



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ  
ΥΠΟΛΟΓΙΣΤΩΝ

Κατανεμημένα Συστήματα  
Εξαμηνιαία Εργασία - BlockChat

Αναστάσιος Στέφανος Αναγνώστου  
03119051

11 Μαρτίου 2024

# Περιεχόμενα

<b>I</b>	<b>Σχεδιασμός Συστήματος</b>	<b>3</b>
1	Δεν είμαι σίγουρος	3
<b>II</b>	<b>Πειράματα</b>	<b>4</b>
2	Απόδοση του συστήματος	4
2.1	Ρυθμαπόδοση . . . . .	4
2.2	Block Time . . . . .	8
3	Κλιμακωσιμότητα του συστήματος	10
4	Δικαιοσύνη	16

Μέρος Ι

# Σχεδιασμός Συστήματος

1 Δεν είμαι σίγουρος

## Μέρος II

# Πειράματα

## 2 Απόδοση του συστήματος

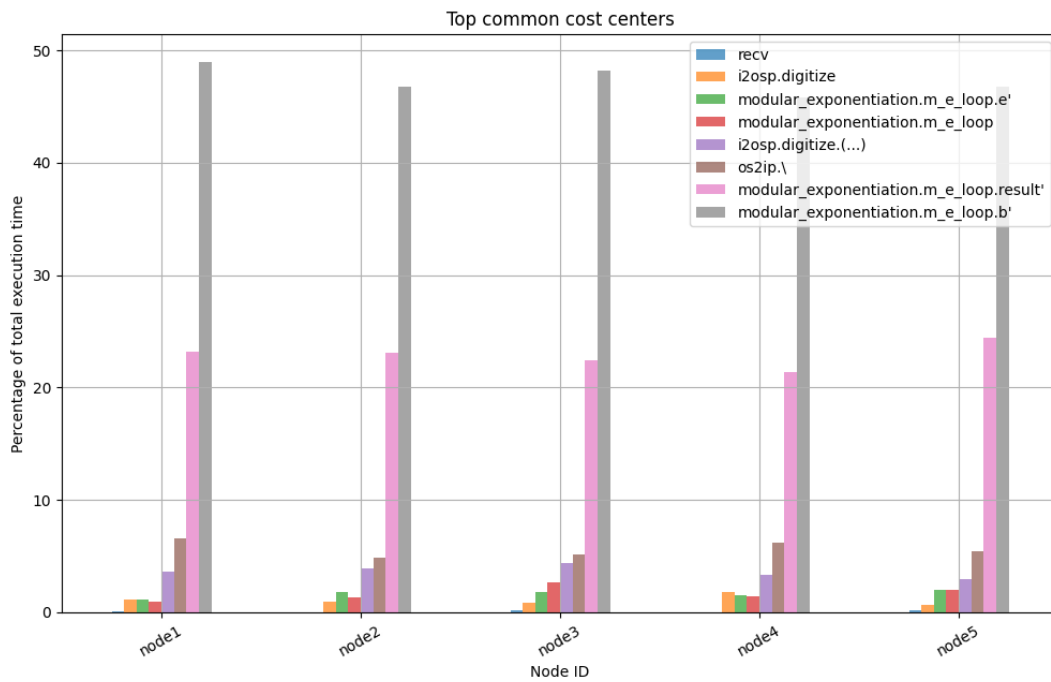
Ανά πείραμα αξιολογούνται, αφενός τα πιο χρονοβόρα κομμάτια του κώδικα, όπως υποδεικνύει το profiling καθενός κόμβου κατά την εκτέλεση του πειράματος, αφετέρου οι συναρτήσεις mint, validateTransaction και processTXs, οι οποίες συνιστούν την λογική λειτουργίας των κόμβων του συστήματος.

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.

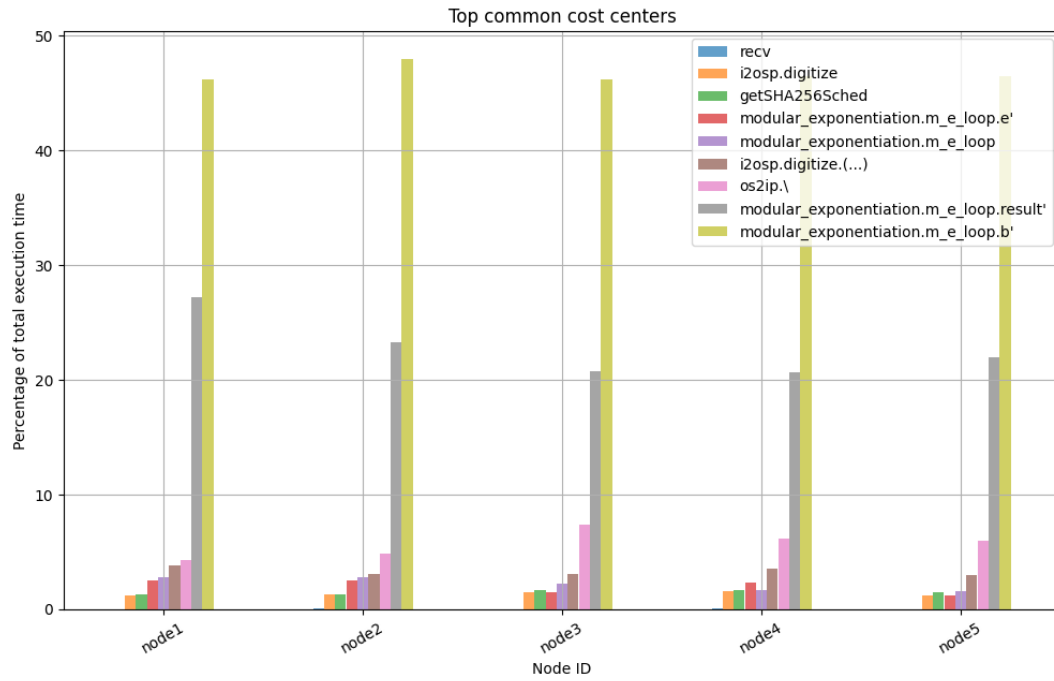
### 2.1 Ρυθμαπόδοση

Στο πείραμα για την αξιολόγηση της ρυθμαπόδοσης του συστήματος, στήνεται ένα δίκτυο 5 κόμβων, καθένας εκ των οποίων εκτελεί 1 staking 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.

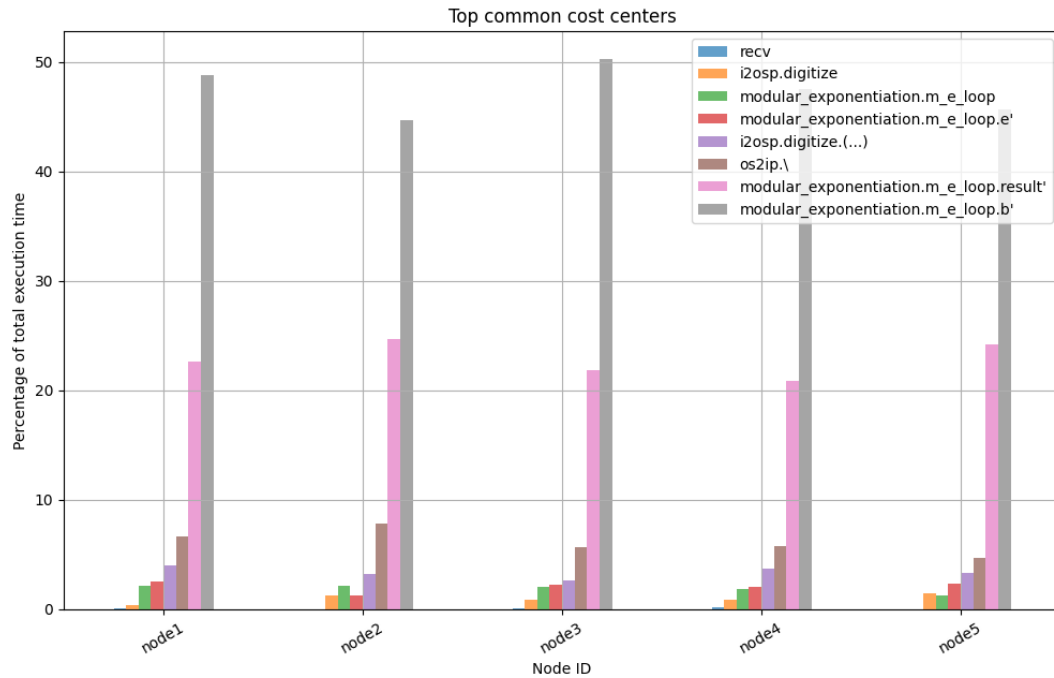
Το πρώτο πράγμα που φαίνεται στο σχήμα 1 είναι ότι το μακράν πιο χρονοβόρο μέρος του κώδικα είναι η συνάρτηση modular.exponentiation που χρησιμοποιείται γενικά για την κρυπτογράφηση / αποκρυπτογράφηση και υπογραφή / επαλήθευση μηνυμάτων. Συγκεκριμένα, φαίνεται να λαμβάνει περίπου το 50% του συνολικού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



(α') capacity=5



( $\beta'$ ) capacity=10



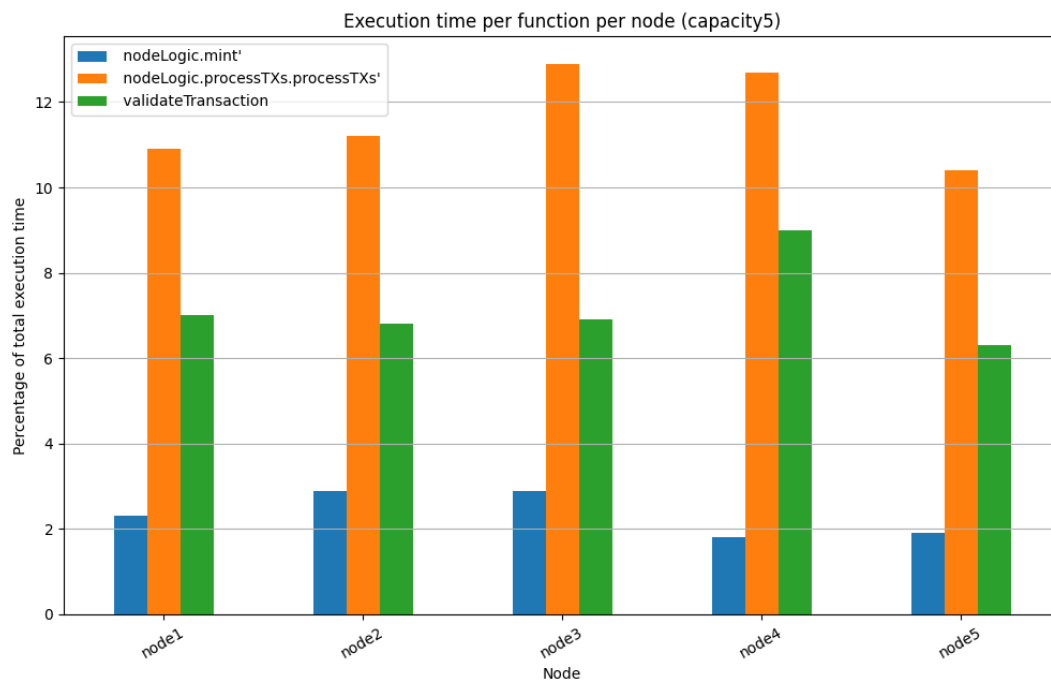
( $\gamma'$ ) capacity=20

Σχήμα 1: Τα πιο χρονοβόρα κομμάτια του κώδικα

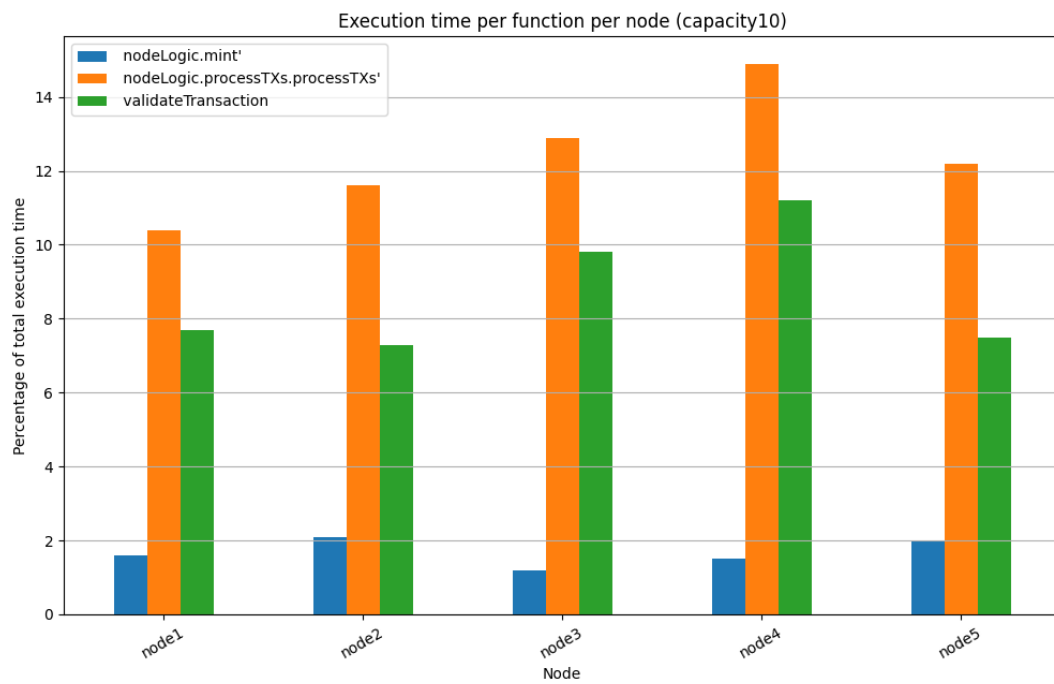
Στον πίνακα 1 παρουσιάζονται ορισμένα στατιστικά σχετικά με τις συναρτήσεις mint, validateTransaction και processTXs. Το πιο σημαντικό να παρατηρηθεί είναι ότι, για κάθε κόμβο, οι κλήσεις στην συνάρτηση validateTransaction είναι ακριβώς τόσες όσες και οι συναλλαγές που αποστέλλονται από όλους τους κόμβους ( $5 + 5 \times 50 = 255$ ). Επίσης,  $\#mint + \#validateTransaction = \#processTXs$  (η -1 διαφορά είναι επειδή έγινε η τελευταία κλήση και τα προγράμματα έλαβαν σήμα τερματισμού). Αυτό σημαίνει ότι οι κόμβοι επιβεβαίωσαν όλες τις συναλλαγές κατά την διάρκεια του πειράματος.

Εξάιρεση αποτελεί ο πίνακας 1α', όπου οι κόμβοι δεν προλαβαίνουν να επιβεβαιώσουν όλες τις συναλλαγές που λαμβάνουν, αλλά εκκρεμούν 5 από τις 255. Αυτό οφείλεται στην μικρή χωρητικότητα του block, το οποίο σημαίνει ότι οι κόμβοι πρέπει συχνά να καλούν την χρονοβόρα συνάρτηση mint και να διακόπτουν την διαδικασία επιβεβαίωσης.

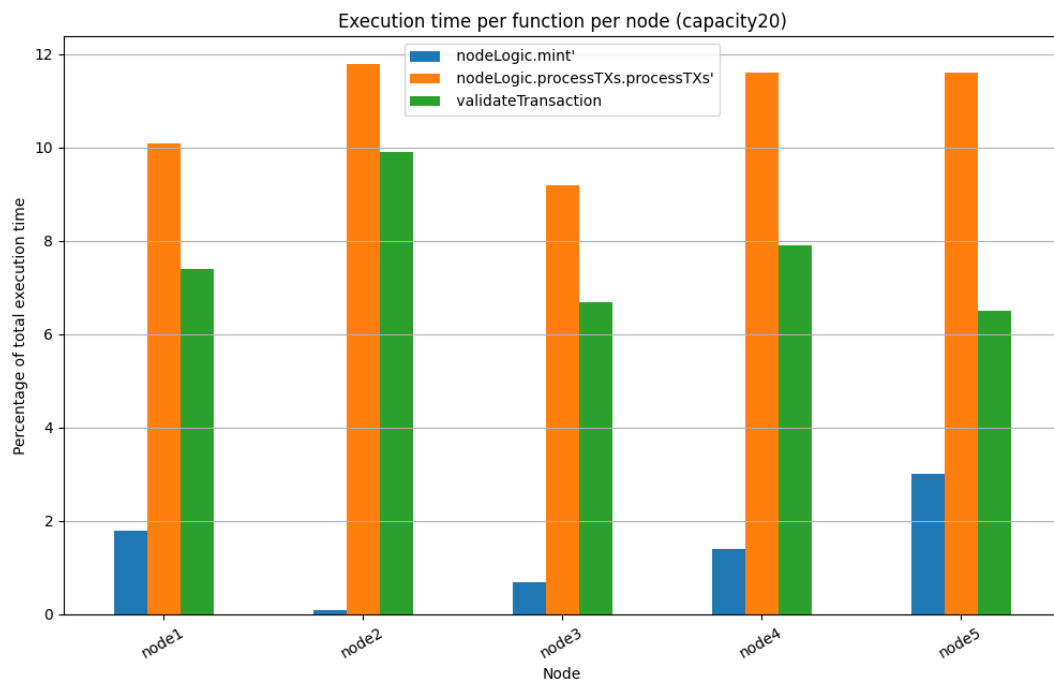
Γενικά, παρατηρείται ότι, με κάποιες μικρές διακυμάνσεις, η processTXs καταναλώνει 9-10% του συνολικού χρόνου εκτέλεσης, η mint 1-3% και η validateTransaction 5-7%.



(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

Πίνακας 1: Στατιστικά συναρτήσεων ανά κόμβο

(α') capacity=5

File	Function	Calls	TimeInh	MemInh
node1.prof	processTXs'	296	10.9	13.2
node1.prof	validateTransaction	250	7.0	9.4
node1.prof	mint'	46	2.3	3.7
node2.prof	processTXs'	295	11.2	12.6
node2.prof	validateTransaction	249	6.8	8.9
node2.prof	mint'	46	2.9	3.7
node3.prof	processTXs'	301	12.9	12.1
node3.prof	validateTransaction	255	6.9	7.6
node3.prof	mint'	45	2.9	4.5
node4.prof	processTXs'	301	12.7	12.0
node4.prof	validateTransaction	255	9.0	9.8
node4.prof	mint'	45	1.8	2.1
node5.prof	processTXs'	301	10.4	9.9
node5.prof	validateTransaction	255	6.3	7.6
node5.prof	mint'	45	1.9	2.3

(β') capacity=10

File	Function	Calls	TimeInh
node1.prof	processTXs'	278	10.4
node1.prof	validateTransaction	255	7.7
node1.prof	mint'	22	1.6
node2.prof	processTXs'	279	11.6
node2.prof	validateTransaction	255	7.3
node2.prof	mint'	23	2.1
node3.prof	processTXs'	278	12.9
node3.prof	validateTransaction	255	9.8
node3.prof	mint'	22	1.2
node4.prof	processTXs'	278	14.9
node4.prof	validateTransaction	255	11.2
node4.prof	mint'	22	1.5
node5.prof	processTXs'	278	12.2
node5.prof	validateTransaction	255	7.5
node5.prof	mint'	22	2.0

(γ') capacity=20

Calls	TimeInh
267	10.1
255	7.4
11	1.8
267	11.8
255	9.9
11	0.1
267	9.2
255	6.7
11	0.7
267	11.6
255	7.9
11	1.4
267	11.6
255	6.5
11	3.0

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. Αυτό σημαίνει ότι η ρυθμαπόδοση του συστήματος μπορεί να υπολογιστεί για κάθε πείραμα ως εξής:

$$\text{Ρυθμαπόδοση} = \frac{\text{Συνολικές συναλλαγές}}{\text{Συνολικός χρόνος}} \Rightarrow \begin{array}{l} th_{capacity=5} = \frac{255}{5} = 51 \frac{txs}{s} \\ th_{capacity=10} = \frac{510}{5} = 102 \frac{txs}{s} \\ th_{capacity=20} = \frac{1010}{5} = 202 \frac{txs}{s} \end{array} \quad (1)$$

## 2.2 Block Time

Ομοίως μπορεί να εκτιμηθεί και το μέσο block time του συστήματος:



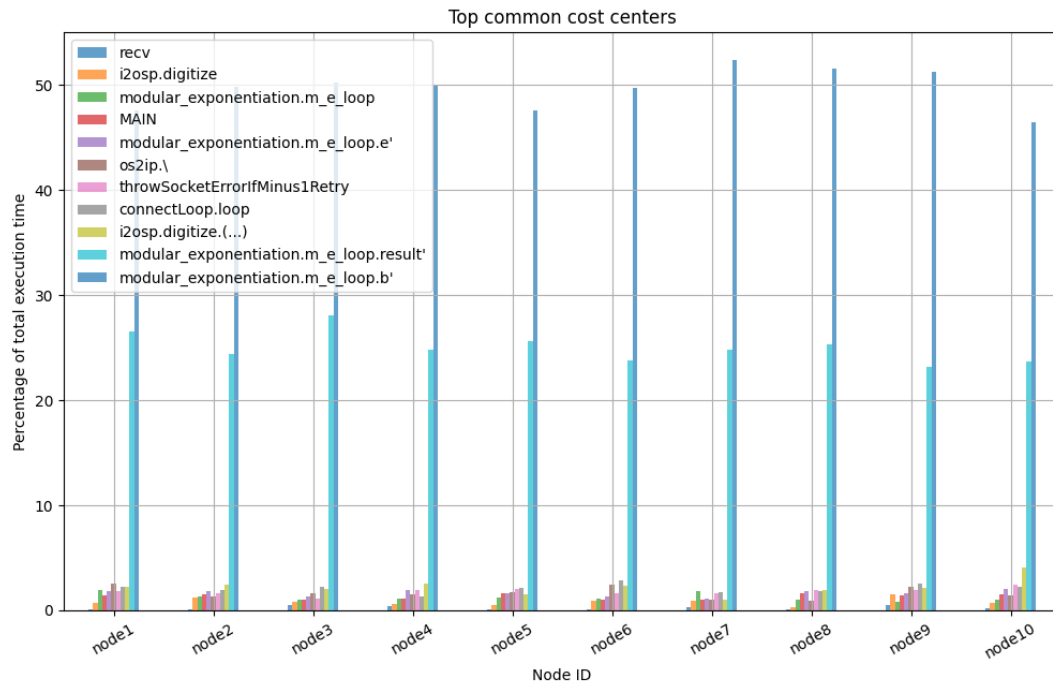
$$\text{Block Time} = \frac{\text{Συνολικός χρόνος}}{\text{Συνολικά blocks}} \Rightarrow \left\{ \begin{array}{l} bt_{capacity=5} = \frac{0.4 \times 46 + 0.6 \times 45}{5} \\ bt_{capacity=10} = \frac{0.8 \times 22 + 0.2 \times 23}{5} \\ bt_{capacity=20} = \frac{11}{5} \end{array} \right\} \Rightarrow$$

$$\begin{array}{l} bt_{capacity=5} = 9.08 \frac{blocks}{s} \\ bt_{capacity=10} = 4.44 \frac{blocks}{s} \\ bt_{capacity=20} = 2.2 \frac{blocks}{s} \end{array}$$

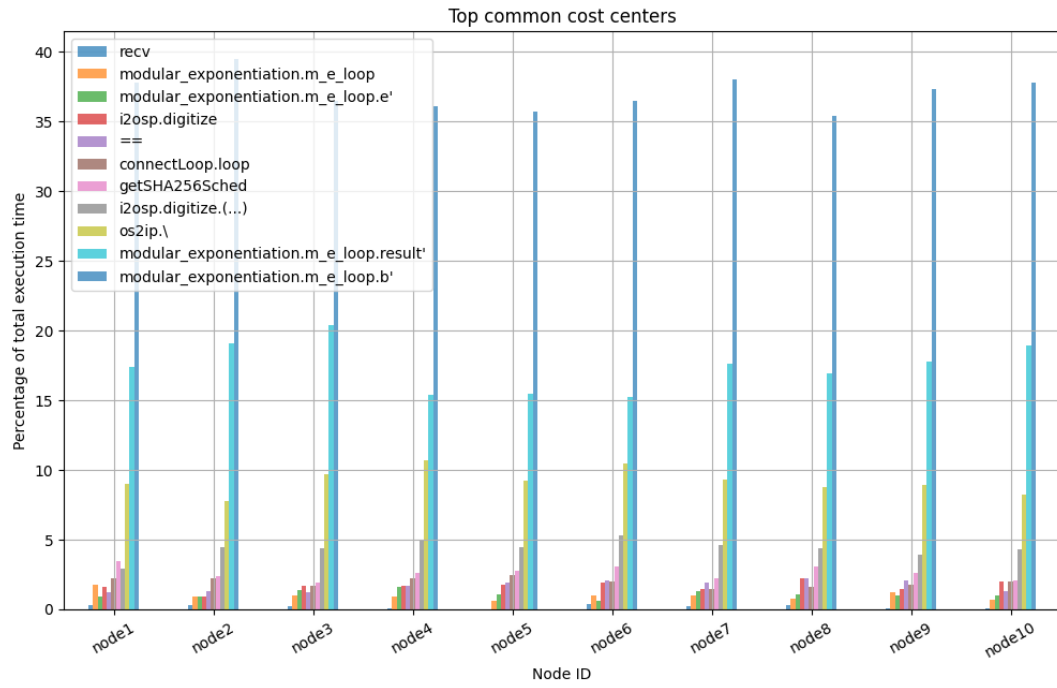
(2)

### 3 Κλιμακωσιμότητα του συστήματος

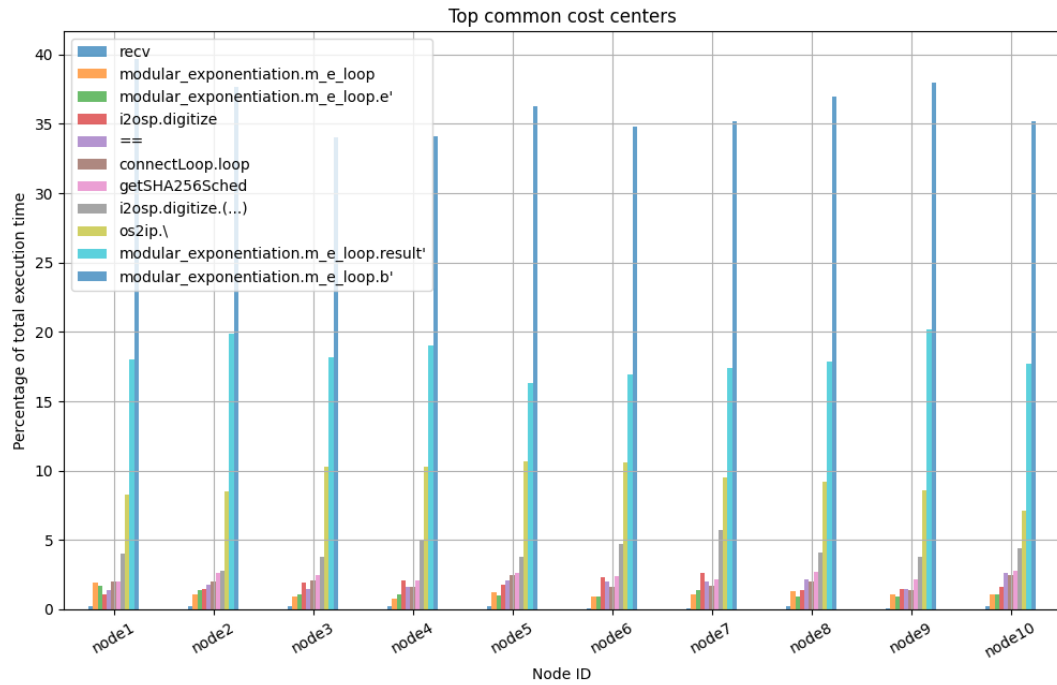
Στα γραφήματα 3 φαίνονται τα πιο χρονοβόρα κομμάτια του κώδικα για κάθε πείραμα κλιμακωσιμότητας. Φαίνεται ότι αυτά είναι τα ίδια με τα πιο χρονοβόρα κομμάτια του κώδικα για το πείραμα ρυθμαπόδοσης, με την συνάρτηση `modular_exponentiation` περίπου το 50% του συνολικού χρόνου εκτέλεσης του προγράμματος, αντί για το 40% που ήταν προηγουμένως.



( $\alpha'$ ) capacity=5



( $\beta'$ ) capacity=10



( $\gamma'$ ) capacity=20

Σχήμα 3: Τα πιο χρονοβόρα κομμάτια του κώδικα

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  $10 + 10 \times 100 = 1010$  το οποίο είναι  $\approx 4$  φορές μεγαλύτερο 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'	74	3.2	3.6
node10.prof	validateTransaction	62	1.5	2.1
node10.prof	mint'	12	1.4	1.5
node1.prof	processTXs'	74	2.3	2.5
node1.prof	validateTransaction	62	1.7	1.8
node1.prof	mint'	12	0.6	0.7
node2.prof	processTXs'	74	2.2	2.0
node2.prof	validateTransaction	62	1.6	1.8
node2.prof	mint'	12	0.4	0.2
node3.prof	processTXs'	74	1.9	1.8
node3.prof	validateTransaction	62	1.5	1.8
node3.prof	mint'	12	0.1	0.0
node4.prof	processTXs'	74	2.4	2.6
node4.prof	validateTransaction	62	1.6	1.9
node4.prof	mint'	12	0.7	0.7
node5.prof	processTXs'	74	1.7	2.2
node5.prof	validateTransaction	62	1.2	2.0
node5.prof	mint'	12	0.4	0.2
node6.prof	processTXs'	74	2.1	2.4
node6.prof	validateTransaction	62	1.2	2.0
node6.prof	mint'	12	0.5	0.4
node7.prof	processTXs'	6	0.2	0.2
node7.prof	validateTransaction	5	0.2	0.2
node7.prof	mint'	1	0.0	0.0
node8.prof	processTXs'	6	0.2	0.2
node8.prof	validateTransaction	5	0.2	0.2
node8.prof	mint'	1	0.0	0.0
node9.prof	processTXs'	74	1.7	2.0
node9.prof	validateTransaction	62	1.4	1.9
node9.prof	mint'	12	0.2	0.0

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 και το μέσο block time μπορούν να υπολογιστούν όπως προηγουμένως:

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

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

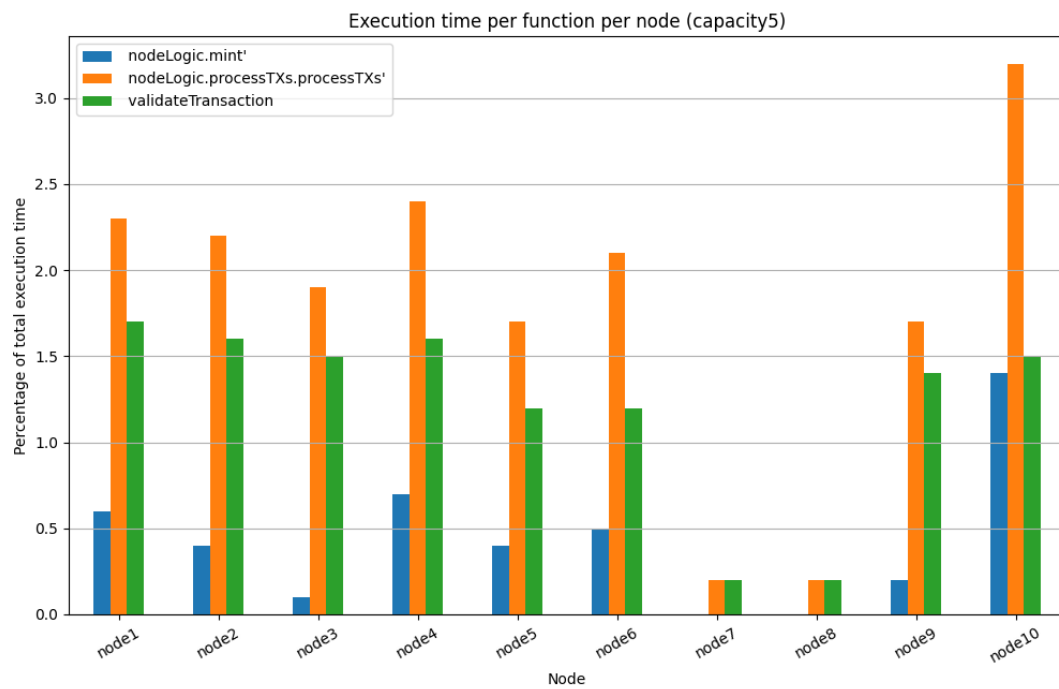
(3)

( $\beta'$ ) capacity=10				( $\gamma'$ ) capacity=20	
File	Function	Calls	TimeInh	Calls	TimeInh
node10.prof	processTXs'	1061	20.7	1036	22.8
node10.prof	validateTransaction	1010	14.5	1010	13.8
node10.prof	mint'	50	1.2	25	1.5
node1.prof	processTXs'	1061	19.3	1036	17.0
node1.prof	validateTransaction	1010	15.1	1010	12.3
node1.prof	mint'	50	1.0	25	1.7
node2.prof	processTXs'	1061	19.6	1036	19.3
node2.prof	validateTransaction	1010	15.0	1010	14.9
node2.prof	mint'	50	1.4	25	0.6
node3.prof	processTXs'	1061	20.1	1036	22.0
node3.prof	validateTransaction	1010	15.4	1010	16.5
node3.prof	mint'	50	1.1	25	1.0
node4.prof	processTXs'	1061	23.6	1036	22.3
node4.prof	validateTransaction	1010	17.2	1010	16.4
node4.prof	mint'	50	0.8	25	1.3
node5.prof	processTXs'	1061	25.3	1036	23.6
node5.prof	validateTransaction	1010	17.2	1010	15.7
node5.prof	mint'	50	1.7	25	1.3
node6.prof	processTXs'	1061	23.7	1036	23.5
node6.prof	validateTransaction	1010	16.9	1010	16.7
node6.prof	mint'	50	0.9	25	1.4
node7.prof	processTXs'	1061	21.6	1036	25.2
node7.prof	validateTransaction	1010	14.4	1010	17.8
node7.prof	mint'	50	2.2	25	1.6
node8.prof	processTXs'	1061	23.6	1034	22.1
node8.prof	validateTransaction	1010	15.6	1010	15.9
node8.prof	mint'	50	1.8	23	0.9
node9.prof	processTXs'	1061	21.8	1033	19.0
node9.prof	validateTransaction	1010	14.3	1010	14.3
node9.prof	mint'	50	1.7	22	0.2

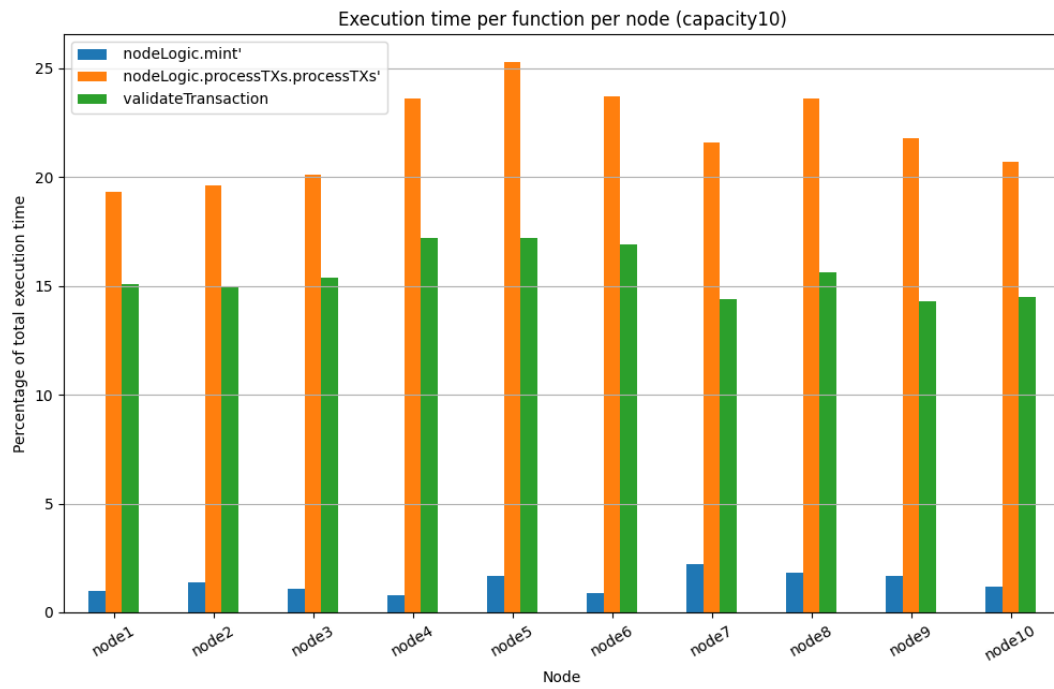
$$\text{Block Time} = \frac{\text{Συνολικός χρόνος}}{\text{Συνολικά blocks}} \Rightarrow \left\{ \begin{array}{l} bt_{\text{capacity}=5} = \frac{0.8 \times 12 + 0.2 \times 1}{10} \\ bt_{\text{capacity}=10} = \frac{50}{10} \\ bt_{\text{capacity}=20} = \frac{0.1 \times 22 + 0.1 \times 23 + 0.8 \times 25}{10}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

$$\begin{aligned} bt_{\text{capacity}=5} &= 0.962 \frac{\text{blocks}}{s} \\ bt_{\text{capacity}=10} &= 5 \frac{\text{blocks}}{s} \\ bt_{\text{capacity}=20} &= 2 \frac{\text{blocks}}{s} \end{aligned}$$

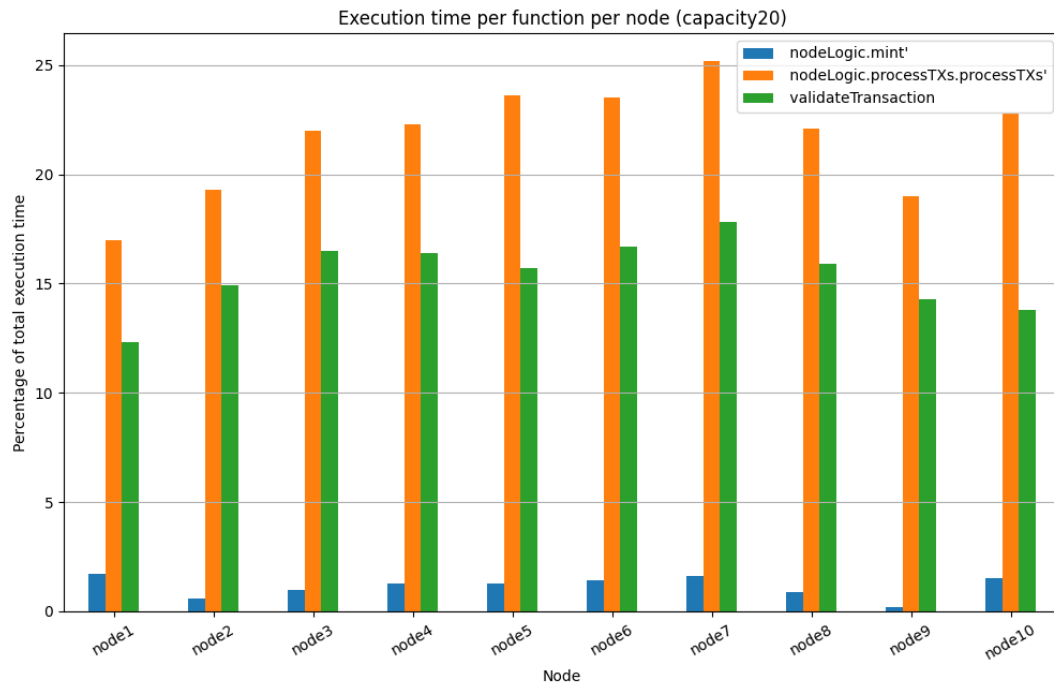
(4)



( $\alpha'$ ) capacity=5



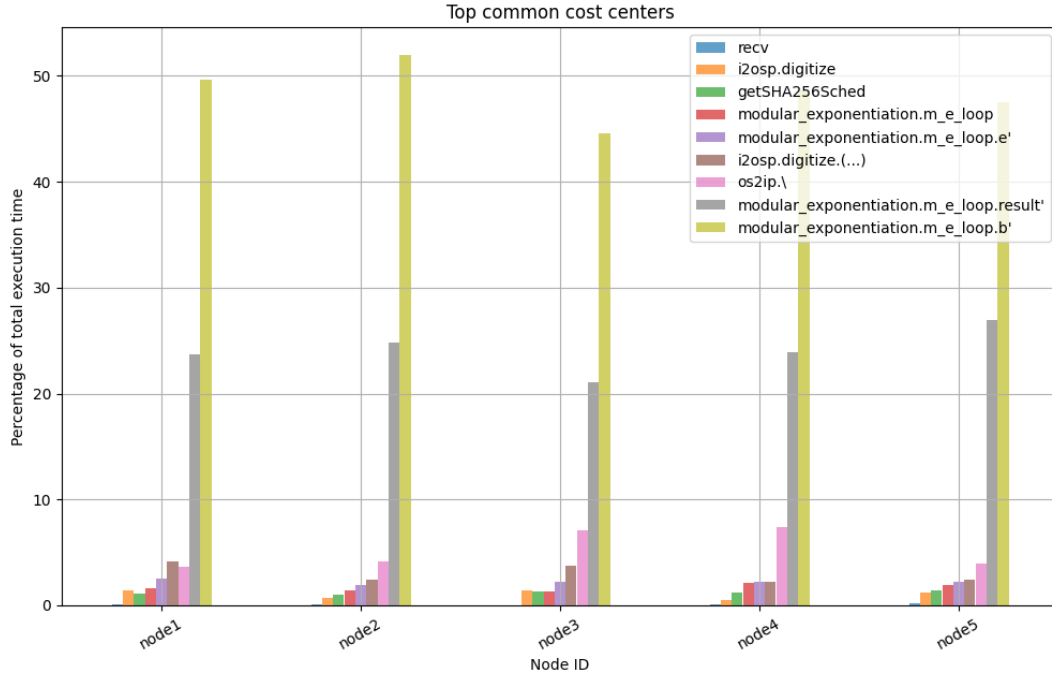
(β') capacity=10



(γ') capacity=20

Σχήμα 4: Ποσοστό χρόνου επί του συνολικού χρόνου εκτέλεσης που λαμβάνει η κάθε συνάρτηση

## 4 Δικαιοσύνη



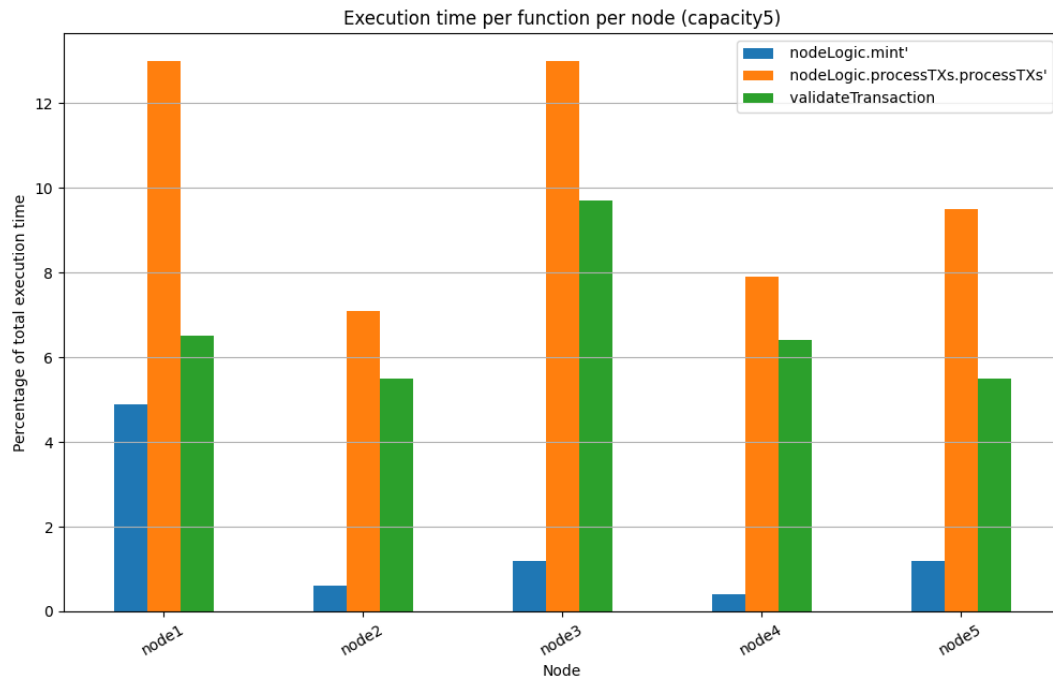
Σχήμα 5: Τα πιο χρονοβόρα κομμάτια του κώδικα capacity=5

Στον πίνακα 3 φαίνονται οι κλήσεις μερικών συναρτήσεων ενδιαφέροντος. Φαίνεται ότι τα πλήθη όλων των κλήσεων είναι ίδια ανά κόμβο, πράγμα που σημαίνει ότι οι κόμβοι εκτελούν τις ίδιες λειτουργίες με την ίδια συχνότητα. Παρόλα αυτά, ο κόμβος με το μεγαλύτερο stake καταναλώνει πολύ περισσότερο χρόνο στην συνάρτηση mint' σε σχέση με τους υπόλοιπους κόμβους, όπως φαίνεται και στο σχήμα 6.

Πίνακας 3: Στατιστικά συναρτήσεων ανά κόμβο capacity=5

File	Function	Calls	TimeInh	MemInh
node1.prof	processTXs'	302	13.0	18.4
node1.prof	validateTransaction	255	6.5	7.6
node1.prof	mint'	46	4.9	10.7
node2.prof	processTXs'	302	7.1	9.5
node2.prof	validateTransaction	255	5.5	8.4
node2.prof	mint'	46	0.6	1.1
node3.prof	processTXs'	301	13.0	11.0
node3.prof	validateTransaction	255	9.7	9.7
node3.prof	mint'	45	1.2	1.3
node4.prof	processTXs'	301	7.9	8.0
node4.prof	validateTransaction	255	6.4	7.6
node4.prof	mint'	45	0.4	0.4
node5.prof	processTXs'	301	9.5	8.7
node5.prof	validateTransaction	255	5.5	7.1
node5.prof	mint'	45	1.2	1.6





Σχήμα 6: Ποσοστό χρόνου επί του συνολικού χρόνου εκτέλεσης που λαμβάνει η κάθε συνάρτηση capacity=5

Στο σχήμα 6 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, παρατηρεί κανείς ότι όντως τα περισσότερα νομίσματα συσσωρεύονται στον κόμβο με το μεγαλύτερο stake.