Εθνικο Μετσοβιο Πολυτεχνειο
Τμημα Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων
Εργαστηριο Λογικης και Επιστημης Υπολογιστων

Προσεγγιστικοί αλγόριθμοι για δίκαιο καταμερισμό υπολογιστικού φόρτου, με εφαρμογές σε κρυπτοσυστήματα κατωφλίου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Κανέλλου Π. Τσίτουρα

Επιβλέπων: Αριστείδης Παγουρτζής
Καθηγητής ΣΗΜΜΥ ΕΜΠ

Αθήνα, 2 Ιουλίου 2025

Εθνικο Μετσοβιο Πολυτεχνειο
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων
Εργαστηριο Λογικης και Επιστημης Υπολογιστων

Προσεγγιστικοί αλγόριθμοι για δίκαιο καταμερισμό υπολογιστικού φόρτου, με εφαρμογές σε κρυπτοσυστήματα κατωφλίου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Κανέλλου Π. Τσίτουρα

Επιβλέπων: Αριστείδης Παγουρτζής
Καθηγητής ΣΗΜΜΥ ΕΜΠ

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 2η Ιουλίου 2025.

.....	.....	.....
Αριστείδης Παγουρτζής	Γεώργιος Στάμου	Νικόλαος Λεονάρδος
Καθηγητής ΣΗΜΜΥ ΕΜΠ	Καθηγητής ΣΗΜΜΥ ΕΜΠ	Επίκουρος Καθηγητής ΣΗΜΜΥ ΕΜΠ

Αθήνα, 2 Ιουλίου 2025.

.....
Κανέλλος Τσίτουρας
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Σε αυτή την εργασία παρουσιάζουμε ένα FPTAS για την k-εκδοχή του προβλήματος k-way Number Partitioning Ratio (k-PART) (το οποίο αποτελεί μια παραλλαγή του κλασικού στη βιβλιογραφία προβλήματος Subset Sum Ratio (SSR)), στο οποίο ζητείται να βρεθεί μια διαμέριση, ενός συνόλου n στοιχείων σε k υποσύνολα, τέτοια ώστε, ο λόγος μεταξύ των αθροισμάτων, του συνόλου με το μεγαλύτερο άθροισμα, προς το σύνολο με το μικρότερο άθροισμα, να είναι ο ελάχιστος δυνατός. Το πρόβλημα αυτό αντιστοιχεί στην ελαχιστοποίηση του λόγου μεγαλύτερου προς μικρότερου αθροίσματος στη γενική οικογένεια προβλημάτων Multiway Number Partitioning. Το FPTAS μας για το πρόβλημα k-PART, το οποίο επιτυγχάνει χρονική πολυπλοκότητα $O(n^{2k}/\varepsilon^{k-1})$, εμφανίζεται σε εργασία της ομάδας μας που έχει υποβληθεί προς δημοσίευση [25].

Στο δεύτερο μέρος της εργασίας μας, παρουσιάζουμε τις εφαρμογές που έχει το FPTAS του πρώτου μέρους, σε ένα (k, t) Κρυπτοσυστημα κατωφλιου, και συγκεκριμένα στο πως μπορεί να χρησιμοποιηθεί για να διαμοιράσει, με προσεγγιστικά βέλτιστο τρόπο, τον υπολογιστικό φόρτο αποκρυπτογράφησης n κρυπτοκειμένων (που έχουν κρυπτογραφηθεί με ένα (k, t) Κρυπτοσυστημα κατωφλιου), σε k οντότητες. Ορίζουμε τυπικά αυτό το νέο πρόβλημα και αφού παρουσιάσουμε τις βασικές αρχές του και την τρέχουσα επιστημονική γνώση σχετικά με αυτό, σχεδιάζουμε και διερευνούμε 3 σενάρια κρυπτογραφικών εφαρμογών γύρω του. Καταλήγουμε με μια μελέτη της εφικτότητας της γενικευμένης χρήσης του FPTAS για το πρόβλημα αυτό, κάτω από τους περιορισμούς στους χειρισμούς των στιγμιοτύπων εισόδου που απαιτούνται στις αντίστοιχες κρυπτογραφικές εφαρμογές.

Λέξεις Κλειδιά: Κρυπτοσυστήματα Κατωφλίου, Multiway Number Partitioning, k-way Number Partitioning Ratio, FPTAS, Προσεγγιστικοί Αλγόριθμοι, Κρυπτογραφία, Άθροισμα Υποσυνόλου

Abstract

In this thesis, we present an FPTAS for the k-version of the k-way Number Partitioning Ratio problem (k-PART) — a variation of the classical Subset Sum Ratio (SSR) problem. The goal of this problem is to partition a set of n elements into k subsets such that the ratio of the largest sumset's sum and the the smallest sumset's sum is minimized. This corresponds to minimizing the maximum-to-minimum sum ratio in the family of Multiway Number Partitioningproblems. Our FPTAS for the k-PART problem, which achieves a time complexity of $O(n^{2k}/\varepsilon^{k-1})$, was published in our joint work [25].

In the second part of the thesis, we present the applications of the FPTAS developed in the first part, in a (k, t)-Threshold Cryptosystem setting — specifically, how it can be used to distribute the computational load of decrypting n ciphertexts (encrypted with a (k, t)-Threshold Cryptosystem scheme), to k entities. We formally define this new problem, and after outlining its core principles and scientific context, we design and explore three cryptographic application scenarios around it. We conclude with a study on the feasibilty of the generalized use of the FPTAS in this problem, under the constraints imposed by the input instance, in these cryptographic applications.

Keywords: Threshold Cryptosystems, Multiway Number Partitioning, k-way Number Partitioning Ratio, FPTAS, Approximation Algorithms, Cryptography, Subset Sum

Ευχαριστίες

Για την εργασία αυτή, αρχικά ευχαριστώ πάρα πολύ τα παιδιά της ομάδας "Subset Sum", με τα οποία περάσαμε πολλές εξαιρετικά παραγωγικές ώρες συζητήσεων και συναντήσεων πριν να καταλήξουμε στην ιδέα και τη δημιουργία του paper "Approximation Schemes for k-Subset Sum Ratio and Multiway Number Partitioning Ratio" ([25]). Συγκεκριμένα, ευχαριστώ τους Σωτήρη Κανελλόπουλο, Γιώργο Μητρόπουλο, Αντώνη Αντωνόπουλο, Χρήστο Περγαμηνέλη, Σταύρο Πετσαλάκη και τους καθηγητές, κύριο Νίκο Λεονάρδο και Άρη Παγουρτζή, καθώς μέσα από τη συνεργασία μας, είδα για πρώτη φορά πως συντονίζεται μια ομάδα για την συγγραφή ενός επιστημονικού άρθρου.

Ευχαριστώ ειδικότερα τον κύριο Παγουρτζή για τη διαρκή καθοδήγηση του και την ανοιχτή σκέψη του στις προτάσεις μου για την ιδέα των κρυπτογραφικών εφαρμογών του FPTAS.

Ευχαριστώ πολύ τους γονείς μου και την οικογένεια μου, για τη διαρκή υποστήριξη και τη βοήθεια τους, σε όλη την εκπαιδευτική μου πορεία, από την αρχή, μέχρι την εισαγωγή μου στο Πολυτεχνείο και την ολοκλήρωση των σπουδών μου. Σας ευχαριστώ πολύ για όλα.

Τέλος, ευχαριστώ την κοπέλα μου, Ηλιάνα, για τη διαρκή στήριξη της και την επιμονή της στο να προσπαθήσω να βρω μια εμπνευσμένη ιδέα για την παρούσα εργασία. Χωρίς τη βοήθεια της, ούτε αυτή η εργασία, ούτε εγώ, θα φτάναμε εδώ που είμαστε σήμερα.

Περιεχόμενα

Κατάλογος πινάκων

Κεφάλαιο 1

Εισαγωγή

Τα πρόβλημα Multiway Number Partitioning, είναι θεμελιώδες στη συνδυαστική βελτιστοποίηση, με πολυάριθμες εφαρμογές στη δίκαιη κατανομή πόρων, την κρυπτογραφία και τον χρονοπρογραμματισμό. Το πρόβλημα έχει τις ρίζες του στο κλασικό πρόβλημα Subset Sum, και σε κλασικές παραλαγές του, όπως το Equal Subset Sum(ESS) [47], το οποίο ζητά δύο ξένα υποσύνολα ενός δοσμένου συνόλου με ίσα αθροίσματα, ενώ η βελτιστοποιημένη εκδοχή του, Subset Sum Ratio (SSR) [47, 3], στοχεύει στην ελαχιστοποίηση του λόγου αθροισμάτων δύο ξένων υποσυνόλων. Αυτά τα NP-δύσκολα προβλήματα προκύπτουν σε πεδία όπως η βιοπληροφορική [16, 15], η θεωρία παιγνίων και η οικονομία [32], καθώς και η κρυπτογραφία [48], μεταξύ άλλων. Μια ειδική περίπτωση του ESS ανήκει στην κλάση PPP [38], υποκλάση της TFNP, γεγονός που ενισχύει το θεωρητικό ενδιαφέρον για αυτά τα προβλήματα.

Σε αυτήν την εργασία ξεκινάμε με τη μελέτη μιας παραλλαγής με περισσότερα υποσύνολα, συγκεκριμένα με το πρόβλημα και k-way Number Partitioning Ratio (k-PART)[1]. Το πρόβλημα αυτό έχει στενή σχέση με το πρόβλημα, k-SSR, (όπου στόχος είναι να βρεθούν k ξένα υποσύνολα που να ελαχιστοποιούν τον λόγο *μέγιστου προς ελάχιστο* των αθροισμάτων τους), καθώς ορίζεται με ακριβώς τον ίδιο τρόπο, με την επιπρόσθετη απαίτηση, τα υποσύνολα να σχηματίζουν διάσπαση του αρχικού συνόλου. Δεδομένου ότι το πρόβλημα είναι NP-δύσκολο, θα εξετάσουμε έναν προσεγγιστικό αλγόριθμο για την επίλυση του και πιο συγκεκριμένα, θα παρουσιάσουμε ένα πλήρως πολυωνυμικό σχήμα προσέγγισης (FPTAS) για το πρόβλημα, όπως αυτό παρουσιάστηκε στο συλλογικό μας έργο [25].

Το FPTAS μας βρίσκει εφαρμογές, μεταξύ άλλων, στην ανάθεση εργασιών σε k επεξεργαστές (π.χ. [41, 18, 22, 17]), καθώς και στη *δίκαιη κατανομή μη διαιρετών αγαθών*, και ειδικότερα στη μελέτη του *envy* σε διακριτές ρυθμίσεις (π.χ. [32, 37, 30] και πιο πρόσφατα [1, 45, 49]). Ειδικά, το πρόβλημα k-PART είναι ισοδύναμο με το πρόβλημα Minimum Envy-Ratio [32] με ταυτόσημες συναρτήσεις αποτίμησης, για το οποίο επιτυγχάνουμε σημαντική βελτίωση σε σχέση με προηγούμενες προσεγγίσεις [37].

Η εφαρμογή στην οποία θα εστιάσουμε, μελετά τον δίκαιο διαμοιρασμό του Υπολογιστικού Φόρτου Αποκρυπτογράφησης σε (k, t) Κρυπτοσυστηματα κατωφλιου. Τα κρυπτοσυστήματα κατωφλίου, έχουν μελετηθεί εκτενώς στη βιβλιογραφία ([23, 6]) και με αφετηρία αυτά, θα ορίσουμε ένα δικό μας πρόβλημα και θα παρουσιάσουμε πολλαπλά σενάρια, στα οποία μπορούμε να χρησιμοποιήσουμε το FPTAS που θα μελετήσουμε, για να πετύχουμε τον συντονισμό μιας ομάδας οντοτήτων για την επίτευξη κρυπτογραφικών στόχων.

Σχετική έρευνα Από την πρωτοποριακή εργασία του Bellman για το Subset Sum [4], δεν υπήρξε σημαντική πρόοδος μέχρι την πρόσφατη έξαρση ενδιαφέροντος που οδήγησε

[1] Το πρόβλημα αυτό αναφέρεται συνήθως ως "multiway", αλλά εμείς χρησιμοποιούμε τον όρο "k-way" για να τονίσουμε ότι το k είναι σταθερό.

σε διάφορους ταχύτερους ψευδοπολυωνυμικούς αλγορίθμους [28, 7, 24, 13] και FPTASs [27, 20].More actions

Για το πρόβλημα Multiway Number Partitioning, οι εξελίξεις περιλαμβάνουν αρκετά (πολύ) πρόσφατα FPTASs [35, 9, 12], με την πιο πρόσφατη εργασία των Chen et al. [14] να επιτυγχάνει ένα FPTAS με χρόνο $\widetilde{O}(n+1/\varepsilon)$, σχεδόν βέλτιστο υπό την Υπόθεση Ισχυρού Εκθετικού Χρόνου. Στο Multiway Number Partitioning [5, 43, 29, 40], διαφορετικές αντικειμενικές συναρτήσεις καθορίζουν διαφορετικές προσεγγίσεις για τη διανομή στοιχείων σε υποσύνολα, τις οποίες θα δούμε περιληπτικά στο Παράρτημα Γ′, μαζί με εφαρμογές για κάθε παραλλαγή. Ο Sahni παρουσιάζει ένα FPTAS για το Multiway Number Partitioning στο [41] που ελαχιστοποιεί το μέγιστο άθροισμα. Ωστόσο, η τεχνική του δεν φαίνεται να εφαρμόζεται για την ελαχιστοποίηση του λόγου, στην οποία εστιάζουμε εμείς.

Στο [32], οι Lipton et al. παρουσιάζουν ένα PTAS για το πρόβλημα Minimum Envy-Ratio όταν όλοι οι πράκτορες έχουν την ίδια συνάρτηση αξιολόγησης (και ο αριθμός των πρακτόρων είναι μέρος της εισόδου), ενώ στο [37] οι Nguyen και Rothe παρουσιάζουν ένα FPTAS με χρόνο $O(n^{4k^2+1}/\varepsilon^{2k^2})$ για σταθερό αριθμό k πρακτόρων με δυνητικά διαφορετικές αξιολογήσεις.

Σχόλιο: Το FPTAS μας για το k-PART ([25]) αποτελεί βελτίωση σε σχέση με αυτά τα αποτελέσματα όταν το πρόβλημα περιορίζεται σε σταθερό αριθμό πρακτόρων και ίδιες συναρτήσεις αξιολόγησης, με πολυπλοκότητα χρόνου $O(n^{2k}/\varepsilon^{k-1})$.

Για την εφαρμογή μας για το δίκαιο καταμερισμό του Υπολογιστικού Φόρτου Αποκρυπτογράφησης σε (k,t) Κρυπτοσυστηματα κατωφλιου, υπάρχουν μελέτες σχετικά με μια γενική παρουσίαση του προβλήματος ([23]), ωστόσο η δουλειά μας δεν φαίνεται να εμφανίζεται κάπου αλλού και η μελέτη που προτείνουμε, με τη χρήση του FPTAS δεν έχει ξαναπαρουσιαστεί.

Η συμβολή μας Σε αυτή την εργασία, περιγράφουμε και αναλύουμε σε μεγαλύτερο βάθος, ένα FPTAS για τα πρόβλημα k-PART, με χρόνο εκτέλεσης $O(n^{2k}/\varepsilon^{k-1})$ για σταθερό k, το οποίο παρουσιάστηκε για πρώτη φορά στο [25] και βασίζεται σε μεθόδους που αναπτύχθηκαν για το SSR [36, 33, 8], εισάγοντας όμως νέες στρατηγικές για την αντιμετώπιση της ρύθμισης με πολλά υποσύνολα και του περιορισμού διαμέρισης. Το αποτέλεσμα αυτό αποτελεί σημαντική βελτίωση σε σύγκριση με το FPTAS του [37]. Σημειώνουμε ότι ο λόγος μέγιστου προς ελάχιστο αθροίσματος είναι η μοναδική αντικειμενική συνάρτηση που εξετάζεται στη βιβλιογραφία η οποία τόσο επιδέχεται FPTAS όσο και έχει σημασία για τη μελέτη του *envy* (βλ. Παράρτημα Γ′ για σύγκριση μεταξύ των συναρτήσεων στόχου). Μελετάμε στη συνέχεια τη χρήση του σε πολλαπλές εφαρμογές κρυπτοσυστημάτων κατωφλίου, και αφού παρουσιάσουμε την ευκολία της εφαρμογής του προβλήματος k-PART, σε σενάρια δίκαιου διαμοιρασμού υπολογιστικών πόρων, παρουσιάζουμε αλγορίθμους που επιτυγχάνουν προσεγγιστικά βέλτιστους διαμοιρασμούς πόρων σε σχέση με τα αποτελέσματα του FPTAS.

Στην Ενότητα 3 ορίζουμε μια περιορισμένη εκδοχή του k-PART, στην οποία απαιτείται κάθε υποσύνολο να περιέχει ένα επαρκώς μεγάλο στοιχείο και εντοπίζουμε μια *τέλεια* παραμετροποίηση του περιορισμού που εξασφαλίζει την ύπαρξη μιας καλώς συμπεριφερόμενης βέλτιστης λύσης. Αν και αυτό δεν οδηγεί σε ψευδοπολυωνυμικό αλγόριθμο (για την αντίστοιχη περιορισμένη περίπτωση), ο συνδυασμός αυτής της ιδιότητας με στρογγυλοποίηση και προσέγγιση οδηγεί σε ένα FPTAS για το k-PART.

Τέλος στην ενότητα 4, παρουσιάζουμε τρείς εφαρμογές του FPTAS σε διαφορετικά σενάρια κρυπτοσυστημάτων κατωφλίου, με μεγαλύτερη έμφαση να δίνεται στην υποενότητα 4.3, στην οποία διερευνάμε την ύπαρξη μιας 1-1 επέκτασης τους FPTAS της ενότητας 3, στο σενάριο του κρυπτοσυστήματος κατωφλίου μας.

Κεφάλαιο 2

Προκαταρκτικά

Έστω $[n] = \{1, \ldots, n\}$ οι αριθμοί από 1 έως έναν θετικό ακέραιο n. Υποθέτουμε ότι η είσοδος των προβλημάτων μας, θα είναι ένα ταξινομημένο[1] πολυσύνολο $A = \{a_1, \ldots, a_n\}$, δηλαδή $a_1 \leq \ldots \leq a_n$.

Ορισμός 1 (Άθροισμα υποσυνόλου δεικτών). Δεδομένου ενός πολυσυνόλου $A = \{a_1, \ldots, a_n\}$ από n θετικούς ακέραιους και ενός συνόλου δεικτών $S \subseteq [n]$ ορίζουμε:

$$\Sigma(S, A) = \sum_{i \in S} a_i$$

το άθροισμα των στοιχείων του S.

Ορισμός 2 (Μέγιστο και ελάχιστο k υποσυνόλων). Έστω $A = \{a_1, \ldots, a_n\}$ ένα πολυσύνολο από n θετικούς ακέραιους και $S_1, \ldots, S_k$, k ξένα μεταξύ τους υποσύνολα του $[n]$. Ορίζουμε τα μέγιστα και ελάχιστα αθροίσματα αυτών των συνόλων, που παράγονται από στοιχεία του A, ως εξής:

$$M(S_1, \ldots, S_k, A) = \max_{i \in [k]} \Sigma(S_i, A) \quad \text{και} \quad m(S_1, \ldots, S_k, A) = \min_{i \in [k]} \Sigma(S_i, A)$$

Ορισμός 3 (Λόγος k υποσυνόλων). Έστω $A = \{a_1, \ldots, a_n\}$ ένα πολυσύνολο από n θετικούς ακέραιους και $S_1, \ldots, S_k$, k ξένα μεταξύ τους υποσύνολα του $[n]$. Ορίζουμε τον λόγο αυτών των k υποσυνόλων στο A ως εξής:

$$\mathcal{R}(S_1, \ldots, S_k, A) = \begin{cases} \frac{M(S_1, \ldots, S_k, A)}{m(S_1, \ldots, S_k, A)} & \text{αν } m(S_1, \ldots, S_k, A) > 0 \\ +\infty & \text{αν } m(S_1, \ldots, S_k, A) = 0 \end{cases}$$

Ορίζουμε το πρόβλημα k-way Number Partitioning Ratio (k-PART) ως εξής:

Ορισμός 4 (k-PART). Δεδομένου ενός πολυσυνόλου $A = \{a_1, \ldots, a_n\}$ από n θετικούς ακέραιους, να βρεθούν k ξένα μεταξύ τους υποσύνολα $S_1, \ldots, S_k$ του $[n]$, με $\bigcup_{i=1}^{k} S_i = [n]$, (δηλαδή μια διαμέριση του A), τέτοια ώστε ο λόγος $\mathcal{R}(S_1, \ldots, S_k, A)$ να είναι ο ελάχιστος δυνατός.

[1] Αν η είσοδος δεν είναι ταξινομημένη, απαιτείται επιπλέον χρόνος $O(n \cdot logn)$. Αυτό δεν επηρεάζει τη χρονική πολυπλοκότητα οποιουδήποτε από τους αλγορίθμους μας, υποθέτοντας ότι ο συμβολισμός $\widetilde{O}$ καλύπτει επίσης πολυλογαριθμικούς παράγοντες.

Ας δούμε τώρα τους ορισμούς για το κρυπτογραφικό μέρος της εργασίας μας. Ξεκινώντας από τα βασικά, είναι ευρέως γνωστό πως σε όλα τα κρυπτοσυστήματα δημοσίου κλειδιού, μια οντότητα μπορεί να χρησιμοποιεί το ιδιωτικό της κλειδί, τόσο για να αποκρυπτογραφεί, όσο και για να υπογράφει μηνύματα, γεγονός που σε αρκετές περιπτώσεις, παρέχει στις οντότητες μεγαλύτερη ισχύ από την επιθυμητή.

Τα κρυπτοσυστήματα κατωφλίου, είναι μια ειδική κατηγορία κρυπτοσυστημάτων δημοσίου κλειδιού, η οποία ορίζεται ως εξής:

Ορισμός 5 ((k,t) Κρυπτοσύστημα κατωφλιού). Ένα (k,t)-Κρυπτοσύστημα κατωφλίου ((k,t) Threshold Cryptosystem), είναι μια ειδική κατηγορία κρυπτοσυστήματος δημοσίου κλειδιού, στην οποία το ιδιωτικό κλειδί θεωρείται ως ένα μυστικό που θα διαχωριστεί σε k κομμάτια και θα διαμοιραστεί σε ένα ισάριθμο πλήθος οντοτήτων με αντικρουόμενα συμφέροντα και απαιτείται η συνεργασία τουλάχιστον t εξ' αυτών για την αποκρυπτογράφηση οποιουδήποτε κρυπτοκειμένου.

Παρατήρηση 1. Καμία από τις οντότητες δεν πρέπει να έχει την ικανότητα, χρησιμοποιώντας μόνο το μέρος του ιδιωτικού κλειδιού που της έχει ανατεθεί, να μπορεί να αποκρυπτογραφήσει από μόνη της οποιοδήποτε κρυπτοκείμενο.

Παρατήρηση 2. Τα (k,t) Threshold Cryptosystem βασίζονται στο σχήμα κατωφλίου, τον ορισμό του οποίου παραθέτουμε στο Α΄.

Η κρυπτογράφηση λειτουργεί όπως στα περισσότερα κρυπτοσυστήματα δημοσίου κλειδιού, χρησιμοποιώντας ένα δημόσιο κλειδί, για το οποίο υπάρχει το αντίστοιχο ιδιωτικό του κλειδί, το οποίο θα διαμοιραστεί σε τμήματα στις οντότητες. Σε ένα τέτοιο πρωτόκολλο, εκτός από τον αριθμό k, που είναι το πλήθος των οντοτήτων, υπάρχει και ο αριθμός κατωφλίου (threshold) t. Η ιδιαιτερότητα των κρυπτοσυστημάτων κατωφλίου, όπως φαίνεται και από τον ορισμό τους, έγκειται στο γεγονός πως για να πραγματοποιηθεί η αποκρυπτογράφηση οποιουδήποτε κρυπτοκειμένου, απαιτείται η συνεργασία τουλάχιστον t από τις k οντότητες, κάτι που ισοδυναμεί με το ότι, αν συνεργαστεί οποιοδήποτε πλήθος από $\leq t$ οντότητες, δεν είναι δυνατή η ανάκτηση του απλού κειμένου.

Ένα κρυπτοσύστημα κατωφλίου αποτελείται από τις εξής φάσεις:

- Δημιουργία Κλειδιών
- Κρυπτογράφηση
- Αποκρυπτογράφηση Μεριδίων
- Συνδυασμός

Παρόλο που εκ πρώτης όψεως, η τεχνική που χρησιμοποιούν τα (k,t) Κρυπτοσυστηματα κατωφλιου, με τον διαμοιρασμό του μυστικού ιδιωτικού κλειδιού, φαίνεται αρκετό για να επιτύχουμε τον περιορισμό της ισχύος των οντοτήτων που επιθυμούμε, αν το σκεφτούμε λίγο παραπάνω, παρουσιάζεται το εξής πρόβλημα: εάν t οντότητες συνεργαστούν για την αποκρυπτογράφηση του μηνύματος και επανασυναρμολογήσουν το κλειδί (μέσω των t κομματιών του), στο εξής όλες οι οντότητες θα έχουν το κλειδί και όλα τα επόμενα κρυπτοκείμενα, που έχουν κρυπτογραφηθεί με το ίδιο πρωτόκολλο, θα μπορούν να αποκρυπτογραφηθούν, κάνοντας πλέον το κρυπτοσύστημα να έχει χάσει την ιδιαιτερότητα του. Άρα χρειαζόμαστε μια τροποποίηση στο σύστημα, με την οποία ακόμα και με τη συνεργασία t οντοτήτων, το κλειδί παραμένει άγνωστο και δεν ανακατασκευάζεται ποτέ πλήρως. Έτσι το ιδιωτικό κλειδί παραμένει μυστικό, και το κρυπτοσύστημα μπορεί να επαναχρησιμοποιηθεί.

Στην βιβλιογραφία, έχουν παρουσιαστεί πολλά τέτοια (k,t) Κρυπτοσυστηματα κατωφλιου, που εμπεριέχουν και την παραπάνω ιδιότητα που αναφέραμε. Ένα από αυτά, το οποίο θα

επιλέξουμε για να χρησιμοποιήσουμε και εμείς για την κρυπτογράφηση των κρυπτοκειμένων μας, είναι το (k, t) Threshold ElGamal, το οποίο προκύπτει από τον συνδυασμό του ElGamal με Σχημα Μυστικης Διανομης του Shamir. Το κλειδί μοιράζεται σε k οντότητες, από τις οποίες πρέπει να συνεργαστούν t για την αποκρυπτογράφηση ενός μηνύματος. Κατά τη διάρκεια της αποκρυπτογράφησης δεν αναδημιουργείται το ιδιωτικό κλειδί, κάτι που διατηρεί την ασφάλεια του συστήματος.

Η διαδικασία που εκτελείται στο (k, t) Threshold ElGamal είναι η εξής:

- **Δημιουργία Κλειδιών** Ένας έμπιστος διανομέας χωρίζει το κλειδί $x \in_R \mathbb{Z}_q$ σε k μερίδια $\{x_i = P(i)\}_{i=1}^{k}$ με το σχήμα του Shamir και τα διανέμει στους οντότητες. Επίσης υπολογίζει και δημοσιοποιεί το δημόσιο κλειδί $y = g^x$.

- **Κρυπτογράφηση** Λειτουργεί όπως στο απλό ElGamal και παράγει κρυπτοκείμενα της μορφής $c = (G, M)$.

- **Αποκρυπτογράφηση Μεριδίων** Κάθε οντότητα εξάγει το πρώτο μέρος του κρυπτοκειμένου και υπολογίζει $c_i = G^{x_i} \mod p$ το οποίο και δημοσιοποιεί.

- **Συνδυασμός** Συνδυάζονται t αποκρυπτογραφημένα μερίδια, τα οποία πολλαπλασιάζονται:

$$\hat{c} = \prod_{i=1}^{t} c_i^{k_i} \mod p$$

  όπου $k_i = \prod_{j \neq i} \frac{j}{j-i} \mod q$ είναι οι συντελεστές Lagrange. Τότε, $\hat{c} = G^x$.

- **Αποκρυπτογράφηση** Η αποκρυπτογράφηση γίνεται και πάλι όπως στο κλασικό ElGamal ως $M/\hat{c}$.

Σύμφωνα και με το Α΄, το παραπάνω κρυπτοσύστημα προϋποθέτει την ύπαρξη ενός έμπιστου διανομέα. Παρόλα αυτά, σε πολλές εφαρμογές, μπορεί να μην επιθυμούμε την ύπαρξη έμπιστου διανομέα, μπορούμε να χρησιμοποιήσουμε την κατασκευή που περιγράφεται στο [39] και παραθέτουμε στο Β΄

Τέλος, θα ορίσουμε το πρόβλημα που θα συναντήσουμε στην παράγραφο 4 και που αποτελεί το σημείο εστίασης της εργασίας μας.

Ορισμός 6 (Δίκαιος διαμοιρασμός υπολογιστικού φόρτου αποκρυπτογράφησης). Στο πρόβλημα του Δίκαιου Διαμοιρασμού Υπολογιστικού Φόρτου Αποκρυπτογράφησης, έχουμε ένα σύνολο κρυπτοκειμένων $C = \{c_1, \ldots, c_n\}$, και ένα σύνολο κόστων αποκρυπτογράφησης $A = \{a_1, \ldots, a_n\}$, και στόχος μας είναι να διαμερίσουμε t αντίγραφα καθενός κρυπτοκειμένου c_i, σε k υποσύνολα $S = \{S_1, \ldots, S_k\}$, με όσο το δυνατόν πιο κοντινά αθροίσματα, δηλαδή τέτοια ώστε ο λόγος $\frac{\max_{j \in [k]} \Sigma(S,A)}{\min_{j \in [k]} \Sigma(S,A)} = \mathcal{R}(S,A)$ να είναι ο ελάχιστος δυνατός.

Παρατηρώντας λίγο πιο προσεκτικά τον παραπάνω ορισμό, μπορούμε να δούμε ότι το πρόβλημα αυτό είναι μια άμεση εφαρμογή του k-way Number Partitioning Ratio, και πάνω σε αυτή την παρατήρηση θα βασίσουμε όλες τις εφαρμογές που θα παρουσιάσουμε στην ενότητα 4.

Κεφάλαιο 3

Ένα FPTAS για το k-way Number Partitioning Ratio

Σε αυτή την ενότητα παρουσιάζουμε το FPTAS για το k-way Number Partitioning Ratio (k-PART), χρόνου $O(n^{2k}/\varepsilon^{k-1})$, όπως αυτό παρουσιάστηκε και στο 4ο μέρος του [25]. Στην επόμενη ενότητα, θα δούμε πως αυτό το FPTAS για το k-PART, μπορεί να χρησιμοποιηθεί αποδοτικά σε κρυπτογραφικές εφαρμογές κρυπτοσυστήματος κατωφλίου.

Σύμφωνα με τον ορισμό 4, κάθε στοιχείο του A, πρέπει να ανήκει σε ένα υποσύνολο S_i (για να έχουμε διαμέριση) και κανένα στοιχείο δεν πρέπει να περισσεύει. Για τη συνέχεια, θα ορίσουμε μια ειδική περιορισμένη εκδοχή του k-PART, ως εξής:

Ορισμός 7 (k-PART$_{\mathrm{R}}$). Δοσμένου ενός ταξινομημένου πολυσυνόλου $A = \{a_1, \ldots, a_n\}$ από n θετικούς ακέραιους, ενός ακεραίου k και ενός ακεραίου p, τέτοιο ώστε, $1 \leq p \leq n - k + 1$, ζητούνται k ξένα υποσύνολα $S_1, \ldots, S_k$ του $[n]$ για τα οποία ισχύει ότι

- $\bigcup_{i=1}^{k} S_i = [n]$ (δηλαδή το S είναι μια διαμέριση του $[n]$)
- $\max(S_1) = p'$
- $\max(S_i) > p$ για $1 < i \leq k$

έτσι ώστε να ελαχιστοποιείται η $\mathcal{R}(S_1, \ldots, S_k, A)$.

Για να βρούμε ένα FPTAS για το k-PART, θα χρησιμοποιήσουμε έναν αλγόριθμο που δίνει λύσεις για το πρόβλημα k-PART$_{\mathrm{R}}$, ως υπορουτίνα για τον αλγόριθμο που θα επιλύει το πλήρες πρόβλημα k-PART. Συγκεκριμένα, θα δείξουμε ότι υπάρχει ένα "τέλειο" p, p^*, τέτοιο ώστε η βέλτιστη λύση $S^* : (A, p^*)$, που αντιστοιχεί στο p^*, συμφωνεί με κάποιες ιδιότητες που απαιτούνται για τη χρήση του αλγορίθμου για το k-PART$_{\mathrm{R}}$, ως υπορουτίνα στο FPTAS του k-PART, που θα κατασκευάσουμε. Αρκεί να διασφαλίσουμε ότι για το p^*, ο αλγόριθμος θα εξετάσει το S^* (ή άλλη ισοδύναμη λύση) ως υποψήφια λύση για το στρογγυλοποιημένο (rounded) στιγμιότυπο του k-PART$_{\mathrm{R}}$.

3.1 Ιδιότητες στιγμιοτύπων k-PART πριν τη στρογγυλοποίηση

Ορισμός 8 (Τέλειο p). Έστω $A = \{a_1, \ldots, a_n\}$ είσοδος του προβλήματος k-PART. Θα λέμε πως ο ακέραιος $p : 1 \leq p \leq n - k + 1$ είναι *τέλειος* για το A αν η βέλτιστη λύση S^* του στιγμιοτύπου (A, p) του k-PART$_{\mathrm{R}}$ (και οι ισοδύναμες της), έχει τον ίδιο λόγο με μια βέλτιστη λύση για το στιγμιότυπο A του k-PART.

Για να χρησιμοποιήσουμε τον αλγόριθμο για το k-PART$_{\mathrm{R}}$, ως υπορουτίνα για το k-PART θα πρέπει να αποδείξουμε πως κάτι τέτοιο μας είναι πράγματι χρήσιμο και παράγει σωστά αποτελέσματα. Για τον σκοπό αυτό, Θα ορίσουμε τα παρακάτω δύο μεγέθη, που θα χρησιμοποιήσουμε στην απόδειξη ορθότητας.

Έστω, $Q = \sum_{i=1}^{p} a_i$, το άθροισμα των p μικρότερων στοιχείων του ταξινομημένου πολυσυνόλου A και έστω $q = \max\{i \mid a_i \leq Q\}$, το μέγιστο στοιχείο με άθροισμα $\leq Q$.

Θεώρημα 1. *Δοθέντος ενός στιγμιοτύπου A, του k-PART, υπάρχει ένα τέλειο p για το A, για το οποίο υπάρχει μία βέλτιστη λύση του στιγμιοτύπου (A, p), του k-PART$_{\mathrm{R}}$, που ικανοποιεί τις ακόλουθες ιδιότητες:*

1. *Για όλα τα υποσύνολα S_i του S, που περιέχουν μόνο στοιχεία $j \leq q$, ισχύει ότι $\Sigma(S_i, A) < 2Q$.*
2. *Όλα τα στοιχεία $j > q$ ανήκουν σε μονοσύνολα.*

Απόδειξη. Για μια αυθαίρετη τιμή του p, έστω $S = (S_1, \ldots, S_k)$ μια αυθαίρετη εφικτή λύση για το στιγμιότυπο (A, p), του k-PART$_{\mathrm{R}}$. Αν το S παραβιάζει την ιδιότητα 2, δηλαδή υπάρχει σύνολο S_i της S, με $|S_i| > 1$, το οποίο περιέχει στοιχείο $j > q$, τότε προχωράμε ως εξής: έστω S_m ένα σύνολο με ελάχιστο άθροισμα[1] και u το μικρότερο στοιχείο του S_i. Ισχύουν οι παρακάτω ανισότητες:

$$m(S, A) = \Sigma(S_m, A) \leq Q < \Sigma(S_i \setminus \{u\}, A)$$

$$\Sigma(S_m \cup \{u\}, A) \leq Q + a_u < \Sigma(S_i, A) \leq M(S, A)$$

από τις οποίες μπορούμε να συμπεράνουμε ότι, μετακινώντας το u από το S_i στο S_m, το ελάχιστο άθροισμα S_{min} δεν μειώνεται (αφού $\Sigma(S_m \cup \{u\}, A) = S_m + a_u > S_m$) και το μέγιστο άθροισμα S_{max} δεν αυξάνεται (αφού $\Sigma(S_m \cup \{u\}, A) < Q$ και υπάρχουν σύνολα, με άθροισμα $> Q$).

Άρα, μετακινώντας το u από το S_i στο S_m, προκύπτει μια νέα λύση S' με $\mathcal{R}(S', A) \leq \mathcal{R}(S, A)$. Θα πρέπει ωστόσο να λάβουμε υπόψη την περίπτωση που το S_m είναι το S_1 και η μετακίνηση του u σε αυτό, καθιστά την S', μη εφικτή λύση για το k-PART$_{\mathrm{R}}$. Αν κάτι τέτοιο συμβαίνει στην S', ορίζουμε ως p', το ελάχιστο μεταξύ των μεγιστικών στοιχείων των k συνόλων της νέας λύσης S' και ως S_1, το σύνολο που περιέχει το p'. Τότε, η S' είναι μια εφικτή λύση για το στιγμιότυπο (A, p') του k-PART$_{\mathrm{R}}$.

Αν τώρα το S' παραβιάζει ακόμη την ιδιότητα 2, εφαρμόζουμε και πάλι την ίδια διαδικασία της μετακίνησης, που περιγράψαμε παραπάνω, για τα νέα μεγέθη $Q' = \sum_{i=1}^{p'} a_i$ και $q' = \max\{i \mid a_i \leq Q'\}$. Εκτελώντας επαναληπτικά αυτή τη διαδικασία, το p δεν μειώνεται και ο λόγος της λύσης $\mathcal{R}(S', A)$ δεν αυξάνεται, όπως είδαμε και στην ανάλυση μια μεμονωμένης μετακίνησης. Παρατηρώντας ότι το p δεν μπορεί να υπερβεί το $n - k + 1$ (αφού είναι $p \leq n - k + 1$, εξ ορισμού), κατά την εκτέλεση αυτής της επαναληπτικής διαδικασίας, βλέπουμε πως κάποια στιγμή το p θα σταματήσει να αυξάνεται. Έστω p_f, αυτή η τελική τιμή "σταθεροποίησης" του p και έστω Q_f και q_f, οι αντίστοιχες τιμές των Q και q. Επαναλαμβάνοντας λοιπόν, (αυτή τη μετακίνηση και τους επανορισμούς, του p σε p', κλπ), κάποια στιγμή θα καταλήξουμε σε μια εφικτή λύση S'' του στιγμιοτύπου (A, p_f), του k-PART$_{\mathrm{R}}$.

Αν τώρα η εφικτή λύση S'' του (A, p_f), που σχηματίσαμε (και ικανοποιεί την ιδιότητα 2), παραβιάζει την ιδιότητα 1, δηλαδή υπάρχει σύνολο S_i στην S'', που αποτελείται μόνο από στοιχεία $j \leq q_f$ και έχει άθροισμα $\Sigma(S_i, A) \geq 2Q$, χρησιμοποιούμε την παρακάτω διαδικασία μετατροπής:

Έστω S_m, ένα σύνολο με ελάχιστο άθροισμα. Παίρνουμε το μικρότερο στοιχείο j από το S_i, και το μετακινούμε στο S_m. Παρατηρούμε ότι $a_j \leq Q_f$ και έτσι προκύπτουν οι παρακάτω ανισότητες:

$$m(S'', A) = \Sigma(S_m, A) \leq Q_f \leq \Sigma(S_i \setminus \{j\}, A)$$

[1] Μπορεί να υπάρχουν πολλά με το ίδιο ελάχιστο άθροισμα.

$$\Sigma(S_m \cup \{j\}, A) \leq 2Q_f \leq \Sigma(S_i, A) \leq M(S'', A)$$

Από την παραπάνω διαδικασία μπορούμε και πάλι να δούμε πως το ελάχιστο σύνολο δεν μειώνεται και το μέγιστο δεν αυξάνεται, έτσι μπορούμε να την εκτελέσουμε επαναλαμβανόμενα, μέχρις ότου να φτάσουμε σε μια λύση που να ικανοποιείται η ιδιότητα 1, όπως κάναμε και παραπάνω με τη μετατροπή για την ιδιότητα 2, αφού ισχύουν τα ίδια επιχειρήματα για την αύξηση του p και τη μη αύξηση του λόγου $\mathcal{R}(S'', A)$. Επιπρόσθετα, η μετατροπή διατηρεί την ιδιότητα 2, αφού το p δεν μειώνεται και προσθέτουμε στοιχεία μόνο στο σύνολο με το ελάχιστο άθροισμα (το οποίο δε μπορεί να είναι συνόλο που περιέχει στοιχείο $j > q$).

Συμπερασματικά, από κάθε εφικτή λύση S, του στιγμιοτύπου (A, p) του k-PART$_{\mathrm{R}}$, μπορούμε να οδηγηθούμε σε μια εφικτή λύση S''' ενός διαφορετικού στιγμιοτύπου (A, p_{new}) του k-PART$_{\mathrm{R}}$, η οποία ικανοποιεί και τις δύο ιδιότητες και έχει ίσο ή μικρότερο λόγο $\frac{S_{max}}{S_{min}}$. Έστω τέλος ότι εφαρμόζουμε τις παραπάνω μετατροπές σε μια βέλτιστη λύση του στιγμιοτύπου (A, p) του k-PART$_{\mathrm{R}}$, με το p να είναι *τέλειο* για το στιγμιότυπο A του k-PART. Αφού ο λόγος της λύσης δεν αυξάνεται μέσω της κατασκευής μας, οποιοσδήποτε νέο p', που αποκτάται από αυτή την κατασκευή, θα πρέπει επίσης να είναι *τέλειο* και η λύση που θα προκύπτει να είναι επίσης βέλτιστη για το στιγμιότυπο (A, p'). ■

Τώρα, έστω p^* ένα από τα *τέλεια* p, το οποίο είναι σίγουρο πως θα υπάρχει, λόγω του θεωρήματος 1, και έστω S^* μια βέλτιστη αντίστοιχη λύση για το στιγμιότυπο (A, p^*) του k-PART$_{\mathrm{R}}$, η οποία ικανοποιεί τις ιδιότητες 1 και 2. Τα ακόλουθα λήμματα δείχνουν ότι μπορούμε να "τριμάρουμε" (δηλαδή αγνοήσουμε) ορισμένες τιμές του p, χωρίς να χάσουμε τη λύση S^*. Θα ορίσουμε μια τιμή $x = n - q$, οπού το x είναι διαισθητικά, ο αριθμός των μεγάλων στοιχείων που πρέπει να περιέχονται σε μονοσύνολα της λύσης S^*, αν $p = p^*$, σύμφωνα με το Θεώρημα 1.

Λήμμα 1. *Αν για κάποιο στιγμιότυπο (A, p), του k-PART$_{\mathrm{R}}$, ισχύει ότι $x > k - 1$, τότε $p \neq p^*$.*

Απόδειξη. Εφόσον το S_1 δεν μπορεί να περιέχει στοιχεία $j > q$, υπάρχουν $k - 1$ σύνολα που μπορούν να περιέχουν τέτοια στοιχεία. Ο αριθμός αυτών των στοιχείων είναι x. Αν $x > k-1$, τότε από την αρχή του περιστερώνα, κάθε εφικτή λύση περιέχει ένα σύνολο με δύο ή περισσότερα από αυτά τα στοιχεία. Αυτό αντιφάσκει με την ιδιότητα 2 του Θεωρήματος 1, άρα $p \neq p^*$. ■

Λήμμα 2. *Αν για κάποιο στιγμιότυπο (A, p), του k-PART$_{\mathrm{R}}$, ισχύει ότι $x = k - 1$ και $p < q$, τότε $p \neq p^*$.*

Απόδειξη. Εφόσον το S_1 δεν μπορεί να περιέχει στοιχεία $j > q$, υπάρχουν $k-1$ σύνολα που μπορούν να περιέχουν τέτοια στοιχεία. Επιπρόσθετα, επειδή $p < q$, το S_1 δεν μπορεί να περιέχει το q. Άρα, υπάρχουν τουλάχιστον $x+1 = k$ στοιχεία που δεν μπορούν να περιέχονται στο S_1, με τα $k - 1$ από αυτά να είναι μεγαλύτερα του q. Από την αρχή του περιστερώνα, κάθε εφικτή λύση περιέχει ένα σύνολο με δύο ή περισσότερα στοιχεία, ένα εκ των οποίων θα είναι μεγαλύτερο του q. Αυτό αντιφάσκει με την ιδιότητα 2 του Θεωρήματος 1, άρα $p \neq p^*$. ■

3.2 Κατασκευή ενός k-PART FPTAS μέσω στρογγυλοποίησης

Σε αυτή την ενότητα θα δούμε του απαραίτητους αλγορίθμους που δομούν το FPTAS για το k-PART.

3.2.1 Ο Αλγόριθμος για το FPTAS του k-PART

Θα παρουσιάσουμε εδώ τον κύριο αλγόριθμο για το k-PART, ο οποίος δημοσιεύτηκε στο 4ο μέρος του [25].

Αλγόριθμος 1 FPTAS_k-PART(A, n, k, ε)

Είσοδος: Ένα ταξινομημένο πολυσύνολο $A = \{a_1, \ldots, a_n\}, a_i \in \mathbb{Z}^+$, το μέγεθος n του A, το πλήθος k των επιθυμητών υποσυνόλων και μια παράμετρος σφάλματος $\varepsilon \in (0, 1)$.

Έξοδος: Μια διαμέριση $(S_1, \ldots, S_k)$ του $[n]$, τέτοια ώστε $\bigcup_{i=1}^{k} S_i = [n]$ και ο λόγος τους να ικανοποιεί τη σχέση $\mathcal{R}(S_1, \ldots, S_k, A) \leq (1+\varepsilon)\mathcal{R}(S_1^*, \ldots, S_k^*, A)$.

for $p = 1, \ldots, n-k+1$ **do**
  $best\_ratio[p] \leftarrow \infty$, $best\_solution[p] \leftarrow (\emptyset, \ldots, \emptyset)$
end for
for $p = 1, \ldots, n-k+1$ **do**
  $Q \leftarrow \sum_{i=1}^{p} a_i$, $q \leftarrow \max\{i \mid a_i \leq Q\}$, $x \leftarrow n - q$
  if $x > k-1 \ \vee \ (x = k-1 \ \wedge \ p < q)$ **then** ▷ Λήμματα 1 και 2
    continue to next p
  end if
  for $y = 1, \ldots, x$ **do**
    $S_{k-x+y} \leftarrow \{q + y\}$
  end for
  $\delta \leftarrow \frac{\varepsilon \cdot a_p}{3 \cdot n}$, $A_r \leftarrow \emptyset$, $k' \leftarrow k - x$
  for $i \leftarrow 1$ to n **do**
    $a_i^r \leftarrow \lfloor \frac{a_i}{\delta} \rfloor$, $A_r \leftarrow A_r \cup \{a_i^r\}$
  end for
  $A'_r = \{a_1^r, \ldots, a_q^r\}$ ▷ q στοιχεία για DP
  $DP\_solutions \leftarrow$ DP_k-PART$_{\mathrm{R}}(A'_r, k', p, Q/\delta)$
  for όλες τις $(S_1, \ldots, S_{k'})$ **στη** $DP\_solutions$ **do**
    $current\_ratio \leftarrow \mathcal{R}(S_1, \ldots, S_k, A_r)$ ▷ Λύση για k-PART$_{\mathrm{R}}$
    if $current\_ratio < best\_ratio[p]$ **then**
      $best\_solution[p] \leftarrow (S_1, \ldots, S_k)$ ▷ Καλύτερη λύση για κάθε A_r
      $best\_ratio[p] \leftarrow current\_ratio$ ▷ και ο αντίστοιχος λόγος της
    end if
  end for
end for
$final\_ratio \leftarrow \infty$, $final\_solution \leftarrow 0$
for $p = 1, \ldots, n-k+1$ **do** ▷ Δοκιμή όλων των p για εύρεση της βέλτιστης λύσης στο A
  $current\_ratio \leftarrow \mathcal{R}(best\_solution[p], A)$
  if $current\_ratio < final\_ratio$ **then**
    $final\_solution \leftarrow best\_solution[p]$, $final\_ratio \leftarrow current\_ratio$
  end if
end for
return $final\_solution$

Αξίζει να σημειωθεί ότι το S^* δεν είναι απαραίτητα βέλτιστη λύση για το στρογγυλοποιημένο στιγμιότυπο, επομένως, με την τεχνική που χρησιμοποιούμε, δεν λαμβάνουμε ***ούτε***

ακριβή αλγόριθμο ούτε FPTAS για την περιορισμένη εκδοχή k-PART$_{\mathrm{R}}$ του προβλήματος k-PART.

Στη γραμμή 17, ο Αλγόριθμος 1 καλεί μία υπορουτίνα δυναμικού προγραμματισμού για κάθε τιμή του p, προκειμένου να βρει υποψήφιες λύσεις για στοιχεία $j \leq q$. Ο αλγόριθμος της υπορουτίνας παρουσιάζεται αμέσως παρακάτω.

3.2.2 Υπορουτίνα Δυναμικού Προγραμματισμού για το k-PART$_{\mathrm{R}}$

Σε αυτή την ενότητα δίνουμε την πλήρη περιγραφή της υπορουτίνας δυναμικού προγραμματισμού (DP).

Αλγόριθμος 2 DP_k-PART$_{\mathrm{R}}(A, k, p, Q)$

Είσοδος: Ένα ταξινομημένο πολύσύνολο $A = \{a_1, \ldots, a_q\},\ a_i \in \mathbb{Z}^+$, το μέγεθος n , το πλήθος των υποσυνόλων διαμέρισης $k \geq 1$, ένας ακέραιος p, $1 \leq p \leq q - k + 1$ και ένας πραγματικός αριθμός $Q > 0$.

Έξοδος: Ένα σύνολο *solutions* που περιέχει k ξένα μεταξύ τους υποσύνολα $(S_1, \ldots, S_k)$ του $[q]$, τέτοια ώστε $\bigcup_{i=1}^{k} S_i = [q]$, $\max(S_1) = p$, και $\max(S_i) > p$ για $1 < i \leq k$.

1: Αρχικά, θέτουμε $T_i[D][V] = \emptyset$ για κάθε $1 \leq i \leq q$, $D = [d_2, \ldots, d_k]$, $V = [v_2, \ldots, v_k]$ με $d_2 \leq d_3 \leq \ldots \leq d_k$, εκτός από την τιμή $T_0[a_p, \ldots, a_p][\texttt{false}, \ldots, \texttt{false}] = (\{p\}, \emptyset, \ldots, \emptyset)$.
2: **for** $i = 1, \ldots, q$ **do**
3: **if** $i = p$ **then**
4: $T_i[D][V] \leftarrow T_{i-1}[D][V]$ **για κάθε** $T_{i-1}[D][V] \neq \emptyset$
5: **continue** to next i
6: **end if**
7: **for κάθε** $T_{i-1}[D][V] \neq \emptyset$ **do**
8: **if** $i < p$ **then**
9: $(S_1, \ldots, S_k), D', V' \leftarrow \texttt{update}(a_i, i, 1, D, V, T_{i-1}[D][V])$
10: **if** $T_i[D'][V'] = \emptyset$ **then**
11: $T_i[D'][V'] \leftarrow (S_1, \ldots, S_k)$
12: **end if**
13: **end if**
14: **for** $j = 2, \ldots, k$ **do**
15: **if** $d_j - a_i > -2Q$ **then**
16: $(S_1, \ldots, S_k), D', V' \leftarrow \texttt{update}(a_i, i, j, D, V, T_{i-1}[D][V])$
17: **if** $T_i[D'][V'] = \emptyset$ **then**
18: $T_i[D'][V'] \leftarrow (S_1, \ldots, S_k)$
19: **end if**
20: **end if**
21: **end for**
22: **end for**
23: **end for**
24: $solutions \leftarrow \emptyset$
25: **for κάθε** $T_q[D][\texttt{true}, \ldots, \texttt{true}] \neq \emptyset$ **do**
26: $(S_1, \ldots, S_k) \leftarrow T_q[D][\texttt{true}, \ldots, \texttt{true}]$
27: $solutions \leftarrow solutions \cup (S_1, \ldots, S_k)$
28: **end for**

29: **return** *solutions*

Ο Αλγόριθμος 1 διατρέχει όλες τις πιθανές τιμές του x, όπου x είναι ο αριθμός των μοναδιαίων συνόλων $\{j\}$, με στοιχεία $a_j > Q$. Για αυτά τα x σύνολα, επιλέγουμε τα μικρότερα στοιχεία του A που είναι μεγαλύτερα ή ίσα με το Q, σύμφωνα με το Θεώρημα 1. Για κάθε x, ο Αλγόριθμος 1 καλεί τον Αλγόριθμο 2 ως υπορουτίνα, ο οποίος χρησιμοποιεί δυναμικό προγραμματισμό (DP) για να παραγάγει μερικές λύσεις για στιγμιότυπα της μορφής $A' = \{a_i \in A \mid a_i \leq Q\}$ και $k' = k - x$. Στη συνέχεια, διατρέχουμε τις μερικές λύσεις που επιστράφηκαν από τον Αλγόριθμο 2, συγχωνευμένες με τα ήδη υπολογισμένα x μονοσύνολα, και βρίσκουμε τον καλύτερο λόγο.

Ο Αλγόριθμος 2 βασίζεται σε τεχνικές από τα [33] και [36]. Δέχεται ως είσοδο ένα πολυσύνολο q ακεραίων, έναν αριθμό k από σύνολα[2] που πρέπει να συμπληρωθούν και έναν ακέραιο p, με $1 \leq p \leq q - k + 1$. Ένας πίνακας δυναμικού προγραμματισμού χρησιμοποιείται για την προσθήκη ενός στοιχείου i σε ένα σύνολο S_j, διασφαλίζοντας ταυτόχρονα ότι πληρούνται ορισμένοι βασικοί περιορισμοί. Συγκεκριμένα, κανένα στοιχείο $i > p$ δεν επιτρέπεται να προστεθεί στο S_1, και ένα σύνολο S_j $(j \neq 1)$ θεωρείται *έγκυρο* αν και μόνο αν περιέχει τουλάχιστον ένα στοιχείο $i > p$. Στο τέλος του αλγορίθμου, μόνο λύσεις με $k - 1$ έγκυρα σύνολα επιστρέφονται.

Οι συντεταγμένες κάθε κελιού $T_i[D][V]$ του πίνακα δυναμικού προγραμματισμού αποτελούνται από τα εξής συστατικά:

1. Έναν δείκτη i, που υποδηλώνει τον αριθμό των στοιχείων που έχουν εξεταστεί μέχρι στιγμής.

2. Ένα *διάνυσμα διαφορών* $D = [d_2, d_3, \ldots, d_k]$ (ταξινομημένο σε μη φθίνουσα σειρά) το οποίο κωδικοποιεί τις διαφορές μεταξύ του αθροίσματος του S_1 και των αθροισμάτων των άλλων συνόλων, δηλαδή $d_j = \sum_{i \in S_1} a_i - \sum_{i \in S_j} a_i$. Παρατηρούμε ότι για οποιοδήποτε $d_j \in D$, ισχύει $d_j \leq Q$, αφού $\sum_{i \in S_1} a_i \leq Q$.

3. Ένα Boolean *διάνυσμα εγκυρότητας* $V = [v_2, v_3, \ldots, v_k]$, όπου κάθε v_j είναι `true` αν το S_j περιέχει ένα στοιχείο i με $i > p$.

Επίσης, η εν λόγω υπορουτίνα, `DP_`k-PART$_{\text{R}}(A'_r, k', p, Q)$, έχει τα εξής 3 χαρακτηριστικά:

1. Κάθε στοιχείο του A προστίθεται σε κάποιο σύνολο.

2. Σύγκρουση σε κάποιο κελί $T_i[D][V]$, συμβαίνει μόνο μεταξύ πλειάδων με ίδιες τιμές i, D, V. Επειδή στο πρόβλημα k-PART$_{\text{R}}$ κάθε στοιχείο πρέπει να μπει σε κάποιο σύνολο (ούτως ώστε η S να είναι διαμέριση του A), αν όλοι οι d_j και το i είναι ίδια, τότε και τα αθροίσματα είναι ίδια. Άρα δεν έχει σημασία ποια πλειάδα θα προτιμηθεί, και επιλέγουμε αυθαίρετα μια για να διατηρήσουμε.

3. Το όριο για αποκοπή μεγάλων αρνητικών διαφορών είναι $-2Q/\delta$, όπου το $Q = \sum_{i=1}^{p} a_i$ υπολογίζεται χρησιμοποιώντας τις *αρχικές* τιμές a_i και όχι τις στρογγυλοποιημένες a_i^r. Αυτό είναι απαραίτητο για την αποφυγή αποκοπής λύσεων οριακής περίπτωσης. Θα επεκταθούμε σε αυτό στο Λήμμα 4.

Ο Αλγόριθμος 2 χρησιμοποιεί μια συνάρτηση `update`, η οποία προσθέτει το στοιχείο i στο S_j και ενημερώνει όλα τα πεδία που επηρεάζονται από αυτήν την προσθήκη. Συγκεκριμένα, το D τροποποιείται είτε με την πρόσθεση του a_i σε όλες τις διαφορές (αν $j = 1$), είτε με την αφαίρεση του a_i από το d_j (αν $j \neq 1$). Στη συνέχεια, ο D ταξινομείται σε αύξουσα

[2]Σε περιπτώσεις όπου ο Αλγόριθμος 2 καλείται με είσοδο $k = 1$, τα διανύσματα διαφορών και εγκυρότητας είναι κενά, οπότε κάθε T_i αποτελείται μόνο από ένα κελί δυναμικού προγραμματισμού.

σειρά και τα σύνολα και τιμές εγκυρότητας αναδιατάσσονται ανάλογα. Σημειώνεται ότι η συνάρτηση εκτελείται σε χρόνο $O(k)$, καθώς (το πολύ) ένα d_j πρέπει να μετακινηθεί. Επειδή η συνάρτηση αυτή είναι σχετικά απλή αλλά εκτενής στην πλήρη της περιγραφή, θα τη χρησιμοποιούμε ως black box, αν και περιλαμβάνεται στο Παράρτημα Δ΄.

3.2.3 Απόδειξη ορθότητας του αλγορίθμου 1

Το παρακάτω λήμμα, αποδεικνύει την εφικτότητα των λύσεων που εξετάζει ο αλγόριθμος 1.

Λήμμα 3. *Κάθε πλειάδα k συνόλων με λόγο $\mathcal{R}(S_1, \ldots, S_k, A_r)$ που εξετάζεται από τον Αλγόριθμο 1, αποτελεί μια εφικτή λύση για το στιγμιότυπο (A_r, p) του k-PART$_\mathrm{R}$.*

Απόδειξη. Τα x μονοσύνολα που κατασκευάζονται από τον Αλγόριθμο 1 είναι ξένα μεταξύ τους και περιέχουν στοιχεία μεγαλύτερα από το q. Στον Αλγόριθμο 2, επεξεργαζόμαστε/εξετάζουμε σε αύξουσα σειρά κάθε στοιχείο $i \leq q$, πριν αυτό προστεθεί σε το πολύ ένα σύνολο. Συνεπώς, όλα τα σύνολα είναι ξένα μεταξύ τους. Σημειώνουμε ότι το p περιέχεται στο S_1 και το S_1 δεν μπορεί να δεχθεί μεγαλύτερο στοιχείο από το p. Οι μόνες λύσεις που επιστρέφει ο Αλγόριθμος 2, είναι αυτές με $V = [\texttt{true}, \ldots, \texttt{true}]$. Υπενθυμίζουμε ότι το v_i τίθεται σε `true` από τη συνάρτηση `update`, του αλγορίθμου 9, όταν το S_i λαμβάνει ένα στοιχείο $j > p$. Επομένως, όλοι οι περιορισμοί που αφορούν το μεγαλύτερο στοιχείο κάθε συνόλου ικανοποιούνται. ■

Υπενθυμίζουμε ότι ορίσαμε τα p^* και S^* ως ένα *τέλειο* p και την αντίστοιχη βέλτιστη λύση του (A, p^*), των οποίων η ύπαρξη είναι εγγυημένη από το Θεώρημα 1.

Λήμμα 4. *Όταν ο Αλγόριθμος 1 συγκρίνει τους λόγους των λύσεων του στιγμιοτύπου (A_r, p^*), του k-PART$_\mathrm{R}$, θα εξετάσει είτε την S^* είτε κάποια άλλη λύση $S = (S_1, \ldots, S_k)$ με $\Sigma(S_i, A_r) = \Sigma(S_i^*, A_r),\ \forall i \in [k]$.*

Απόδειξη. Από τα Λήμματα 1 και 2, για $p = p^*$, δεν μπορούμε να έχουμε ούτε $x > k - 1$, ούτε $x = k - 1$ με $p < q$. Τα μονοσύνολα που επιβάλλονται από τον Αλγόριθμο 1 είναι ακριβώς αυτά που περιέχονται στην S^*, σύμφωνα με την ιδιότητα 2 του Θεωρήματος 1.

Θυμίζουμε ότι μια λύση, τριμάρεται από την υπορουτίνα δυναμικού προγραμματισμού 2, εάν έχει μια διαφορά $d_j \leq -2Q/\delta$. Από την ιδιότητα 1 του Θεωρήματος 1 ισχύει ότι $\Sigma(S_i^*, A) < 2Q,\ \forall i \in [k - x]$ και επειδή τα στοιχεία του A_r είναι *στρογγυλοποιημένα προς τα κάτω*, κατά δ, έχουμε:

$$\Sigma(S_i^*, A_r) \leq \Sigma(S_i^*, A)/\delta < 2Q/\delta.$$

Αυτό συνεπάγεται πως για την S^*, δεν θα υπάρχει d_j με $d_j \leq -2Q/\delta$.

Οι συγκρούσεις στον πινακα δυναμικού προγραμματισμού, στον αλγόριθμο 2, για το πρόβλημα k-PART$_\mathrm{R}$, συμβαίνουν μόνο μεταξύ πλειάδων με ίδια αθροίσματα.

Όταν δύο πλειάδες $C_1 = (S_1, \ldots, S_k)$ και $C_2 = (S'_1, \ldots, S'_k)$ προσπαθούν να ενημερώσουν το ίδιο κελί του πίνακα δυναμικού προγραμματισμού, οι C_1 και C_2 έχουν ακριβώς τα ίδια αθροίσματα (αλλά αναφέρονται σε διαφορετικά σύνολα), χρησιμοποιούν μόνο στοιχεία έως κάποιο στοιχείο i και έχουν τα ίδια διανύσματα D, V. Αυτό συνεπάγεται ότι οποιοσδήποτε συνδυασμός στοιχείων μεγαλύτερων από το i που μπορεί να προστεθεί στα σύνολα του C_1 μπορεί επίσης να προστεθεί και στα σύνολα του C_2, και αντίστροφα. Άρα, οποιοσδήποτε συνδυασμός αθροισμάτων που μπορεί να επιτευχθεί μέσω του C_1 μπορεί επίσης να επιτευχθεί μέσω του C_2, και αντίστροφα (δηλαδή οποιοδήποτε συνδυασμός στοιχείων που μπορεί να προστεθεί σε σύνολα της μιας πλειάδας, θα μπορεί να προστεθεί και σε σύνολα της άλλης) και οι δυνατές προσθέσεις είναι ίδιες. Άρα το S^* μπορεί να αντικατασταθεί μόνο από λύση $S = \{S_1, \ldots . S_k\}$ με ίδια αθροίσματα ανά σύνολο, δηλαδή με $\Sigma(S_i, A_r) = \Sigma(S_i^*, A_r), \forall i \in [k]$.

Η υπορουτίνα δυναμικού προγραμματισμού του αλγορίθμου 2, εξετάζει κάθε πιθανό συνδυασμό συνόλων $S_1, \ldots, S_{k-x}$, εκτός από αυτούς που τριμάρονται από τις περιπτώσεις που αναφέρθηκαν στην παραπάνω παράγραφο. Αυτό συμβαίνει γιατί στην i-οστή επανάληψη ($i \neq p$), ο αλγόριθμος παίρνει κάθε συνδυασμό συνόλων που έχει κατασκευαστεί μέχρι εκείνο το σημείο και προσπαθεί να προσθέσει το i σε κάθε σύνολο (εκτός από το S_1, αν $i > p$) για να κατασκευάσει νέο συνδυασμό. Προσπαθεί επίσης να αγνοήσει το i και να μην το προσθέσει σε κανένα σύνολο. Ο νέος συνδυασμός εισάγεται στον πίνακα T_i, του 2, εκτός αν απορριφθεί όπως περιγράφηκε παραπάνω. Στον p-οστή επανάληψη, ο αλγόριθμος απλώς αντιγράφει το T_{i-1} στο T_i, αφού το p περιλαμβάνεται στο S_1 εξ ορισμού.

Επομένως, σε κάθε περίπτωση ο Αλγόριθμος 1 θα εντοπίσει μια βέλτιστη λύση κατά την αναζήτηση του βέλτιστου λόγου. ■

Είμαστε πλέον έτοιμοι να παρουσιάσουμε το κύριο θεώρημα αυτής της ενότητας.

Θεώρημα 2. *Ο Αλγόριθμος 1 αποτελεί FPTAS για το πρόβλημα* k-PART.

Απόδειξη. Αρχικά, θεωρούμε την περίπτωση $p \neq p^*$. Αν ικανοποιούνται οι συνθήκες του Λήμματος 1 ή 2, τότε το συγκεκριμένο p αγνοείται. Διαφορετικά, θα βρεθεί μια πλειάδα από k σύνολα, η οποία θα αποτελεί εφικτή λύση για το στιγμιότυπο (A_r, p) του k-PART$_{\text{R}}$, σύμφωνα με το Λήμμα 3.

Στη συνέχεια, θεωρούμε την περίπτωση $p = p^*$. Από το Λήμμα 4, ο Αλγόριθμος 1 θα εξετάσει τη βέλτιστη λύση S^* ή κάποια άλλη λύση S με $\mathcal{R}(S, A_r) = \mathcal{R}(S^*, A_r)$ ως λύση για το στιγμιότυπο (A_r, p^*), του k-PART$_{\text{R}}$. Επομένως, για τη λύση S_{alg} που θα επιστρέψει ο αλγόριθμος σε αυτή την επανάληψη, ισχύει ότι $\mathcal{R}(S_{alg}, A_r) \leq \mathcal{R}(S^*, A_r)$. Με βάση αυτή την παρατήρηση, στο παράρτημα Ε′, αποδεικνύεται η παρακάτω ανισότητα για το FPTAS:

$$\mathcal{R}(S_{alg}, A) \leq (1 + \varepsilon)\mathcal{R}(S^*, A)$$

Λαμβάνοντας υπόψη και τις δύο περιπτώσεις, η τελική λύση που επιστρέφει ο αλγόριθμος, είναι μια εφικτή λύση για το στιγμιότυπο A του προβλήματος k-PART, και έχει λόγο μικρότερο ή ίσο από αυτόν της λύσης που βρέθηκε στην p^*-ιοστη επανάληψη. Συνεπώς, ο Αλγόριθμος 1 είναι μια $(1+\varepsilon)$-προσέγγιση για το k-PART, δηλαδή αποτελεί FPTAS για το πρόβλημα k-PART. ■

Θεώρημα 3. *Ο Αλγόριθμος 1 εκτελείται σε χρόνο* $O(n^{2k}/\varepsilon^{k-1})$.

Απόδειξη. Αναλύουμε τώρα την πολυπλοκότητα του αλγορίθμου. Ο αλγόριθμος 1, καλεί την υπορουτίνα δυναμικού προγραμματισμού (αλγόριθμο 2), για μια τιμή του Q στρογγυλοποιημένη προς τα κάτω, συγκεκριμένα για Q/δ και μία φορά για κάθε τιμή του x στο σύνολο $\{0, \ldots, n-q\}$, όπου x είναι ο αριθμός των μοναδιαίων συνόλων $\{j\}$ τέτοιων ώστε $a_j > Q$. Για σταθερό x, ο Αλγόριθμος 2 εφαρμόζεται σε ένα στιγμιότυπο με $k' = k - x$ σύνολα και $q = O(n)$ στοιχεία.

Λόγω του τριμαρίσματος που εκτελείται στον Αλγόριθμο 2, ισχύει ότι $\forall j \in \{2, \ldots, k'\}$: $-2(Q/\delta) < d_j \leq (Q/\delta)$. Εφόσον ο αλγόριθμος αποθηκεύει μόνο τα D σε μη φθίνουσα σειρά (δηλ. $d_2 \leq d_3 \leq \cdots \leq d_{k'}$), υπάρχουν $O((3(Q/\delta))^{k'-1}/(k'-1)!)$ διακριτά διανύσματα διαφορών. Λαμβάνοντας υπόψη το διάνυσμα εγκυρότητας V και όλα τα T_i, προκύπτει το ακόλουθο άνω φράγμα για τον αριθμό των κελιών δυναμικού προγραμματισμού: $O(n(6(Q/\delta))^{k'-1}/(k'-1)!)$.

Για σταθερό k, αυτό το φράγμα γίνεται $O(n(Q/\delta)^{k'-1})$. Παρατηρούμε ότι όλες οι λειτουργίες του Αλγορίθμου 2 πάνω στα κελιά DP (όπως η συνάρτηση `update` του αλγορίθμου 9) εκτελούνται σε χρόνο $O(k')$, επομένως η πολυπλοκότητα χρόνου του Αλγορίθμου 2 είναι $O(n(Q/\delta)^{k'-1}) = O(n(Q/\delta)^{k-x-1})$. Συνεπώς, η χρονική πολυπλοκότητα του Αλγορίθμου 1 φράσσεται από:

$$\sum_{x=0}^{k-1} n(Q/\delta)^{k-x-1} = O(n(Q/\delta)^{k-1})$$

Για τις τιμές του Q τώρα, εφόσον κλιμακώνουμε τα δεδομένα εισόδου κατά δ, οι τιμές του Q φράσσονται ως εξής:

$$Q^{k-1} = \left(\sum_{i=1}^{p} a_i^r\right)^{k-1} \leq \left(n \cdot a_p^r\right)^{k-1} \leq \left(\frac{n \cdot a_p}{\delta}\right)^{k-1} = \left(\frac{3 \cdot n^2}{\varepsilon}\right)^{k-1} = \frac{3^{k-1} \cdot n^{2k-2}}{\varepsilon^{k-1}}$$

Συνεπώς, ο Αλγόριθμος 1 εκτελείται σε χρόνο $O(n^{2k-1}/\varepsilon^{k-1})$. Λαμβάνοντας υπόψη την επανάληψη για όλες τις δυνατές τιμές του p ($1 \leq p \leq n - k + 1$), καταλήγουμε στο ότι ο Αλγόριθμος 1 εκτελείται σε χρόνο $O(n^{2k}/\varepsilon^{k-1})$. ■

Κεφάλαιο 4

Εφαρμογές του FPTAS του k-way Number Partitioning Ratio σε κρυπτοσυστήματα κατωφλίου

Σε αυτή την ενότητα θα δούμε πως μπορεί να αξιοποιηθεί το FPTAS για το k-PART (που είδαμε στο προηγούμενο κεφάλαιο), χρόνου $O(n^{2k}/\varepsilon^{k-1})$, για να ισομοιράσουμε τον φόρτο αποκρυπτογράφησης n κρυπτοκειμένων, ενός (k,t) κρυπτοσυστήματος κατωφλίου, σε k οντότητες. Ας δούμε αρχικά, πώς μπορεί το FPTAS μας για το k-PART, να χρησιμοποιηθεί σε τέτοια κρυπτοσυστήματα.

Μια ομάδα 5 μαθηματικών, ο Paul, η Kate, η Helen, η Tania και ο Kan, έχουν καταφέρει να αποκτήσουν από ένα μέρος, ενος ιδιωτικού κλειδιού p_k ο καθένας, και θέλουν να αποκρυπογραφήσουν ένα σύνολο 100 κρυπτοκειμένων που έχουν υποκλέψει από μια αντίπαλη ομάδα μαθηματικών. Δυστυχώς, δεν μπορούν να στείλουν όλοι τα κομμάτια τους σε όλους, ώστε να σχηματίσουν το ενωμένο ιδιωτικό κλειδί, καθώς οι συζητήσεις τους ενδέχεται να παρακολουθούνται, και εάν οι αντίπαλοι μαθηματικοί αντιληφθούν πως το ιδιωτικό τους κλειδί έχει διαρρεύσει, θα σημάνει συναγερμός στο πανεπιστήμιο τους και τα κρυπτοκείμενα θα καταστραφούν. Αποφασίζουν λοιπόν να βρούν κάποιον εναλλακτικό τρόπο.

Μια συνάδελφος τους η Iliana, κατάφερε και επισκέφτηκε τον καθένα τους ξεχωριστά και τους έδωσε τις εξής πληροφορίες για το κρυπτοσύστημα που χρησιμοποιούν οι αντίπαλοι τους. Καθένα από τα 100 υποκλεμμένα κρυπτοκείμενα, $c_1, \ldots, c_{100}$, έχει κόστος αποκρυπτογράφησης $a_1, \ldots, a_{100}$, αντίστοιχα. Οι αντίπαλοι έχουν χρησιμοποιήσει ένα κρυπτοσύστημα (k,t) Threshold ElGamal, με $(k=5, t=3)$, για την κρυπτογράφηση, άρα για να μπορέσει να ανακτηθεί το απλό κείμενο, που αντιστοιχεί σε κάθε κρυπτοκείμενο, αρκεί να συγκεντρωθούν 3 μερικές αποκρυπτογραφήσεις (partial decryptions) για το καθένα. Ο καθένας τους έχει από έναν υπολογιστή, ίδιας αποκρυπτογραφικής ισχύος, ο οποίος είναι ικανός να αντέξει ένα συγκεκριμένο κόστος αποκρυπτογράφησης, πρίν να εξαντληθεί η μπαταρία του. Η Iliana επίσης τους ενημερώνει, πως αν μοιράσουν με κατάλληλο τρόπο τα κρυπτοκείμενα μεταξύ τους, όλοι τους οι υπολογιστές θα είναι ικανοί να πραγματοποιήσουν την αποκρυπτογράφηση χωρίς πρόβλημα. Συγκεκριμένα, εαν όλος ο απαραίτητος φόρτος εργασίας (αποκρυπτογράφησης), μοιραστεί δίκαια μεταξύ τους, τότε κανένας δε θα αντιμετωπίσει πρόβλημα, και όλα τα κρυπτοκείμενα θα αποκρυπτογραφηθούν.

Πιο τυπικά, έχουμε ένα σύνολο $C = \{c_1, \ldots, c_n\}$ από n κρυπτοκείμενα, που έχουν κρυπτογραφηθεί με το πρωτόκολλο (k,t) Threshold ElGamal, και υποθέτουμε πως στο κάθε κρυπτοκείμενο c_i, αντιστοιχεί ένα βάρος αποκρυπτογράφησης a_i. Έτσι έχουμε το σύνολο $A = \{a_1, \ldots, a_n\}$ των βαρών των κρυπτοκειμένων. Στόχος μας είναι να μοιράσουμε όσο πιο δίκαια γίνεται το κόστος αποκρυπτογράφησης των n κρυπτοκειμένων σε k οντότητες, ούτως ώστε καμία οντότητα να μην αδικηθεί, στην ποσότητα του φόρτου εργασίας που θα

αναλάβει.

Οι φίλοι μας, αναρωτιούνται αν υπάρχει κάποιος αποδοτικός τρόπος να βρούν ένα αρκετά καλό μοίρασμα των κρυπτοκειμένων μεταξύ τους, χωρίς προφανώς να χρειαστεί να δοκιμάσουν όλους τους πιθανούς συνδυασμούς αθροισμάτων των κόστων των κρυπτοκειμένων. Ξαφνικά λοιπόν σκέφτονται το εξής: εάν διαμερίσουν αποδοτικά τα n κόστη κρυπτοκειμένων σε 5 σύνολα, ένα για τον καθένα τους, και τα κόστη που βρίσκονται σε κάθε σύνολο, αθροίζονται σε περίπου τον ίδιο αριθμό με τα υπόλοιπα υποσύνολα, τότε ο λόγος τους αθροίσματος του συνόλου με το μεγαλύτερο άθροισμα, προς το άθροισμα του συνόλου με το μικρότερο άθροισμα, θα είναι πολύ μικρός. Συγκεκριμένα, εάν καταφέρουν και ελαχιστοποιήσουν αυτό τον λόγο, θα έχουν καταφέρει να βρούν τη βέλτιστη διαμέριση κρυπτοκειμένων προς αποκρυπτογράφηση μεταξύ τους.

Όμως, όπως είδαμε στη συζήτηση της προηγούμενης παραγράφου, το FPTAS για το k-PART, μπορεί να χρησιμοποιηθεί ακριβώς για αυτόν το σκοπό, την εύρεση δηλαδή ενός προσεγγιστικά βέλτιστου λόγου $\mathcal{R}(S, A)$, άρα και για να μοιράσουμε, με όσο πιο δίκαιο τρόπο μπορούμε, τον υπολογιστικό φόρτο αποκρυπτογράφησης μεταξύ k οντοτήτων.

Ωστόσο, χρειάζεται να κάνουμε μια κομβική παρατήρηση. Για να μπορέσουμε να αποκτήσουμε όλες τις αποκρυπτογραφήσεις, το κάθε κρυπτοκείμενο πρέπει να αποκρυπτογραφηθεί ακριβώς t φορές, ούτως ώστε να έχουμε τις ελάχιστες δυνατές αποκρυπτογραφήσεις που απαιτούνται για κάθε κρυπτοκείμενο (που ισούνται με t λόγω του threshold). Έτσι, το να δίναμε απλώς στον αλγόριθμο του FPTAS το σύνολο A των n κόστων αποκρυπτογράφησης, δεν θα είχε πρακτική χρησιμότητα για εμάς, καθώς έτσι θα παίρναμε απλά από μια μερική αποκρυπτογράφηση, αντί για τις t που χρειαζόμαστε για το κάθε κρυπτοκείμενο και άρα δε θα μαθαίναμε κανένα απλό κείμενο.

Για να το αντιμετωπίσουμε αυτό, θα εξετάσουμε 3 διαφορετικές προσεγγίσεις, τροποποιώντας κατάλληλα τον αλγόριθμο 1, από τις οποίες, οι 2 πρώτες επιλύουν το πρόβλημα μας, δίνοντας ένα κατάλληλο FPTAS, που θα μπορέσουμε να χρησιμοποιήσουμε στα σενάρια που θα περιγράψουμε, ενώ η τρίτη, εξετάζει το πρόβλημα που αντιμετωπίζει μια 1-1 γενίκευση του αλγορίθμου του FPTAS, του προηγούμενου κεφαλαίου, στη δική μας εφαρμογή.

4.1 Πρώτη Εφαρμογή: Κυκλική ανάθεση t αντιγράφων μιας σχεδόν βέλτιστης λύσης $\mathcal{R}(S, A)$

Μια πρώτη προσέγγιση που θα ακολουθήσουμε είναι η εξής: γνωρίζουμε πως το FPTAS, εάν έπαιρνε ως είσοδο το σύνολο A, θα μας επέστρεφε μια διαμέριση $S = \{S_1, S_2, \ldots, S_k\}$, των κρυπτοκειμένων σε k οντότητες, δηλαδή θα είχε αναθέσει κάθε κρυπτοκείμενο σε μία από τις k οντότητες, με όσο δικαιότερο τρόπο γίνεται, συγκεκριμένα με μια διαμέριση που είναι $(1 + \varepsilon) \cdot R(S^*, A)$ της βέλτιστης. Με αυτόν τον τρόπο όμως, κάθε κρυπτοκείμενο θα λαμβάνει ακριβώς μια μερική κρυπτογράφηση, αντί για t που επιθυμούμε, όπως συζητήσαμε και παραπάνω.

Για να το αντιμετωπίσουμε αυτό, θα προσθέσουμε μια επέκταση στο τέλος του αλγορίθμου 1 της προηγούμενης παραγράφου, με την οποία η λύση S που επιστρέφει ο αλγόριθμος, θα "αντιγράφεται" t φορές και σε κάθε μια από τις k οντότητες, θα αναλαμβάνει t σύνολα από τα k με κυκλικό τρόπο.

Συγκεκριμένα ο αλγόριθμος θα τροποποιηθεί ως εξής:

Αλγόριθμος 3 `Threshold_`k-PART(A, n, k, ε)

Είσοδος: Ένα ταξινομημένο πολυσύνολο $A = \{a_1, \ldots, a_n\}, a_i \in \mathbb{Z}^+$, το μέγεθος n του A, το πλήθος k των επιθυμητών υποσυνόλων και μια παράμετρο σφάλματος $\varepsilon \in (0, 1)$.

Έξοδος: Ένα σύνολο συνόλων $S_{ext} = (S'_1, \ldots, S'_k)$ του $[n]$, με $\bigcup_{i=1}^{k} S_i = [n]$, τέτοια ώστε κάθε S'_j να είναι μια ένωση από t διαφορετικά υποσύνολα S_i και ο λόγος τους να ικανοποιεί τη σχέση $\mathcal{R}(S_1, \ldots, S_k, A) \leq (1+\varepsilon)\mathcal{R}(S^*_1, \ldots, S^*_k, A)$.

1: **for** $p = 1, \ldots, n-k+1$ **do**
2: $best\_ratio[p] \leftarrow \infty$, $best\_solution[p] \leftarrow (\emptyset, \ldots, \emptyset)$
3: **end for**
4: **for** $p = 1, \ldots, n-k+1$ **do**
5: $Q \leftarrow \sum_{i=1}^{p} a_i$, $q \leftarrow \max\{i \mid a_i \leq Q\}$, $x \leftarrow n - q$
6: **if** $x > k-1 \ \vee \ (x = k-1 \ \wedge \ p < q)$ **then** ▷ Λήμματα 1 και 2
7: **continue** to next p
8: **end if**
9: **for** $y = 1, \ldots, x$ **do**
10: $S_{k-x+y} \leftarrow \{q + y\}$
11: **end for**
12: $\delta \leftarrow \frac{\varepsilon \cdot a_p}{3 \cdot n}$, $A_r \leftarrow \emptyset$, $k' \leftarrow k - x$
13: **for** $i \leftarrow 1$ to n **do**
14: $a^r_i \leftarrow \lfloor \frac{a_i}{\delta} \rfloor$, $A_r \leftarrow A_r \cup \{a^r_i\}$
15: **end for**
16: $A'_r = \{a^r_1, \ldots, a^r_q\}$ ▷ q στοιχεία για DP
17: $DP\_solutions \leftarrow$ `DP_`k-PART$_{\mathrm{R}}(A'_r, k', p, Q/\delta)$
18: **for όλες τις** $(S_1, \ldots, S_{k'})$ **στη** $DP\_solutions$ **do**
19: $current\_ratio \leftarrow \mathcal{R}(S_1, \ldots, S_k, A_r)$ ▷ Λύση για k-PART$_{\mathrm{R}}$
20: **if** $current\_ratio < best\_ratio[p]$ **then**
21: $best\_solution[p] \leftarrow (S_1, \ldots, S_k)$ ▷ Καλύτερη λύση για κάθε A_r
22: $best\_ratio[p] \leftarrow current\_ratio$ ▷ και ο αντίστοιχος λόγος της
23: **end if**
24: **end for**
25: **end for**
26: $final\_ratio \leftarrow \infty, final\_solution \leftarrow 0$
27: **for** $p = 1, \ldots, n-k+1$ **do** ▷ Δοκιμή όλων των p για εύρεση της βέλτιστης λύσης στο A
28: $current\_ratio \leftarrow \mathcal{R}(best\_solution[p], A)$
29: **if** $current\_ratio < final\_ratio$ **then**
30: $final\_solution \leftarrow best\_solution[p]$, $final\_ratio \leftarrow current\_ratio$
31: **end if**
32: **end for**
33: $threshold\_solution \leftarrow Rotate\_Solution(final\_solution, A, k, t)$
34: **return** $threshold\_solution$

Στην γραμμή 33, ο αλγόριθμος 3, καλεί την υπορουτίνα `Rotate_Solution`(S, A, k, t), που παρουσιάζεται στον αλγόριθμο 4 που ακολουθεί, ούτως ώστε να επιστρέψει την τελική ανάθεση κρυπτοκειμένων προς αποκρυπτογράφηση, στις k οντότητες.

Ο αλγόριθμος `Rotate_Solution`(S, A, k, t) είναι ο εξής:

Αλγόριθμος 4 `Rotate_Solution`(S, A, k, t)

Είσοδος: Μια διαμέριση $S = \{S_1, \ldots, S_k\}$ από σύνολα S_i και το κατώφλι t
Έξοδος: Ένα σύνολο συνόλων $S_{ext} = (S'_1, \ldots, S'_k)$ του $[n]$, με $\bigcup_{i=1}^{k} S_i = [n]$, τέτοια ώστε κάθε S'_j να είναι μια ένωση από t διαφορετικά υποσύνολα S_i

```
1: q = 1
2: S_ext = ∅
3: for i = 1, ..., k do
4:     for j = i, ..., t + i − 1 do
5:         S'_i = S'_i ⋃ S_q
6:         q = q + 1
7:         if q > k then
8:             q = 1
9:         end if
10:    end for
11:    S_ext = S_ext ⋃ S'_i
12: end for
13: return S_ext
```

Τι έχουμε πετύχει με τη χρήση αυτής της ρουτίνας στον αλγόριθμο 3; Με την ανάθεση του S_{ext} στις k οντότητες, κάθε μια έχει αναλάβει να αποκρυπτογραφήσει έναν αριθμό $|S'_i|$ από διαφορετικά κρυπτοκείμενα, όπως φαίνεται και στο παρακάτω λήμμα:

Λήμμα 5. *Η έξοδος S_{ext} του αλγορίθμου 4, αποτελείται από k σύνολα, S'_i, καθένα από τα οποία περιέχουν $|S'_i|$ διαφορετικά μεταξύ τους κρυπτοκείμενα.*

Απόδειξη. Εφόσον καθένα από τα αρχικά σύνολα S_i που έχει δώσει το FPTAS του αλγορίθμου 1 είναι ξένα μεταξύ τους (αφού η έξοδος S του αλγορίθμου είναι μια διαμέριση του $[n]$), κανένα κρυπτοκείμενο c_j δε θα περιέχεται σε δυο διαφορετικά σύνολα της διαμέρισης S. Έτσι, αφού $t < k$ σε ένα κρυπτοσύστημα κατωφλίου, η ένωση t εξ αυτών των k υποσυνόλων, θα περιέχει μοναδικά μεταξύ τους κρυπτοκείμενα c_j. ■

Έτσι, κάθε μια από τις k οντότητες, αναλαμβάνει να αποκτυπτογραφήσει $|S'_i|$ διαφορετικά κρυπτοκείμενα. Όμως ο αλγόριθμος 4 μας προσφέρει μια ακόμα ιδιότητα. Όταν κάθε μια από τις k οντότητες αποκρυπτογραφήσει τα κρυπτοκείμενα που τις έχουν ανατεθεί από το S_{ext}, θα έχουμε αποκτήσει ακριβώς t μερικές αποκρυπτογραφήσεις καθενός από τα n κρυπτοκείμενα, όπως δείχνει το λήμμα που ακολουθεί:

Λήμμα 6. *Μετά την αποκρυπτογράφηση των κρυπτοκειμένων του συνόλου S'_i που έχει ανατεθεί σε κάθε οντότητα i, από τον αλγόριθμο 4, έχουμε αποκτήσει ακριβώς t διαφορετικές μερικές αποκρυπτογραφήσεις, για καθένα από τα n κρυπτοκείμενα.*

Απόδειξη. Σύμφωνα με το λήμμα 5, κανένα σύνολο S'_i δεν περιέχει δύο αντίγραφα του ίδιου κρυπτοκειμένου, άρα κάθε οντότητα θα δώσει το πολύ μια μερική αποκρυπτογράφηση για κάθε κρυπτοκείμενο. Επιπλέον, με την εκτέλεση του αλγορίθμου 4, το υπερσύνολο S_{ext} περιέχει ακριβώς t αντίγραφα του κάθε κρυπτοκειμένου c_i. Συνδυάζοντας τις δύο παραπάνω προτάσεις, συμπεραίνουμε ότι κάθε κρυπτοκείμενο θα αποκρυπτογραφηθεί ακριβώς t φορές από ακριβώς t διαφορετικές οντότητες, και άρα πως στο τέλος των αποκρυπτογραφήσεων, θα έχουμε αποκτήσει ακριβώς t μερικές αποκρυπτογραφήσεις των n κρυπτοκειμένων, δηλαδή τις ελάχιστες δυνατές αποκρυπτογραφήσεις που χρειαζόμαστε για να ανακτήσουμε τα n αρχικά κείμενα. ■

Έτσι, συνδυάζοντας αυτές τις μερικές αποκρυπτογραφήσεις, θα αποκτήσουμε καθένα από τα n αρχικά κείμενα, χάρη στην ιδιότητα του κρυπτοσυστήματος κατωφλίου με το οποίο είχαν κρυπτογραφηθεί.

Μένει μόνο να δούμε, εάν η ανάθεση S_{ext} των $n \times t$ πλέον κρυπτοκειμένων, παραμένει $(1+\varepsilon)$ προσέγγιση της βέλτιστης δυνατής ανάθεσης και έτσι καμία οντότητα δεν "αδικείται" υπολογιστικά, αναλαμβάνοντας σημαντικά μεγαλύτερο φόρτο εργασίας από τις υπόλοιπες.

Θεώρημα 4. *Ο αλγόριθμος 4, δίνει μια λύση (S_{ext},A) τέτοια ώστε*

$$\mathcal{R}(S_{ext}, A) \leq (1+\varepsilon)\mathcal{R}(S^*, A).$$

Απόδειξη. Η ένωση t από τα k υποσύνολα της λύσης S του αλγορίθμου 1, δίνει σύνολα $S'_i = \bigcup_{j=q}^{(q+t-1)} S_{(j \bmod q)}$ (για $1 \leq q \leq k$) καθένα από τα οποία έχει άθροισμα $\Sigma(S'_i, A) = \sum_{j=q}^{q+t-1} S_{(j \bmod q)}$. Είναι προφανές ότι:

$$t \cdot \min_{j\in[k]} \Sigma(S_j, A) \leq \sum_{j=q}^{q+t-1} S_{(j \bmod q)} \leq t \cdot \max_{j\in[k]} \Sigma(S_j, A)$$

και άρα:

$$t \cdot \min_{j\in[k]} \Sigma(S_j, A) \leq \Sigma(S'_i, A) \leq t \cdot \max_{j\in[k]} \Sigma(S_j, A)$$

Άρα στη χειρότερη περίπτωση,

$$\frac{M(S'_1, \ldots, S'_k, A)}{m(S'_1, \ldots, S'_k, A)} = \frac{\max_{j\in[k]} \Sigma(S'_j, A)}{\min_{j\in[k]} \Sigma(S'_j, A)} = \frac{t \cdot \max_{j\in[k]} \Sigma(S_j, A)}{t \cdot \min_{j\in[k]} \Sigma(S_j, A)}$$

και όπως αποδείχτηκε στο θεώρημα 2:

$$\frac{\max_{j\in[k]} \Sigma(S_j, A)}{\min_{j\in[k]} \Sigma(S_j, A)} = \mathcal{R}(S_{alg}, A) \leq (1+\varepsilon)\mathcal{R}(S^*, A)$$

Άρα

$$\mathcal{R}(S_{ext}, A) = \mathcal{R}(S_{alg}, A) \leq (1+\varepsilon)\mathcal{R}(S^*, A)$$

■

Το παραπάνω θεώρημα μας εξασφαλίζει ότι η ανάθεση κρυπτοκειμένων προς αποκρυπτογράφηση, από το S_{ext} στις k οντότητες, διατηρεί τον βέλτιστο προσεγγιστικά λόγο που δίνει το FPTAS του αλγορίθμου 1 και ο υπολογιστικός φόρτος θα μοιραστεί όσο δικαιότερα γίνεται στις k οντότητες, σύμφωνα πάντα με τις εγγυήσεις του FPTAS της προηγούμενης παραγράφου. Μάλιστα, η τεχνική μας με την κυκλική ένωση των t από τα k κρυπτοκείμενα, κατά τη διάρκεια της δημιουργίας των S'_i, μας εξασφαλίζει ότι στις περισσότερες περιπτώσεις, τα αθροίσματα θα βρίσκονται ακόμα πιο κοντά μεταξύ τους (με εξαίρεση κάποιες οριακές περιπτώσεις όπου τα t μικρότερα S_i είναι ίσα με το S_{min} και τα t μεγαλύτερα S_i, ίσα με το S_{max}).

Τέλος, ας δούμε τη χρονική πολυπλοκότητα του αλγορίθμου μας, που επιλύει το πλήρες πρόβλημα της δίκαια μοιρασμένης αποκρυπτογράφησης.

Θεώρημα 5. *Ο Αλγόριθμος μας για την επίλυση του προβλήματος του δίκαιου καταμερισμού υπολογιστικού κόστους αποκρυπτογράφησης, με κυκλική ανάθεση t αντιγράφων της S, εκτελείται σε χρόνο $O(n^{2k}/\varepsilon^{k-1})$.*

Απόδειξη. Για την επίλυση του προβλήματος, αρχικά τρέχουμε τον αλγόριθμο 1, οπότε και παίρνουμε την αρχική προσεγγιστικά βέλιστη διαμέριση S σε χρόνο $O(n^{2k}/\varepsilon^{k-1})$, όπως αποδείχτηκε στο θεώρημα 2. Στη συνέχεια τρέχουμε τον αλγόριθμο 4 ο οποίος τρέχει σε χρόνο $O(k \cdot t)$. Άρα, η συνολική πολυπλοκότητα της λύσης μας θα είναι:

$$O(n^{2k}/\varepsilon^{k-1}) + O(k \cdot t) = O(n^{2k}/\varepsilon^{k-1})$$

■

4.2 Δεύτερη Εφαρμογή: Αλγόριθμος για την επίλυση του προβλήματος του δίκαιου καταμερισμού υπολογιστικού κόστους αποκρυπτογράφησης, με χρήση ομάδων μεγέθους k/t

Σε αυτή τη δεύτερη προσέγγιση του προβλήματος θα οργανώσουμε το πρόβλημα ως εξής: ας υποθέσουμε ότι δεν έχουμε τους 5 μαθηματικούς του παραδείγματος μας και πως αντιθέτως, είμαστε οι συντονιστές μιας ομάδας από k μαθηματικούς, στους οποίους θέλουμε να αναθέσουμε την αποκρυπτογράφηση n κρυπτοκειμένων ενός κρυπτοσυστήματος κατωφλίου. Θα υποθέσουμε επίσης πως το κατώφλι t διαιρεί ακριβώς το πλήθος k των οντοτήτων[1], και πως στόχος μας είναι να διαμοιράσουμε και πάλι όσο πιο δίκαια γίνεται τον υπολογιστικό φόρτο αποκρυπτογράφησης στις k οντότητες, με σκοπό να αποκτήσουμε t μερικές αποκρυπτογραφήσεις και από αυτές, τα n αρχικά κείμενα. Έστω ότι $k' = k/t$. Ως συντονιστές θα εκτελέσουμε μυστικά την εξής διαδικασία:

1. Χωρίζουμε τις k οντότητες εν άγνοια τους σε k' ομάδες, καθεμία από τις οποίες αποτελείται από t μαθηματικούς, χωρίς κανένας τους να έχει γνώση αυτής της ομαδοποίησης. Έτσι έχουμε αποκτήσει k' ομαδοποιημένες οντότητες.

2. Εκτελούμε τον αλγόριθμο 1 για είσοδο (A, n, k', ε), ο οποίος θα μας επιστρέψει μια διαμέριση $S = (S_1, \ldots, S_{k'})$ των n κρυπτοκειμένων στις k' ομάδες οντοτήτων.

3. Τέλος, σε καθέναν από από τους t μαθηματικούς της καθεμίας από τις k' ομάδες, αναθέτουμε το σύνολο κρυπτοκειμένων S_i.

Συγκεκριμένα, ο αλγόριθμος που θα εκτελέσουμε είναι ο εξής:

[1] Στην επόμενη ενότητα (ενότητα 5) θα αναφέρουμε ενδεικτικά προσεγγίσεις για το ενδεχόμενο $t \nmid k$

Αλγόριθμος 5 `Threshold_Decryption_with_Entity_Grouping`(A, k, t, ε)

Είσοδος: Ένα σύνολο $A = \{a_1, a_2, \ldots, a_n\}$ από n βάρη κρυπτοκειμένων, το πλήθος οντοτήτων k, το κατώφλι t και μια παράμετρο σφάλματος $\varepsilon \in (0, 1)$..

Έξοδος: Μια ανάθεση $S = (S_1, \ldots, S_k)$ δεικτών κρυπτοκειμένων προς αποκρυπτογράφηση στις k οντότητες και τέτοια ώστε, ο λόγος τους να ικανοποιεί τη σχέση $\mathcal{R}(S_1, \ldots, S_k, A) \leq (1 + \varepsilon)\mathcal{R}(S_1^*, \ldots, S_k^*, A)$.

$k' = k/t$
$S_{groups} =$ `FPTAS_`k-PART(A, n, k', ε)
$S = \emptyset$
for $i = 1 \ldots, k'$ **do**
    for $j = 1, \ldots, t$ **do**
        $S = S \bigcup S_{groups_i}$
    end for
end for
return S

Ο αλγόριθμος επιστρέφει μια ανάθεση S, για $n \cdot t$ συνολικά κρυπτοκείμενα (t αντίγραφα καθενός από τα n αρχικά κρυπτοκείμενα), στην οποία έχει πρακτικά αναθέσει στους πρώτους t μαθηματικούς, το σύνολο κρυπτοκειμένων S_{groups_1}, στους επόμενους t το σύνολο S_{groups_2}, κ.ο.κ., μέχρις ότου κάθε μαθηματικός να έχει ένα σύνολο $Sgroups_i$ να αποκρυπτογραφήσει.

Με αυτή τη διαδικασία που εκτελέσαμε, κάθε μαθηματικός θα αποκρυπτογραφήσει τα κρυπτοκείμενα c_j που περιέχονται στο S_i που του ανατέθηκε. Επειδή κάθε ομάδα έχει t μαθηματικούς, στο τέλος των αποκρυπτογραφήσεων, θα έχουμε αποκτήσει ακριβώς t διαφορετικές μερικές αποκρυπτογραφήσεις, για καθένα από τα n κρυπτοκείμενα, και έτσι συνδυάζοντας τες, θα μπορέσουμε να αποκτήσουμε τα n αρχικά κείμενα.

Άρα και με αυτή τη προσέγγιση, επιτυγχάνουμε την αποκρυπτογράφηση n κρυπτοκειμένων από k οντότητες, τα οποία έχουν κρυπτογραφηθεί από ένα (k, t)-κρυπτοσύστημα κατωφλίου. Μένει να δούμε κατά πόσο ο αλγόριθμος 5 δίνει πράγματι μια ανάθεση S, με λόγο $(1 + \varepsilon)$ του βέλτιστου. Αυτό αποδεικνύεται στο παρακάτω θεώρημα.

Θεώρημα 6. *Ο αλγόριθμος 5, δίνει πράγματι μια ανάθεση $S = (S_1, \ldots, S_k)$ (δεικτών κρυπτοκειμένων προς αποκρυπτογράφηση στις k οντότητες), τέτοια ώστε,*

$$\mathcal{R}(S, A) \leq (1 + \varepsilon)\mathcal{R}(S_{groups}^*, A).$$

Απόδειξη. Στον αλγόριθμο 5, η διαμέριση του A σε k υποσύνολα γίνεται από τον αλγόριθμο 1, ο οποίος επιστρέφει μια λύση $\mathcal{R}(S_{groups}, A) \leq (1 + \varepsilon)\mathcal{R}(S_{groups}^*, A)$. Η λύση μας S που προκύπτει κατασκευαστικά από την S_{groups}, έχει το ίδιο S_{min} και το ίδιο S_{max} με την S_{groups} (απλώς αντεγραμμένη t φορές), άρα θα είναι:

$$\mathcal{R}(S, A) = \frac{M(S_1, \ldots, S_k, A)}{m(S_1, \ldots, S_k, A)} = \frac{\max_{j \in [k]} \Sigma(S_j, A)}{\min_{j \in [k]} \Sigma(S_j, A)} = \frac{\max_{j \in [k']} \Sigma(S_{groups_j}, A)}{\min_{j \in [k']} \Sigma(S_{groups_j}, A)} = \mathcal{R}(S_{groups}, A)$$

Άρα

$$\mathcal{R}(S, A) = \mathcal{R}(S_{groups}, A) \leq (1 + \varepsilon)\mathcal{R}(S_{groups}^*, A)$$

■

Το παραπάνω θεώρημα μας εξασφαλίζει ότι η ανάθεση κρυπτοκειμένων προς αποκρυπτογράφηση, στους k μαθηματικούς (που εκτελεί ο αλγόριθμος 5), διατηρεί τον βέλτιστο προσεγγιστικά λόγο που δίνει το FPTAS του αλγορίθμου 1, για τον διαμοιρασμό του φόρτου

εργασίας μεταξύ k' ομάδων μαθηματικών, και επειδή ο υπολογιστικός φόρτος θα μοιραστεί όσο δικαιότερα γίνεται στις k' ομάδες (σύμφωνα πάντα με τις εγγυήσεις του FPTAS της προηγούμενης παραγράφου), ο φόρτος αυτός θα μοιραστεί εξίσου δίκαια στους k μαθηματικούς, καθώς ακριβώς t από αυτούς αναλαμβάνουν ένα σύνολο με υπολογιστικό φόρτο S_{groups_i}. Μάλιστα, χάρη στην τεχνική μας της αντιγραφής συνόλων, η λύση μας είναι ακριβώς ίδια (αριθμητικά) με τη λύση για τις k' οντότητες που δίνει ο αλγόριθμος 1.

Τέλος, ας δούμε τη χρονική πολυπλοκότητα του αλγορίθμου μας, που επιλύει το πλήρες πρόβλημα της δίκαια μοιρασμένης αποκρυπτογράφησης.

Θεώρημα 7. *Ο Αλγόριθμος 5 για την επίλυση του προβλήματος του δίκαιου καταμερισμού υπολογιστικού κόστους αποκρυπτογράφησης, με χρήση ομάδων μεγέθους k/t, εκτελείται σε χρόνο $O(n^{2k}/\varepsilon^{k-1})$.*

Απόδειξη. Ο Αλγόριθμος 5, στη γραμμή 2, καλεί τον αλγόριθμο 1, ο οποίος επιστρέφει την διαμέριση S_{groups} σε χρόνο $O(n^{2k}/\varepsilon^{k-1})$ (όπως αποδείχτηκε στο θεώρημα 2). Στη συνέχεια ο βρόγχος των γραμμών 4 και 5, εκτελείται σε $k' \cdot t = k$ βήματα (από τον ορισμό του k'), δηλαδή σε χρόνο $O(k \cdot t)$. Άρα, ο συνολικός χρόνος εκτέλεσης του αλγορίθμου 5 θα είναι:

$$O(n^{2k}/\varepsilon^{k-1}) + O(k) = O(n^{2k}/\varepsilon^{k-1})$$

■

4.3 Τρίτη Εφαρμογή: Μια 1-1 επέκταση του αλγορίθμου 1, για $n \cdot t$ κρυπτοκείμενα

Μια εύλογη παρατήρηση που μπορεί να κάνει κάποιος, από τις δύο παραπάνω εφαρμογές, είναι πως προσεγγίζουν τη δημιουργία και την ανάθεση των t αντιγράφων των n κρυπτοκειμένων στις k οντότητες (η οποία είναι απαραίτητη προϋπόθεση για την απόκτηση των t μερικών αποκρυπτογραφήσεων, που απαιτούνται για να πάρουμε επιτυχώς όλα τα αρχικά κείμενα), με πλάγιο τρόπο. Χρησιμοποιούν δηλαδή αυτούσιο τον αλγόριθμο 1 και στη συνέχεια χρησιμοποιούν είτε κάποια επιπρόσθετη τεχνική (αντιγραφή της λύσης και κυκλική ανάθεση, στον αλγόριθμο 4), είτε κάποιο χαρακτηριστικό του ορισμού της εφαρμογής (ομαδοποίηση των οντοτήτων σε ομάδες, στον αλγόριθμο 5), για να επιτύχουν την επιθυμητή, και όσο το δυνατόν προσεγγιστικά δικαιότερη, ανάθεση των κρυπτοκειμένων στις k οντότητες.

Αξίζει λοιπόν να αναρωτηθούμε, εάν υπάρχει τρόπος να αλλάξουμε τον ίδιο τον αλγόριθμο 1, και τις υπορουτίνες του (αλγόριθμοι 2 και 9) για να διαχειρίζεται απευθείας το πρόβλημα μας.

4.3.1 Αλγόριθμοι για το k-PART με Περιορισμό Αντιγράφων

Αρχικά, ένα λεπτό ζήτημα, το οποίο αν και δεν τονίζουμε ιδιαίτερα στις δύο παραπάνω εφαρμογές, οφείλουμε να προσέξουμε πολύ, είναι ο τρόπος με τον οποίο θέλουμε να γίνει η ανάθεση των $n \times t$ συνολικά κρυπτοκειμένων, στις k οντότητες.

Αρχικά, θα ορίσουμε το *επαυξημένο σύνολο* A_{ext}:

Ορισμός 9 (*Επαυξημένο Σύνολο* του A). Ορίζουμε A_{ext}, το *επαυξημένο σύνολο* του A, ως το σύνολο $A_{ext} = \{a'_1, \ldots, a'_N\} = \{a_1, \ldots, a_1, a_2, \ldots, a_2, \ldots, a_n, \ldots, a_n\}$, με $N = n \times t$, δηλαδή ως το ταξινομημένο πολυσύνολο που αποτελείται από t αντίγραφα κάθε στοιχείου του A.

Προφανώς, σε ένα επαυξημένο πολυσυνόλο A_{ext}, κάθε $a_i \in A$ εμφανίζεται ακριβώς t φορές και το μέγεθος του συνόλου είναι $|A_{ext}| = N = n \times t$. Έτσι, το σύνολο A_{ext} που κατασκευάσαμε, περιέχει εκείνα ακριβώς τα κρυπτοκείμενα για τα οποία θέλουμε αποκρυπτογραφήσεις, δηλαδή t φορές το κάθε κρυπτοκείμενο $c_i, i \in [n]$, αφού μόνο εάν πάρουμε ακριβώς t αποκρυπτογραφήσεις για το κάθε c_i, θα ικανοποιήσουμε την ιδιότητα του κατωφλίου του κρυπτοσυστήματος, και θα μπορέσουμε να αποκτήσουμε καθένα από τα n αρχικά κείμενα.

Αναρωτιόμαστε τώρα, εάν θα μπορούσαμε να δώσουμε το σύνολο A_{ext}, στον αλγόριθμο 1, ώστε να μας διαμερίσει τα επιθυμητά μας κρυπτοκείμενα στις k οντότητες, με βέλτιστο προσεγγιστικά τρόπο. Εκ πρώτης όψεως, ο αλγόριθμος 1, είναι ικανότατος για μια τέτοια εργασία. Εξ' ορισμού μπορεί να διαχειριστεί πολυσύνολα, και εμείς μπορούμε να αντιμετωπίσουμε κάλλιστα το επαυξημένο σύνολο A_{ext}, ως ένα απλό σύνολο εισόδου A για τον αλγόριθμο 1, που απλώς έχει μέγεθος $N = n \times t$. Επειδή ο αλγόριθμος 1, έχει άγνοια για την ύπαρξη του t, μπορεί απλώς να αντιμετωπίσει αφαιρετικά το μέγεθος N του A_{ext}, με τον ίδιο τρόπο που αντιμετωπίζει σύνολα μεγέθους n. Με αυτή την αφαίρεση (abstraction), οι ορισμοί, τα θεωρήματα και τα λήμματα της προηγούμενης ενότητας 3, θα ίσχυαν όπως είδαμε και προηγουμένως, και θα παίρναμε μια λύση S, με λόγο $\mathcal{R}(S_{alg}, A_{ext}) \leq (1+\varepsilon)\mathcal{R}(S^*, A_{ext})$ και σε χρόνο $O(N^{2k}/\varepsilon^{k-1})$.

Όμως αυτή η αλόγιστη χρήση του αλγορίθμου 1, δε μας δίνει πράγματι τη λύση που επιθυμούμε. Θυμίζουμε πως πλέον στόχος μας είναι να επιλύσουμε το πρόβλημα του δίκαιου καταμερισμού Υπολογιστικού κόστους αποκρυπτογράφησης, (που ορίζεται στον ορισμό 6) και το συγκεκριμένο πρόβλημα έχει μια ιδιαιτερότητα για να έχει πραγματική αξία στις εφαρμογές μας. Εάν η λύση S, που θα επιστρέψει ο αλγόριθμος 1, έχει τοποθετήσει δύο (ή και περισσότερα) αντίγραφα, οποιουδήποτε κρυπτοκειμένου c_i, στο ίδιο υποσύνολο S_j, τότε στην πράξη, έχουμε αναθέσει δύο φορές το ίδιο κρυπτοκείμενο c_i προς αποκρυπτογράφηση, σε μια οντότητα j. Η οντότητα j, έχοντας μόνο ένα τμήμα του ιδιωτικού κλειδιού, θα χρησιμοποιήσει αυτό το τμήμα για την αποκρυπτογράφηση του c_i και θα μας δώσει την ίδια μερική αποκρυπτογράφηση για αυτό. Έτσι, στο τέλος των αποκρυπτογραφήσεων, για το κρυπτοκείμενο c_i, θα έχουμε το πολύ $t-1$ ***διαφορετικές*** μερικές αποκρυπτογραφήσεις, και άρα για το συγκεκριμένο c_i, δε θα ικανοποιείται η συνθήκη κατωφλίου (που απαιτεί ακριβώς t μερικές αποκρυπτογραφήσεις) και άρα δεν θα ανακτήσουμε το αντίστοιχο απλό κείμενο του c_i.

Δίνουμε λοιπόν έναν εναλλακτικό ορισμό του προβλήματος k-PART, που ικανοποιεί τις συνθήκες που θέλουμε να σεβαστούμε:

Ορισμός 10 (k-PART με Περιορισμο Αντιγραφων). Δεδομένου ενός *επαυξημένου συνόλου* $A_{ext} = \{a'_1, \ldots, a'_N\} = \{a_1, \ldots, a_1, a_2, \ldots, a_2, \ldots, a_n, \ldots, a_n\}$, του $A = \{a_1, \ldots, a_n\}$, με $N = n \times t$, να βρεθούν k ξένα μεταξύ τους υποσύνολα $S_1, \ldots, S_k$ του $[N]$, τέτοια ώστε:

1. $\bigcup_{i=1}^{k} S_i = [N]$, (δηλαδή το S είναι μια διαμέριση του A_{ext})

2. Ο λόγος $\mathcal{R}(S_1, \ldots, S_k, A)$ να είναι ο ελάχιστος δυνατός

3. *Κανένα* $S_i \in S$ *δεν περιέχει δύο αντίγραφα του ίδου κρυπτοκειμένου* c_j, δηλαδή $\forall a'_j \in S_i, \nexists a'_{j_1}, a'_{j_2}$, τέτοια ώστε $j_1 \equiv j_2 \bmod t$

Ο περιορισμός ίδιου αντιγράφου $j_1 \not\equiv j_2 \bmod t, \forall j_i \in [n]$, της συνθήκης 3, μας εξασφαλίζει πως θα αποκτήσουμε ακριβώς t διαφορετικές αποκρυπτογραφήσεις, για κάθε κρυπτοκείμενο c_j, αφού έτσι βεβαιωνόμαστε πως κάθε αντίγραφο θα δοθεί σε μια διαφορετική οντότητα i από τις k. Ακολουθεί ο βασικός αλγόριθμος για το k-PART με Περιορισμο Αντιγραφων.

Αλγόριθμος 6 FPTAS_k-PARTME ΠΕΡΙΟΡΙΣΜΟ ΑΝΤΙΓΡΑΦΩΝ(A, n, k, ε)

Είσοδος: Ένα επαυξημένο σύνολο $A_{ext} = \{a'_1, \dots a'_N\}$, του A, το μέγεθος N του A_{ext}, το πλήθος k των επιθυμητών υποσυνόλων και μια παράμετρος σφάλματος $\varepsilon \in (0, 1)$.

Έξοδος: k ξένα υποσύνολα $(S_1, \dots, S_k)$ του $[N]$, τέτοια ώστε $\bigcup_{i=1}^{k} S_i = [N]$ και ο λόγος τους ικανοποιεί την σχέση $\mathcal{R}(S_1, \dots, S_k, A_{ext}) \leq (1+\varepsilon)\mathcal{R}(S^*_1, \dots, S^*_k, A_{ext})$, ενώ συγχρόνως, κανένα S_i δεν περιέχει δύο φορές το ίδιο κρυπτοκείμενο.

1: **for** $p = 1, \dots, N-k+1$ **do**
2: $best\_ratio[p] \leftarrow \infty$, $best\_solution[p] \leftarrow (\emptyset, \dots, \emptyset)$
3: **end for**
4: **for** $p = 1, \dots, N-k+1$ **do**
5: $Q \leftarrow \sum_{i=1}^{p} a_i$, $q \leftarrow \max\{i \mid a_i \leq Q\}$, $x \leftarrow N - q$
6: **if** $x > k-1 \ \vee \ (x = k-1 \ \wedge \ p < q)$ **then** ▷ Lemmas 1 and 2
7: **continue** to next p
8: **end if**
9: **for** $y = 1, \dots, x$ **do**
10: $S_{k-x+y} \leftarrow \{q+y\}$
11: **end for**
12: $\delta \leftarrow \frac{\varepsilon \cdot a_p}{3 \cdot N}$, $A_r \leftarrow \emptyset$, $k' \leftarrow k - x$
13: **for** $i \leftarrow 1$ to $n \times t$ **do**
14: $a^r_i \leftarrow \lfloor \frac{a_i}{\delta} \rfloor$,
15: $A_r \leftarrow A_{ext,r} \cup \{a^r_i\}$
16: **end for**
17: $A'_{ext,r} = \{a^r_1, \dots, a^r_q\}$ ▷ q items for DP
18: $DP\_solutions \leftarrow$ DP_k-PARTME ΠΕΡΙΟΡΙΣΜΟ ΑΝΤΙΓΡΑΦΩΝ$(A'_r, k', p, Q/\delta)$
19: **for all** $(S_1, \dots, S_{k'})$ **in** $DP\_solutions$ **do**
20: $current\_ratio \leftarrow \mathcal{R}(S_1, \dots, S_k, A_{ext,r})$ ▷ k-PART$_{\mathrm{R}}$ solution
21: **if** $current\_ratio < best\_ratio[p]$ **then**
22: $best\_solution[p] \leftarrow (S_1, \dots, S_k)$ ▷ Βέλτιστη λύση για κάθε $A_{ext,r}$
23: $best\_ratio[p] \leftarrow current\_ratio$ ▷ και ο λόγος της
24: **end if**
25: **end for**
26: **end for**
27: $final\_ratio \leftarrow \infty$, $final\_solution \leftarrow 0$
28: **for** $p = 1, \dots, N-k+1$ **do** ▷ Iterate for all p to find best sol. for A_{ext}
29: $current\_ratio \leftarrow \mathcal{R}(best\_solution[p], A_{ext})$
30: **if** $current\_ratio < final\_ratio$ **then**
31: $final\_solution \leftarrow best\_solution[p]$, $final\_ratio \leftarrow current\_ratio$
32: **end if**
33: **end for**
34: **return** $final\_solution$

Ο παραπάνω αλγόριθμος αποτελεί μια 1-1 τροποποίηση του αλγορίθμου 1, η οποία σέβεται τη συνθήκη 3 του ορισμού 10. Ο έλεγχος για τη συνθήκη 3, πραγματοποιείται στην υπορουτίνα του αλγορίθμου 7 που ακολουθεί, η οποία καλείται στη γραμμή 19 του αλγορίθμου 6.

Αλγόριθμος 7 DP_ k-PARTME ΠΕΡΙΟΡΙΣΜΟ ΑΝΤΙΓΡΑΦΩΝ(A_{ext}, k, p, Q)

Είσοδος: Ένα επαυξημένο σύνολο $A_{ext} = \{a'_1, \dots a'_N\}$, του A, το πλήθος οντοτήτων k, ένας ακέραιος p, με $1 \leq p \leq q - k + 1$ και ένας πραγματικός αριθμός $Q > 0$.

Έξοδος: Ένα σύνολο $solutions$ το οποίο περιέρχει πλειάδες από k, ξένα μεταξύ τους, υποσύνολα $(S_1, \dots, S_k)$ του $[q]$, τέτοιο ώστε $\bigcup_{i=1}^{k} S_i = [q]$, $\max(S_1) = p$, $\max(S_i) > p$ για $1 < i \leq k$, ενώ συγχρόνως, κανένα S_i δεν περιέχει δύο φορές το ίδιο κρυπτοκείμενο.

Αρχικά, έστω $T_i[D][V] = \emptyset$ για κάθε $1 \leq i \leq q$, $D = [d_2, \dots, d_k]$, $V = [v_2, \dots, v_k]$ με $d_2 \leq d_3 \leq \dots \leq d_k$, εκτός $T_0[a_p, \dots, a_p][\text{false}, \dots, \text{false}] = (\{p\}, \emptyset, \dots, \emptyset)$
for $i = 1, \dots, q$ **do**
    if $i = p$ **then**
        $T_i[D][V] \leftarrow T_{i-1}[D][V]$ **for all** $T_{i-1}[D][V] \neq \emptyset$
        continue to next i
    end if
    ..... ▷ Ίδιο με τον αλγόριθμο 1
    for all $T_{i-1}[D][V] \neq \emptyset$ **do**
        if $i < p$ **then**
            skip $\leftarrow$ false
            for $s \in S_1$ **do**
                if $s \equiv i \bmod t$ **then** ▷ Ελεγχος για ύπαρξη αντιγράφου
                    skip $\leftarrow$ true ▷ Μην αλλάξεις τον πίνακα DP
                    break
                end if
            end for
            if not skip then
                $(S_1, \dots, S_k), D', V' \leftarrow$ **Copy_Restricted_Update**$(a_i, i, 1, D, V, T_{i-1}[D][V])$
                if $T_i[D'][V'] = \emptyset$ **then**
                    $T_i[D'][V'] \leftarrow (S_1, \dots, S_k)$
                end if
            end if
        end if
        for $j = 2, \dots, k$ **do**
            if $d_j - a_i > -2Q$ **then**
                skip $\leftarrow$ false
                for $s \in S_j$ **do**
                    if $s \equiv i \bmod t$ **then** ▷ Ελεγχος για ύπαρξη αντιγράφου
                        skip $\leftarrow$ true ▷ Μην αλλάξεις τον πίνακα DP
                        break
                    end if
                end for
                if not skip then
                    $(S_1, \dots, S_k), D', V' \leftarrow$ **Copy_Restricted_Update**$(a_i, i, j, D, V, T_{i-1}[D][V])$
                    if $T_i[D'][V'] = \emptyset$ **then**
                        $T_i[D'][V'] \leftarrow (S_1, \dots, S_k)$
                    end if
                end if
            end if
        end for

41: **end for**
42: **end for**

Ο αλγόριθμος 7, είναι πανομοιότυπος με τον αλγόριθμο 2, με μια διαφοροποίηση στις γραμμές 12 και 28. Στην πράξη, η συνθήκη ελέγχου των γραμμών αυτών, ελέγχει αν κάποιο αντίγραφο του κρυπτοκειμένου c_s υπάρχει ήδη στο σύνολο που ετοιμαζόμαστε να προσθέσουμε τον δείκτη s, και εάν υπάρχει, δεν καλεί την **Copy_Restricted_Update** και προχωρά στην επόμενη επανάληψη του εξωτερικού βρόγχου. Η δε συνάρτηση **Copy_Restricted_Update** είναι η ίδια με τον αλγόριθμο 9 που παρουσιάζεται στο παράρτημα Δ΄.

Αλγόριθμος 8 Copy_Restricted_Update$(a_i, i, j, D, V, S_1, \ldots, S_k)$

Είσοδος: Ένα στοιχείο a_i, ο δείκτη του i, ένας δείκτη j $(1 \leq j \leq k)$, ένας (ταξινομημένος) πίνακας διαφορών $D = [d_2, \ldots, d_k]$, ένας Boolean πίνακας εγκυρότητας (ή πίνακας ορθότητας) $V = [v_2, \ldots, v_k]$, και k ξένα μεταξύ τους υποσύνολα S_j που περιέχουν στοιχεία στο $[i-1]$.

Έξοδος: k ενημερωμένα υποσύνολα S_j, τον ενημερωμένο (ταξινομημένο) πίνακα διαφορών D και τον πίνακα εγκυρότητας V.

1: **if** j == 1 **then**
2: $S_1 = S_1 \cup \{i\}$,
3: **for all** d_j in D **do**
4: $d_j = d_j + a_i$
5: **end for**
6: **return** $(S_1, \ldots, S_k),\ D,\ V$
7: **end if**
8: $S_j = S_j \cup \{i\},\ d_j = d_j - a_i$
9: **if** $i > p$ **then**
10: $v_j =$ true
11: **end if**
12: **while** $d_{j-1} > d_j$ **do** ▷ Ενάλλαξε τα σύνολα και τα αντίστοιχα τους D, V
13: $tempS = S_j,\ tempD = d_j,\ tempV = v_j$
14: $S_j = S_{j-1},\ d_j = d_{j-1},\ v_j = v_{j-1}$
15: $S_{j-1} = tempS,\ d_{j-1} = tempD,\ v_{j-1} = tempV$
16: $j = j - 1$
17: **end while**
18: **return** $(S_1, \ldots, S_k),\ D,\ V$

Έχουμε πλέον επεκτείνει επιτυχώς τους αλγορίθμους της ενότητας 3. Μένει μόνο να επιβεβαιώσουμε την ισχύ των θεωρήματων που στηρίζουν τους προηγούμενους αλγορίθμους του k-PART με Περιορισμο Αντιγραφων.

4.3.2 Ανάλυση των αλγορίθμων για το k-PART με Περιορισμό Αντιγράφων

Για την ανάλυση μας, αρχικά θα κρατήσουμε τον ορισμό 7 του k-PART$_{\text{R}}$ όπως είναι στην παράγραφο 3, με μια μικρή διαφορά στο μέγεθος του A_{ext} που πλέον χρησιμοποιούμε.

Ορισμός 11 (k-PART$_{\text{R}}$με Περιορισμο Αντιγραφων). Δοσμένου ενός ταξινομημένου πολυσυνόλου $A_{ext} = \{a'_1, \ldots, a'_N\}$ από $N = n \cdot t$ θετικούς ακέραιους, ενός ακεραίου k και ενός ακεραίου

p, τέτοιο ώστε, $1 \leq p \leq N - k + 1$, ζητούνται k ξένα υποσύνολα $S_1, \ldots, S_k$ του $[N]$ για τα οποία ισχύει ότι $\bigcup_{i=1}^{k} S_i = [N]$ (δηλαδή το S είναι μια διαμέριση του $[N]$), $\max(S_1) = p'$ και $\max(S_i) > p$ για $1 < i \leq k$ έτσι ώστε να ελαχιστοποιείται η $\mathcal{R}(S_1, \ldots, S_k, A_{ext})$.

Σχεδόν ίδιοι θα παραμείνουν και οι ορισμοί του τέλειου p καθώς και των Q και q, με μικρές τροποποιήσεις στο μέγεθος τους, καθώς το n είναι πλέον $N = n \cdot t$:

Ορισμός 12 (Τέλειο p στο k-PART με Περιορισμο Αντιγραφων)**.** Έστω $A_{ext} = \{a'_1, \ldots, a'_N\}$, με $N = n \cdot t$, η είσοδος του προβλήματος k-PART με Περιορισμο Αντιγραφων. Θα λέμε πως ο ακέραιος $p : 1 \leq p \leq N - k + 1$, είναι *τέλειος* για το A_{ext}, αν η βέλτιστη λύση S^* του στιγμιοτύπου (A_{ext}, p) του k-PART$_{\text{R}}$ με Περιορισμο Αντιγραφων (και οι ισοδύναμες της), έχει τον ίδιο λόγο με μια βέλτιστη λύση για το στιγμιότυπο A_{ext} του k-PART με Περιορισμο Αντιγραφων.

Καθώς επίσης και έστω, $Q = \sum_{i=1}^{p} a'_i$, το άθροισμα των p μικρότερων στοιχείων του ταξινομημένου πολυσυνόλου A_{ext} και έστω $q = \max\{i \mid a'_i \leq Q\}$, το μέγιστο στοιχείο με άθροισμα $\leq Q$.

Όπως προαναφέραμε, επειδή ο αλγόριθμος 1, λειτουργούσε απροβλημάτιστα για πολυσύνολα, το ότι το A_{ext}, αποτελεί ένα πολυσύνολο συγκεκριμένης δομής (περιέχει ακριβώς t αντίγραφα κάθε στοιχείου a_i), σε καμία περίπτωση δεν το εμποδίζει να δοθεί ως είσοδος στον αλγόριθμο 1. Το βασικό όμως θεώρημα που θα πρέπει να ισχύει για να ισχύει ο αλγόριθμος 6 και οι υπορουτίνες του, είναι το θεώρημα 1 της υποενότητας 3.1. Συγκεκριμένα όμως, θα πρέπει να ισχύει μια ενισχυμένη εκδοχή του θεωρήματος 1, που να λαμβάνει υπόψιν του τον περιορισμό αντιγράφων και να είναι έγκυρο για το πρόβλημα k-PART με Περιορισμο Αντιγραφων.

Η ενισχυμένη εκδοχή του θεωρήματος 1 συνοψίζεται στην παρακάτω εικασία:

Εικασία 1. *Δοθέντος ενός στιγμιοτύπου A_{ext}, του k-PART με Περιορισμο Αντιγραφων, υπάρχει ένα τέλειο p για το A_{ext}, για το οποίο υπάρχει μία βέλτιστη λύση του στιγμιοτύπου (A_{ext}, p), του k-PART$_{\text{R}}$ με Περιορισμο Αντιγραφων, που ικανοποιεί τις ακόλουθες ιδιότητες:*

1. *Για όλα τα υποσύνολα S_i του S, που περιέχουν μόνο στοιχεία $j \leq q$, ισχύει ότι $\Sigma(S_i, A_{ext}) < 2Q$.*
2. *Όλα τα στοιχεία $j > q$ ανήκουν σε μονοσύνολα.*
3. *Κάθε σύνολο S_i του S περιέχει το πολύ ένα αντίγραφο του κρυπτοκειμένου c_j, δηλαδή $\forall a'_j \in S_i$, $\nexists a'_{j_1}, a'_{j_2}$, τέτοια ώστε $j_1 \equiv j_2 \bmod t$*

■

Ας προσπαθήσουμε να αναπαράγουμε βήμα βήμα την απόδειξη της υποενότητας 3.1. Για μια αυθαίρετη τιμή του p, έστω $S = (S_1, \ldots, S_k)$ μια αυθαίρετη εφικτή λύση για το στιγμιότυπο (A_{ext}, p), του k-PART$_{\text{R}}$με Περιορισμο Αντιγραφων. Αν το S παραβιάζει την ιδιότητα 2, δηλαδή υπάρχει σύνολο S_i της S, με $|S_i| > 1$, το οποίο περιέχει στοιχείο $j > q$, τότε προχωράμε ως εξής: έστω S_m ένα σύνολο με ελάχιστο άθροισμα[2] και u το μικρότερο στοιχείο του S_i. Σε αυτό το σημείο θα θέλαμε να μπορέσουμε να μετακινήσουμε το μικρότερο στοιχείο του S_i, στο σύνολο S_m με το μικρότερο άθροισμα. Ωστόσο, λόγω της ιδιότητας 3, που πρέπει να τηρεί η εκδοχή του θεωρήματος μας με τον περιορισμό αντιγράφου, επειδή το σύνολο S_m, ενδέχεται να έχει ήδη ένα αντίγραφο του μικρότερου στοιχείου[3] c_u του S_i,

[2]Μπορεί να υπάρχουν πολλά με το ίδιο ελάχιστο άθροισμα.

[3]εδώ το μικρότερο στοιχείο u είναι το κρυπτοκείμενο του S_i με το μικρότερο κόστος αποκρυπτογράφησης, οπότε στο εξής θα λέμε ως μικρότερο στοιχείο του S_i το c_u, εννοώντας ότι έχει το μικρότερο κόστος αποκρυπτογράφησης

η μετακίνηση του c_u στο S_m, θα καθιστούσε τη λύση μας S, μη εφικτή για το πρόβλημα k-PARTME ΠΕΡΙΟΡΙΣΜΟ ΑΝΤΙΓΡΑΦΩΝ. Για να μπορέσουμε να πραγματοποιήσουμε με επιτυχία τη μετακίνηση του c_u και να καταφέρουμε μετά από επαναλήψεις το S_i, που περιέχει ένα $j > q$, να είναι μονοσύνολο, θα πρέπει να αποφύγουμε τα $t-1$ άλλα σύνολα, εκτός του S_i που έχουν αντίγραφο του c_u, καθώς μια μετακίνηση του c_u, σε σύνολο S_m που περιέχει ήδη ένα αντίγραφο του, χαλάει την εφικτότητα της λύσης S που αρχικά υποθέσαμε.

Άρα, για να μπορούμε με ασφάλεια να πραγματοποιούμε τη μετατροπή του θεωρήματος 1, θα πρέπει σε κάθε απαραίτητη μετακίνηση, να υπάρχει κάποιο σύνολο $S_j \in S$, το οποίο αφού λάβει το u, δε θα χειροτερεύει τον λόγο $\mathcal{R}(S, A_{ext})$ της S.

Για να συμβαίνει αυτό, θα πρέπει να τηρούνται δύο προϋποθέσεις:

1. Η μετακίνηση του u στο S_j να μην ελαττώνει το μικρότερο άθροισμα S_{min}, της S.

2. Η μετακίνηση του u στο S_j να μην αυξάνει το μεγαλύτερο άθροισμα S_{max}, της S.

Για την ιδιότητα 1, δεν χρειάζεται να σκεφτούμε πολύ. Πριν την αφαίρεση του u από το S_i, η λύση S περιείχε και σύνολα με άθροισμα μικρότερο του Q (συγκεκριμένα, σίγουρα το S_1, από τον ορισμό του Q), ενώ μετά τη μετακίνηση, το S_i εξακολουθεί να περιέχει ένα $j > q$, άρα ακόμα και μετά την αφαίρεση, το S_j δε θα γίνει το S_{min}, άρα το ελάχιστο άθροισμα δεν μειώνεται.

Δυστυχώς όμως, στην απόδειξη του θεωρήματος 1, το βασικότερο μας επιχείρημα για την ύπαρξη κατάλληλου εφικτού συνόλου[4] μετακίνησης, είναι πως χρησιμοποιούμε το σύνολο S_m επειδή ξέρουμε πως είναι $\Sigma(S_m, A) \leq Q$ και άρα η προσθήκη του u σε αυτό, δε θα περάσει σε άθροισμα το S_{max}.

Η ύπαρξη ενός S_m στη λύση S, είναι εγγυημένη επειδή το Q έχει οριστεί ως το άθροισμα των p μικρότερων στοιχείων του A. Αφού υπάρχει σίγουρα το S_1 στη λύση S του k-PART$_{\text{R}}$, και το S_1, έχει την ιδιότητα να έχει για μέγιστο στοιχείο το a_p, υπάρχει σίγουρα ένα σύνολο S_m (το S_1), τέτοιο ώστε $\Sigma(S_m, A) \leq \sum_{i=1}^{p} a'_i = Q$.

Στο επαυξημένο μας θεώρημα όμως, η χρήση του S_m μπορεί να μην είναι εφικτή, όπως αναλύσαμε παραπάνω, και δεν υπάρχει τρόπος να εγγυηθούμε ότι κάποιο από τα υπόλοιπα $k-t+1$ σύνολα, που δεν έχουν αντίγραφο του u, θα έχουν κατάλληλο άθροισμα, τέτοιο ώστε η προσθήκη του u σε αυτό, να μην υπερβαίνει το S_{max} της S και άρα να χειροτερεύει τον λόγο $\mathcal{R}(S, A_{ext})$.

Μάλιστα, η ίδια η δομή της νέου μας προβλήματος, ενδέχεται να μας δημιουργήσει πρόβλημα. Με το k να παραμένει σταθερό και το μέγεθος του συνόλου να αυξάνεται t φορές, παρόλο που η αφαίρεση του πολυσυνόλου, μας επιτρέπει να αντιμετωπίσουμε την είσοδο ανεξάρτητα, η λύση που προσπαθούμε να επιτύχουμε με τη χρήση του τέλειου p, προϋποθέτει να έχουμε αρκετά μονοσύνολα k στη διαμέριση, τα οποία θα περιέχουν τα "μεγάλα" στοιχεία $j > q$. Και ενώ αυτό το επιχείρημα μπορούσε να χρησιμοποιηθεί υπερ μας στα λήμματα 1 και 2 του κανονικού προβλήματος k-PART, η δημιουργία t αντιγράφων κάθε μεγάλου στοιχείου $j > q$, μπορεί να οδηγήσει στο γρήγορο γέμισμα των k συνόλων της διαμέρισης, από στοιχεία $j > q$, που πρέπει να μπουν σε μονοσύνολα και την εξάντληση του διαθέσιμου χώρου για τα υπόλοιπα στοιχεία μας.

Ο λόγος προφανώς που συμβαίνει κάτι τέτοιο, είναι πως τώρα δεν επιτρέπεται με βεβαιότητα να μετακινήσουμε ένα στοιχείο u σε οποιοδήποτε σύνολο S_m έχει ελάχιστο άθροισμα, λόγω της ενδεχόμενης παραβίασης της ιδιότητας 3 της εικασίας 1. Εάν μπορούσαμε και πάλι να εγγυηθούμε ότι πάντα υπάρχει ένα έγκυρο σύνολο S_m (δηλαδή που δεν έχει αντίγραφο του u), με $\Sigma(S_m, A) \leq Q$, τότε θα μπορούσαμε να μετακινούμε το u με ασφάλεια, χωρίς να χαλάμε τον λόγο $\mathcal{R}(S, A_{ext})$. Χωρίς αυτή την εγγύηση όμως, και λόγω της t

[4] Με τον όρο εφικτό σύνολο, εννοούμε εδώ ένα σύνολο που να μην έχει αντίγραφο του u, ώστε να μπορέσουμε να πραγματοποιήσουμε τη μετακίνηση του u σε αυτό, χωρίς να χαλάσουμε την εφικτότητα της λύσης S

πολλαπλότητας των μεγάλων στοιχείων του A στο A_{ext}, στις περισσότερες περιπτώσεις θα είναι $x \geq k-1$ και οι λύσεις για αυτά τα p θα τριμάρονται.

Αν λοιπόν τρέξουμε αλόγιστα τον αλγόριθμο 6, οι αρχικά εφικτές λύσεις S που τηρούν τη συνθήκη 3 της εικασίας 1, θα τριμάρονται στις γραμμές 12 και 28 του αλγορίθμου 7, ως μη εφικτές για το πρόβλημα μας.

Για τον ίδιο λόγο καταρρέει και το ανάλογο επιχείρημα μετακίνησης του θεωρήματος 1 για τη διόρθωση των συνόλων που παραβιάζουν την ιδιότητα 1 στο θέωρημα 1.

Συμπερασματικά, οι αλγόριθμοι μας, της παραγράφου 4.3.1, δεν δουλεύουν όπως θα θέλαμε για το γενικό πρόβλημα k-PARTme Περιορισμο Αντιγραφων.

4.3.3 Ειδική περίπτωση: Ισχύς των αλγορίθμων για το k-PART με Περιορισμό Αντιγράφων υπό προϋποθέσεις

Έχει ακόμα κάποια χρησιμότητα ο αλγόριθμος 7; Φυσικά και έχει. Μπορεί τα θεωρήματα να μην ισχύουν σε κάθε περίπτωση για το πρόβλημα k-PART με Περιορισμο Αντιγραφων, όπως ο αλγόριθμος 1 για το πρόβλημα k-PART, όμως αν τρέξουμε τον αλγόριθμο 7 για είσοδο A_{ext}, και μας επιστρέψει μια λύση S_{alg} στην έξοδο του, τότε μπορούμε να εγγυηθούμε πως η λύση αυτή S_{alg} θα είναι μια ορθή λύση για το στιγμιότυπο A_{ext} του προβλήματος 7, και μάλιστα μια λύση με βέλτιστο προσεγγιστικά λόγο $\mathcal{R}(S_{alg}, A_{ext}) \leq (1+\varepsilon)\mathcal{R}(S^*, A_{ext})$.

Θα ορίσουμε λοιπόν το νέο θεώρημα 1, για το τέλειο p και θα αποδείξουμε την ορθότητα του κάτω από την προϋπόθεση του να υπάρχει πάντα έγκυρο σύνολο S_m για τη μετακίνηση του u σε αυτό, με $\Sigma(S_m, A) \leq Q$

Θεώρημα 8. *Δοθέντος ενός στιγμιοτύπου A_{ext}, του k-PART με Περιορισμο Αντιγραφων, υπάρχει ένα τέλειο p για το A_{ext}, για το οποίο υπάρχει μία βέλτιστη λύση του στιγμιοτύπου (A_{ext}, p), του k-PART$_{\text{R}}$ με Περιορισμο Αντιγραφων, που ικανοποιεί τις ακόλουθες ιδιότητες, εάν τηρούνται κάποιες συγκεκριμένες προϋποθέσεις (που θα αναλύσουμε εντός της απόδειξης):*

1. *Για όλα τα υποσύνολα S_i του S, που περιέχουν μόνο στοιχεία $j \leq q$, ισχύει ότι $\Sigma(S_i, A_{ext}) < 2Q$.*
2. *Όλα τα στοιχεία $j > q$ ανήκουν σε μονοσύνολα.*
3. *Κάθε σύνολο S_i του S περιέχει το πολύ ένα αντίγραφο του κρυπτοκειμένου c_j, δηλαδή $\forall a'_j \in S_i, \nexists a'_{j_1}, a'_{j_2}$, τέτοια ώστε $j_1 \equiv j_2 \bmod t$*

Απόδειξη. Για μια αυθαίρετη τιμή του p, έστω $S = (S_1, \ldots, S_k)$ μια αυθαίρετη εφικτή λύση για το στιγμιότυπο (A_{ext}, p), του k-PART$_{\text{R}}$με Περιορισμο Αντιγραφων. Αν το S παραβιάζει την ιδιότητα 2, δηλαδή υπάρχει σύνολο S_i της S, με $|S_i| > 1$, το οποίο περιέχει στοιχείο $j > q$, τότε προχωράμε ως εξής: έστω u το μικρότερο στοιχείο του S_i και S_m ένα σύνολο με άθροισμα $\Sigma(S_m, A_{ext}) \leq Q$ που δεν περιέχει αντίγραφο του u. Ισχύουν οι παρακάτω ανισότητες:

$$m(S, A_{ext}) = \Sigma(S_m, A_{ext}) \leq Q < \Sigma(S_i \setminus \{u\}, A_{ext})$$

$$\Sigma(S_m \cup \{u\}, A_{ext}) \leq Q + a_u < \Sigma(S_i, A_{ext}) \leq M(S, A_{ext})$$

από τις οποίες μπορούμε να συμπεράνουμε ότι, μετακινώντας το u από το S_i στο S_m, το ελάχιστο άθροισμα S_{min} δεν μειώνεται (αφού $\Sigma(S_m \cup \{u\}, A_{ext}) = S_m + a_u > S_m$) και το μέγιστο άθροισμα S_{max} δεν αυξάνεται (αφού $\Sigma(S_m \cup \{u\}, A_{ext}) < Q$ και υπάρχουν σύνολα, με άθροισμα $> Q$).

Άρα, μετακινώντας το u από το S_i στο S_m, προκύπτει μια νέα λύση S' με $\mathcal{R}(S', A_{ext}) \leq \mathcal{R}(S, A_{ext})$. Θα πρέπει ωστόσο να λάβουμε υπόψη την περίπτωση που το S_m είναι το S_1 και η μετακίνηση του u σε αυτό, καθιστά την S', μη εφικτή λύση για το k-PART$_{\text{R}}$ με Περιορισμο

Αντιγραφων. Αν κάτι τέτοιο συμβαίνει στην S', ορίζουμε ως p', το ελάχιστο μεταξύ των μεγιστικών στοιχείων των k συνόλων της νέας λύσης S' και ως S_1, το σύνολο που περιέχει το p'. Τότε, η S' είναι μια εφικτή λύση για το στιγμιότυπο (A_{ext}, p') του k-PART$_{\text{R}}$.

Αν τώρα το S' παραβιάζει ακόμη την ιδιότητα 2, εφαρμόζουμε και πάλι την ίδια διαδικασία της μετακίνησης, που περιγράψαμε παραπάνω, υποθέτοντας πάντα πως υπάρχει κατάλληλο S_m με $\Sigma(S_m, A_{ext}) \leq Q$, για τα νέα μεγέθη $Q' = \sum_{i=1}^{p'} a'_i$ και $q' = \max\{i \mid a'_i \leq Q'\}$. Εκτελώντας επαναληπτικά αυτή τη διαδικασία, το p δεν μειώνεται και ο λόγος της λύσης $\mathcal{R}(S', A_{ext})$ δεν αυξάνεται, όπως είδαμε και στην ανάλυση μια μεμονωμένης μετακίνησης. Παρατηρώντας ότι το p δεν μπορεί να υπερβεί το $N - k + 1$ (αφού είναι $p \leq N - k + 1$, εξ ορισμού), κατά την εκτέλεση αυτής της επαναληπτικής διαδικασίας, βλέπουμε πως κάποια στιγμή το p θα σταματήσει να αυξάνεται. Έστω p_f, αυτή η τελική τιμή "σταθεροποίησης" του p και έστω Q_f και q_f, οι αντίστοιχες τιμές των Q και q. Επαναλαμβάνοντας λοιπόν, αυτή τη μετακίνηση και τους επανορισμούς (του p σε p', κλπ), κάποια στιγμή θα καταλήξουμε σε μια εφικτή λύση S'' του στιγμιοτύπου (A_{ext}, p_f), του k-PART$_{\text{R}}$ με Περιορισμο Αντιγραφων.

Αν τώρα η εφικτή λύση S'' του (A_{ext}, p_f), που σχηματίσαμε (και ικανοποιεί την ιδιότητα 2), παραβιάζει την ιδιότητα 1, δηλαδή υπάρχει σύνολο S_i στην S'', που αποτελείται μόνο από στοιχεία $j \leq q_f$ και έχει άθροισμα $\Sigma(S_i, A_{ext}) \geq 2Q$, χρησιμοποιούμε την παρακάτω διαδικασία μετατροπής:

Έστω πως υπάρχει ένα κατάλληλο σύνολο S_m, με άθροισμα $\Sigma(S_m, A_{ext}) \leq Q_f$. Παίρνουμε το μικρότερο στοιχείο j από το S_i, και το μετακινούμε στο S_m. Παρατηρούμε ότι $a'_j \leq Q_f$ και έτσι προκύπτουν οι παρακάτω ανισότητες:

$$m(S'', A_{ext}) = \Sigma(S_m, A_{ext}) \leq Q_f \leq \Sigma(S_i \setminus \{j\}, A_{ext})$$

$$\Sigma(S_m \cup \{j\}, A_{ext}) \leq 2Q_f \leq \Sigma(S_i, A_{ext}) \leq M(S'', A_{ext})$$

Από την παραπάνω διαδικασία μπορούμε και πάλι να δούμε πως το ελάχιστο σύνολο δεν μειώνεται και το μέγιστο δεν αυξάνεται, έτσι μπορούμε να την εκτελέσουμε επαναλαμβανόμενα, υποθέτοντας πάντα πως υπάρχει κατάλληλο S_m με $\Sigma(S_m, A_{ext}) \leq Q_f$, μέχρις ότου να φτάσουμε σε μια λύση που να ικανοποιείται η ιδιότητα 1, όπως κάναμε και παραπάνω με τη μετατροπή για την ιδιότητα 2, αφού ισχύουν τα ίδια επιχειρήματα για την αύξηση του p και τη μη αύξηση του λόγου $\mathcal{R}(S'', A_{ext})$. Επιπρόσθετα, η μετατροπή θα διατηρεί την ιδιότητα 2, αφού το p δεν μειώνεται και προσθέτουμε στοιχεία μόνο στο σύνολο S_m με άθροισμα $\Sigma(S_m, A_{ext}) \leq Q_f$ (το οποίο δε μπορεί να είναι συνόλο που περιέχει στοιχείο $j > q$).

Συμπερασματικά, από κάθε εφικτή λύση S, του στιγμιοτύπου (A_{ext}, p) του k-PART$_{\text{R}}$ με Περιορισμο Αντιγραφων μπορούμε να οδηγηθούμε σε μια εφικτή λύση S''' ενός διαφορετικού στιγμιοτύπου $(A_{ext}, p_{\text{new}})$ του k-PART$_{\text{R}}$, η οποία ικανοποιεί και τις τρεις ιδιότητες και έχει ίσο ή μικρότερο λόγο $\frac{S_{max}}{S_{min}}$, αρκεί να τηρούνται οι προϋποθέσεις που περιγράψαμε κατά την απόδειξη. Έστω τέλος ότι εφαρμόζουμε τις παραπάνω μετατροπές σε μια βέλτιστη λύση του στιγμιοτύπου (A_{ext}, p) του k-PART$_{\text{R}}$, με το p να είναι *τέλειο* για το στιγμιότυπο A_{ext} του k-PART με Περιορισμο Αντιγραφων. Αφού ο λόγος της λύσης δεν αυξάνεται μέσω της κατασκευής μας, οποιοδήποτε νέο p', που αποκτάται από αυτή τη κατασκευή, θα πρέπει επίσης να είναι *τέλειο* και η λύση που θα προκύπτει να είναι επίσης βέλτιστη για το στιγμιότυπο (A_{ext}, p'). ■

Όταν τηρούνται λοιπόν οι προϋποθέσεις κατάλληλα μικρών S_m με $\Sigma(S_m, A_{ext}) \leq Q$, ο αλγόριθμος μας για το πρόβλημα k-PART με Περιορισμο Αντιγραφων, δουλεύει ακριβώς όπως και ο αλγόριθμος 1 για το k-PART. Τα λήμματα 1 και 2, ισχύουν ακριβώς όπως στην παράγραφο 3.1, και το ίδιο ισχύει και για το λήμμα 4 της παραγράφου 3.2.3. Για την εφικτότητα των λύσεων του αλγορίθμου 6, έχουμε το ακόλουθο λήμμα:

Λήμμα 7. *Κάθε πλειάδα k συνόλων με λόγο $\mathcal{R}(S_1, \ldots, S_k, A_r)$ που εξετάζεται από τον Αλγόριθμο 1, αποτελεί μια εφικτή λύση για το στιγμιότυπο (A_r, p) του k-PART$_{\text{R}}$* με Περιορισμο Αντιγραφων.

Απόδειξη. Η εφικτότητα όσον αφορά τα τμήματα του k-PART, του προβλήματος μας, αποδεικνύεται με τον ίδιο τρόπο όπως στο λήμμα 3. Για τη συνθήκη της μη ύπαρξης αντιγράφων σε ένα σύνολο S_i, το "τριμάρισμα" που εκτελείται στις γραμμές 12 και 28 του αλγορίθμου 2, εγγυάται πως κάθε μη εφικτή λύση του k-PART με Περιορισμο Αντιγραφων, που θα πήγαινε να δημιουργηθεί από τις μετακινήσεις στοιχείων, που αποτελούν αντίγραφο του ίδιου κρυπτοκειμένου, μεταξύ συνόλων της διαμέρισης, αποκόπτονται. Έτσι, κάθε λύση που εξετάζεται αποτελεί εφικτή λύση. ■

Βλέπουμε λοιπόν πως κάτω από αυτή την υπόθεση ότι πάντα υπάρχει κατάλληλα μικρό S_m για να πραγματοποιούμε τις μετακινήσεις που θέλουμε, ο αλγόριθμος 6 αποτελεί FPTAS για το πρόβλημα k-PART με Περιορισμο Αντιγραφων, καθώς τα θεωρήματα 2 και 3 ισχύουν με τον ίδιο τρόπο όπως και στην παράγραφο 3.2.3.

Η ορθότητα του αποτελέσματος μας βασίζεται στο ότι, μπορεί ο αλγόριθμος μας να τριμάρει λύσεις πολύ "βίαια", στην περίπτωση του προβλήματος k-PART με Περιορισμο Αντιγραφων, όμως οι λύσεις που θα "επιβιώσουν" από το τριμάρισμα, αποτελούν πάντα εφικτές λύσεις του προβλήματος k-PART με Περιορισμο Αντιγραφων, και μάλιστα λύσεις με λόγο $\mathcal{R}(S_{alg}, A_{ext}) \leq (1+\varepsilon)\mathcal{R}(S^*, A_{ext})$.

Κεφάλαιο 5

Ανοιχτά Προβλήματα

Σε αυτή την ενότητα θα παρουσιάσουμε ανοιχτά προβλήματα που προκύπτουν από τα αποτελέσματά μας.

Ανοιχτό παραμένει το ερώτημα που εξετάζουν οι παράγραφοι 4.3.1 έως 4.3.3, του εάν υπάρχει δηλαδή τρόπος να βρεθεί μια απόδειξη για την εικασία 1 της παραγράφου 4.3.2. Μια 1-1 επέκταση των αποτελεσμάτων της ενότητας 3, για το πρόβλημα k-PART με Περιορισμο Αντιγραφων, θα ήταν πολύ χρήσιμη, καθώς θα έδινε ένα FPTAS για την εκδοχή του προβλήματος Multiway Number Partitioning, με περιορισμό αντιγράφων.

Εικάζουμε πως ίσως κάτι τέτοιο δεν είναι εφικτό με την παρούσα μορφή του FPTAS του αλγορίθμου 1, και ίσως η λύση μας να απαιτεί ριζική αναθεώρηση του αλγορίθμου και των θεωρημάτων ορθότητας του. Μια ιδέα που ίσως να είναι υποσχόμενη, είναι η εγκατάλειψη της χρήσης διανυσμάτων διαφορών $D = [d_2, d_3, \ldots, d_k]$, κάτι που όμως θα εξέταζε το πρόβλημα υπό νέο πρίσμα, και θα απαιτούσε μια νέα λύση χτισμένη "από το 0".

Ενδιαφέρον παρουσιάζει και μια εκδοχή της αντιμετώπισης του σεναρίου της υποενότητας 4.2, για την περίπτωση που $t \nmid k$. Μια πρώτη ιδέα σε αυτή την περίπτωση είναι πως αρχικά οι μαθηματικοί είναι $k = (\lfloor k/t \rfloor) * t + k \bmod t$. Μπορούμε λοιπόν να ορίσουμε $k' = \lfloor k/t \rfloor$ ομάδες και να εξετάσουμε σενάρια για να αξιοποιήσουμε την υπολογιστική δυνατότητα των $k \bmod t$ μαθηματικών που περισσεύουν.

Μια πρώτη σκέψη θα ήταν να χρησιμοποιήσουμε τους $k \bmod t$ περισσευόμενους μαθηματικούς για να υποστηρίξουν τις ομάδες με το μεγαλύτερο φόρτο αποκρυπτογράφησης. Κάτι τέτοιο ωστόσο, θα περιείχε τον κίνδυνο να αδικηθούν σημαντικά οι υπόλοιπες ομάδες, σε κατάλληλα "κακά" στιγμιότυπα εισόδου, οπότε απαιτείται εκτενέστερη μελέτη με πιθανοτική ανάλυση.

Μια άλλη ιδέα θα ήταν να χωρίσουμε τους μαθηματικούς σε $k' = \lceil k/t \rceil$ ομάδες, από τις οποίες $\lfloor k/t \rfloor$ από αυτές θα έχουν t μαθηματικούς και $k' - \lfloor k/t \rfloor$ από αυτές θα έχουν $t - 1$ μαθηματικούς. Αφού οι k' ομάδες θα ολοκληρώσουν το έργο τους, θα χρειαζόμαστε ακόμα $k' - \lfloor k/t \rfloor$ μερικές αποκρυπτογραφήσεις, τις οποίες πρέπει να αναθέσουμε σε κάποιες από τις k οντότητες.

Ποσο δίκαιο θα ήταν να πάρουμε, για παράδειγμα, τους μαθηματικούς με τον λιγότερο φόρτο εργασίας και να τους αναθέσουμε να κάνουν τις εναπομείνασες $(k' - \lfloor k/t \rfloor)$ αποκρυπτογραφήσεις; Και πάλι υπάρχει περιθώριο για πολλές προσεγγίσεις και εκτενείς αναλύσεις, του πιο αποτελεσματικού τρόπου διαμοιρασμού, του υπολειπόμενου απαιτούμενου υπολογιστικού φόρτου.

Μια ακόμα ενδιαφέρουσα σκέψη για μελλοντική δουλειά, θα ήταν μια πιθανοτική ανάλυση της χαλάρωσης του ορίου προς τα πάνω, για ακριβώς t αντίγραφα του c_i. Αν δηλαδή φτιάχναμε $t' > t$ αντίγραφα κάθε κρυπτοκειμένου, και δίναμε αυτό το σύνολο A'_{Ext} ως είσοδο στον αλγόριθμο 1, χωρίς να εφαρμόσουμε περιορισμούς διπλής ανάθεσης αντιγράφων, ποιο θα ήταν το ελάχιστο $t' > t$ πλήθος αντιγράφων, που θα έπρεπε φτιάξουμε και να δώσουμε ως είσοδο στον A_{ext}, ούτως ώστε να παίρναμε τουλάχιστον t μερικές αποκρυπτο-

γραφήσεις κάθε κρυπτοκειμένου (και έτσι όλα τα αρχικά κείμενα, συνδυάζοντας τες), με καλή πιθανότητα;

Για τη βελτίωση του κόστους αποκρυπτογραφήσεων που έχει αναλάβει η κάθε οντότητα, θα μπορούσαμε να βάζαμε κάθε οντότητα να κρατάει τους δείκτες των κρυπτοκειμένων για τα οποία έχει δώσει μερική αποκρυπτογράφηση, και να μην αποκρυπτογραφεί ξανά τα κρυπτοκείμενα που έχει ήδη αποκρυπτογραφήσει, μειώνοντας έτσι το συνολικό κόστος αποκρυπτογραφήσης που έχει αναλάβει κάθε οντότητα. Για άλλη μια φορά, αναδύονται πολύ ενδιαφέροντα ερωτήματα, της μελέτης της επίδρασης τέτοιων χειρισμών, στον βέλτιστο προσεγγιστικά λόγο, $\mathcal{R}(S_{alg}, A'_{Ext})$ που βρίσκουμε από το FPTAS.

Παράρτημα

Α΄ Διανομή Απορρήτων (Secret Sharing)

Στην παρούσα ενότητα θα παρουσιάσουμε το θεωρητικό υπόβαθρο των κρυπτογραφικών ορολογιών, που χρησιμοποιούνται στη βιβλιογραφία για τα κρυπτοσυστήματα κατωφλίου. Για μια αναλυτικότερη παρουσίαση των θεμάτων, δείτε τα [50, 26], από τα οποία και εμπνευστήκαμε.

Ορισμός 13 (Σχημα Διανομης Απορρητου). Ένα **σχήμα διανομής απορρήτων** παρέχει έναν τρόπο για την επίλυση του εξής προβλήματος: Έστω ότι έχουμε l πρόσωπα (οντότητες) P_i. Κάθε πρόσωπο P_i έχει στην κατοχή του κάποια πληροφορία p_i η οποία είναι άγνωστη σε όλους τους άλλους P_j, όπου $j \neq i$. Θέλουμε να δώσουμε ένα σχήμα με το οποίο κάποια πληροφορία s μπορεί εύκολα να υπολογιστεί από κάθε t-άδα εκ των p_i, αλλά η γνώση λιγότερων από $t-1$ από τα p_i, δεν αρκεί για να υπολογιστεί το s. Ο κάτοχος της πληροφορίας αυτής ονομάζεται **διανομέας (dealer)**.

Ορισμός 14 (Σχημα κατωφλιου). Έστω t, l θετικοί ακέραιοι, με $t \leq l$. Το σύνολο $P = \{p_1, \ldots, p_l\}$ λέγεται (t, l)-σχήμα κατωφλίου αν ισχύουν τα παρακάτω:

- Το s μπορεί να υπολογιστεί εύκολα από οποιαδήποτε t στοιχεία του συνόλου A.
- Η γνώση οποιωνδήποτε $t-1$ στοιχείων του A δεν αρκεί για να υπολογιστεί το s.

Α΄.1 Διανομή Απορρήτων Shamir (Shamir's Secret Sharing)

Η πρώτη μέθοδος υλοποίησης ενός τέτοιου σχήματος προτάθηκε από τον Shamir στο [44]. Επιτρέπει τον διαμοιρασμό ενός τέτοιου μυστικού στοιχείου s μεταξύ l παικτών, ώστε οποιοδήποτε υποσύνολο από t να μπορεί να το ανακτήσει, αλλά όχι οποιοδήποτε υποσύνολο από $t-1$ οντότητες. Για να το πετύχει αυτό, χρησιμοποιεί την ιδέα της πολυωνυμικής παρεμβολής, με τη μέθοδο των συντελεστών Lagrange:

- Ένα πολυώνυμο P βαθμού $t-1$ μπορεί να ανακατασκευαστεί από t σημεία $\{(x_i, y_i)\}_{i=1}^{t}$.
- $P(x) = \sum_{i=1}^{t} y_i \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{x - x_j}{x_i - x_j}$

Κατά συνέπεια:

- Ο διανομέας διαλέγει ένα τυχαίο πολυώνυμο P βαθμού $t-1$ ώστε $P(0) = s$.
- Παράγει και διανέμει l ζεύγη $(x_i, P(x_i))$, με $x_i \neq 0$.
- Σύμφωνα με την πολυωνυμική παρεμβολή, t οντότητες μπορούν να συνεννοηθούν και να ανακτήσουν το s, όχι όμως $t-1$.

Για τη διανομή ενός ακέραιου μυστικού, το παραπάνω σχήμα μπορεί να εφαρμοστεί στο $\mathbb{Z}_p$, όπου p είναι ένας πρώτος μεγαλύτερος από το μυστικό. Έτσι όλοι οι συντελεστές επιλέγονται τυχαία από το $\mathbb{Z}_p$ και όλες οι αριθμητικές πράξεις γίνονται $\mathrm{mod} p$.

Το παραπάνω σχήμα υποθέτει ότι όλες οι οντότητες είναι τίμιες. Κάτι τέτοιο όμως μπορεί να μην ισχύει πάντοτε [42]:

- Ο διανομέας μπορεί να δώσει λανθασμένα μερίδια σε όλους ή σε τμήμα των παικτών. Αυτό θα έχει ως συνέπεια να μην μπορεί να ανακατασκευαστεί το μυστικό. Για να αποφευχθεί κάτι τέτοιο, πρέπει να δοθεί δυνατότητα στις οντότητες να επαληθεύσουν την ορθότητα των μεριδίων τους.

- Κάποια οντότητα μπορεί να μην παρέχει τα σωστά μερίδια κατά τη διάρκεια της ανακατασκευής. Πρέπει λοιπόν, όλές να μπορούν να ελέγξουν ότι είναι αυτά που δόθηκαν αρχικά από τον διανομέα.

Για να αντιμετωπιστούν οι παραπάνω απειλές, προτάθηκαν κάποιες παραλλαγές του [44].

Α΄.2 Επαληθεύσιμη Διανομή Απορρήτων [19]

Ο διανομέας εισάγει επιπλέον πληροφορία ώστε να είναι δυνατή η επαλήθευση της ορθότητας των μεριδίων. Πιο συγκεκριμένα, υπολογίζει και διανέμει δεσμεύσεις στους συντελεστές του πολυωνύμου. Αν δηλαδή $P(x) = \sum_{i=0}^{t} a_i x^i$, ο διανομέας μοιράζει τις δεσμεύσεις $\{c_i = g^{a_i}\}_{i=0}^{t}$, όπου g είναι ο γεννήτορας μιας ομάδας με δύσκολο πρόβλημα διακριτού λογαρίθμου.

Για να επαληθεύσει κανείς ότι το μερίδιο (x_i, y_i) είναι έγκυρο, δηλαδή ότι ισχύει $y_i = P(x_i)$, ελέγχει ότι:

$$g^{y_i} = \prod_{j=0}^{t} c_j^{x_i^j}$$

το οποίο πρέπει να είναι έγκυρο καθότι:

$$\prod_{j=0}^{t} c_j^{x_i^j} = \prod_{j=0}^{t} g^{a_j x_i^j} = g^{\sum_{j=0}^{t} a_j x_i^j} = g^{P(x_i)} = g^{y_i}$$

Α΄.3 Public Verifiable Secret Sharing

Για να αντιμετωπίσουμε την περίπτωση όπου μια οντότητα δεν παρουσιάζει τα σωστά μερίδια κατά τη φάση της ανακατασκευής, μπορεί να χρησιμοποιηθεί το παρακάτω πρωτόκολλο, το οποίο προτάθηκε στο [42] και αποτελείται από τις φάσεις της διανομής και της ανακατασκευής:

Διανομή

- Υποθέτουμε πάλι ότι το πολυώνυμο είναι $P(x) = \sum_{i=0}^{t} a_i x^i$.

- Κάθε χρήστης έχει ένα μυστικό κλειδί sk_i και ένα δημόσιο κλειδί $pk_i = G^{sk_i}$, όπου G είναι ένας γεννήτορας μιας ομάδας με δύσκολο πρόβλημα DLOG. Εκτός του G, χρησιμοποιείται και ένας άλλος γεννήτορας g.

- Ο διανομέας δεσμεύεται στους συντελεστές: $\{c_i = g^{a_i}\}_{i=0}^{t}$

- Στη συνέχεια, κρυπτογραφεί τα μερίδια χρησιμοποιώντας το δημόσιο κλειδί κάθε συμμετέχοντα: $\{Y_i = pk_i^{p(i)}\}_{i=1}^n$
- Ο διανομέας χρησιμοποιεί το πρωτόκολλο Chaum Pedersen [11] για να δείξει ότι τα μερίδια είναι έγκυρα. Συγκεκριμένα, αποδεικνύει ότι:

$$\log_g \left(\prod_{j=0}^{t} c_j^{x_i^j} \right) = \log_{pk_i} Y_i = p(i), \quad \text{για } i = 1, \ldots, n$$

Ανακατασκευή

- Η αποκρυπτογράφηση των μεριδίων γίνεται με τα ιδιωτικά κλειδιά:

$$Y_i^{1/sk_i} = pk_i^{p(i)/sk_i} = G^{p(i)}$$

- Το μυστικό ανακατασκευάζεται χρησιμοποιώντας τους συντελεστές Lagrange.

Β΄ (k, t) Threshold ElGamal χωρίς την ύπαρξη έμπιστου διανομέα

Όπως είδαμε και στο κυρίως κείμενο, στην παράγραφο 2, σε πολλές εφαρμογές, που μπορεί να μην επιθυμούμε την ύπαρξη έμπιστου διανομέα, μπορούμε να χρησιμοποιήσουμε την κατασκευή που περιγράφεται στο [39] ως εξής:

- Κάθε οντότητα i διαλέγει $x_i \in_R \mathbb{Z}_q$ ως το μερίδιό της.
- Επιπλέον διαλέγει το πολυώνυμο $f_i(z) = \sum_{j=0}^{t-1} f_{ij} z^j$ με $f_{i0} = x_i$, $f_{ij} \in_R \mathbb{Z}_q$.
- Για κάθε συντελεστή δημοσιοποιεί $F_{ij} = g^{f_{ij}}$.
- Κάθε οντότητα i στέλνει το $s_{ij} = f_i(j)$ στον συμμετέχοντα j.
- Όλοι επαληθεύουν τα μερίδιά τους ελέγχοντας αν $g^{s_{ij}} = \prod_{l=0}^{t-1} F_{jl}^i$.
- Κάθε οντότητα i δεσμεύεται στα μερίδια δίνοντας $y_i = g^{x_i}$.

Η κρυπτογράφηση προχωράει όπως στο ElGamal.

Για την αποκρυπτογράφηση του (G, M), κάθε οντότητα υπολογίζει $w_i = G^{x_i}$. Το μήνυμα δίνεται από τη σχέση:

$$M = \frac{M}{\prod_{i \in \Lambda} w_i^{\lambda_i}}$$

Επαληθεύεται ότι $x_i = \log_G w_i = \log_g y_i$ με το πρωτόκολλο Chaum-Pedersen [11].

Γ΄ Συναρτήσεις στόχου για το Multiway Number Partitioning

Υπάρχουν τέσσερις διαφορετικές συναρτήσεις στόχου (objective functions) που χρησιμοποιούνται για το πρόβλημα Multiway Number Partitioning, στη βιβλιογραφία, κάθε μία με εφαρμογές σε διαφορετικούς τομείς:

1. Ελαχιστοποίηση του μέγιστου αθροίσματος. Είναι γνωστή και ως ελαχιστοποίηση του makespan και χρησιμοποιείται στο πρόβλημα Ελάχιστου Χρόνου Ολοκλήρωσης στον προγραμματισμό ίδιων μηχανών [41, 22]. Επίσης, εμφανίζεται σε προβλήματα bin packing [31].

2. Μεγιστοποίηση του ελάχιστου αθροίσματος. Η συνάρτηση αυτή έχει μελετηθεί σε εργασίες σχετικές με προγραμματισμό [18] και bin packing [31], καθώς και σε μελέτες που σχετίζονται με το *maximin share* [21, 30, 10], το οποίο αποτελεί κριτήριο δίκαιης κατανομής αντικειμένων. Σημειώνεται ότι αλγόριθμοι για τη συνάρτηση στόχου (2) μπορούν να τροποποιηθούν ώστε να λειτουργούν και για την (1) και αντιστρόφως [29].

3. Ελαχιστοποίηση της διαφοράς μεταξύ του μεγαλύτερου και του μικρότερου αθροίσματος. Η συνάρτηση αυτή είναι λιγότερο συνηθισμένη, αλλά έχει μελετηθεί σε εργασίες σχετικές με το Multiway Number Partitioning, όπως οι [34, 29]. Δεν υπάρχει FPTAS για αυτήν τη συγκεκριμένη συνάρτηση στόχου [3, 46].

4. Ελαχιστοποίηση του λόγου μεταξύ του μεγαλύτερου και του μικρότερου αθροίσματος (π.χ. [2]). Αν και αυτή η συνάρτηση συχνά παραλείπεται σε μελέτες για το Multiway Number Partitioning (π.χ. [29, 43]), έχει μελετηθεί στον τομέα της δίκαιης κατανομής, υπό τη μορφή του προβλήματος Minimum Envy-Ratio [32, 37], καθώς και στον προγραμματισμό [17].

Σε αυτή την εργασία, εξετάζουμε τη (4) ως τη συνάρτηση στόχου για το πρόβλημα k-way Number Partitioning Ratio. Παρατηρούμε ότι και οι τέσσερις συναρτήσεις είναι ισοδύναμες όταν $k = 2$· αυτό δεν ισχύει για $k \geq 3$. Αυτό είναι γνωστό για τις τρεις πρώτες συναρτήσεις [29], αλλά για πληρότητα θα παρουσιάσουμε κάποια αντιπαραδείγματα που αποδεικνύουν ότι η (4) δεν είναι ισοδύναμη με καμία από τις άλλες τρεις, όταν $k \geq 3$.

Είναι εύκολο να βρει κανείς αντιπαραδείγματα για τις δύο πρώτες. Σκεφτείτε το σύνολο εισόδου $\{1, 2, 3, 10\}$ για $k = 3$. Η διαμέριση $(\{1\}, \{2, 3\}, \{10\})$ ελαχιστοποιεί το μέγιστο άθροισμα, αλλά δεν επιτυγχάνει τον βέλτιστο λόγο. Ομοίως, για το σύνολο $\{5, 5, 5, 10\}$ με $k = 3$, η διαμέριση $(\{5\}, \{5\}, \{5, 10\})$ μεγιστοποιεί το ελάχιστο άθροισμα, αλλά δεν έχει τον βέλτιστο λόγο.

Τέλος, εξετάζουμε την είσοδο $\{16, 16, 18, 20, 24, 27, 29, 40\}$ για $k = 4$. Ο παρακάτω πίνακας συγκρίνει τη λύση που ελαχιστοποιεί τη διαφορά με αυτή που ελαχιστοποιεί τον λόγο.

Λύση	**Αθροίσματα**	**Διαφορά**	**Λόγος**
$\{40\}, \{16, 16, 18\}, \{24, 27\}, \{29, 20\}$	{40, 50, 51, 49}	**11**	1.275
$\{40, 16\}, \{24, 20\}, \{16, 29\}, \{18, 27\}$	{56, 44, 45, 45}	12	**1.27273**

Πίνακας 1: Σύγκριση λύσεων για το πρόβλημα Multiway Number Partitioning

Δ΄ Ψευδοκώδικας για τη συνάρτηση `update`

Σε αυτή την ενότητα παρουσιάζουμε την πλήρη περιγραφή της συνάρτησης `update`, η οποία καλείται από τον Αλγόριθμο 2. Ο σκοπός της εξηγείται στην υποενότητα 3.2.2.

Αλγόριθμος 9 update$(a_i, i, j, D, V, S_1, \ldots, S_k)$

Είσοδος: Ένα στοιχείο a_i, ο δείκτης του i, ένας δείκτης j $(1 \leq j \leq k)$, ένας (ταξινομημένος) πίνακας διαφορών $D = [d_2, \ldots, d_k]$, ένας Boolean πίνακας εγκυρότητας (ή πίνακας ορθότητας) $V = [v_2, \ldots, v_k]$, και k ξένα μεταξύ τους υποσύνολα S_j που περιέχουν στοιχεία στο $[i-1]$.

Έξοδος: k ενημερωμένα υποσύνολα S_j, τον ενημερωμένο (ταξινομημένο) πίνακα διαφορών D και τον πίνακα εγκυρότητας V.

if j == 1 **then**
  $S_1 = S_1 \cup \{i\}$,
  for all d_j in D **do**
    $d_j = d_j + a_i$
  end for
  return $(S_1, \ldots, S_k),\ D,\ V$
end if
$S_j = S_j \cup \{i\},\ d_j = d_j - a_i$
if $i > p$ **then**
  $v_j =$ true
end if
while $d_{j-1} > d_j$ **do** ▷ Ενάλλαξε τα σύνολα και τα αντίστοιχα τους D, V
  $tempS = S_j,\ tempD = d_j,\ tempV = v_j$
  $S_j = S_{j-1},\ d_j = d_{j-1},\ v_j = v_{j-1}$
  $S_{j-1} = tempS,\ d_{j-1} = tempD,\ v_{j-1} = tempV$
  $j = j - 1$
end while
return $(S_1, \ldots, S_k),\ D,\ V$

Από τα παραπάνω, έπεται ότι η συνάρτηση `update` λειτουργεί όπως περιγράφεται στην υποενότητα 3.2.2. Μπορεί να επαληθευτεί ότι εκτελείται σε χρόνο $O(k)$, καθώς στη χειρότερη περίπτωση πρέπει να τροποποιήσει κάθε θέση των διανυσμάτων *διαφοράς* και *εγκυρότητας*.

Ε΄ Απόδειξη της ανισότητας του Θεωρήματος 2

Σε αυτή την ενότητα θα αποδείξουμε την ανισότητα

$$\mathcal{R}(S_{alg}, A) \leq (1+\varepsilon)\mathcal{R}(S^*, A)$$

που είναι απαραίτητη για την απόδειξη του θεωρήματος 2

Έστω $(A = \{a_1, \ldots, a_n\}, p)$ ένα στιγμιότυπο του k-PART$_\text{R}$ και $\varepsilon \in (0,1)$ μία παράμετρος σφάλματος. Ορίζουμε μια παράμετρο $\delta = \frac{\varepsilon \cdot a_p}{3 \cdot n}$, που θα χρησιμοποιήσουμε για την κλιμάκωση της εισόδου (scale factor).

Έστω $A_r = \{a_i^r = \lfloor \frac{a_i}{\delta} \rfloor :$ για κάθε $i \in \{1, \ldots, n\}\}$. Έστω $S_{alg} = (S_1, \ldots, S_k)$ η λύση που επιστρέφει ο Αλγόριθμος 1 και $S_{opt} = (S_1^*, \ldots, S_k^*)$ μία βέλτιστη λύση για το στιγμιότυπο (A, p). Ορίζουμε

$$S_M = \arg\max_{S_i \in S_{alg}} \Sigma(S_i, A), \quad S_m = \arg\min_{S_i \in S_{alg}} \Sigma(S_i, A),$$

$$S_M^* = \arg\max_{S_i^* \in S_{opt}} \Sigma(S_i^*, A), \quad S_m^* = \arg\min_{S_i^* \in S_{opt}} \Sigma(S_i^*, A).$$

Συνεχίζουμε αποδεικνύοντας τα ακόλουθα λήμματα.

Λήμμα 8. *Για κάθε* $S \in \{S_M, S_m, S_M^*, S_m^*\}$ *ισχύει ότι*

$$\sum_{i \in S} a_i - n \cdot \delta \leq \sum_{i \in S} \delta \cdot a_i^r \leq \sum_{i \in S} a_i.$$

Απόδειξη. Υπενθυμίζουμε ότι $a_i^r = \lfloor \frac{a_i}{\delta} \rfloor$ για κάθε $i \in \{1, \ldots, n\}$. Άρα

$$\frac{a_i}{\delta} - 1 \leq a_i^r \leq \frac{a_i}{\delta} \Rightarrow a_i - \delta \leq \delta \cdot a_i^r \leq a_i \Rightarrow \sum_{i \in S}(a_i - \delta) \leq \sum_{i \in S} \delta \cdot a_i^r \leq \sum_{i \in S} a_i.$$

Αφού $S \in \{S_M, S_m, S_M^*, S_m^*\}$, έχουμε $|S| \leq n$, άρα αποδεικνύεται το λήμμα. ■

Λήμμα 9. *Για κάθε* $S \in \{S_M, S_m, S_M^*, S_m^*\}$ *ισχύει ότι*

$$n \cdot \delta \leq \frac{\varepsilon}{3} \cdot \sum_{i \in S} a_i.$$

Απόδειξη. Παρατηρούμε ότι $\max S \geq p$ για κάθε $S \in \{S_M, S_m, S_M^*, S_m^*\}$, λόγω του ορισμού του k-PART$_\mathrm{R}$. Εφόσον η είσοδος είναι ταξινομημένη, ισχύει ότι

$$n \cdot \delta = \frac{\varepsilon \cdot a_p}{3} \leq \frac{\varepsilon}{3} \cdot \sum_{i \in S} a_i.$$

■

Λήμμα 10. *Ισχύει ότι* $\mathcal{R}(S_{alg}, A) \leq \mathcal{R}(S_{alg}, A_r) + \varepsilon/3$.

Απόδειξη. Μπορούμε να δείξουμε ότι

$$\begin{aligned}
\mathcal{R}(S_{alg}, A) = \frac{\sum_{i \in S_M} a_i}{\sum_{j \in S_m} a_j} &\leq \frac{\sum_{i \in S_M} \delta \cdot a_i^r + \delta \cdot n}{\sum_{j \in S_m} a_j} && \text{[Λήμμα 8]} \\
&\leq \frac{\sum_{i \in S_M} a_i^r}{\sum_{j \in S_m} a_j^r} + \frac{\delta \cdot n}{\sum_{j \in S_m} a_j}. && \text{[Λήμμα 8]}
\end{aligned}$$

Από το Λήμμα 9 έχουμε ότι $\delta \cdot n \leq (\varepsilon/3) \cdot \sum_{j \in S_m} a_j$. Άρα

$$\mathcal{R}(S_{alg}, A) \leq \frac{\sum_{i \in S_M} a_i^r}{\sum_{j \in S_m} a_j^r} + \frac{\varepsilon}{3} = \mathcal{R}(S_{alg}, A_r) + \frac{\varepsilon}{3}. \qquad \text{[ορισμός 3]}$$

■

Λήμμα 11. *Για κάθε* $\varepsilon \in (0, 1)$, *ισχύει ότι*

$$\mathcal{R}(S_{opt}, A_r) \leq \left(1 + \frac{\varepsilon}{2}\right) \cdot \mathcal{R}(S_{opt}, A).$$

Απόδειξη. Μπορούμε να αποδείξουμε ότι

$$\begin{aligned}
\mathcal{R}(S_{opt}, A_r) &= \frac{\sum_{i \in S_M^*} a_i^r}{\sum_{j \in S_m^*} a_j^r} \\
&\leq \frac{\sum_{i \in S_M^*} a_i}{\sum_{j \in S_m^*} a_j - n \cdot \delta} && \text{[Λήμμα 8]} \\
&= \frac{\sum_{j \in S_m^*} a_j}{\sum_{j \in S_m^*} a_j - n \cdot \delta} \cdot \frac{\sum_{i \in S_M^*} a_i}{\sum_{j \in S_m^*} a_j} \\
&= \left(1 + \frac{n \cdot \delta}{\sum_{j \in S_m^*} a_j - n \cdot \delta}\right) \cdot \frac{\sum_{i \in S_M^*} a_i}{\sum_{j \in S_m^*} a_j}.
\end{aligned}$$

Χρησιμοποιώντας το Λήμμα 9 για $S = S_m^*$ και απλοποιώντας το κλάσμα, παίρνουμε

$$\mathcal{R}(S_{opt}, A_r) \leq \left(1 + \frac{\varepsilon}{3-\varepsilon}\right) \cdot \frac{\sum_{i \in S_M^*} a_i}{\sum_{j \in S_m^*} a_j}.$$

Αφού $\varepsilon \in (0, 1)$, έχουμε

$$\begin{aligned} \mathcal{R}(S_{opt}, A_r) &\leq \left(1 + \frac{\varepsilon}{2}\right) \cdot \frac{\sum_{i \in S_M^*} a_i}{\sum_{j \in S_m^*} a_j} \\ &= \left(1 + \frac{\varepsilon}{2}\right) \cdot \mathcal{R}(S_{opt}, A). \qquad \text{[ορισμός 3]} \end{aligned}$$

■

Είμαστε πλέον έτοιμοι να αποδείξουμε το Θεώρημα 9.

Θεώρημα 9 (Προσέγγιση k-PART$_\mathrm{R}$). *Έστω $(S_1, \ldots, S_k)$ τα σύνολα που επιστρέφει ο Αλγόριθμος 1 για ένα στιγμιότυπο του k-PART$_\mathrm{R}$ με $(A = \{a_1, \ldots, a_n\}, p)$ και παράμετρο σφάλματος ε. Έστω $(S_1^*, \ldots, S_k^*)$ μια βέλτιστη λύση για το ίδιο στιγμιότυπο. Τότε*

$$\mathcal{R}(S_1, \ldots, S_k, A) \leq (1 + \varepsilon) \cdot \mathcal{R}(S_1^*, \ldots, S_k^*, A).$$

Απόδειξη. Η απόδειξη προκύπτει άμεσα από τα Λήμματα 10 και 11. Για απλότητα γράφουμε $S_{alg} = (S_1, \ldots, S_k)$ και $S_{opt} = (S_1^*, \ldots, S_k^*)$. Επίσης, χρησιμοποιούμε το γεγονός ότι το S_{alg} είναι βέλτιστη λύση για την περίπτωση (A_r, p), σύμφωνα με τα θεωρήματα του κυρίως κειμένου.

$$\begin{aligned} \mathcal{R}(S_{alg}, A) &\leq \mathcal{R}(S_{alg}, A_r) + \frac{\varepsilon}{3} \leq \mathcal{R}(S_{opt}, A_r) + \frac{\varepsilon}{3} \\ &\leq \left(1 + \frac{\varepsilon}{2}\right) \cdot \mathcal{R}(S_{opt}, A) + \frac{\varepsilon}{3} \\ &\leq (1 + \varepsilon) \cdot \mathcal{R}(S_{opt}, A) \end{aligned}$$

■

Βιβλιογραφία

[1] Georgios Amanatidis, Aris Filos-Ratsikas, and Alkmini Sgouritsa. Pushing the frontier on approximate efx allocations. In *Proceedings of the 25th ACM Conference on Economics and Computation*, EC '24, page 1268–1286, New York, NY, USA, 2024. Association for Computing Machinery.

[2] Antonis Antonopoulos, Aris Pagourtzis, Stavros Petsalakis, and Manolis Vasilakis. Faster algorithms for k-subset sum and variations. *J. Comb. Optim.*, 45(1):24, 2023.

[3] Cristina Bazgan, Miklos Santha, and Zsolt Tuza. Efficient approximation algorithms for the SUBSET-SUMS EQUALITY problem. *J. Comput. Syst. Sci.*, 64(2):160–170, 2002.

[4] Richard E. Bellman. *Dynamic programming*. Princeton University Press, Princeton, NJ, 1957.

[5] Samuel Bismuth, Vladislav Makarov, Erel Segal-Halevi, and Dana Shapira. Partitioning problems with splittings and interval targets. In Julián Mestre and Anthony Wirth, editors, *35th International Symposium on Algorithms and Computation, ISAAC 2024, December 8-11, 2024, Sydney, Australia*, volume 322 of *LIPIcs*, pages 12:1–12:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024.

[6] Dan Boneh, Rosario Gennaro, Steven Goldfeder, Aayush Jain, Sam Kim, Peter M. R. Rasmussen, and Amit Sahai. Threshold cryptosystems from threshold fully homomorphic encryption. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 565–596. Springer, 2018.

[7] Karl Bringmann. A near-linear pseudopolynomial time algorithm for subset sum. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1073–1084. SIAM, 2017.

[8] Karl Bringmann. Approximating subset sum ratio faster than subset sum. In *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024, Alexandria, VA, USA, January 7-10, 2024*, pages 1260–1277. SIAM, 2024.

[9] Karl Bringmann and Vasileios Nakos. A fine-grained perspective on approximating subset sum and partition. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 1797–1815. SIAM, 2021.

[10] Eric Budish. The combinatorial assignment problem: approximate competitive equilibrium from equal incomes. In Moshe Dror and Greys Sosic, editors, *Proceedings of the Behavioral and Quantitative Game Theory - Conference on Future Directions, BQGT '10, Newport Beach, California, USA, May 14-16, 2010*, page 74:1. ACM, 2010.

[11] David Chaum and Torben P. Pedersen. Wallet databases with observers. In Ernest F. Brickell, editor, *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*, volume 740 of *Lecture Notes in Computer Science*, pages 89–105. Springer, 1992.

[12] Lin Chen, Jiayi Lian, Yuchen Mao, and Guochuan Zhang. Approximating partition in near-linear time. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, STOC 2024, page 307–318, New York, NY, USA, 2024. Association for Computing Machinery.

[13] Lin Chen, Jiayi Lian, Yuchen Mao, and Guochuan Zhang. An improved pseudopolynomial time algorithm for subset sum. In *65th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2024, Chicago, IL, USA, October 27-30, 2024*, pages 2202–2216. IEEE, 2024.

[14] Lin Chen, Jiayi Lian, Yuchen Mao, and Guochuan Zhang. A note on deterministic fptas for partition, 2025.

[15] Mark Cieliebak and Stephan J. Eidenbenz. Measurement errors make the partial digest problem np-hard. In Martin Farach-Colton, editor, *LATIN 2004: Theoretical Informatics, 6th Latin American Symposium, Buenos Aires, Argentina, April 5-8, 2004, Proceedings*, volume 2976 of *Lecture Notes in Computer Science*, pages 379–390. Springer, 2004.

[16] Mark Cieliebak, Stephan J. Eidenbenz, and Paolo Penna. Noisy data make the partial digest problem np-hard. In Gary Benson and Roderic D. M. Page, editors, *Algorithms in Bioinformatics, Third International Workshop, WABI 2003, Budapest, Hungary, September 15-20, 2003, Proceedings*, volume 2812 of *Lecture Notes in Computer Science*, pages 111–123. Springer, 2003.

[17] Ed Coffman and Michael Langston. A performance guarantee for the greedy set-partitioning algorithm. *Acta Informatica*, 21:409–415, 11 1984.

[18] Bryan Deuermeyer, Donald Friesen, and Michael Langston. Scheduling to maximize the minimum processor finish time in a multiprocessor system. *SIAM Journal on Algebraic and Discrete Methods*, 3, 06 1982.

[19] Paul Feldman. A practical scheme for non-interactive verifiable secret sharing. In *28th Annual Symposium on Foundations of Computer Science, Los Angeles, California, USA, 27-29 October 1987*, pages 427–437. IEEE Computer Society, 1987.

[20] George Gens and Eugene Levner. A fast approximation algorithm for the subset-sum problem. *INFOR: Information Systems and Operational Research*, 32(3):143–148, 1994.

[21] Theodore P. Hill. Partitioning General Probability Measures. *The Annals of Probability*, 15(2):804 – 813, 1987.

[22] Dorit S. Hochbaum and David B. Shmoys. Using dual approximation algorithms for scheduling problems theoretical and practical results. *J. ACM*, 34(1):144–162, January 1987.

[23] Jeongdae Hong, Jinil Kim, Jihye Kim, Matthew K. Franklin, and Kunsoo Park. Fair threshold decryption with semi-trusted third parties. *Int. J. Appl. Cryptogr.*, 2(2):139–153, 2010.

[24] Ce Jin and Hongxun Wu. A simple near-linear pseudopolynomial time randomized algorithm for subset sum. In Jeremy T. Fineman and Michael Mitzenmacher, editors,

2nd Symposium on Simplicity in Algorithms, SOSA 2019, January 8-9, 2019, San Diego, CA, USA, volume 69 of *OASIcs*, pages 17:1–17:6. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019.

[25] Sotiris Kanellopoulos, Giorgos Mitropoulos, Antonis Antonopoulos, Nikos Leonardos, Aris Pagourtzis, Christos Pergaminelis, Stavros Petsalakis, and Kanellos Tsitouras. Approximation schemes for k-subset sum ratio and multiway number partitioning ratio. *CoRR*, abs/2503.18241, 2025.

[26] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography, Second Edition*. CRC Press, 2014.

[27] Hans Kellerer, Renata Mansini, Ulrich Pferschy, and Maria Grazia Speranza. An efficient fully polynomial approximation scheme for the subset-sum problem. *J. Comput. Syst. Sci.*, 66(2):349–370, 2003.

[28] Konstantinos Koiliaris and Chao Xu. Faster pseudopolynomial time algorithms for subset sum. *ACM Trans. Algorithms*, 15(3):40:1–40:20, 2019.

[29] Richard E. Korf. Objective functions for multi-way number partitioning. In Ariel Felner and Nathan R. Sturtevant, editors, *Proceedings of the Third Annual Symposium on Combinatorial Search, SOCS 2010, Stone Mountain, Atlanta, Georgia, USA, July 8-10, 2010*, pages 71–72. AAAI Press, 2010.

[30] David Kurokawa, Ariel D. Procaccia, and Junxing Wang. Fair enough: Guaranteeing approximate maximin shares. *J. ACM*, 65(2):8:1–8:27, 2018.

[31] Joseph Y-T. Leung. Bin packing with restricted piece sizes. *Information Processing Letters*, 31(3):145–149, 1989.

[32] Richard J. Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. In Jack S. Breese, Joan Feigenbaum, and Margo I. Seltzer, editors, *Proceedings 5th ACM Conference on Electronic Commerce (EC-2004), New York, NY, USA, May 17-20, 2004*, pages 125–131. ACM, 2004.

[33] Nikolaos Melissinos and Aris Pagourtzis. A faster FPTAS for the subset-sums ratio problem. In Lusheng Wang and Daming Zhu, editors, *Computing and Combinatorics - 24th International Conference, COCOON 2018, Qing Dao, China, July 2-4, 2018, Proceedings*, volume 10976 of *Lecture Notes in Computer Science*, pages 602–614. Springer, 2018.

[34] Stephan Mertens. The easiest hard problem: Number partitioning. In Allon G. Percus, Gabriel Istrate, and Cristopher Moore, editors, *Computational Complexity and Statistical Physics*, Santa Fe Institute Studies in the Sciences of Complexity, pages 125–140. Oxford University Press, 2006.

[35] Marcin Mucha, Karol Wegrzycki, and Michal Wlodarczyk. A subquadratic approximation scheme for partition. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 70–88. SIAM, 2019.

[36] Danupon Nanongkai. Simple FPTAS for the subset-sums ratio problem. *Inf. Process. Lett.*, 113(19-21):750–753, 2013.

[37] Trung Thanh Nguyen and Jörg Rothe. Minimizing envy and maximizing average nash social welfare in the allocation of indivisible goods. *Discret. Appl. Math.*, 179:54–68, 2014.

[38] Christos H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. Syst. Sci.*, 48(3):498–532, 1994.

[39] Torben P. Pedersen. A threshold cryptosystem without a trusted party (extended abstract). In Donald W. Davies, editor, *Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8-11, 1991, Proceedings*, volume 547 of *Lecture Notes in Computer Science*, pages 522–526. Springer, 1991.

[40] Diego Recalde, Daniel Severín, Ramiro Torres, and Polo Vaca. An exact approach for the balanced k-way partitioning problem with weight constraints and its application to sports team realignment. *J. Comb. Optim.*, 36(3):916–936, 2018.

[41] Sartaj Sahni. Algorithms for scheduling independent tasks. *J. ACM*, 23(1):116–127, 1976.

[42] Berry Schoenmakers. A simple publicly verifiable secret sharing scheme and its application to electronic. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 148–164. Springer, 1999.

[43] Ethan L. Schreiber, Richard E. Korf, and Michael D. Moffitt. Optimal multi-way number partitioning. *J. ACM*, 65(4):24:1–24:61, 2018.

[44] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[45] Max Springer, Mohammad Taghi Hajiaghayi, and Hadi Yami. Almost envy-free allocations of indivisible goods or chores with entitlements. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024.

[46] Gerhard J. Woeginger. When does a dynamic programming formulation guarantee the existence of a fully polynomial time approximation scheme (fptas)? *INFORMS J. Comput.*, 12(1):57–74, 2000.

[47] Gerhard J. Woeginger and Zhongliang Yu. On the equal-subset-sum problem. *Inf. Process. Lett.*, 42(6):299–302, 1992.

[48] Yair Zadok, Nadav Voloch, Noa Voloch-Bloch, and Maor Meir Hajaj. Multiple subset problem as an encryption scheme for communication. *CoRR*, abs/2401.09221, 2024.

[49] Houyu Zhou, Tianze Wei, Biaoshuai Tao, and Minming Li. Fair allocation of items in multiple regions. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024.

[50] Ευστάθιος Ζάχος, Αριστείδης Παγουρτζής, and Παναγιώτης Γροντάς. *Υπολογιστική Κρυπτογραφία*. Κάλλιπος, Ανοικτές Ακαδημαϊκές Εκδόσεις, Αθήνα, 2015. Προπτυχιακό εγχειρίδιο, Άδεια: CC BY-NC-SA 4.0.