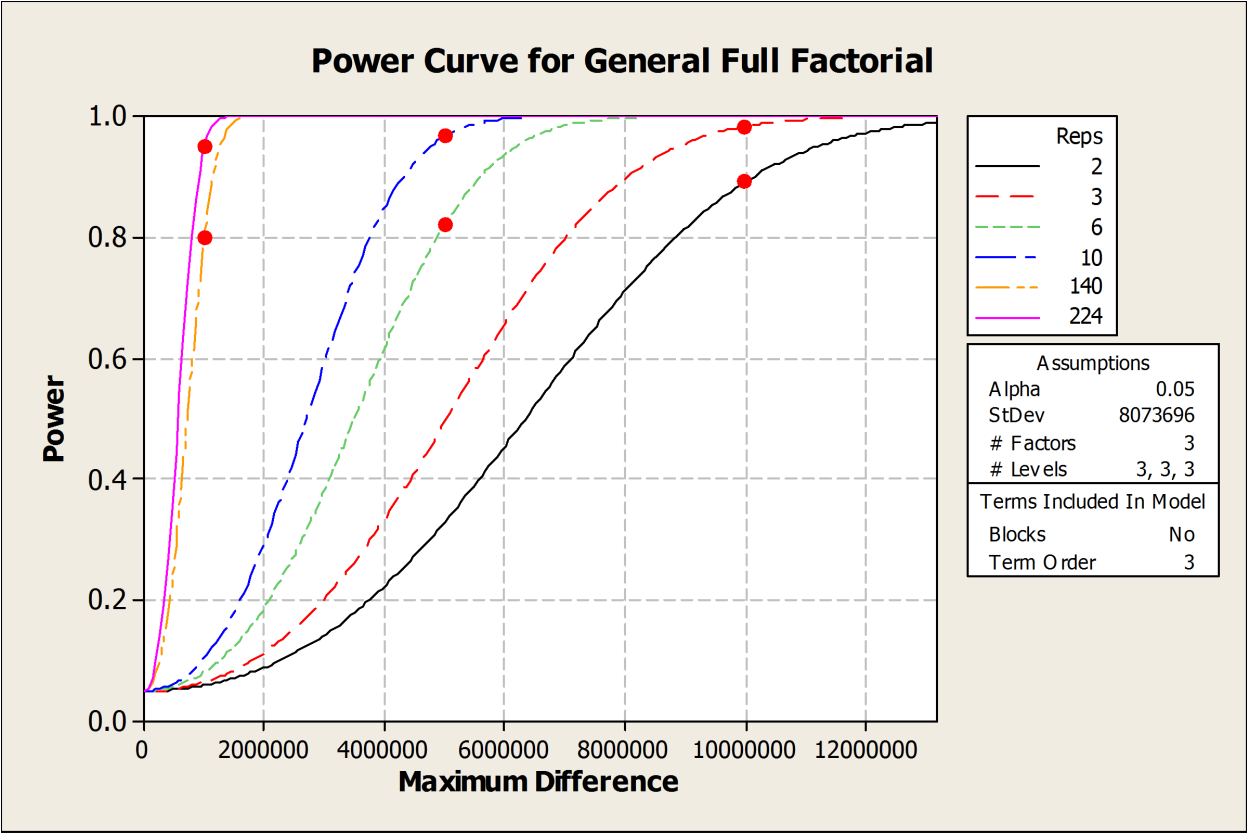


Figure 2: Power Analysis Results

Grouping Information Using Tukey Method and 95.0% Confidence

op	N	Mean	Grouping
INSERT	20	76963632	A
DELETE	20	56011496	B

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals

Response Variable nanoTime

All Pairwise Comparisons among Levels of op

op = DELETE subtracted from:

op	Lower	Center	Upper	
INSERT	15774155	20952136	26130117	(-----*-----)

18000000 21000000 24000000

Grouping Information Using Tukey Method and 95.0% Confidence

type	op	N	Mean	Grouping
TRIE	INSERT	10	81073760	A
BINARY_SEARCH_TREE	INSERT	10	72853504	A B
BINARY_SEARCH_TREE	DELETE	10	65319945	B
TRIE	DELETE	10	46703047	C

Means that do not share a letter are significantly different.

Tukey 95.0% Simultaneous Confidence Intervals

Response Variable nanoTime

All Pairwise Comparisons among Levels of type*op

type = BINARY_SEARCH_TREE

op = DELETE subtracted from:

type	op	Lower	Center	Upper
BINARY_SEARCH_TREE	INSERT	-2193854	7533559	17260973
TRIE	DELETE	-28344311	-18616897	-8889484
TRIE	INSERT	6026402	15753816	25481229

type	op	
BINARY_SEARCH_TREE	INSERT	(---*---)
TRIE	DELETE	(---*---)
TRIE	INSERT	(---*---)

-2.5E+07 0 25000000 50000000

type = BINARY_SEARCH_TREE

op = INSERT subtracted from:

type	op	Lower	Center	Upper
TRIE	DELETE	-35877870	-26150457	-16423043
TRIE	INSERT	-1507157	8220257	17947670

type	op	
TRIE	DELETE	(---*---)
TRIE	INSERT	(---*---)

-----+-----+-----+-----+--
-2.5E+07 0 25000000 50000000

type = TRIE
op = DELETE subtracted from:

type	op	Lower	Center	Upper
TRIE	INSERT	24643300	34370713	44098126

type	op	
TRIE	INSERT	(---*---)

-----+-----+-----+-----+--
-2.5E+07 0 25000000 50000000

Power analysis

Power and Sample Size

General Full Factorial Design

Alpha = 0.05 Assumed standard deviation = 8073696

Factors: 3 Number of levels: 3, 3, 3

Include terms in the model up through order: 3
Not including blocks in model.

Maximum Difference	Reps	Total Runs	Target Power	Actual Power
1000000	140	3780	0.80	0.800978
1000000	224	6048	0.95	0.950165
5000000	6	162	0.80	0.820545
5000000	10	270	0.95	0.967069
10000000	2	54	0.80	0.891741
10000000	3	81	0.95	0.983354