



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

Κατανεμημένα Συστήματα
Εξαμηνιαία Εργασία - BlockChat

Αναστάσιος Στέφανος Αναγνώστου
03119051

13 Μαρτίου 2024

Περιεχόμενα

I	Σχεδιασμός Συστήματος	3
1	Δεν είμαι σίγουρος	3
II	Πειράματα	4
2	Απόδοση του συστήματος	4
2.1	Χρονοβόρα τμήματα του κώδικα	4
2.2	Συναρτήσεις του συστήματος	6
2.3	Ρυθμαπόδοση και Block time	9
3	Κλιμακωσιμότητα του συστήματος	10
3.1	Χρονοβόρα τμήματα του κώδικα	10
3.2	Συναρτήσεις του συστήματος	12
3.3	Block Time	16
4	Δικαιοσύνη	17

Μέρος Ι

Σχεδιασμός Συστήματος

1 Δεν είμαι σίγουρος

Μέρος II

Πειράματα

Ανά πείραμα αξιολογούνται, αφενός τα πιο χρονοβόρα κομμάτια του κώδικα, όπως υποδεικνύει το profiling καθενός κόμβου κατά την εκτέλεση του πειράματος, αφετέρου οι συναρτήσεις mint, validateTransaction και processTXs, οι οποίες συνιστούν την λογική λειτουργίας των κόμβων του συστήματος. Επίσης, εκτιμάται η ρυθμαπόδοση του συστήματος και το μέσο block time.

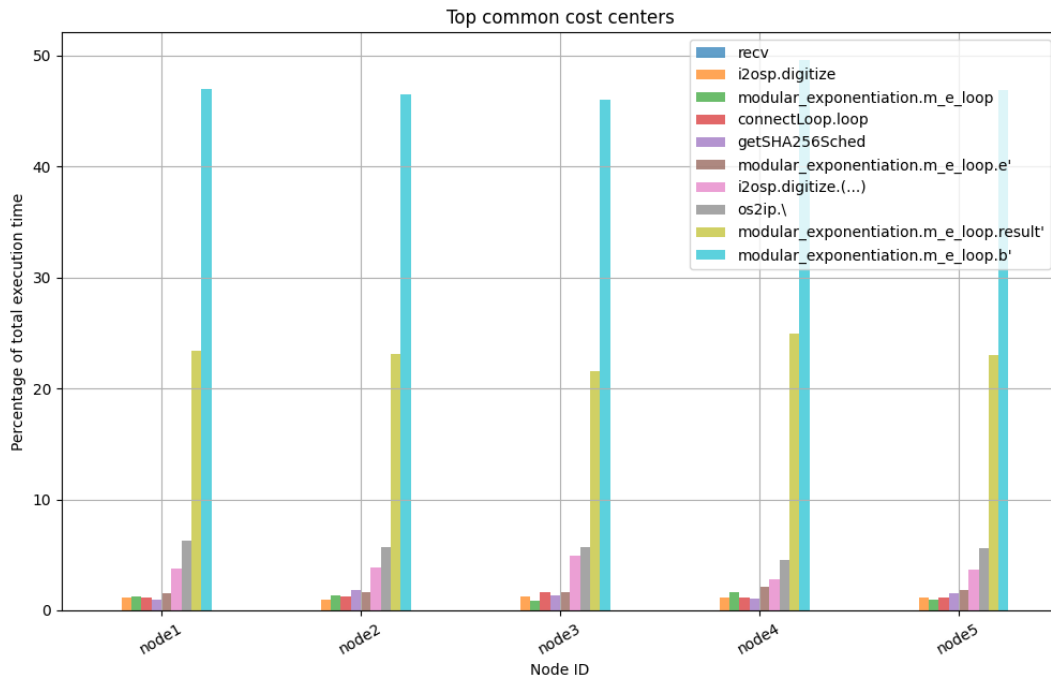
2 Απόδοση του συστήματος

Σημειώνεται ότι το ποσοστό του χρόνου εκτέλεσης των σημείων που υποδεικνύει το profiling δεν είναι κληρονομημένο, δηλαδή δεν εμπεριέχονται στο ποσοστό οι χρόνοι εκτέλεσης των συναρτήσεων που καλούνται από τις συναρτήσεις που εμφανίζονται στο profiling.

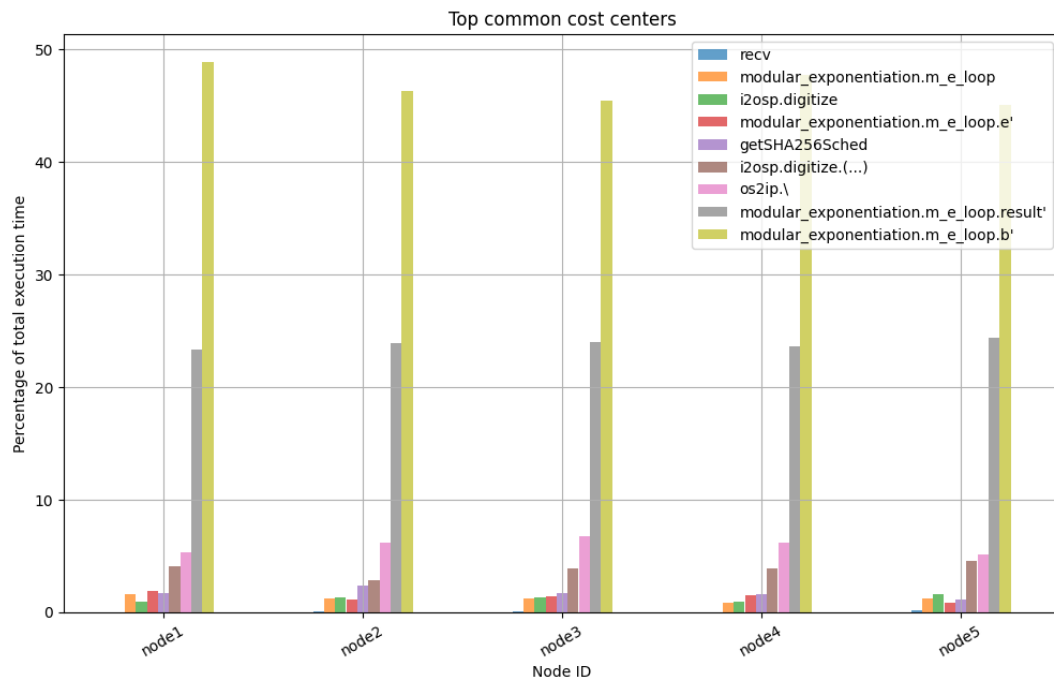
2.1 Χρονοβόρα τμήματα του κώδικα

Στο πείραμα για την αξιολόγηση της ρυθμαπόδοσης του συστήματος, στήνεται ένα δίκτυο 5 κόμβων, καθένας εκ των οποίων εκτελεί 1 staking, με stake 10 BCC συναλλαγή και 50 συναλλαγές (συγκεκριμένα αποστολές μηνυμάτων) προς τους άλλους κόμβους. Η ταχύτητα αποστολής συναλλαγών είναι ίδια μεταξύ των κόμβων, ίση με $10 \frac{txs}{s}$ και παραμένει σταθερή μεταξύ όλων των πειραμάτων.

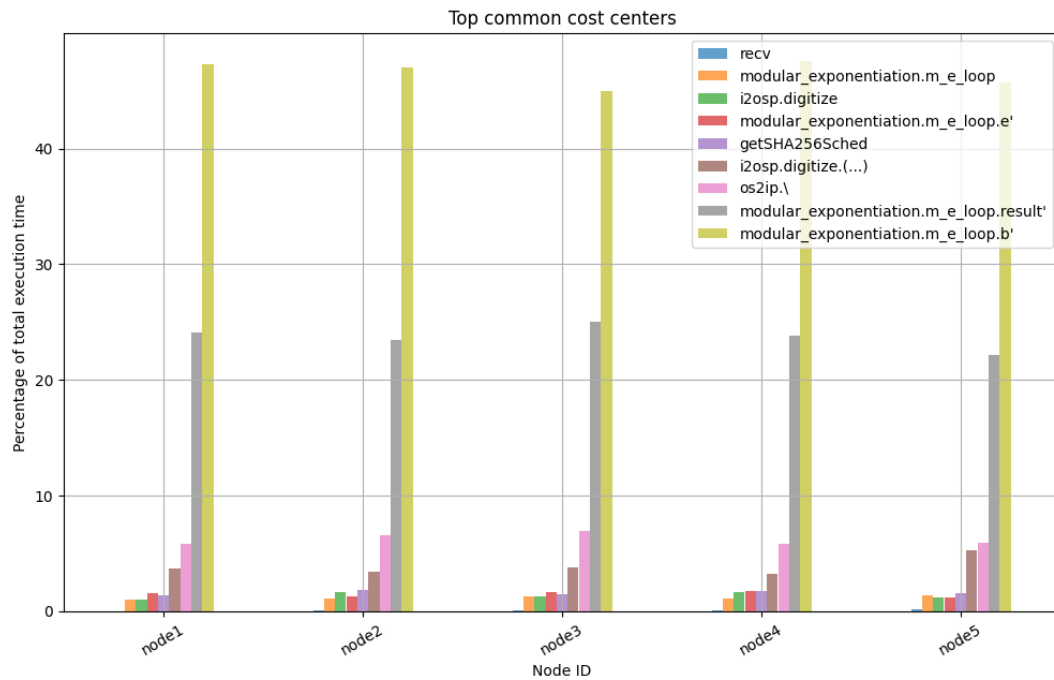
Το πρώτο πράγμα που φαίνεται στο σχήμα 1 είναι ότι το μακράν πιο χρονοβόρο μέρος του κώδικα είναι η συνάρτηση modular_exponentiation που χρησιμοποιείται γενικά για την κρυπτογράφηση / αποκρυπτογράφηση και υπογραφή / επαλήθευση μηνυμάτων. Συγκεκριμένα, φαίνεται να λαμβάνει περίπου το 50% του συνολικού χρόνου εκτέλεσης του προγράμματος.



(α') capacity=5



(β') capacity=10

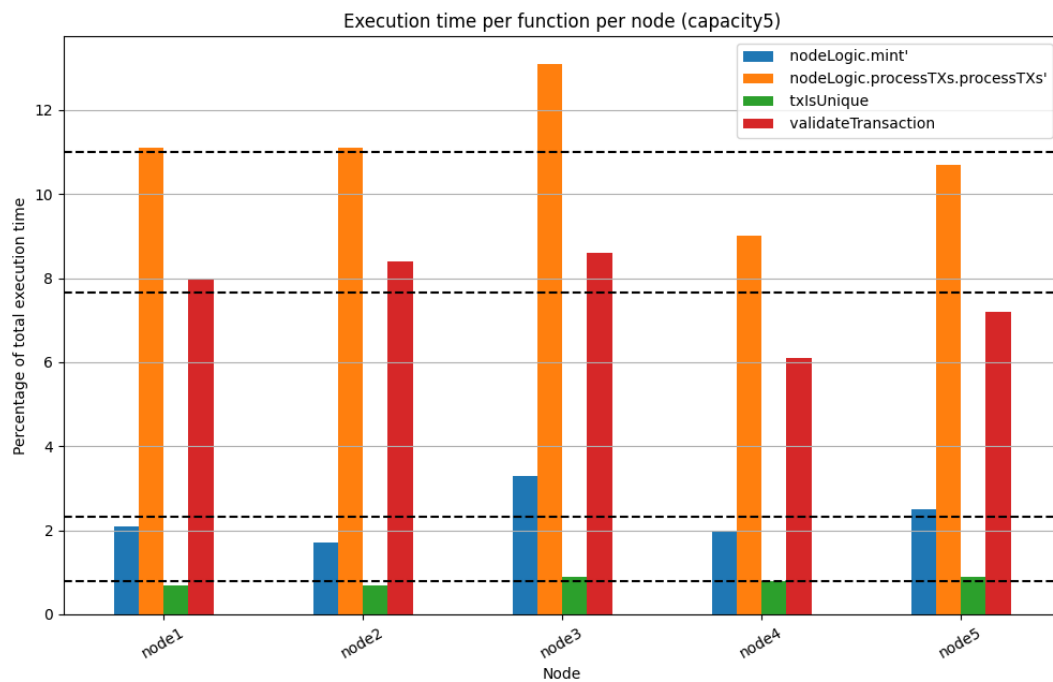


(γ') capacity=20

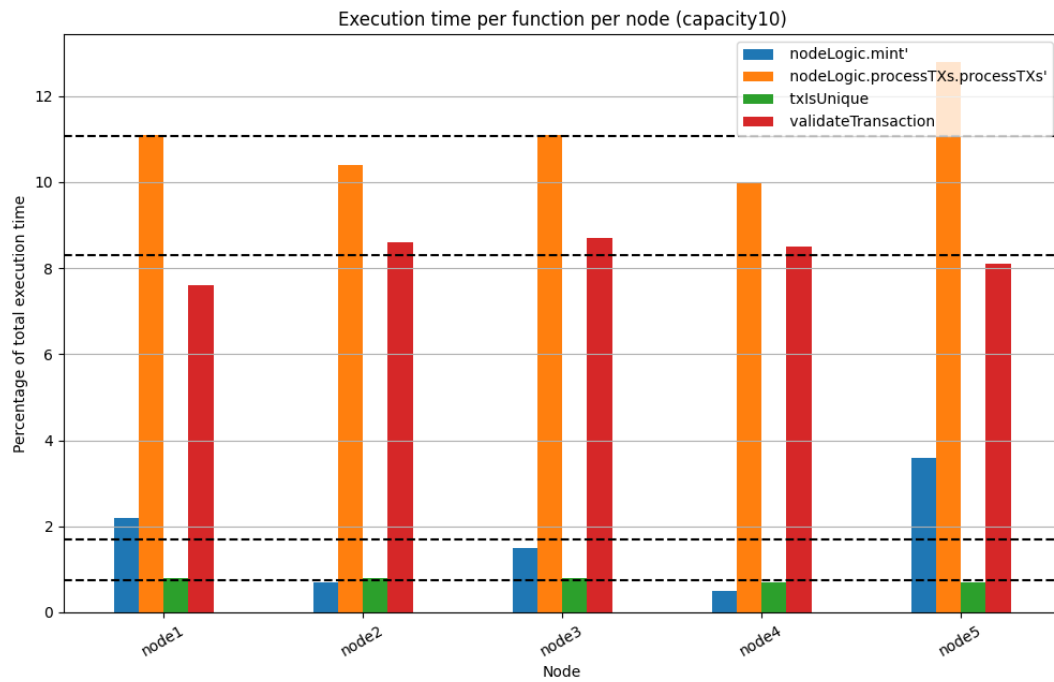
Σχήμα 1: Τα πιο χρονοβόρα κομμάτια του κώδικα

2.2 Συναρτήσεις του συστήματος

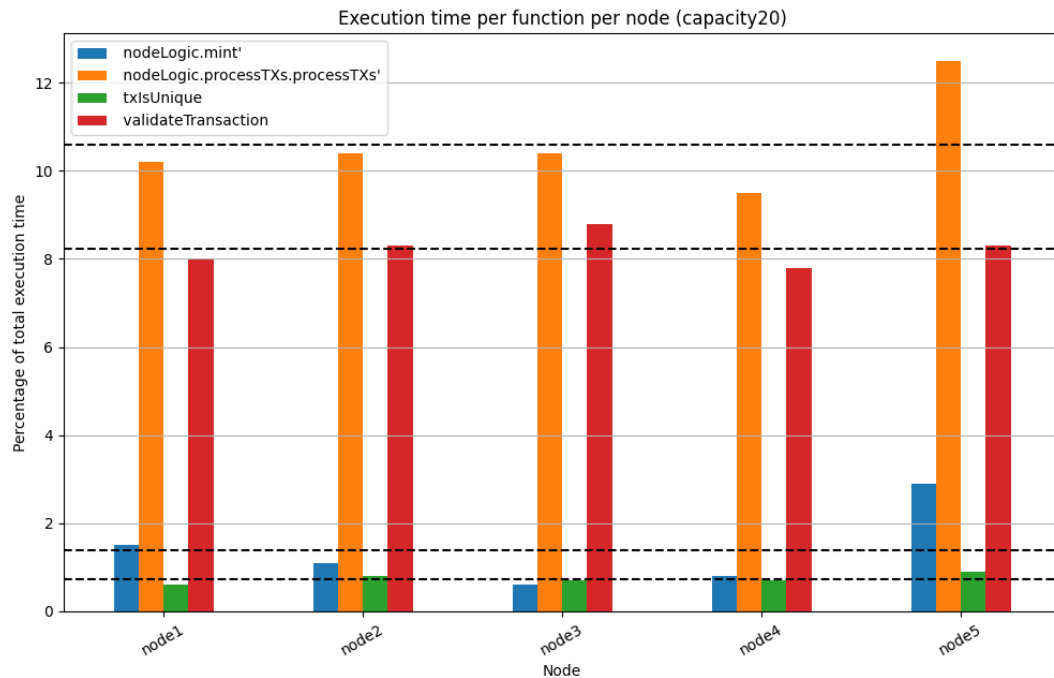
Σχετικά με τις top level συναρτήσεις του συστήματος, παρατηρείται ότι, με κάποιες μικρές διακυμάνσεις, η processTXs καταναλώνει 9-10% του συνολικού χρόνου εκτέλεσης, η mint 1-3% και η validateTransaction 7-9%.



(α') Ρυθμαπόδοση capacity=5



(β') Ρυθμαπόδοση capacity=10



(γ') Ρυθμαπόδοση capacity=20

Σχήμα 2: Ποσοστό χρόνου επί του συνολικού χρόνου εκτέλεσης που λαμβάνει η κάθε συνάρτηση

Στον πίνακα 1 παρουσιάζονται ορισμένα στατιστικά σχετικά με τις συναρτήσεις `processTXs`, `validateTransaction`, `txIsUnique` και `mint`. Το πιο σημαντικό να παρατηρηθεί είναι ότι, για κάθε κόμβο, οι κλήσεις στην συνάρτηση `validateTransaction` είναι ακριβώς τόσες όσες και οι συναλλαγές που αποστέλλονται από όλους τους κόμβους ($5 + 5 \times 50 = 255$). Επίσης, $\#mint + \#validateTransaction = \#processTXs$ ¹. Παρότι φαίνεται σαν να επικυρώθηκαν όλες οι συναλλαγές, αυτό δεν ισχύει. Στην πραγματικότητα, επειδή οι κόμβοι δεν παραλαμβάνουν κατ'ανάγκην τις συναλλαγές με την σειρά αποστολή τους, είναι πιθανό κάποιος validator να ακυρώσει κάποια συναλλαγή η οποία με διαφορετική σειρά θα είχε επιβεβαιωθεί. Για αυτόν τον λόγο φαίνεται ότι ένα υποσύνολο των συναλλαγών εξετάζεται για την μοναδικότητά τους από την συνάρτηση `txIsUnique`². Έτσι αιτιολογείται και το γεγονός ότι το blockchain έχει μήκος μικρότερο από το μέγιστο δυνατό του δεδομένων των συναλλαγών. Παραδείγματος χάριν, για capacity 5 έχει μήκος $45 = \frac{228}{5}e^{\frac{255}{5}} = 51$.

Πίνακας 1: Στατιστικά συναρτήσεων ανά κόμβο

(α') capacity=5

File	Function	Calls	TimeInh	MemInh
node1.prof	processTXs'	301	11.1	11.2
node1.prof	validateTransaction	255	8.0	7.9
node1.prof	txIsUnique	228	0.7	0.0
node1.prof	mint'	45	2.1	3.4
node2.prof	processTXs'	301	11.1	11.3
node2.prof	validateTransaction	255	8.4	8.9
node2.prof	txIsUnique	230	0.7	0.0
node2.prof	mint'	45	1.7	2.3
node3.prof	processTXs'	301	13.1	14.2
node3.prof	validateTransaction	255	8.6	9.4
node3.prof	txIsUnique	227	0.9	0.0
node3.prof	mint'	45	3.3	4.8
node4.prof	processTXs'	301	9.0	8.2
node4.prof	validateTransaction	255	6.1	5.9
node4.prof	txIsUnique	227	0.8	0.0
node4.prof	mint'	45	2.0	2.3
node5.prof	processTXs'	301	10.7	10.5
node5.prof	validateTransaction	255	7.2	7.9
node5.prof	txIsUnique	229	0.9	0.0
node5.prof	mint'	45	2.5	2.6

¹H -1 διαφορά είναι επειδή έγινε η τελευταία κλήση και τα προγράμματα έλαβαν σήμα τερματισμού

²H συνάρτηση είναι βέβαιο ότι επιστρέφει true για όλες τις συναλλαγές, αφού εκ κατασκευής είναι οι συναλλαγές μοναδικές

(β') capacity=10				(γ') capacity=20	
File	Function	Calls	TimeInh	Calls	TimeInh
node1.prof	processTXs'	278	11.1	267	10.2
node1.prof	validateTransaction	255	7.6	255	8.0
node1.prof	txIsUnique	228	0.8	230	0.6
node1.prof	mint'	22	2.2	11	1.5
node2.prof	processTXs'	278	10.4	267	10.4
node2.prof	validateTransaction	255	8.6	255	8.3
node2.prof	txIsUnique	229	0.8	231	0.8
node2.prof	mint'	22	0.7	11	1.1
node3.prof	processTXs'	278	11.1	267	10.4
node3.prof	validateTransaction	255	8.7	255	8.8
node3.prof	txIsUnique	227	0.8	228	0.7
node3.prof	mint'	22	1.5	11	0.6
node4.prof	processTXs'	278	10.0	267	9.5
node4.prof	validateTransaction	255	8.5	255	7.8
node4.prof	txIsUnique	227	0.7	228	0.7
node4.prof	mint'	22	0.5	11	0.8
node5.prof	processTXs'	278	12.8	267	12.5
node5.prof	validateTransaction	255	8.1	255	8.3
node5.prof	txIsUnique	229	0.7	230	0.9
node5.prof	mint'	22	3.6	11	2.9

2.3 Ρυθμαπόδοση και Block time

Χρησιμοποιώντας τα στατιστικά από το profiling καθενός κόμβου μπορούν να εκτιμηθούν η ρυθμαπόδοση και το μέσο block time.

```

(α') capacity=5
node1.prof: total time = 0.56 secs txIsUnique = 228 nodeLogic.mint' = 45
node2.prof: total time = 0.57 secs txIsUnique = 230 nodeLogic.mint' = 45
node3.prof: total time = 0.56 secs txIsUnique = 227 nodeLogic.mint' = 45
node4.prof: total time = 0.80 secs txIsUnique = 227 nodeLogic.mint' = 45
node5.prof: total time = 0.61 secs txIsUnique = 229 nodeLogic.mint' = 45
Time = 3.1, TXs = 1141, Blocks = 45 => Throughput = 368.06, Blocktime = .06

(β') capacity=10
node1.prof: total time = 0.71 secs txIsUnique = 228 nodeLogic.mint' = 22
node2.prof: total time = 0.65 secs txIsUnique = 229 nodeLogic.mint' = 22
node3.prof: total time = 0.65 secs txIsUnique = 227 nodeLogic.mint' = 22
node4.prof: total time = 0.66 secs txIsUnique = 227 nodeLogic.mint' = 22
node5.prof: total time = 0.69 secs txIsUnique = 229 nodeLogic.mint' = 22
Time = 3.36, TXs = 1140, Blocks = 22 => Throughput = 339.28, Blocktime = .15

(γ') capacity=20
node1.prof: total time = 0.60 secs txIsUnique = 230 nodeLogic.mint' = 11
node2.prof: total time = 0.59 secs txIsUnique = 231 nodeLogic.mint' = 11
node3.prof: total time = 0.59 secs txIsUnique = 228 nodeLogic.mint' = 11
node4.prof: total time = 0.69 secs txIsUnique = 228 nodeLogic.mint' = 11
node5.prof: total time = 0.61 secs txIsUnique = 230 nodeLogic.mint' = 11
Time = 3.08, TXs = 1147, Blocks = 11 => Throughput = 372.40, Blocktime = .28

```

Σχήμα 3: Χρόνοι εκτέλεσης κόμβων

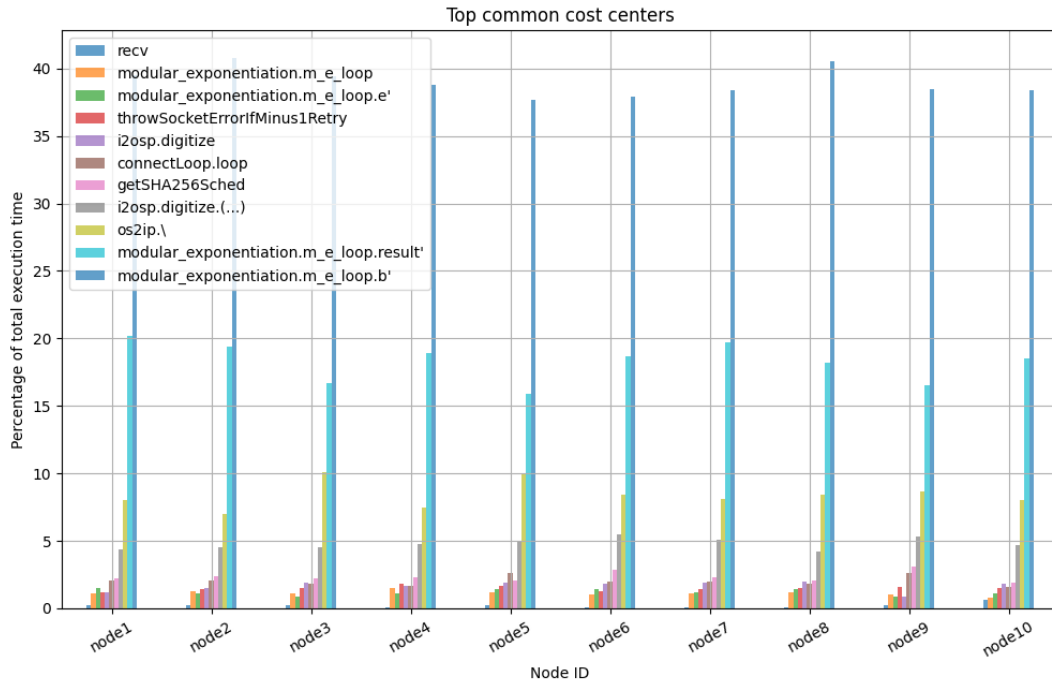
Στον πίνακα 1 φαίνεται ότι, με εξαίρεση λίγους κόμβους στο πείραμα χωρητικότητας 5, το δίκτυο φτάνει εντός πειράματος σε σταθερή κατάσταση, έχοντας επαληθεύει όλες τις συναλλαγές.

3 Κλιμακωσιμότητα του συστήματος

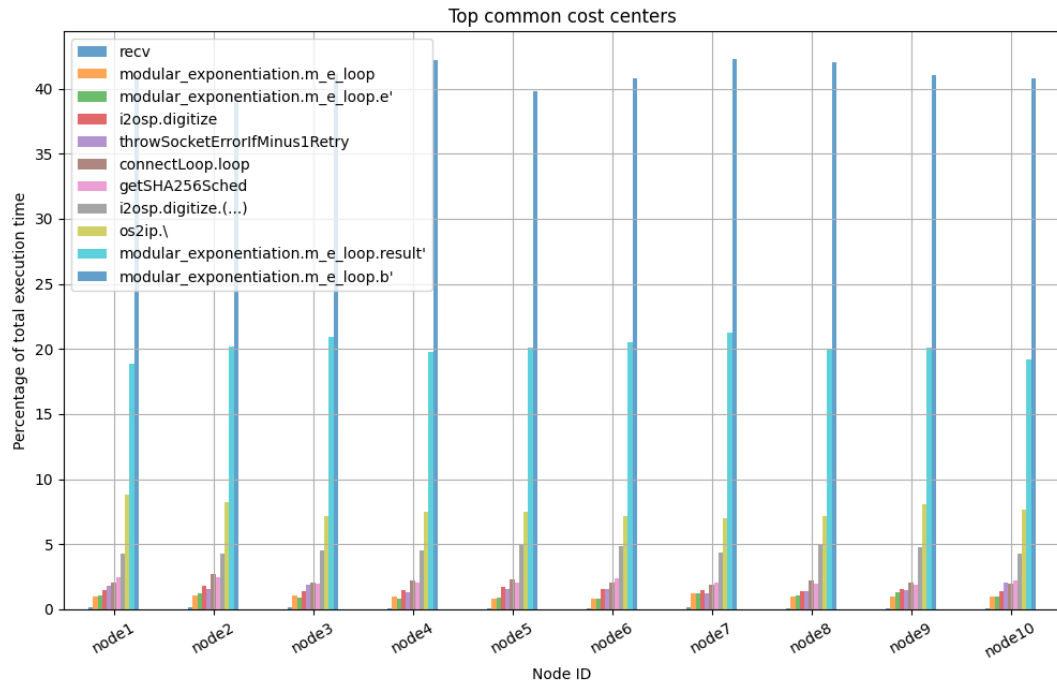
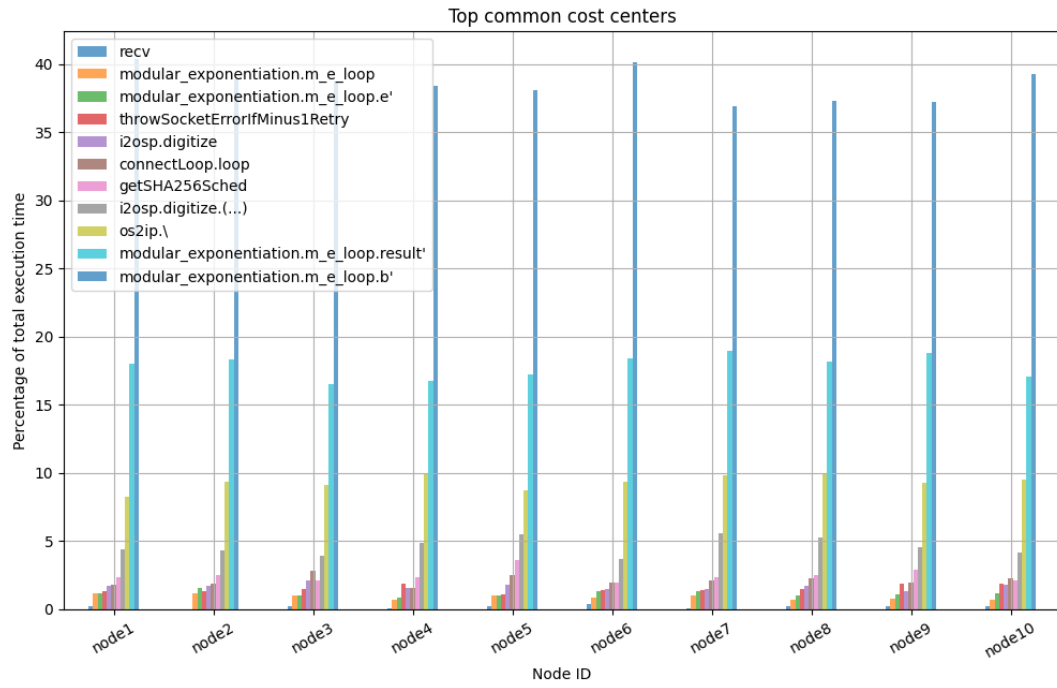
Στο πείραμα κλιμακωσιμότητας, το δίκτυο εκκινείται με 10 κόμβους, καθένας εκ των οποίων εκτελεί 1 staking συναλλαγή, με stake 10 BCC και 100 συναλλαγές (συγκεκριμένα αποστολές μηνυμάτων) προς τους άλλους κόμβους. Σκοπός είναι να εξεταστεί η κλιμάκωση του συστήματος ως προς το πλήθος των συμμετεχόντων κόμβων.

3.1 Χρονοβόρα τμήματα του κώδικα

Στα γραφήματα 4 φαίνονται τα πιο χρονοβόρα κομμάτια του κώδικα για κάθε πείραμα κλιμακωσιμότητας. Φαίνεται ότι αυτά είναι τα ίδια με τα πιο χρονοβόρα κομμάτια του κώδικα για το πείραμα ρυθμαπόδοσης, με την συνάρτηση modular_exponentiation περίπου το 50% του συνολικού χρόνου εκτέλεσης του προγράμματος, αντί για το 40% που ήταν προηγουμένως.



(α') capacity=5

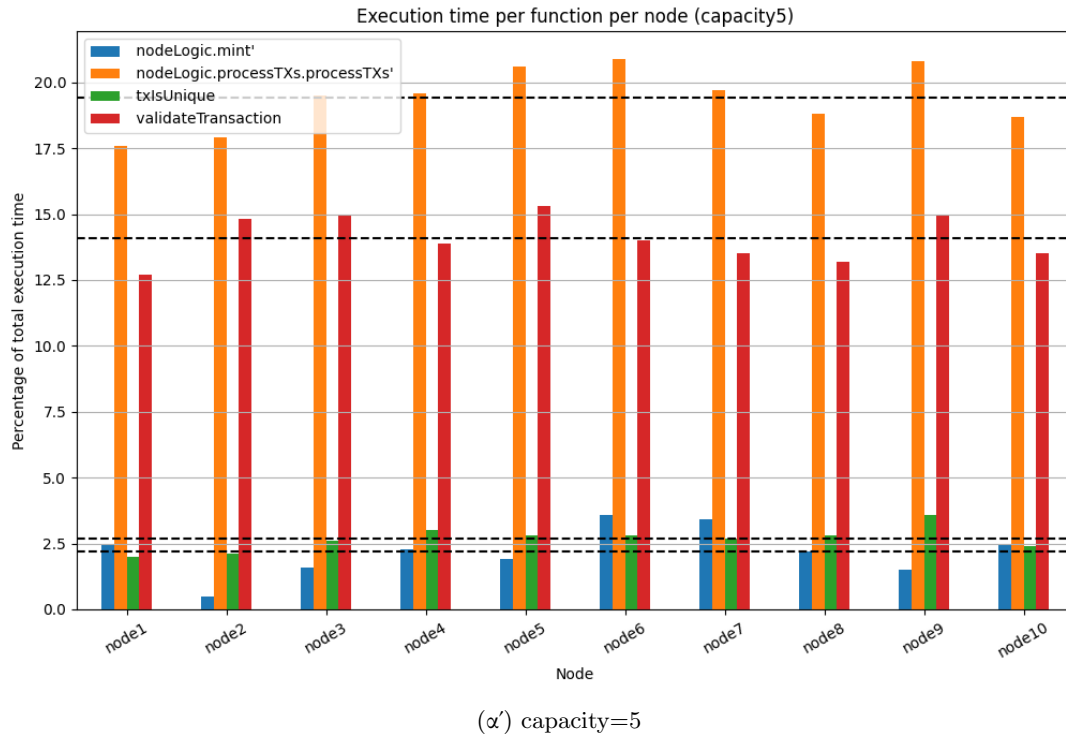


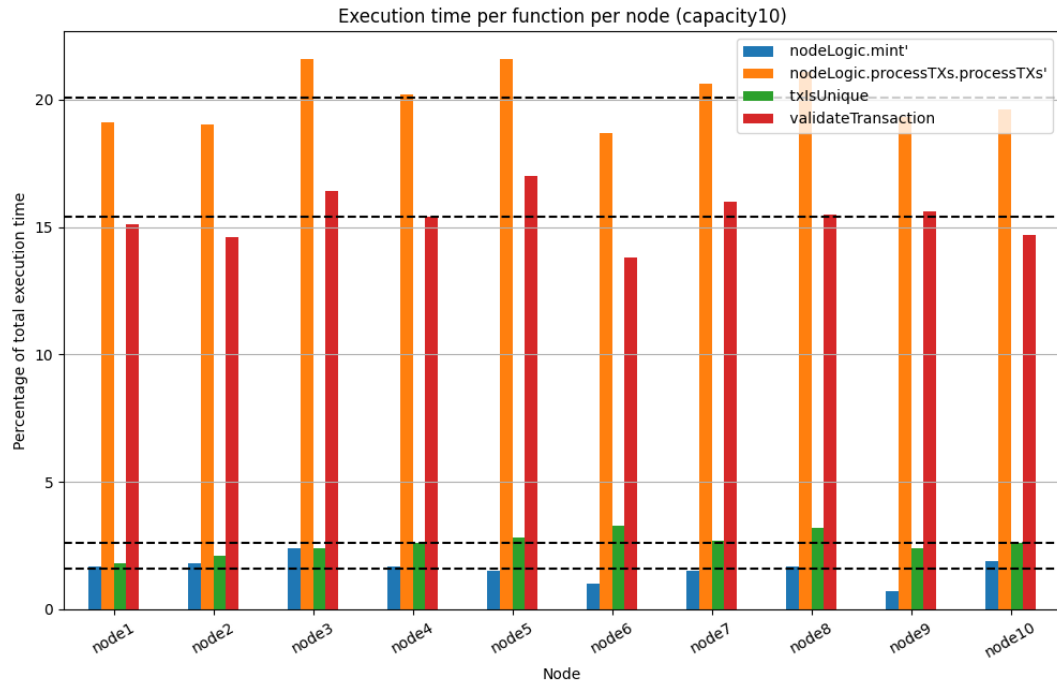
Σχήμα 4: Τα πιο χρονοβόρα κομμάτια του κώδικα

3.2 Συναρτήσεις του συστήματος

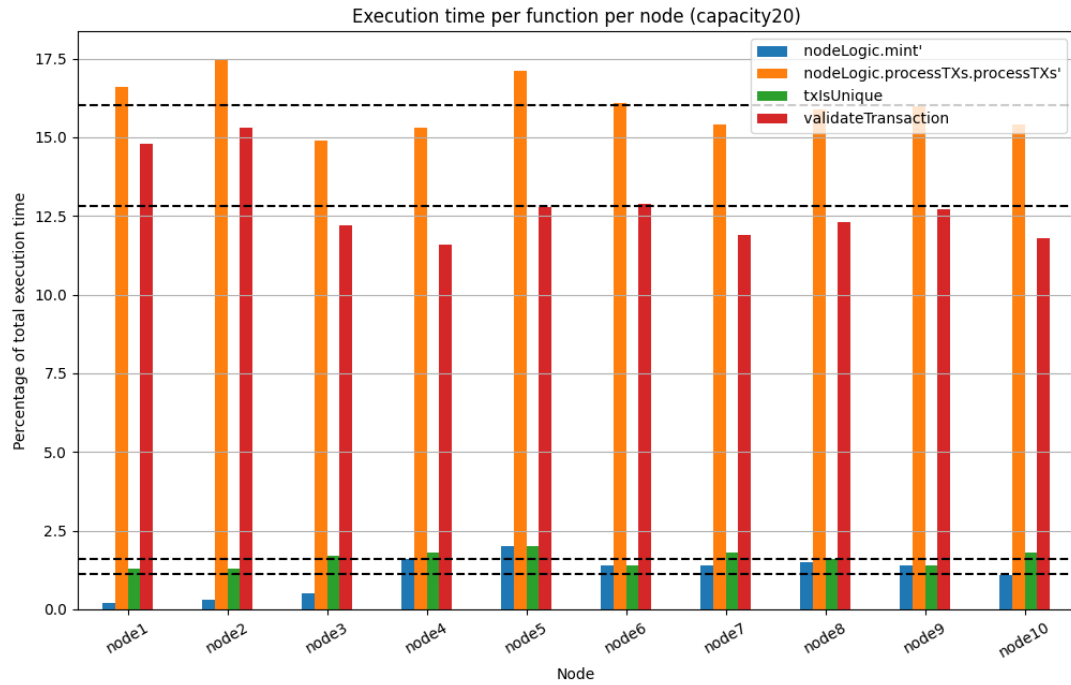
Αρχικά, στο γράφημα 5α' αποτυπώνεται σαφώς το γεγονός

Στα γραφήματα 5 παρατηρείται ότι η `validateTransaction` καταλαμβάνει περίπου 15% του συνολικού χρόνου εκτέλεσης, ενώ προηγουμένως κατελάμβανε περίπου 8%. Δεδομένου ότι οι συναλλαγές έχουν τετραπλασιαστεί, ο πολλαπλασιασμός του ποσοστού επί 2 φαίνεται προβληματικός εκ πρώτης όψews. Ωστόσο, ο χρόνος του τρέχοντος πειράματος είναι διπλάσιος από τον χρόνο του προηγούμενου πειράματος, οπότε στην πραγματικότητα ο χρόνος είναι ο τετραπλάσιος. Επομένως, ο χρόνος που ξοδεύει συνολικά ένας κόμβος στην επικύρωση συναλλαγών αυξάνεται γραμμικά με το πλήθος των συμμετέχοντων κόμβων, σε μία διάταξη όπως η παρούσα, όπου κάθε κόμβος στέλνει ένα σταθερό πλήθος συναλλαγών ανά κόμβο.





(β') capacity=10



(γ') capacity=20

Σχήμα 5: Ποσοστό χρόνου επί του συνολικού χρόνου εκτέλεσης που λαμβάνει η κάθε συνάρτηση

Στον πίνακα 2α' φαίνονται οι κλήσεις ενδιαφέροντος των κόμβων. Παρατηρώντας το πλήθος των κλήσεων ανά συνάρτηση, διαπιστώνεται ότι οι κόμβοι δεν προλαβαίνουν να επικυρώσουν όλες τις συναλλαγές που λαμβάνουν. Αυτό οφείλεται, αφενός στον μεγαλύτερο όγκο συναλλαγών $10 + 10 \times 100 = 1010$ ο οποίος είναι ≈ 4 φορές μεγαλύτερος από προηγουμένως, και αφετέρου στην μικρή χωρητικότητα του block, το οποίο σημαίνει ότι οι κόμβοι πρέπει συχνά να καλούν την χρονοβόρα συνάρτηση mint και να διακόπτουν την διαδικασία επικύρωσης.

Επίσης, παρατηρείται ότι οι κόμβοι 7-8 έχουν μείνει πολύ πίσω σε σχέση με τους υπόλοιπους κόμβους. Αυτό είναι μάλλον συνέπεια της πειραματικής διάταξης, αφού όλοι οι κόμβοι τρέχουν στο ίδιο μηχάνημα.

Πίνακας 2: Στατιστικά συναρτήσεων ανά κόμβο

(α') capacity=5

File	Function	Calls	TimeInh	MemInh
node10.prof	processTXs'	1112	18.7	20.3
node10.prof	validateTransaction	1010	13.5	18.1
node10.prof	txIsUnique	511	2.4	0.0
node10.prof	mint'	101	2.5	2.1
node1.prof	processTXs'	1112	17.6	20.2
node1.prof	validateTransaction	1010	12.7	17.1
node1.prof	txIsUnique	513	2.0	0.0
node1.prof	mint'	101	2.5	3.1
node2.prof	processTXs'	1103	17.9	18.1
node2.prof	validateTransaction	1010	14.8	17.9
node2.prof	txIsUnique	468	2.1	0.0
node2.prof	mint'	92	0.5	0.1
node3.prof	processTXs'	1112	19.5	21.7
node3.prof	validateTransaction	1010	15.0	20.2
node3.prof	txIsUnique	515	2.6	0.0
node3.prof	mint'	101	1.6	1.5
node4.prof	processTXs'	1112	19.6	19.5
node4.prof	validateTransaction	1010	13.9	17.1
node4.prof	txIsUnique	518	3.0	0.0
node4.prof	mint'	101	2.3	2.4
node5.prof	processTXs'	1112	20.6	21.9
node5.prof	validateTransaction	1010	15.3	19.9
node5.prof	txIsUnique	515	2.8	0.0
node5.prof	mint'	101	1.9	2.0
node6.prof	processTXs'	1112	20.9	20.8
node6.prof	validateTransaction	1010	14.0	18.1
node6.prof	txIsUnique	517	2.8	0.0
node6.prof	mint'	101	3.6	2.7
node7.prof	processTXs'	1112	19.7	19.6
node7.prof	validateTransaction	1010	13.5	16.3
node7.prof	txIsUnique	515	2.7	0.0
node7.prof	mint'	101	3.4	3.3
node8.prof	processTXs'	1112	18.8	18.4
node8.prof	validateTransaction	1010	13.2	16.4
node8.prof	txIsUnique	516	2.8	0.0
node8.prof	mint'	101	2.2	2.0
node9.prof	processTXs'	1112	20.8	21.7
node9.prof	validateTransaction	1010	15.0	20.2
node9.prof	txIsUnique	516	3.6	0.0
node9.prof	mint'	101	1.5	1.5

Αντιθέτως, στους πίνακες 2β' και 2γ' φαίνεται από τις κλήσεις των συναρτήσεων ότι έχουν επικυρωθεί όλες οι συναλλαγές και έχουν παραχθεί τα αντίστοιχα blocks. Η μεγαλύτερη χωρητικότητα των blocks επιτρέπει στους κόμβους μεγαλύτερα χρονικά παράθυρα για την επικύρωση των συναλλαγών και η διακοπή για την παραγωγή των blocks δεν καθυστερεί την εξέλιξη του δικτύου.

Η ρυθμαπόδοση του συστήματος μπορεί να υπολογιστεί όπως προηγουμένως:

(β') capacity=10				(γ') capacity=20	
File	Function	Calls	TimeInh	Calls	TimeInh
node10.prof	processTXs'	1061	19.6	782	15.4
node10.prof	validateTransaction	1010	14.7	758	11.8
node10.prof	txIsUnique	504	2.6	481	1.8
node10.prof	mint'	50	1.9	24	1.1
node1.prof	processTXs'	1061	19.1	1033	16.6
node1.prof	validateTransaction	1010	15.1	1010	14.8
node1.prof	txIsUnique	506	1.8	454	1.3
node1.prof	mint'	50	1.7	22	0.2
node2.prof	processTXs'	1061	19.0	1034	17.5
node2.prof	validateTransaction	1010	14.6	1010	15.3
node2.prof	txIsUnique	507	2.1	480	1.3
node2.prof	mint'	50	1.8	23	0.3
node3.prof	processTXs'	1061	21.6	785	14.9
node3.prof	validateTransaction	1010	16.4	761	12.2
node3.prof	txIsUnique	508	2.4	483	1.7
node3.prof	mint'	50	2.4	24	0.5
node4.prof	processTXs'	1061	20.2	773	15.3
node4.prof	validateTransaction	1010	15.4	749	11.6
node4.prof	txIsUnique	509	2.6	482	1.8
node4.prof	mint'	50	1.7	24	1.6
node5.prof	processTXs'	1061	21.6	777	17.1
node5.prof	validateTransaction	1010	17.0	753	12.8
node5.prof	txIsUnique	507	2.8	482	2.0
node5.prof	mint'	50	1.5	24	2.0
node6.prof	processTXs'	1061	18.7	801	16.1
node6.prof	validateTransaction	1010	13.8	777	12.9
node6.prof	txIsUnique	507	3.3	482	1.4
node6.prof	mint'	50	1.0	24	1.4
node7.prof	processTXs'	1061	20.6	810	15.4
node7.prof	validateTransaction	1010	16.0	786	11.9
node7.prof	txIsUnique	507	2.7	482	1.8
node7.prof	mint'	50	1.5	24	1.4
node8.prof	processTXs'	1061	21.1	769	15.9
node8.prof	validateTransaction	1010	15.5	745	12.3
node8.prof	txIsUnique	509	3.2	481	1.6
node8.prof	mint'	50	1.7	24	1.5
node9.prof	processTXs'	1061	19.3	796	16.0
node9.prof	validateTransaction	1010	15.6	772	12.7
node9.prof	txIsUnique	508	2.4	481	1.4
node9.prof	mint'	50	0.7	24	1.4

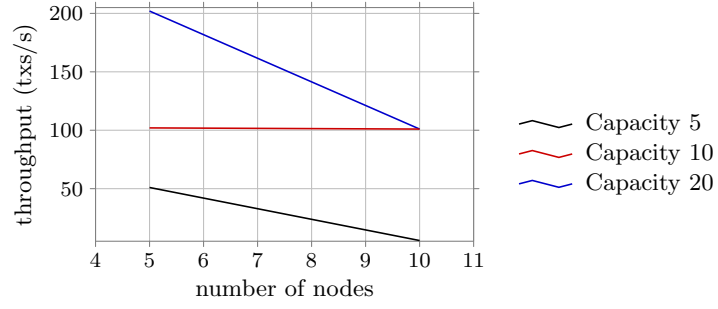
$$\text{Ρυθμαπόδοση} = \frac{\text{Συνολικές συναλλαγές}}{\text{Συνολικός χρόνος}} \Rightarrow \left\{ \begin{array}{l} th_{capacity=5} = \frac{0.8 \times 62 + 0.2 \times 5}{10} \\ th_{capacity=10} = \frac{1010}{10} \\ th_{capacity=20} = \frac{1010}{10} \end{array} \right\} \Rightarrow$$

$$\begin{aligned}
th_{capacity=5} &= 5.6 \frac{txs}{s} \\
th_{capacity=10} &= 101 \frac{txs}{s} \\
th_{capacity=20} &= 101 \frac{txs}{s}
\end{aligned}$$

(1)

Στο παρακάτω γράφημα φαίνεται η ρυθμαπόδοση του συστήματος για κάθε πείραμα:

Σχήμα 6: Throughput του συστήματος



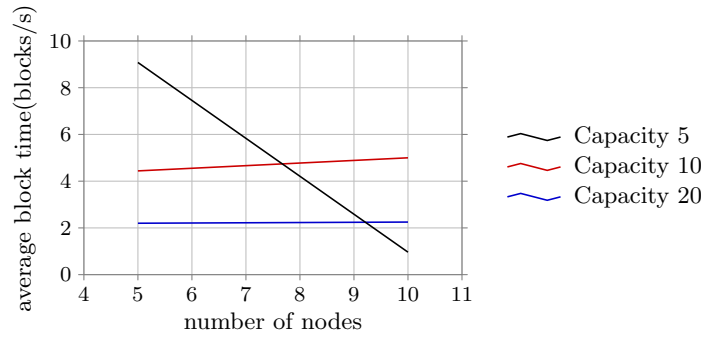
3.3 Block Time

Το μέσο block time του συστήματος υπολογίζεται πάλι όπως προηγουμένως:

$$\text{Block Time} = \frac{\text{Συνολικός χρόνος}}{\text{Συνολικά blocks}} \Rightarrow \left\{ \begin{array}{l} bt_{capacity=5} = \frac{0.8 \times 12 + 0.2 \times 1}{10} \\ bt_{capacity=10} = \frac{50}{10} \\ bt_{capacity=20} = \frac{0.1 \times 22 + 0.1 \times 23 + 0.8 \times 25}{10} \end{array} \right\} \Rightarrow \quad (2)$$

$$\begin{aligned} bt_{capacity=5} &= 0.962 \frac{\text{blocks}}{s} \\ bt_{capacity=10} &= 5 \frac{\text{blocks}}{s} \\ bt_{capacity=20} &= 2.25 \frac{\text{blocks}}{s} \end{aligned}$$

Σχήμα 7: Μέσο block time του συστήματος



(α') capacity=5

```
node10.prof: total time = 0.75 secs  txIsUnique = 511 nodeLogic.mint' = 101
node1.prof: total time = 0.76 secs  txIsUnique = 513 nodeLogic.mint' = 101
node2.prof: total time = 0.71 secs  txIsUnique = 468 nodeLogic.mint' = 92
node3.prof: total time = 0.68 secs  txIsUnique = 515 nodeLogic.mint' = 101
node4.prof: total time = 0.78 secs  txIsUnique = 518 nodeLogic.mint' = 101
node5.prof: total time = 0.72 secs  txIsUnique = 515 nodeLogic.mint' = 101
node6.prof: total time = 0.77 secs  txIsUnique = 517 nodeLogic.mint' = 101
node7.prof: total time = 0.82 secs  txIsUnique = 515 nodeLogic.mint' = 101
node8.prof: total time = 0.77 secs  txIsUnique = 516 nodeLogic.mint' = 101
node9.prof: total time = 0.70 secs  txIsUnique = 516 nodeLogic.mint' = 101
Time = 7.46, TXs = 5104, Blocks = 101 => Throughput = 684.18, Blocktime = .07
( $\beta'$ ) capacity=10
```

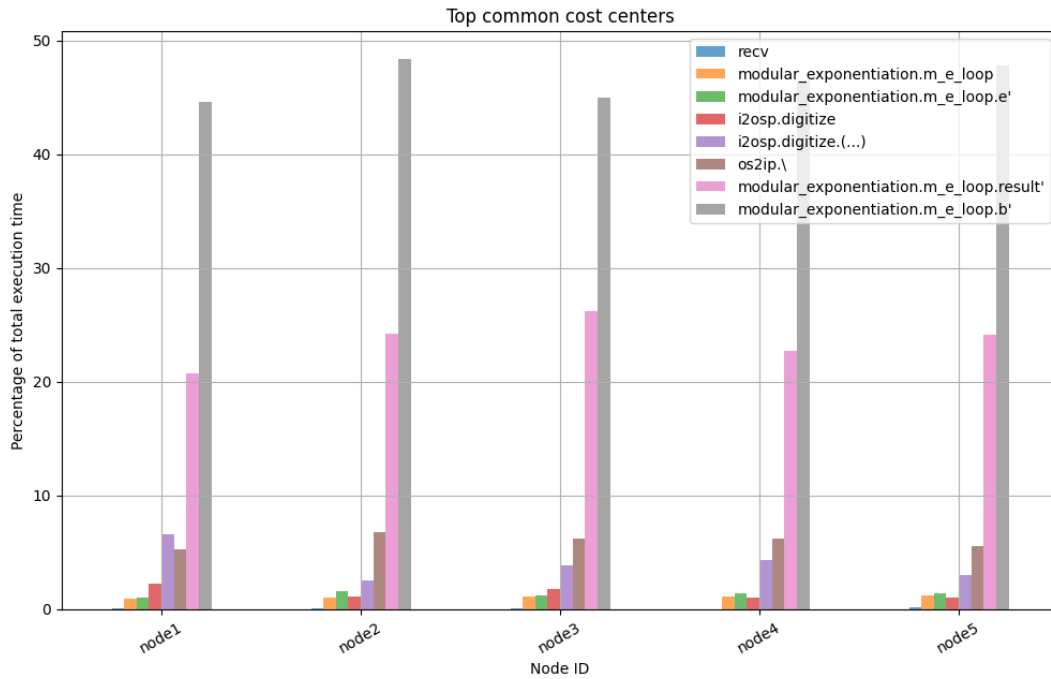
```
node10.prof: total time = 0.69 secs  txIsUnique = 504 nodeLogic.mint' = 50
node1.prof: total time = 0.67 secs  txIsUnique = 506 nodeLogic.mint' = 50
node2.prof: total time = 0.74 secs  txIsUnique = 507 nodeLogic.mint' = 50
node3.prof: total time = 0.68 secs  txIsUnique = 508 nodeLogic.mint' = 50
node4.prof: total time = 0.71 secs  txIsUnique = 509 nodeLogic.mint' = 50
node5.prof: total time = 0.70 secs  txIsUnique = 507 nodeLogic.mint' = 50
node6.prof: total time = 0.75 secs  txIsUnique = 507 nodeLogic.mint' = 50
node7.prof: total time = 0.70 secs  txIsUnique = 507 nodeLogic.mint' = 50
node8.prof: total time = 0.70 secs  txIsUnique = 509 nodeLogic.mint' = 50
node9.prof: total time = 0.69 secs  txIsUnique = 508 nodeLogic.mint' = 50
Time = 7.03, TXs = 5072, Blocks = 50 => Throughput = 721.47, Blocktime = .14
( $\gamma'$ ) capacity=20
```

```
node10.prof: total time = 1.08 secs  txIsUnique = 481 nodeLogic.mint' = 24
node1.prof: total time = 1.04 secs  txIsUnique = 454 nodeLogic.mint' = 22
node2.prof: total time = 1.07 secs  txIsUnique = 480 nodeLogic.mint' = 23
node3.prof: total time = 1.04 secs  txIsUnique = 483 nodeLogic.mint' = 24
node4.prof: total time = 1.06 secs  txIsUnique = 482 nodeLogic.mint' = 24
node5.prof: total time = 1.04 secs  txIsUnique = 482 nodeLogic.mint' = 24
node6.prof: total time = 1.05 secs  txIsUnique = 482 nodeLogic.mint' = 24
node7.prof: total time = 1.17 secs  txIsUnique = 482 nodeLogic.mint' = 24
node8.prof: total time = 1.08 secs  txIsUnique = 481 nodeLogic.mint' = 24
node9.prof: total time = 1.12 secs  txIsUnique = 481 nodeLogic.mint' = 24
Time = 10.75, TXs = 4788, Blocks = 24 => Throughput = 445.39, Blocktime = .44
```

Σχήμα 8: Χρόνοι εκτέλεσης κόμβων

4 Δικαιοσύνη

Στο πείραμα δικαιοσύνης, το δίκτυο εκκινείται με 5 κόμβους και ο υπάριθμὸν 1 ἀπὸ αὐτοὺς κάνει stake 100 BCC, ἐνῶ οἱ ὑπόλοιποι κάνουν stake 10 BCC.

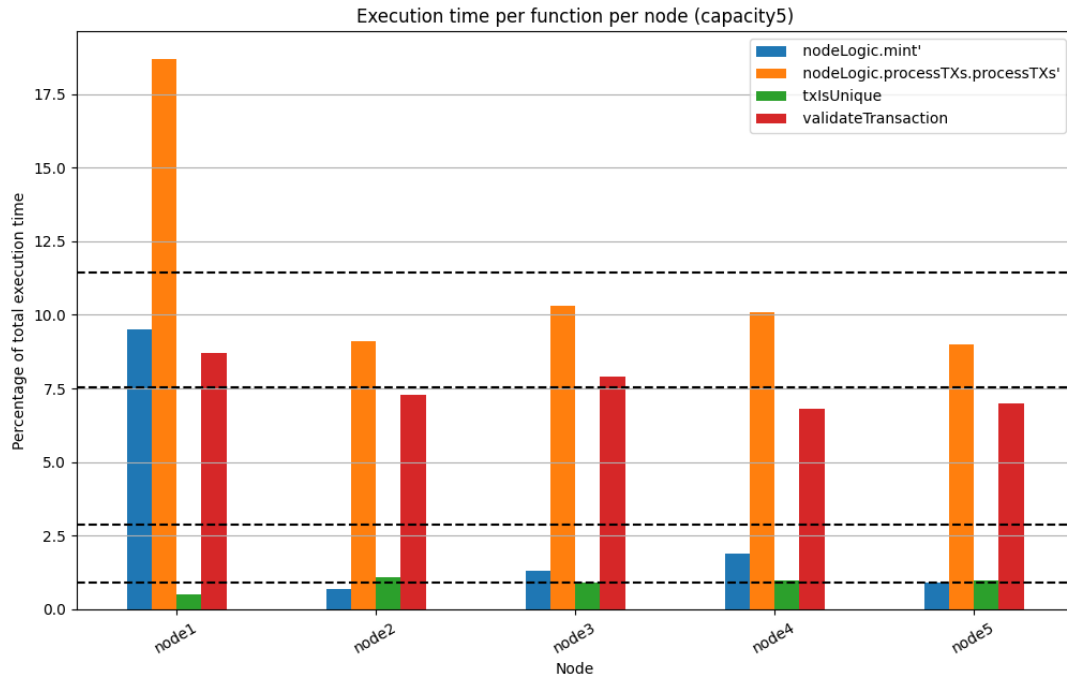


Σχήμα 9: Τα πιο χρονοβόρα κομμάτια του κώδικα capacity=5

Στον πίνακα 3 φαίνονται οἱ κλήσεις μερικῶν συναρτήσεων ενδιαφέροντος. Φαίνεται ὅτι τὰ πλήθη ὅλων τῶν κλήσεων εἶναι ἴδια ἀνὰ κόμβο, πράγμα που σημαίνει ὅτι οἱ κόμβοι ἐκτελοῦν τὶς ἴδιες λειτουργίες με τὴν ἴδια συχρότητα. Παρ'ὅλα αὐτά, ὁ κόμβος με τὸ μεγαλύτερο stake καταναλώνει πολὺ περισσότερο χρόνο στὴν συνάρτηση mint' σε σχέση με τοὺς ὑπόλοιπους κόμβους, ὅπως φαίνεται καὶ στὸ σχῆμα 9.

Πίνακας 3: Στατιστικά συναρτήσεων ανά κόμβο capacity=5

File	Function	Calls	TimeInh	MemInh
node1.prof	processTXs'	267	18.7	20.7
node1.prof	validateTransaction	225	8.7	8.7
node1.prof	txIsUnique	208	0.5	0.0
node1.prof	mint'	41	9.5	12.0
node2.prof	processTXs'	267	9.1	10.0
node2.prof	validateTransaction	225	7.3	9.4
node2.prof	txIsUnique	207	1.1	0.0
node2.prof	mint'	41	0.7	0.5
node3.prof	processTXs'	268	10.3	9.9
node3.prof	validateTransaction	225	7.9	8.7
node3.prof	txIsUnique	212	0.9	0.0
node3.prof	mint'	42	1.3	1.1
node4.prof	processTXs'	268	10.1	11.4
node4.prof	validateTransaction	225	6.8	8.8
node4.prof	txIsUnique	212	1.0	0.0
node4.prof	mint'	42	1.9	2.6
node5.prof	processTXs'	268	9.0	7.3
node5.prof	validateTransaction	225	7.0	6.3
node5.prof	txIsUnique	213	1.0	0.0
node5.prof	mint'	42	0.9	1.0



Σχήμα 10: Ποσοστό χρόνου επί του συνολικού χρόνου εκτέλεσης που λαμβάνει η κάθε συνάρτηση capacity=5

Στο σχήμα 9 φαίνεται, πράγματι, ότι ο κόμβος με το μεγαλύτερο stake καταναλώνει πολύ περισσότερο χρόνο στην συνάρτηση `mint` σε σχέση με τους υπόλοιπους κόμβους, ενδεικτικό του γεγονότος ότι πράγματι αυτός αναλαμβάνει συχνότερα την δημιουργία των νέων blocks. Μάλιστα, επισκοπώντας τα υπόλοιπα των λογαριασμών των κόμβων στο σχήμα 10, παρατηρεί κανείς ότι όντως τα περισσότερα νομίσματα συσσωρεύονται στον κόμβο με το μεγαλύτερο stake.

Σχήμα 11: Υπόλοιπα λογαριασμών κόμβων `capacity=5`