Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Ομομορφική Κρυπτογραφία με Ιδεώδη Δικτυώματα** **Homomorphic Encryption using Ideal Lattices**
Ονοματεπώνυμο Φοιτητή	**Δημήτριος Μπαλτάς**
Πατρώνυμο	**Κωνσταντίνος**
Αριθμός Μητρώου	**ΜΠΠΛ 13055**
Επιβλέπων	**Παναγιώτης Κοτζανικολάου, Επίκουρος Καθηγητής**

Ημερομηνία Παράδοσης **Νοέμβριος 2016**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
Όνομα Επώνυμο Βαθμίδα	Όνομα Επώνυμο Βαθμίδα	Όνομα Επώνυμο Βαθμίδα

Περίληψη

Η παρούσα μεταπτυχιακή διατριβή αποτελεί μια μελέτη για την ανάπτυξη της πλήρους ομομορφικής κρυπτογραφίας. Μελετάμε το σχήμα του Gentry το οποίο αποτελεί το πρώτο πλήρες ομομορφικό σχήμα, λύνοντας έτσι ένα ανοιχτό πρόβλημα εδώ και δεκαετίες στον τομέα της Κρυπτογραφίας. Η κατασκευή του Gentry μας επιτρέπει την εκτέλεση αυθαίρετα μεγάλου αριθμού υπολογισμών με κρυπτογραφημένα δεδομένα χωρίς να απαιτείται η αποκρυπτογράφηση τους πρώτα. Η χρήση της πλήρους ομομορφικής κρυπτογράφησης έχει ποικίλες εφαρμογές. Για παράδειγμα, μας επιτρέπει να εκτελέσουμε ερωτήματα σε μια μηχανή αναζήτησης διασφαλίζοντας την ιδιωτικότητα της αναζήτησής μας. Ας υποθέσουμε πως ένας χρήστης επιθυμεί να κάνει μια αναζήτηση σε μια μηχανή αναζήτησης. Για να το πραγματοποιήσει θέτει ένα ερώτημα στην μηχανή σε κρυπτογραφημένη όμως μορφή. Η μηχανή υλοποιεί ένα πλήρες ομομορφικό σχήμα έτσι μπορεί να χειριστεί το κρυπτογραφημένο ερώτημα από τον χρήστη και να του επιστρέψει τα αποτελέσματα σε επίσης κρυπτογραφημένη μορφή. Με αυτό τον σχεδιασμό η μηχανή εκτέλεσε το ερώτημα χωρίς να γνωρίζει ποιό είναι αυτό αφού δεν το αποκρυπτογράφησε πρώτα. Ο χρήστης λαμβάνει το κρυπτογραφημένο αποτελέσματα της αναζήτησης του και στην συνέχεια ο ίδιος το αποκρυπτογραφεί και το διαβάζει. Έτσι εξασφαλίζει την ιδιωτικότητα της αναζήτησής του.

Η κατασκευή του Gentry ξεκινά με ένα *κάπως ομομορφικό σχήμα (somewhat homomorphic scheme)*. Στην συνέχεια τροποποιεί κατάλληλα το σχήμα με την τεχνική του *squashing* ώστε να το εφοδιάσει με μια πολύ χρήσιμη ιδιότητα που καλείται *bootstrappability*. Τέλος, αποδεικνύει πως κάθε *εκκινήσιμο(bootstrappable)* σχήμα το οποίο είναι επιπλέον και κάπως ομομορφικό μπορεί να μετατραπεί σε ένα πλήρως ομομορφικό σχήμα με μια αναδρομική διαδικασία αυτό-ενσωμάτωσης. Η ασφάλεια του συστήματος Gentry στηρίζεται σε δύσκολα προβλήματα της θεωρίας δικτυωμάτων και σε μια παραλλαγή του προβλήματος αθροίσματος υποσυνόλων που καλείται *sparse subset sum problem*.

ABSTRACT

In this Master's Thesis we do a survey about the development of fully homomorphic encryption. We study Gentry's scheme which is the first fully homomorphic encryption scheme, solving a central open problem in cryptography. Gentry with his construction allows us to compute arbitrary functions over encrypted data without decrypt them first. Fully homomorphic encryption has numerous applications. For example allows to us to make private queries to a search engine. Suppose that a user wants to search for something so he commits a query in encrypted format to the search engine. The engine implements a fully homomorphic scheme so it can handle the query in absolutely encrypted mode and return the result in also encrypted format without knowing what it returned or what the engine searched about. This provides the users with full search privacy.

Gentry's construction begins with a *somewhat homomorphic encryption* scheme. Gentry then shows how to slightly modify this scheme to make it *bootstrappable*. Finally, he shows that any bootstrappable somewhat homomorphic encryption scheme can be converted into a fully homomorphic encryption through a recursive *self-embedding*. Gentry based the security of his scheme on hard problems over ideal lattices and the sparse subset sum problem.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα της μεταπτυχιακής διατριβής μου, Καθηγητή κ. Παναγιώτη Κοτζανικολάου, για την ενθάρρυνσή του να μελετήσω ένα τόσο ενδιαφέρον επιστημονικά θέμα και την πολύτιμη καθοδήγησή του σε κάθε στάδιο της δημιουργίας της. Επιθυμώ να εκφράσω τη βαθιά ευγνωμοσύνη μου για την εμπιστοσύνη που επέδειξε στο πρόσωπό μου και την ουσιαστική στήριξή του σε όλη την διάρκεια συγγραφής της διατριβής αυτής.

Επίσης, ιδιαίτερες ευχαριστίες θέλω να απευθύνω στον καθηγητή κ. Πατσάκη Κωνσταντίνο χάρη στον οποίο ήρθα για πρώτη φορά σε επαφή με τον θαυμαστό κόσμο της Κρυπτογραφίας και για την καθοριστική του βοήθεια σε κάθε φάση της πορείας μου.

Θερμές ευχαριστίες θα ήθελα να εκφράσω και στα υπόλοιπα μέλη της εξεταστικής επιτροπής: τους καθηγητές του τμήματος Πληροφορικής του Πανεπιστημίου Πειραιώς κ. Δουληγέρη Χρήστο και κ. Ψαράκη Μιχαήλ για την πρόθυμη συμμετοχή τους στην κρίση της μεταπτυχιακής αυτής διατριβής.

Ιδιαίτερα επίσης, ευχαριστώ τον Δρ. Αρούκατο Νικόλαο για την πολύτιμη στήριξη και καθοδήγηση που μου παρείχε καθ' όλη την διάρκεια των μεταπτυχιακών μου σπουδών.

Επίσης, ένα πολύ μεγάλο ευχαριστώ στον συνάδελφο αλλά πάνω από όλα φίλο μου Καραβάτσο Γεώργιο ο οποίος με στήριξε και ήταν δίπλα μου όποτε τον χρειάστηκα.

Ακόμη, θα ήθελα να ευχαριστήσω τους γονείς μου, Κωνσταντίνο και Γαρυφαλλιά για όλη την αγάπη και την αμέριστη συμπαράσταση τους καθ' όλη τη διάρκεια των ακαδημαϊκών μου σπουδών. Ιδιαίτερες ευχαριστίες στην αδελφή μου Ειρήνη, η οποία είναι πάντα δίπλα μου σε κάθε δύσκολη στιγμή με όλη της την αγάπη και την αφοσίωση.

Τέλος, το πιο μεγάλο ευχαριστώ το οφείλω στην σύντροφο μου Ιωάννα γιατί με την υπομονή, το χαμόγελό και την αγάπη της υπήρξε το μεγαλύτερο στήριγμα μου σε όλη την πορεία και διάρκεια των ακαδημαϊκών μου σπουδών αλλά και γενικότερα στην ζωή μου.

Η παρούσα μεταπτυχιακή διατριβή αφιερώνεται στην Ιωάννα.

Περιεχόμενα

1. Εισαγωγή

Από την φύση της η πληροφορία είναι άμεσα συνδεδεμένη με την έννοια του απορρήτου. Το πόσο κρίσιμη είναι μία πληροφορία και η ανάγκη διασφάλισης της εμπιστευτικότητας της αποτελούν μεγέθη ανάλογα. Όσο πιο κρίσιμη τόσο πιο επιτακτική ανάγκη είναι η διασφάλιση του απορρήτου της και μέσα σε αυτό το θεωρητικό πλαίσιο αναπτύχθηκε και εξελίχθηκε ο κλάδος της Κρυπτογραφίας στην πορεία των ετών.

Κρυπτογραφία είναι ο επιστημονικός κλάδος που ασχολείται με τη μελέτη και ανάπτυξη μαθηματικών τεχνικών, με σκοπό την ασφαλή αποθήκευση, μετάδοση και επεξεργασία των πληροφοριών, όπου σε συνεργασία με τον κλάδο της Κρυπτανάλυσης η οποία ασχολείται με τη μελέτη τρόπων παραβίασης αυτών, απαρτίζει την επιστήμη της Κρυπτολογίας. Η Κρυπτογραφία αποτελεί έναν θεμελιώδη καταλύτη για ασφαλείς και αξιόπιστες υποδομές και ένα διεπιστημονικό ερευνητικό τομέα με υψηλές, στρατηγικές επιπτώσεις στην βιομηχανία και για την κοινωνία στο σύνολό της.

Το σύνολο των διαδικασιών που ακολουθούνται προκειμένου να επιτευχθεί μία κρυπτογράφηση ή αποκρυπτογράφηση καλείται *κρυπτογραφικός αλγόριθμος*. Ένα σύστημα που παρέχει έναν τέτοιο αλγόριθμο αναφέρεται ως *κρυπτογραφικό σύστημα*. Οι αλγόριθμοι αυτοί είναι πολύπλοκοι μαθηματικοί τύποι που εφαρμόζονται στο προς κρυπτογράφηση μήνυμα. Οι περισσότεροι μέθοδοι χρησιμοποιούν ένα μυστικό αριθμό που καλείται κλειδί και η τιμή του επηρεάζει την κρυπτογράφηση ή αποκρυπτογράφηση του μηνύματος αφού εφαρμοστεί μέσω του κρυπτογραφικού αλγόριθμου. Αν και οι αλγόριθμοι είναι εκείνοι οι οποίοι υπαγορεύουν το πώς θα πραγματοποιηθεί η κρυπτογράφηση ή η αποκρυπτογράφηση δεν αποτελούν εν γένει το μυστικό κομμάτι της καθώς τις περισσότερες φορές είναι γνωστή η υλοποίησή τους. Επομένως, για να έχει νόημα η έννοια της κρυπτογράφησης ενός μηνύματος θα πρέπει να υπάρχει κάτι μυστικό. Τον ρόλο αυτό παίζει το μυστικό κλειδί που αναφέραμε πριν το οποίο είναι μία μεγάλη ακολουθία τυχαίων bit. Ο κάθε κρυπτογραφικός αλγόριθμος περιέχει ένα συγκεκριμένο σύνολο δυνατών τιμών για τα κλειδιά που ονομάζεται πεδίο τιμών. Ένα μεγάλο πεδίο τιμών επιτρέπει περισσότερα πιθανά κλειδιά με συνέπεια να είναι δυσκολότερο για τον ενδεχόμενο επιβουλέα να τα υπολογίσει.

Πιο φορμαλιστικά ένα κρυπτοσύστημα CS (Crypto System) ορίζεται ως εξής:

$$CS = (M, K, C, KeyGen, Encrypt, Decrypt)$$

- M: Το σύνολο των μηνυμάτων προς κρυπτογράφηση.
- K: Το σύνολο των δυνατών κλειδιών
- C: Το σύνολο των κρυπτοκειμένων
- $KeyGen = (Key_{enc}, Key_{dec}) \in K^2 \quad , \quad K = \{0,1\}^{\lambda}$
  - Πιθανοτικός Αλγόριθμος
  - Το κλειδί συνήθως επιλέγεται ομοιόμορφα από το K
  - λ : Παράμετρος ασφάλειας – το πλήθος των bits του κλειδιού
- $Encrypt(Key_{enc}, m) = c \in C$
  - Ντετερμινιστικός Αλγόριθμος: Κάθε μήνυμα αντιστοιχεί σε ένα κρυπτοκείμενο
  - Πιθανοτικός Αλγόριθμος: Κάθε μήνυμα αντιστοιχεί σε ένα σύνολο πιθανών κρυπτοκειμένων

- $Decrypt(Key_{dec}, c) = m \in M$

Το κλειδί αποκρυπτογράφησης που χρησιμοποιείται σε έναν κρυπτογραφικό αλγόριθμο εν γένει δεν είναι ίδιο με το κλειδί κρυπτογράφησης. Όταν τα δύο κλειδιά είναι ίδια δηλαδή $Key_{enc} = Key_{dec}$ μιλάμε για συμμετρική κρυπτογράφηση ενώ αν είναι διαφορετικά η κρυπτογράφηση καλείται κρυπτογράφηση δημοσίου κλειδιού.

Έτσι η κρυπτογραφία χωρίζεται σε δύο βασικούς κλάδους με βάση τον τύπο της κρυπτογράφησης που χρησιμοποιείται: *Συμμετρική κρυπτογραφία* και *Κρυπτογραφία δημοσίου κλειδιού*.
Η συμμετρική κρυπτογραφία λοιπόν χαρακτηρίζεται από τη χρήση ενός κοινού κλειδιού μεταξύ του αποστολέα και του παραλήπτη. Τα περισσότερα συστήματα κρυπτογράφησης δημοσίου κλειδιού είναι πολύ αποδοτικά ως προς τους υπολογιστικούς πόρους που απαιτούνται ενώ ταυτόχρονα είναι εξαιρετικά δύσκολο να παραβιαστούν σε λογικό χρονικό διάστημα.

Ωστόσο, η συμμετρική κρυπτογραφία έχει κάποια μειονεκτήματα. Η κύρια πρόκληση είναι η διανομή του κοινού κλειδιού. Ο αποστολέας και ο παραλήπτης θα πρέπει με κάποιο τρόπο να διασφαλίσει κάθε φορά ότι μοιράζονται το ίδιο κλειδί και ότι κατά τη διαδικασία του διαμοιρασμού καμία άλλη οντότητα δεν μπορεί να έχει γνώση του κλειδιού. Αυτό θα ισχύει για κάθε ζεύγος από οντότητες που θέλουν να επικοινωνήσουν που σημαίνει ότι κάθε οντότητα θα πρέπει να διατηρεί αποθηκευμένα πολλά κοινά κλειδιά για την ασφαλή επικοινωνία της με άλλους. Για λόγους ασφαλείας ένα κοινό κλειδί χρησιμοποιείται μόνο για σύντομο χρονικό διάστημα που συνήθως καλείται συνεδρία.

Η κρυπτογραφία δημοσίου κλειδιού εν τω μεταξύ, χρησιμοποιεί δύο διαφορετικά κλειδιά όπως είπαμε, ένα δημόσιο και ένα ιδιωτικό. Το δημόσιο κλειδί δημοσιεύεται από μία οντότητα που θέλει να λάβει ένα μήνυμα και ενώ κατέχει και το αντίστοιχο ιδιωτικό κλειδί που κανείς άλλος δεν γνωρίζει. Αυτά τα δύο κλειδιά είναι μαθηματικά συσχετισμένα με τέτοιο τρόπο ώστε αν μία άλλη οντότητα γνωρίζει το δημόσιο κλειδί δεν μπορεί να υπολογίσει με εύκολο τρόπο το ιδιωτικό της άλλης. Το κύριο λοιπόν πλεονέκτημα της κρυπτογραφίας δημοσίου κλειδιού είναι ότι η διανομή των κλειδιών είναι πολύ εύκολη. Η κάθε οντότητα το μόνο που έχει να κάνει είναι να δημοσιεύσει το δημόσιο κλειδί της σε έναν server που όλοι εμπιστεύονται και όποιος άλλος θέλει να επικοινωνήσει μαζί της χρησιμοποιεί αυτό το δημόσιο κλειδί για να κρυπτογραφήσει το προς μετάδοση μήνυμα και στην συνέχεια να το αποστείλει. Εν συνεχεία ο παραλήπτης χρησιμοποιεί το ιδιωτικό του κλειδί για να αποκρυπτογραφήσει το μήνυμα. Το βασικό μειονέκτημα της κρυπτογραφίας δημοσίου κλειδιού είναι ότι είναι λιγότερο αποδοτική υπό την έννοια των υπολογιστικών πόρων σε σχέση με τα συστήματα συμμετρικής κρυπτογράφησης.

1.1. Ανάλυση Προβλήματος – Εφαρμογές Πλήρης Ομομορφικής κρυπτογραφίας

Η κρυπτογράφηση ξεκίνησε σαν εργαλείο κυρίως για στρατιωτικούς σκοπούς όπως το να στείλεις κρυπτογραφημένα μηνύματα που περιέχουν απόρρητη στρατιωτική πληροφορία. Με την πάροδο των ετών όμως άρχισε να χρησιμοποιείται πολύ ευρύτερα, ώσπου στις μέρες μας χρησιμοποιούμε κρυπτογράφηση χωρίς να το αντιλαμβανόμαστε καθημερινά, από τις ηλεκτρονικές πληρωμές μέσω της τράπεζας μας, μέχρι την επικοινωνία μας με άλλους μέσω email ή άμεσων μηνυμάτων (instant messaging).

Μία από τις πλέον ενδιαφέρουσες χρήσεις της κρυπτογράφησης αποτελεί το *cloud computing*. Με μία πρώτη ματιά η υπολογιστική νέφους (cloud computing) [1] φαίνεται να αποτελεί έναν πολύ βολικό τρόπο για την αποθήκευση δεδομένων και στη συνέχεια η χρήση των cloud services με σκοπό την επεξεργασία των αποθηκευμένων αυτών δεδομένων. Ωστόσο, οι τρέχουσες υποδομές του cloud computing απαιτούν την εμπιστοσύνη του χρήστη προς τους παρόχους αυτών των υπηρεσιών. Από την άλλη η αποθήκευση των δεδομένων σε κρυπτογραφημένη μορφή δεν έχει νόημα αν δεν μπορούμε να εκτελέσουμε λειτουργίες πάνω σε αυτά χωρίς την αποκρυπτογράφηση τους πρώτα.

Εάν ήταν εφικτό λοιπόν να αποθηκεύσουμε τα δεδομένα σε κρυπτογραφημένη μορφή και στη συνέχεια οι πάροχοι cloud υπηρεσιών να μπορούν να εκτελέσουν λειτουργίες πάνω σε αυτά χωρίς να απαιτείται η αποκρυπτογράφηση τους πρώτα, αυτομάτως θα μπορούσε να καταργηθεί η ανάγκη της εμπιστοσύνης από τους χρήστες, καθώς δεν θα μπορούσε κανείς να δει τα δεδομένα τους εκτός από τους ίδιους. Η χρήση της πλήρους ομομορφικής κρυπτογράφησης θα μπορούσε να αποτελέσει τη λύση σε αυτό το πρόβλημα.

Με τον όρο πλήρης ομομορφική κρυπτογράφηση εννοούμε ένα κρυπτογραφικό σύστημα όπου μία οντότητα μπορεί να λάβει κρυπτογραφημένα δεδομένα και να εκτελέσει οποιεσδήποτε πράξεις πάνω σε αυτά με σκοπό την επεξεργασία τους. Τα δεδομένα παραμένουν κρυπτογραφημένα αλλά οι πράξεις μπορούν να πραγματοποιηθούν χωρίς να αποκρυπτογραφηθούν πρώτα. Αυτό θα μπορούσε να εξασφαλίσει την ιδιωτικότητα των δεδομένων όταν αυτά αποστέλλονται και αποθηκεύονται σε μία υπηρεσία cloud computing. Στην επιστημονική βιβλιογραφία υπάρχουν διάφορα τέτοια συστήματα όπως του **Pailier** και του **ElGamal** αλλά σε κανένα από αυτά δεν είναι δυνατή η εκτέλεση όλων των δυνατών πράξεων παρά μόνο συγκεκριμένων. Μιλάμε δηλαδή για ομομορφική κρυπτογράφηση χωρίς να είναι πλήρης.

Η πλήρης ομομορφική κρυπτογράφηση αν και σαν πρόβλημα είχε διατυπωθεί εδώ και πολλά χρόνια εν τέλει αποτελεί μία πολύ νέα ερευνητική περιοχή στον τομέα της κρυπτογραφίας. Ένα πρώτο τέτοιο κρυπτογραφικό σύστημα αναπτύχθηκε από τον **Gentry** το 2009. Το σύστημα αυτό στηρίζεται σε ιδεώδη δικτυώματα και η ασφάλεια του στηρίζεται σε δυσεπίλυτα προβλήματα της θεωρίας δικτυωμάτων. Η κρυπτογραφία δικτυωμάτων κερδίζει τεράστιο επιστημονικό ενδιαφέρον καθώς η κατασκευή κβαντικών υπολογιστών είναι προ των πυλών και η ίδια παρουσιάζει αντίσταση σε επιθέσεις κβαντικών υπολογιστών. Ωστόσο, το βασικό πρόβλημα που παραμένει είναι ότι τα συστήματα που στηρίζονται σε αυτό το είδος κρυπτογράφησης δεν είναι ακόμα αρκετά αποδοτικά ώστε να χρησιμοποιηθούν άμεσα.

2. Μαθηματικό Υπόβαθρο

Το κεφάλαιο αυτό αποτελείται από τις απαραίτητες μαθηματικές έννοιες που απαιτούνται για την κατανόηση των όσων θα αναφερθούν παρακάτω. Οι ορισμοί και τα θεωρήματα που ακολουθούν αφορούν το πεδίο της αλγεβρικής θεωρίας αριθμών [3], της θεωρίας ομάδων [4] και της θεωρίας πλεγμάτων [2]. Δεδομένου ότι η συγκεκριμένη διατριβή επικεντρώνεται στην ανάλυση ομομορφικών κρυπτογραφικών συστημάτων η εφαρμογή τους θα πραγματοποιηθεί στα αντίστοιχα κεφάλαια.

2.1. Αλγεβρικές δομές και αλγεβρική θεωρία αριθμών

2.1.1. Ομάδες

Θα ορίσουμε κάποιες σημαντικές αλγεβρικές δομές.

Ορισμός 2.1. Έστω G ένα μη κενό σύνολο και ένας εσωτερικός νόμος. Η δομή (G, *) καλείται **ομάδα** αν και μόνο αν ισχύει:

- Ο νόμος είναι προσεταιριστικός: $\forall a,b,c\in G:\quad a*(b*c)=(a*b)*c$
- Υπάρχει ουδέτερο στοιχείο e: $(\exists e\in G)(\forall a\in G):\quad a*e=e*a=a$
- Υπάρχει αντίστροφο στοιχείο a': $(\forall a\in G)(\exists a'\in G):\quad a*a'=a'*a=e$

Εάν ενισχύσουμε τον ορισμό με την ιδιότητα της αντιμεταθετικότητας δηλαδή $\forall a,b\in G\ :\ a*b=b*a$ τότε η ομάδα καλείται **αβελιανή**. Το πλήθος των στοιχείων μιας ομάδας καλείται **τάξη** της ομάδας και συμβολίζεται με $|G|$.Έτσι εάν $|G|<\infty$ η ομάδα καλείται **πεπερασμένη** ενώ εάν $|G|=\infty$ η ομάδα καλείται **άπειρη**. Για το υπόλοιπο της διατριβής θα εργαζόμαστε πάντα μέσα σε πεπερασμένες αλγεβρικές δομές εκτός αν αναφέρεται διαφορετικά.
Μια αλγεβρική δομή A με έναν ή δυο εσωτερικούς νόμους (παρακάτω θα ορίσουμε και τέτοιες δομές) μπορεί να περιέχεται σε μια άλλη δομή G με την έννοια ότι $A\subseteq G$ και οι νόμοι στην A είναι οι περιορισμοί των νόμων της G στην A.
Έτσι μπορούμε να ορίσουμε την έννοια της **υποομάδας** A μιας ομάδας G, με την έννοια ότι αν (G, *) είναι ομάδα και $A\subseteq G$, τότε A υποομάδα της G $\Leftrightarrow$ (A, *) είναι επίσης ομάδα.
Έστω $a\in G.$ Αν $\exists n\in\mathbb{Z},n\geq 0\ :\ a^n=e,$ τότε το α λέμε ότι έχει πεπερασμένη τάξη. Ο μικρότερος τέτοιος θετικός ακέραιος ονομάζεται **τάξη** του α και συμβολίζεται με $|a|=n$.
Ορισμός 2.2 Έστω (G, *) ομάδα και $a\in G$. Το σύνολο $\langle\alpha\rangle=\{x\in G:x=a^n,n\in\mathbb{Z}\}$ καλείται **κυκλική υποομάδα** τάξης n που γεννάται από το α.

Πιο γενικά μια ομάδα G καλείται **κυκλική ομάδα** αν υπάρχει στοιχείο $a\in G\ :\ G=\langle a\rangle$. Σε αυτή την περίπτωση το α καλείται **γεννήτορας** της ομάδας.

Γενικά αξίζει να σημειωθεί ότι αν G ομάδα και $S\subseteq G$, τότε έχει νόημα να αναζητήσουμε όλες τις υποομάδες της G που περιέχουν το S. Μια τέτοια υποομάδα είναι η ίδια η G. Η τομή τους είναι μια υποομάδα που περιέχει το S εφόσον το S περιέχεται σε όλες τις υποομάδες που θεωρήσαμε. Η υποομάδα αυτή συμβολίζεται με $\langle S\rangle$ και καλείται υποομάδα της

G παραγόμενη από το S. Τα στοιχεία του S ονομάζονται **γεννήτορες** της $\langle S \rangle$ Η κυκλική ομάδα λοιπόν είναι μια ειδική περίπτωση όπου $S=\{a\}$.

Ορισμός 2.3 Μία σχέση ενός συνόλου A λέγεται **σχέση ισοδυναμίας** αν είναι αυτοπαθητική συμμετρική και μεταβατική. Συνήθως συμβολίζεται με ~. Δηλαδή:

- x ~ x (*αυτοπαθητική*)
- x ~ y ⇒ y ~ x (*συμμετρική*)
- x ~ y, y ~ z ⇒ x ~ z (*μεταβατική*)

Ορισμός 2.4 Έστω A ένα σύνολο και ~ μια σχέση ισοδυναμίας επί του A. **Κλάση** ενός στοιχείου $a\in A$ ως προς την σχέση ~ καλείται το σύνολο $C_a=\{x\in A : x\sim a\}$.

Να σημειώσουμε πως το σύνολο όλων των κλάσεων ως προς μια σχέση ισοδυναμίας ενός συνόλου A αποτελεί μια *διαμέριση* του συνόλου A. Κάθε στοιχείο μιας κλάσης ονομάζεται **αντιπρόσωπος** μιας κλάσης.

Έστω ότι G ομάδα. Ορίζουμε μία σχέση ισοδυναμίας επί των στοιχείων της G ως εξής:
x ~ y ⇔ $x^{-1}y\in G$ με κλάσεις ισοδυναμίας τα σύνολα aG που λέγονται **αριστερές κλάσεις** ή **αριστερά σύμπλοκα**. Ομοίως ορίζουμε και την σχέση x ~ y ⇔ $xy^{-1}\in G$ με κλάσεις ισοδυναμίας τα σύνολα Ga που λέγονται **δεξιές κλάσεις** ή **δεξιά σύμπλοκα**.

Έστω μια σχέση ισοδυναμίας επί ενός συνόλου A. Το σύνολο των κλάσεων ως προς την σχέση αυτή καλείται **σύνολο πηλίκο** του A με την σχέση ~ και συμβολίζεται με A / ~ δηλαδή A / ~ = $\{C_a : a\in A\}$

Θεώρημα 2.5 (Lagrange). Έστω G ομάδα πεπερασμένης τάξης και H υποομάδα αυτής. Η τάξη της H διαιρεί την τάξη της G.

Απόδειξη. Ισχυρόμαστε αρχικά ότι κάθε αριστερό σύμπλοκο και κάθε δεξιό σύμποκο της H έχει το ίδιο πλήθος στοιχείων με την H. Δηλαδή $|gH|=|Hg|=|H|$. Για να το αποδείξουμε αυτό θα ορίσουμε μια 1-1 και επί απεικόνιση επί αριστερού συμπλόκου gH της H $\forall\, g\in G$

Μια προφανής επιλογή είναι η απεικόνιση φ: H → gH, φ(h) = gh.

- Είναι επί προφανώς από τον τρόπο που ορίστηκε.
- Είναι 1-1 διότι: Έστω $\varphi(h_1)=\varphi(h_2)$, $h_1,h_2\in H$ Τότε $gH_1=gH_2\Rightarrow h_1=h_2$

Έστω τώρα n η τάξη της G και m η τάξη της H. Από τον ισχυρισμό που μόλις αποδείξαμε φαίνεται ότι κάθε σύμπλοκο της H έχει επίσης m στοιχεία. Έστω r το πλήθος των υποσυνόλων στην διαμέριση της G σε αριστερά σύμπλοκα. Τότε n=rm, άρα ο m | n.

2.1.2. Ομομορφισμοί

Στην ενότητα αυτή θα ορίσουμε μια πολύ σημαντική έννοια της θεωρίας ομάδων αλλά και της άλγεβρας γενικότερα, την έννοια του ομομορφισμού. Οι ομομορφισμοί παίζουν πολύ σημαντικό ρόλο στην ανάπτυξη κρυπτογραφικών συστημάτων με την ιδιότητα της ομομορφίας.

Στην άλγεβρα, και γενικότερα στα μαθηματικά, ορίζουμε διαφόρων ειδών αντικείμενα, με συγκεκριμένη δομή το καθένα, και απεικονίσεις ή μορφισμούς μεταξύ τους, που σέβονται αυτή τη δομή. Μια οικογένεια αντικειμένων και μορφισμών (που ικανοποιούν ορισμένες προφανείς ιδιότητες, λέγεται **κατηγορία**. Υπάρχει μία γενική γλώσσα *«θεωρία κατηγοριών»* που περιγράφει διάφορες γενικές ιδιότητες και κατασκευές σε μία κατηγορία ή μεταξύ κατηγοριών [5].

Παραδείγματα κατηγοριών είναι:

Αντικείμενα	Μορφισμοί
Σύνολα	Απεικονίσεις
Τοπολογικοί χώροι	Συνεχείς απεικονίσεις
Ομαλές (C∞) πολλαπλότητες	Ομαλές απεικονίσεις
Ομάδες	Ομομορφισμοί ομάδων
Διανυσματικοί χώροι	Γραμμικοί τελεστές
Δακτύλιοι	Ομομορφισμοί δακτυλίων

Ορισμός 2.6. Μια απεικόνιση $f: G \to H$, όπου $(G, *_G)$ και $(H, *_H)$ ομάδες καλείται **ομομορφισμός** ομάδων αν και μόνο αν $f(g_1 *_G g_2) = f(g_1) *_H f(g_2)$, $\forall g_1, g_2 \in G$.

Με άλλα λόγια οι ομομορφισμοί είναι απεικονίσεις που *"σέβονται"* την αλγεβρική δομή. Πρόκειται για μια πολύ σπουδαία ιδιότητα στα Μαθηματικά.

Να σημειώσουμε σε αυτό το σημείο πως ένας ομομορφισμός ομάδων "ταξιδεύει" το ουδέτερο στοιχείο της ομάδας G στο ουδέτερο στοιχείο της ομάδας H. Δηλαδή: $f(e_G) = f(e_H)$

Επίσης μεταφέρει τα αντίστροφα των στοιχείων στις αντίστροφες εικόνες τους αφού:

$$f(g) f(g^{-1}) = f(g\, g^{-1}) = f(e_G) \Rightarrow f(g)^{-1} = f(g^{-1}).$$

Ορίζουμε **πυρήνα** ενός ομομορφισμού το σύνολο $Kerf = \{g \in G : f(g) = e_H\}$.

Η **εικόνα** ενός ομομορφισμού όπως και για κάθε απεικόνιση ορίζεται το σύνολο:

$$Imf = \{h \in H : \exists g \in G \text{ ώστε } f(g) = h\}$$

Στην περίπτωση που ένας ομομορφισμός είναι αμφιμονοσήμαντος (1-1) τότε καλείται **μονομορφισμός** ενώ αν είναι επί καλείται **επιμορφισμός**.

Ορισμός 2.7. Ένας ομομορφισμός που είναι αμφιμονοσήμαντος και επί καλείται **ισομορφισμός**.

Έστω λοιπόν $f: G \to H$, όπου είναι αμφιμονοσήμαντος και επί. Τότε οι δυο ομάδες είναι ισόμορφες και συμβολίζουμε με $G \cong H$ αυτόν τον ισομορφισμό.

Το ενδιαφέρον των ισομορφισμών έγκειται στο γεγονός ότι δύο ισομορφικά αντικείμενα δεν μπορούν να διαχωριστούν, χρησιμοποιώντας μόνο τις ιδιότητες που χρησιμοποιούνται για να καθορίσουν τον ομομορφισμό: έτσι ισομορφικά αντικείμενα μπορούν να θεωρηθούν το ίδιο, αρκεί να αναλογιστεί κανείς μόνο, τις ιδιότητες αυτές καθώς και τις συνέπειές τους.

2.2. Θεωρία αριθμών

Ορισμός 2.8. Έστω $\mathbb{Z}$ σύνολο των ακεραίων και $n \in \mathbb{N}$. Ορίζουμε επί του $\mathbb{Z}$ την σχέση ~ ως εξής: $a \sim b \Leftrightarrow n/a-b$. Οι ακέραιοι a, b καλούνται **ακέραιοι ισοδύναμοι modulo n** και γράφουμε $a \equiv b mod n$.

Η παραπάνω σχέση αποτελεί σχέση ισοδυναμίας διότι :

- $a \sim a$ *εφόσον* $n/a-a$ (αυτοπαθητική)
- $a \equiv b mod n \Rightarrow n/a-b \Rightarrow a-b=kn \Rightarrow b-a=n(-k) \Rightarrow n/b-a \Rightarrow b \equiv a mod n$ (συμμετρική)
- $a \equiv b mod n, b \equiv c mod n \Rightarrow a \sim c mod n$ (μεταβατική) αφού: $n/a-b, n/b-c \Rightarrow a-b=nk, b-c=nt \Rightarrow a-c=(a-b)+(b-c)=na+nb=n(a+b) \Rightarrow n/a-c \Rightarrow a \equiv c mod n$

Η κλάση ενός στοιχείου $x \in \mathbb{Z}$ ως προς την σχέση ισοδυναμίας που ορίσαμε ανωτέρω συμβολίζεται με $\bar{x}$ και είναι το σύνολο των ακεραίων που είναι ισοδύναμοι modulo n με τον x. Πιο φορμαλιστικά: $\bar{x}=\{y \in \mathbb{Z} : y \equiv x mod n\} = x+n\mathbb{Z}$

Ορισμός 2.9. Το σύνολο $\mathbb{Z}_n = \{\bar{0}, \bar{1}, ..., \overline{n-1}\}$, $n \neq 0$ καλείται **σύνολο ακεραίων modulo n** και αποτελείται από τις κλάσεις modulo n. Να σημειωθεί ότι $|\mathbb{Z}_n| = n$

Ορισμός 2.10. Ο **μέγιστος κοινός διαιρέτης** δυο μη μηδενικών ακέραιων a, b γράφεται gcd(a,b) και είναι ο μεγαλύτερος θετικός ακέραιος που διαιρεί τον a και τον b.

Το **ελάχιστο κοινό πολλαπλάσιο** δύο ακέραιων γράφεται lcm(a,b) είναι ο ελάχιστος θετικός ακέραιος που είναι πολλαπλάσιο των a, b.

Ορίζονται οι πράξεις $\bar{a}+\bar{b}=\overline{a+b}$, $\bar{a}*\bar{b}=\overline{a*b}$

Ορισμός 2.11. Πολλαπλασιαστικός αντίστροφος modulo n ενός ακεραίου a είναι ένας ακέραιος b (εάν υπάρχει) τέτοιος ώστε $a*b=1 mod n$

Η **πολλαπλασιαστική τάξη ενός ακεραίου a modulo n** είναι ο μικρότερος θετικός ακέραιος k τέτοιος ώστε $a^k = 1 mod n$

Πρόταση 2.12. Ένας ακέραιος a έχει πολλαπλασιαστικό αντίστροφο modulo n αν και μόνο αν gcd(a,n) =1

Ορισμός 2.12. Ορίζουμε το σύνολο των στοιχείων του $\mathbb{Z}_n$ που έχουν πολλαπλασιαστικό αντίστροφο δηλαδή $\mathbb{Z}_n^* = \{\bar{x} \in \mathbb{Z}_n : gcd(x,n)=1\}$. Το σύνολο αυτό εφοδιασμένο με τον πολλαπλασιαστικό νόμο που ορίσαμε στον ορισμό 2.11 αποδεικνύεται ότι αποτελεί ομάδα που καλείται **πολλαπλασιαστική ομάδα** του $\mathbb{Z}_n$.

Ορισμός 2.13. Για κάθε θετικό ακέραιο n, ορίζεται η **συνάρτηση του Euler** φ(n) ως το σύνολο των ακεραίων b όπου $1 \le b \le n$ και gcd(b,n)=1.

Για παράδειγμα φ(1) = 1, φ(2) = 1, φ(3) = 2 και φ(4) = 2. Σε αυτό το σημείο αξίζει να σημειωθεί πως

$\varphi(n) = |\mathbb{Z}_n^*|$.

2.3. Πρώτοι Αριθμοί

Με δεδομένο ότι οι πρώτοι αριθμοί αποτελούν ένα αναπόσπαστο κομμάτι της μελέτης σχεδόν κάθε κρυπτογραφικού συστήματος η ενότητα αυτή περιλαμβάνει μία εισαγωγή στους πρώτους αριθμούς ως κομμάτι της αλγεβρικής θεωρίας αριθμών.

Ορισμός 2.14. Ένας θετικός ακέραιος p καλείται **πρώτος αριθμός** αν και μόνο αν οι μόνοι θετικοί διαιρέτες του είναι το 1 και ο p.

Υπάρχει ένα πολύ ενδιαφέρον θεώρημα που αφορά την ύπαρξη πρώτων αριθμών μέσα σε συγκεκριμένο διάστημα. Αυτό είναι πολύ χρήσιμο κατά την αναζήτηση πρώτων παραγόντων ενός σύνθετου αριθμού.

Θεώρημα 2.15. Ένας θετικός ακέραιος n είναι πρώτος αν και μόνο αν δεν έχει διαιρέτες μέσα στο διάστημα (1, $\sqrt{n}$].

Το θεώρημα αυτό μας υποδεικνύει μια διαδικασία για τον έλεγχο πρώτων αριθμών που ονομάζεται μέθοδος δοκιμαστικής διαίρεσης. Δηλαδή δοθέντος ενός σύνθετου αριθμού n για να διαπιστώσουμε αν είναι πρώτος μπορούμε να ελέγξουμε αν υπάρχουν διαιρέτες του μέσα στο διάστημα (1, $\sqrt{n}$].

Θεώρημα 2.16. (Θεμελιώδες θεώρημα της Αριθμητικής). Κάθε φυσικός n > 1 γράφεται μονοσήμαντα (όταν δεν λαμβάνουμε υπόψιν την σειρά τους) σαν γινόμενο πρώτων παραγόντων.

Δηλαδή $n = q_1 q_2 .... q_m$

Απόδειξη. Η απόδειξη χωρίζεται σε δύο σκέλη. Στο πρώτο σκέλος θα αποδείξουμε ότι κάθε φυσικός αριθμός αναλύεται σε γινόμενο πρώτων παραγόντων και στο δεύτερο θα αποδείξουμε ότι αυτή η ανάλυση είναι μοναδική για κάθε φυσικό αριθμό.

Αν ο n δεν είναι πρώτος τότε έχει διαιρέτη ένα πρώτο αριθμό q_1 . Έστω $n = q_1 p_1$. Ο p_1 έχει επίσης ένα πρώτο διαιρέτη q_2 οπότε $p_1 = q_2 p_2$ και $n = q_1 q_2 p_2$. Συνεχίζοντας αφού $n > p_1 > p_2 > ....,$ βλέπουμε ότι ο n γράφεται σε γινόμενο πρώτων παραγόντων.

Έστω δύο αναλύσεις του σε γινόμενο πρώτων παραγόντων δηλαδή $n = q_1 q_2 ... q_m = p_1 p_2 ... p_s$.

Θα αποδείξουμε ότι m=s και ότι $\{q_1, q_2, ..., q_m\} = \{p_1, p_2, ..., p_s\}$.

Με επαγωγή ως προς m έχουμε: Αν m=1, ο n είναι πρώτος και συνεπώς $n = q_1 = p_1$ δηλαδή το θεώρημα ισχύει. Θα υποθέσουμε ότι σ'ένα γινόμενο πρώτων αριθμών με πλήθος παραγόντων < m το σύνολο των πρώτων παραγόντων ορίζεται μονοσήμαντα.

Αν $q_1 \dots q_m = p_1 \dots p_s$ το δεύτερο μέλος διαιρείται με q_1 . Συνεπώς q_1 / p_i για κάποιο p_i . Αφού ο p_i είναι πρώτος πρέπει $q_1 = p_i$. Μπορούμε να υποθέσουμε ότι i=1, διαφορετικά αλλάζουμε την αρίθμηση. Τότε $q_1 = p_1$ και επομένως $q_2 \dots q_m = p_2 \dots p_s$. Το πλήθος των πρώτων παραγόντων είναι < m. Κατά την υπόθεση της επαγωγής m-1 = s-1 και $\{q_2, \dots, q_m\} = \{p_2, \dots, p_s\}$. Άρα m=s και $\{q_1, \dots, q_m\} = \{p_1, \dots, p_s\}$. Άρα η ανάλυση είναι και μοναδική.

Ορισμός 2.17. Δύο ακέραιοι a, b καλούνται **σχετικά πρώτοι** αν και μόνο αν gcd(a, b) =1.

Θεώρημα 2.18. Ένας πρώτος p διαιρεί το γινόμενο ab αν και μόνο αν p/a ή p/b.

Απόδειξη. Αν p/a τότε η πρόταση είναι προφανής, διαφορετικά εφόσον p είναι πρώτος τότε $gcd(p, a) \neq p$ και $gcd(p, a) = 1$. Από την ταυτότητα bezout προκύπτει ότι $1 = rp + sa \quad , r, s \in \mathbb{Z}$. Επίσης $ab = kp$ λόγω του ότι p/ab. Οπότε $b = b * 1 = b(rp + sa) = p * (rb + sk)$ άρα ο b πολλαπλάσιο του p.

2.4. Ο μικρός Fermat και το θεώρημα του Euler

Αυτή η παράγραφος παρουσιάζει τρία πολύ σημαντικά θεωρήματα στο πεδίο της κρυπτογραφίας και της θεωρίας αριθμών. Το πρώτο αποκαλείται μικρό θεώρημα του Fermat το οποίο διατυπώθηκε 373 χρόνια πριν και συγκεκριμένα το 1640 από τον Pierre de Fermat όπως άλλωστε μαρτυρά και το όνομά του.

Η πρώτη απόδειξη του θεωρήματος δημοσιεύθηκε από τον μαθηματικό Leonard Euler το 1836 με τίτλο “Theorematum Quorundam ad Numeros Primos Spectantium Demonstratio”. Την ίδια χρονιά ο Euler παρουσίασε μία γενίκευση του “μικρού Fermat” όπως έχει επικρατήσει να ονομάζεται στην βιβλιογραφία. Το μικρό θεώρημα του Fermat αποτελεί ένα από τα πιο βασικά **τεστ ελέγχου πρώτων αριθμών (primality test)**.

Θεώρημα 2.19. (Μικρό θεώρημα Fermat). Για κάθε πρώτο p και κάθε ακέραιο $a \neq 0 \, mod p$ τότε $a^{p-1} \equiv 1 mod p$

Απόδειξη. Αποδεικνύεται εύκολα ότι το σύνολο $\mathbb{Z}_p^* = \{1,2,.., p-1\}$ εφοδιασμένο με την πράξη του πολλαπλασιασμού αποτελεί ομάδα. Έστω $1 \leq a \leq p-1$, $k = |a|$ δηλαδή $a^k \equiv 1 mod p$. Από το θεώρημα Lagrange προκύπτει ότι το k διαιρεί την τάξη της ομάδας $\mathbb{Z}_p^*$, που είναι p-1. Τότε

$$a^{p-1} \equiv a^{km} \equiv (a^k)^m \equiv 1^m \equiv 1 mod p$$

Πόρισμα

Αν $a \in \mathbb{Z}$ τότε $a^p \equiv a mod p$ για κάθε p πρώτο αριθμό.

Θεώρημα 2.20. (Κινέζικο Θεώρημα Υπολοίπων)

Έστω $n_{1,..}, n_k$, $n_i \in \mathbb{Z}$ σχετικά πρώτοι ανά δύο. Έστω επίσης $a_{1,} \dots, a_k \in \mathbb{Z}$.

Τότε $\exists z \in \mathbb{Z}$ τέτοιος ώστε $z \equiv a_i \bmod n_i (i = 1, \ldots, k)$. Επίσης κάθε άλλος ακέραιος $z' \in \mathbb{Z}$ αποτελεί λύση των παραπάνω ισοτιμιών αν και μόνο αν $z \equiv z' \bmod n$ όπου $n = \prod_{i=1}^{k} n_i$.

Σημείωση: Σε πολλά βιβλία το κινέζικο θεώρημα συναντάται στην μορφή:

Έστω n = p*q , $p, q \in \mathbb{Z}$ και σχετικά πρώτοι μεταξύ τους. Τότε $\mathbb{Z}_n \simeq \mathbb{Z}_p \, x \, \mathbb{Z}_q$.

και $\mathbb{Z}_n^* = \mathbb{Z}_p^* \, x \, \mathbb{Z}_q^*$. Ο ισομορφισμός των παραπάνω χώρων καθίσταται δυνατός υπό την δράση της απεικόνισης $f : x \rightarrow (x_p, x_q), x \in 0, \ldots, n-1$ και $x_p \in 0, \ldots, p-1 \, \kappa \alpha \iota \, x_q \in 0, \ldots, q-1$ με τύπο

$f(x) = ([x \bmod p], [x \bmod q])$

Το κινέζικο θεώρημα είναι από τα πιο σπουδαία θεωρήματα της θεωρίας αριθμών. Η μεγάλη του χρησιμότητα έγκειται στο γεγονός πως αφενός με χρήση του αποδεικνύεται το θεώρημα του Euler, αφετέρου υποδεικνύει πως οι πράξεις της πρόσθεσης και του πολλαπλασιασμού modulo n μπορούν να "μετασχηματιστούν" στις αντίστοιχες πράξεις modulo p και q κάνοντας έτσι γρηγορότερους τους υπολογισμούς.

Ιστορικά, οι Κινέζοι χρησιμοποιούσαν αυτό το θεώρημα για να υπολογίσουν το πλήθος των στρατιωτών. Μετά από μια μάχη, οι στρατιώτες στοιχίζονταν σε σειρές (για παράδειγμα) των τριών, μετά των πέντε και μετά των εφτά. Υπολογίζοντας τους περισσευούμενους σε κάθε στοίχιση, οι στρατηγοί μπορούσαν να υπολογίσουν γρήγορα τον αριθμό των ανδρών και έτσι να υπολογίσουν και τις απώλειες.

Πόρισμα 2.21. Έστω φ(n) η συνάρτηση του Euler και $n \in \mathbb{Z}$ Τότε η φ(n) έχει τις ακόλουθες ιδιότητες:

- η φ είναι πολλαπλαστική συνάρτηση δηλαδή : $\varphi(mn) = \varphi(m)\varphi(n) \quad \forall m, n \in \mathbb{Z} : gcd(m, n) = 1$.
- Για κάθε πρώτο και $k \geq 1 : \varphi(p^a) = (p-1)p^{a-1}$.
- $\varphi(n^k) = n^{k-1}\varphi(n).$

Θεώρημα 2.22 (Θεώρημα του Euler).

Αν α είναι ένας ακέραιος σχετικά πρώτος με τον n, τότε $\alpha^{\varphi(n)} = 1 \bmod n$, $\forall n \in \mathbb{N}$

Στην ειδική περίπτωση όπου n=p, p πρώτος αριθμός τότε φ(p) = p-1 επομένως προκύπτει άμεσα ως συμπέρασμα το μικρό θεώρημα του Fermat.

Απόδειξη. Το σύνολο των αριθμών α οι οποίοι είναι σχετικά πρώτοι με τον n αποτελούν δομή ομάδας με την πράξη του πολλαπλασιασμού mod n $(\mathbb{Z}_n^*)$. Η τάξη της ομάδας $\mathbb{Z}_n^*$ είναι φ(n). Έστω $\alpha \in \mathbb{Z}_n^*$. Έστω κ η τάξη του α. Δηλαδή $|\alpha| = \kappa$.Από το θεώρημα Lagrange προκύπτει πως η τάξη του α διαιρεί την τάξη της $\mathbb{Z}_n^*$, δηλαδή $\exists m > 0 : mk = \varphi(n)$. Οπότε $\alpha^{\varphi(n)} = \alpha^{mk} = 1^m = 1 \bmod n$.

2.5. Δακτύλιοι

Ορισμός 2.23. Έστω R ένα σύνολο εφοδιασμένο με δυο διμελής πράξεις που αποκαλούμε πρόσθεση και πολλαπλασιασμό αντίστοιχα. Η δομή (R, +, *) όπου $+: G x G \rightarrow G, (a,b) \rightarrow a+b$ και $*: GxG \rightarrow G, (a,b) \rightarrow a*b$ καλείται **αντιμεταθετικός δακτύλιος** εάν ισχύουν οι ακόλουθες συνθήκες:

- (R, +) είναι μια αβελιανή ομάδα
- Προσεταιριστικοί Νόμοι :
  $$\forall a,b,c \in R: a+(b+c)=(a+b)+c$$
  $$\forall a,b,c \in R: a*(b*c)=(a*b)*c$$
- Αντιμεταθετικοί Νόμοι :
  $$\forall a,b \in R: a+b=b+a$$
  $$\forall a,b \in R: a*b=b*a$$
- Επιμεριστικοί Νόμοι :
  $$\forall a,b,c \in R: a*(b+c)=a*b+a*c$$

Ο αντιμεταθετικός δακτύλιος R καλείται **αντιμεταθετικός δακτύλιος με μοναδιαίο** αν υπάρχει στοιχείο 1 τέτοιο ώστε $\forall \alpha \in R, a*1=1*a=a$

Ένα μη μηδενικό στοιχείο $\alpha \in R$ καλείται **διαιρέτης του μηδενός** αν και μόνο αν υπάρχει στοιχείο $b \neq 0$ τέτοιο ώστε $a*b=0$ ή $b*a=0$.

Ο αντιμεταθετικός δακτύλιος R με μοναδιαίο στοιχείο καλείται **ακέραια περιοχή** αν και μόνο αν $\forall a,b \in R$, $a*b=0 \Rightarrow a=0 \eta \beta=0$.

Ένα στοιχείο u του R καλείται **μονάδα** αν έχει πολλαπλασιαστικό αντίστροφο στο R. Αν κάθε μη μηδενικό στοιχείο του R είναι μονάδα τότε ο R καλείται **δακτύλιος διαίρεσης**.

Σώμα καλείται ένας αντιμεταθετικός δακτύλιος διαίρεσης.

Μια απεικόνιση $\varphi: R \rightarrow R'$ καλείται **ομομορφισμός δακτυλίων** εάν ισχύουν τα εξής:

- $\varphi(a +_R b) = \varphi(\alpha) +_R' \varphi(b)$
- $\varphi(\alpha *_R b) = \varphi(\alpha) *_R' \varphi(b)$

Εάν επιπλέον η φ είναι 1-1 θα ονομάζεται **μονομορφισμός** δακτυλίων, ενώ εάν είναι επί θα ονομάζεται **επιμορφισμός** δακτυλίων. Εάν τυχαίνει η φ να είναι 1-1 και επί, τότε καλείται **ισομορφισμός** δακτυλίων όπως αναφέρθηκε και στις ομάδες.

Ορισμός 2.24. Έστω ο δακτύλιος (R, +, ·) . Μια υποομάδα (I, +) της προσθετικής υποομάδας (R, +) καλείται **ιδεώδες** του δακτυλίου R και συμβολίζεται με $I \lhd R$ αν ισχύει:

- $\forall x \in I$ και $\forall r \in R, x r \kappa \alpha \iota r x \in I$

Ένα ιδεώδες καλείται **κύριο** αν παράγεται από ένα στοιχείο του α του R.

Δύο ιδεώδη I, J ενός αντιμεταθετικού δακτυλίου είναι καλούνται **πρώτα ή σχετικά πρώτα** αν και μόνο αν I+J=R, όπου $I+J=\{i+j:, i\in I, j\in J\}$

Ορισμός 2.25. Έστω R ένας αντιμεταθετικός δακτύλιος με μοναδιαίο. Έστω ένα ιδεώδες I του R, $I \lhd R$. Ορίζουμε την σχέση ισοδυναμίας ~ στο R ως εξής: a~b $\Leftrightarrow a-b\in I$, $a, b\in R$.
Ορίζουμε το **σύνολο πηλίκο** R / ~ = $\{r+I : i\in I\}$.
Ορίζουμε τις πράξεις :

- (r+I) + (s+I) = (r+s) + I
- (r+I) · (s+I) = (r·s) +I

Η δομή (R /~, +, ·) αποδεικνύεται πως αποτελεί δακτύλιο και πιο συγκεκριμένα ονομάζεται **δακτύλιος πηλίκο**.

Ορισμός 2.25. Έστω R ένας αντιμεταθετικός δακτύλιος. Το σύνολο των πολυωνύμων

$$c_n x^n + c_{n-1} x^{n-1} + \ldots + c_1 x + c_0$$

με ανεξάρτητη μεταβλητή το x και συντελεστές $c_i \in R$, i=1,...,n εφοδιασμένο με τις γνωστές πράξεις της πρόσθεσης και του πολλαπλασιασμού πολυωνύμων αποτελεί δακτύλιο που συμβολίζεται με $R[x]$ και ονομάζεται **δακτύλιος πολυωνύμων**.
Έστω $p(x)\in R[x]$ ένα **μονικό** (ο συντελεστής του μεγιστοβάθμιου όρου ισούται με 1) πολυώνυμο βαθμού n $p(x)=x^n+p_{n-1}x^{n-1}+\ldots+p_0$.
Ξεχωριστό ενδιαφέρον παρουσιάζει ο δακτύλιος των πολυωνύμων modulo p(x) που συμβολίζεται με: $R_p=R[x]/p(x)$.

2.6. Διανυσματικοί Χώροι (Vector Space)

Ορισμός 2.26. Διανυσματικός χώρος V επί ενός σώματος F είναι μια αβελιανή ομάδα (V, +), μαζί με μια πράξη πολλαπλασιασμού : F × V → V τέτοια, ώστε για κάθε $a, b\in F$ και $u, v\in V$, να ικανοποιούνται τα ακόλουθα αξιώματα:

- α(u + w) = αu + αw.
- (α + b)u = αu + bu.
- (αb)u = α(bu).
- 1u = u.

Τα στοιχεία του διανυσματικού χώρου V καλούνται **διανύσματα**, ενώ τα στοιχεία του σώματος F καλούνται **βαθμωτά**. Τέλος, η πράξη ομάδας + καλείται **διανυσματική πρόσθεση**, ενώ η πράξη πολλαπλασιασμού καλείται **βαθμωτός πολλαπλασιασμός**.

Ορισμός 2.27. Έστω V ένας διανυσματικός χώρος επί ενός σώματος F. Ένας **υπόχωρος** του V είναι μία προσθετική υποομάδα U της V η οποία είναι κλειστή ως προς τον βαθμωτό πολλαπλασιασμό, δηλαδή $au\in U$ για κάθε $a\in F$ και $u\in U$.

Ένας υπόχωρος ενός διανυσματικού χώρου είναι επίσης ένας διανυσματικός χώρος. Με άλλα λόγια, εάν W μη κενό υποσύνολο ενός δ.χ. V, εφοδιασμένο με τις πράξεις του V είναι και αυτό δ.χ., τότε λέμε ότι είναι **διανυσματικός υπόχωρος** του V.

Ορισμός 2.28. Έστω $S=\{u_{1,}u_{2,..},u_n\}$ ένα πεπερασμένο υποσύνολο ενός διανυσματικού χώρου V επί ενός σώματος F. Γραμμικός συνδυασμός του S είναι μια έκφραση της μορφής $a_1u_1+a_2u_2+...+a_nu_n$, όπου κάθε $a_i\in F$.

Εάν $a_1+a_2+...+a_n=1$, τότε το παραπάνω διάνυσμα λέγεται **ομοπαραλληλικός συνδυασμός** (affine combination) των $u_{1,}u_{2,}...,u_n$.

Ορισμός 2.29. Το σύνολο S είναι **γραμμικά εξαρτημένο** επί ενός σώματος F εάν υπάρχουν $a_{1,}a_{2,}...,a_n$ όχι όλα μηδέν , τέτοια ώστε $a_1u_1+a_2u_2+...+a_nu_n=0$. Εάν δεν υπάρχουν τέτοια βαθμωτά, τότε το S είναι **γραμμικά ανεξάρτητα** επί του F.

Ορισμός 2.30. Βάση (B) ενός διανυσματικού χώρου V είναι ένα υποσύνολο του, $B\subset V$, γραμμικά ανεξάρτητο, το οποίο παράγει ολόκληρο τον V.

Ορισμός 2.31. Διάσταση (dimV) ενός δ.χ V ονομάζεται το πλήθος στοιχείων μιας οποιασδήποτε βάσης του.

2.7. Θεωρία Δικτυωμάτων

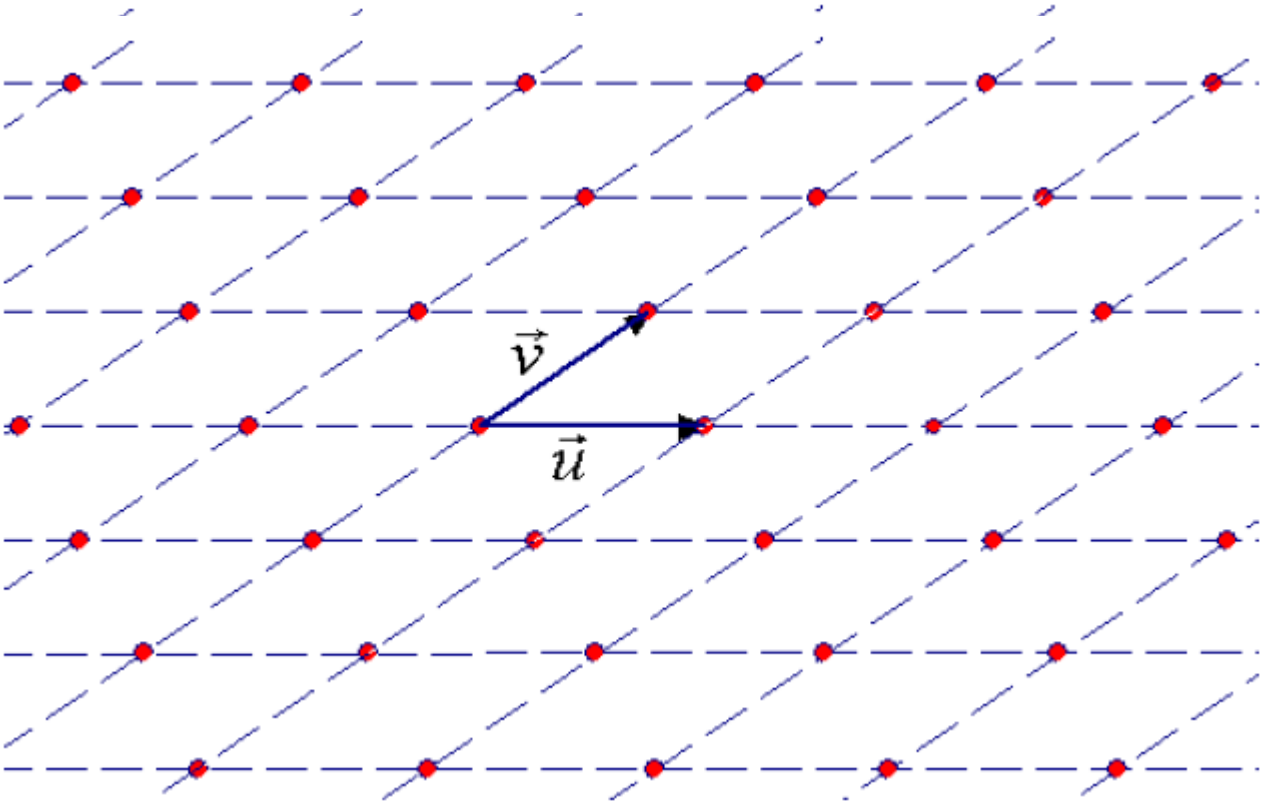
Τα δικτυώματα παίζουν πολύ σημαντικό ρόλο στη σύγχρονη κρυπτογραφία, ειδικά από τότε που ο κλάδος της μετα-κβαντικής κρυπτογραφίας έχει συγκεντρώσει τεράστιο ερευνητικό ενδιαφέρον. Παρόλο που τα δικτυώματα έχουν μια σχετική απλότητα κρύβουν μέσα τους μία πολύ πλούσια και ενδιαφέρουσα δομή. Θεωρείται άλλωστε πως η κρυπτογραφία που βασίζεται στα δικτυώματα καθώς και τα “δύσκολα προβλήματα” που στηρίζονται σε αυτήν, είναι το είδος εκείνο το οποίο παρουσιάζει την μεγαλύτερη αντίσταση σε επίθεση από κβαντικούς υπολογιστές, σε αντίθεση με τα προβλήματα παραγοντοποίησης που δεν παρουσιάζουν καμία αντίσταση σε μια πιθανή επίθεση από έναν κβαντικό υπολογιστή που τρέχει τον αλγόριθμο του Shor [2].

Με τον όρο «δικτύωμα» αναφερόμαστε σε ένα σύνολο σημείων σε χώρο-διαστάσεων με περιοδική δομή. Από ιστορικής απόψεως, τα δικτυώματα άρχισαν να διερευνούνται γύρω στα τέλη του 18ου αιώνα από διακεκριμένους μαθηματικούς όπως οι Lagrange, Gauss, και αργότερα από τον Minkowski. Η θεωρία δικτυώματος (lattice theory) είναι η μελέτη των συνόλων των αντικειμένων που είναι γνωστά ως δικτυώματα. Είναι αποτέλεσμα της μελέτης αλγεβρών Boole και παρέχει ένα πλαίσιο για την ενοποίηση της μελέτης των κατηγοριών (categories) ή των διατεταγμένων συνόλων (ordered sets) στα μαθηματικά. Η μελέτη της θεωρίας δικτυώματος γνώρισε μεγάλη ώθηση από μία σειρά άρθρων και ένα επακόλουθο βιβλίο–ορόσημο που γράφτηκε από τον G.Birkhoff το 1967 και εκδόθηκε το 1979. Στη σύγχρονη εποχή, τα δικτυώματα αποτελούν αντικείμενο ενεργού έρευνας στην επιστήμη των υπολογιστών. Χρησιμοποιούνται ως ένα αλγοριθμικό εργαλείο για την επίλυση μιας ευρείας ποικιλίας προβλημάτων.

Έχουν, επίσης, πολλές εφαρμογές στην κρυπτογραφία και στην κρυπτανάλυση και από τη σκοπιά της υπολογιστικής πολυπλοκότητας διαθέτουν κάποιες μοναδικές ιδιότητες. Η θεωρητική μελέτη των δικτυωμάτων (και ειδικότερα η σύνδεσή τους με τα κυρτά σώματα) συχνά καλείται γεωμετρία αριθμών, όνομα που αποδόθηκε σε αυτήν από τον Hermann Minkowski το 1910 στο ομότιτλο βιβλίο του [3]. Η γεωμετρία των αριθμών αποτελεί ένα σημαντικό τμήμα της θεωρίας αριθμών, το οποίο έχει τις ρίζες του σε ιστορικά προβλήματα, όπως είναι οι γενικεύσεις σε υψηλές διαστάσεις του Ευκλείδειου αλγόριθμου για τον υπολογισμό του Μ.Κ.Δ. δύο ακεραίων και η διαδικασία της διευθέτησης μη επικαλυπτόμενων σφαιρών μέσα σε ένα γραμμικό χώρο που τις περιέχει, η οποία είναι ευρέως γνωστή ως *sphere packing*.

Ορισμός 2.32. Έστω $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n$ είναι μια βάση του $\mathbb{R}^n$ και $n \geq 1$. **Δικτύωμα** L διάστασης n και βάσης $\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n$ είναι το σύνολο όλων των γραμμικών συνδυασμών των διανυσμάτων της βάσης με ακέραιους συντελεστές, δηλαδή:

$$L = \mathbb{Z}\vec{x}_1 + \mathbb{Z}\vec{x}_2 + \ldots + \mathbb{Z}\vec{x}_n = \{\sum_{i=1}^{n} a_i \vec{x}_i \mid a_1, a_2, \ldots, a_n \in \mathbb{Z}\}$$



Εικόνα 2.32. : Γραφική αναπαράσταση του δικτυώματος στον $\mathbb{R}^2$

Ορισμός 2.33. Έστω $\mathbb{R}^m$ ο m – διάστατος ευκλείδειος χώρος και $\vec{b}_1, \vec{b}_2, \ldots, \vec{b}_n \in \mathbb{R}^m$ γραμμικώς ανεξάρτητα.

Το δικτύωμα που παράγεται από τα διανύσματα $\vec{b}_i$ ορίζεται ως

$L(\vec{b}_1, \vec{b}_2, \ldots, \vec{b}_n) = \{\sum_{i=1}^{n} x_i \vec{b}_i : x_i \in \mathbb{Z}\} = \{B\vec{x} : \vec{x} \in \mathbb{Z}^n\} \subset span(B) = \{B\vec{x} : x \in \mathbb{R}^n\}$. Το n καλείται **βαθμός** και το m **διάσταση** του δικτυώματος.

Η ακολουθία διανυσμάτων $\{\vec{b}_1, \ldots, \vec{b}_n\}$ καλείται **βάση** του δικτυώματος. Μπορούμε να αναπαραστήσουμε την βάση δικτυώματος ως έναν πίνακα:

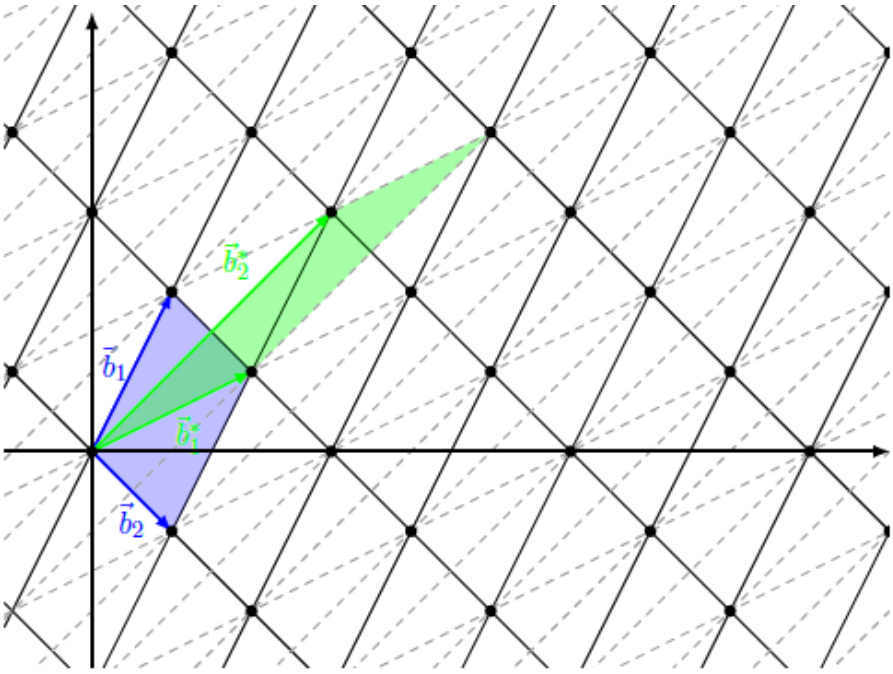
$B=[\vec{b}_{1,}\vec{b}_{2,}\ldots,\vec{b}_n]\in\mathbb{R}^{mxn}$, $m\geq n$. Στην ειδική περίπτωση που n=m το δικτύωμα καλείται **πλήρες**.

Μία βάση $B=(\vec{b}_i,\ldots,\vec{b}_n)\in\mathbb{Z}^{nxn}$ λέμε πως είναι σε **Ερμιτιανή Κανονική μορφή (Hermite Normal Form – HNF)** εάν:

$b_{i,j}=\begin{Bmatrix}0,i>j\\0<b_{i,j}<b_{i,i},\alpha\lambda\lambda\iota\acute{\omega}\varsigma\end{Bmatrix}$. Μια HNF βάση είναι μοναδική για κάθε δικτύωμα και μπορεί να υπολογιστεί σε πολυωνυμικό χρόνο, άρα αποτελεί μια "κακή" (bad basis) επιλογή για βάση για κρυπτογραφικούς σκοπούς.

Μια βάση θα λέγεται **"καλή" (good)** αν τα διανύσματά της είναι σχεδόν ορθογώνια ενώ αν είναι σχεδόν παράλληλα θα λέγεται **"κακή" (bad)**.

Να σημειωθεί ότι ένα δικτύωμα μπορεί να έχει πολλές βάσεις. Για παράδειγμα εάν $\{b_{1,}b_2\}$ είναι βάση ενός δικτυώματος L του $\mathbb{R}^2$ τότε $\{b_{1,}b_1+b_2\}$ είναι επίσης βάση του L. Πιο γενικά αν B βάση ενός δικτυώματος L διάστασης n και U_{nxn} είναι ένας πίνακας με στοιχεία από το $\mathbb{Z}$ με ορίζουσα 1 τότε BU είναι επίσης βάση του L.



Εικόνα 2.33.: Παράδειγμα 2 διαφορετικών βάσεων ενός ίδιου δικτυώματος

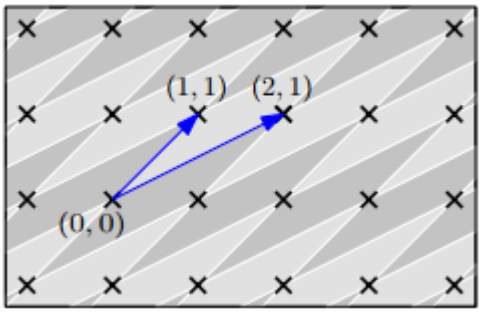
Ένα δικτύωμα μπορεί διαφορετικά να οριστεί σαν ένα μη κενό σύνολο Λ του $\mathbb{R}^n$ το οποίο είναι κλειστό ως προς την αφαίρεση (εάν $\vec{x}\in\Lambda\,\kappa\alpha\iota\,\vec{y}\in\Lambda\,,\tau\acute{o}\tau\varepsilon\,\vec{x}-\vec{y}\in\Lambda$) , την πρόσθεση (εάν $\vec{x}\in\Lambda\,\kappa\alpha\iota\,\vec{y}\in\Lambda\,,\tau\acute{o}\tau\varepsilon\,\vec{x}+\vec{y}\in\Lambda$) και είναι

διακριτό (υπάρχει θετικός ακέραιος Λ>0 τέτοιος ώστε η απόσταση μεταξύ δυο οποιονδήποτε διανυσμάτων του δικτυώματος είναι μεγαλύτερη ή το πολύ ίση με Λ).

Ορισμός 2.34. Θεμελιώδες Παραλληλεπίπεδο (Fundamental Parallelepiped)

Για οποιασδήποτε βάση B ενός δικτυώματος L ορίζουμε ως **θεμελιώδες παραλληλεπίπεδο (Fundamental Parallelepiped)** το σύνολο $P(B)=\{B\vec{x}, \vec{x}\in\mathbb{R}^n, \forall i: 0\le x_i\le 1\}$.

Παραδείγματα θεμελιωδών παραλληλεπιπέδων απεικονίζονται στις γκρι περιοχές του παρακάτω σχήματος.



Εικόνα 2.34.1. : Θεμελιώδες Παραλληλεπίπεδο

Έστω $\vec{v}\in\mathbb{R}^n$. Ορίζουμε την πράξη $(\vec{v},B)\rightarrow \vec{v}\ mod B=\vec{u}: \vec{v}-\vec{u}\in L$. Το διάνυσμα $\vec{u}$ μπορεί να υπολογιστεί από την σχέση $\vec{u}=\vec{v}-B*\lceil B^{-1}*\vec{v}\rfloor$ όπου ⌈*⌋ είναι το σύμβολο της στρογγυλοποίησης προς τον πλησιέστερο ακέραιο.

Επίσης με την βοήθεια του θεμελιώδες παραλληλεπιπέδου ορίζετα η έννοια της **ορίζουσας** ενός δικτυώματος L να είναι ο όγκος του n-διάστατου του P(B) και δίδεται από την σχέση $det(L)=\sqrt{det(B^T B)}$.

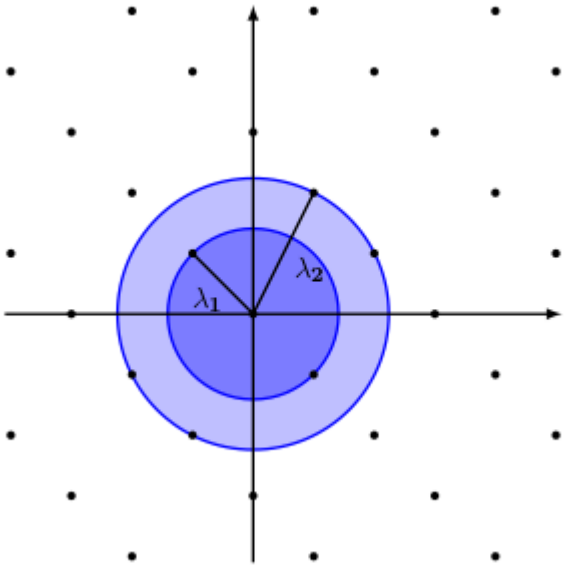
Δυικό δικτύωμα (dual lattice) ενός δικτυώματος L είναι το σύνολο $L^*=\{\vec{x}\in span(B): \forall \vec{v}\in L, \langle \vec{x},\vec{v}\rangle\in\mathbb{Z}\}$, όπου $\langle . \rangle$ συμβολίζει ένα εσωτερικό γινόμενο.

Ορίζουμε **m-διάστατη ανοιχτή σφαίρα** ακτίνας r και κέντρου $\vec{0}$ και συμβολίζουμε με $B_m(\vec{0},r)$ το σύνολο $B_m(\vec{0},r)=\{\vec{x}\in\mathbb{R}^m: ||\vec{x}||<r\}$.

Ορίζουμε επίσης το **i-οστό διαδοχικό ελάχιστο (i-th successive minimum)** και συμβολίζουμε με $\lambda_i(L)$ ως την ακτίνα της μικρότερης σφαίρας που περιέχει i γραμμικά ανεξάρτητα διανύσματα του δικτώματος L.

Δηλαδή: $\lambda_i(L)=inf\{r: dim[span(L\cap B(\vec{0},r))]\ge i\}$.

Η τιμή $\lambda_1(L)$ καλείται **ελάχιστη απόσταση** του L.



Εικόνα 2.34.2.: Δυο διαδοχικά ελάχιστα $\lambda_{1,}\lambda_2$

Ορισμός 2.35. 'Εστω $R=\mathbb{Z}[x]/f(x)$ ο δακτύλιος πολυωνύμων modulo κάποιο μονικό πολυώνυμο f(x) βαθμού n. Χρησιμοποιώντας το σύνολο $\{gmodf : g\in\mathbb{Z}[x]\}$ μπορούμε κάθε στοιχείο του να το ταυτίσουμε με ένα διάνυσμα μέσω μιας ισομορφικής απεικόνισης $\varphi:\mathbb{Z}[x]/f(x)\rightarrow\mathbb{Z}^n$. Η φ αποδεικνύεται πως είναι 1-1 και επί και έτσι κάθε ιδεώδες $I\subset\mathbb{Z}[x]/f(x)$ καθορίζει ένα αντίστοιχο υποδικτύωμα ακεραίων $L(I)\subset\mathbb{Z}^n$. Ένα **ιδεώδες δικτυώμα** είναι ένα δικτύωμα ακεραίων τέτοιο ώστε $B=\{gmodf : g\in I, g\ \mu ονικό\ \beta αθμού\ n\}$ και ένα ιδεώδες $I\subset\mathbb{Z}[x]/f(x)$.

Στη σύγχρονη κρυπτογραφία μία από τις βασικές απαιτήσεις είναι η απόδειξη της ασφάλειας των χρησιμοποιούμενων κρυπτογραφικών αλγορίθμων. Ένας από τους βασικότερους τρόπους είναι με το να βασιστούν τα κρυπτογραφικά συστήματα σε γνωστά δύσκολα μαθηματικά προβλήματα, επιδιώκοντας να φτιάξουμε κρυπτογραφικούς αλγόριθμους οι οποίοι να μπορούν να υπολογιστούν εύκολα από τον οποιοδήποτε αλλά η αντίστροφη τους να μπορεί να γίνει μόνο από εξουσιοδοτημένες οντότητες.

Έτσι στην περίπτωση του RSA αλγόριθμου το πρόβλημα είναι η *παραγοντοποίηση μεγάλων ακεραίων*, στο κρυπτοσύστημα McEliece το πρόβλημα είναι η *εύρεση της ελάχιστης απόστασης* για οποιονδήποτε γραμμικό κώδικα, ενώ στην κρυπτογράφηση ElGamal το πρόβλημα είναι η *εύρεση του διακριτού λογαρίθμου* μιας κυκλικής ομάδας της μορφής $(\mathbb{Z}_p)^x$.

Στην περίπτωση της χρήσης δικτυωμάτων τα πιο γνωστά τέτοια προβλήματα είναι το *πρόβλημα του βραχύτερου διανύσματος (Shortest Vector Problem – SVP)* και το *πρόβλημα του εγγύτατου διανύσματος (Closest Vector Problem – CVP)* [8].

Shortest Vector Problem (SVP): Δοθέντος ενός δικτυώματος $L = L(b_{1,} b_{2,\ldots}, b_n) \subset \mathbb{R}^n$ παραγόμενου από τα γραμμικώς ανεξάρτητα διανύσματα $b_{1,} b_{2,\ldots}, b_n \in \mathbb{Q}^n$ και ενός ρητού αριθμού r>0, να βρεθεί ένα μη μηδενικό διάνυσμα $u \in L$ τέτοιο ώστε $\|u\|_2 < r$.

Σύμφωνα με το πρώτο **θεώρημα Minkowski**, κάθε δικτύωμα L τάξης n περιέχει ένα μη μηδενικό διάνυσμα με μήκος το πολύ $\sqrt{n}(detL)^{\frac{1}{n}}$. Η ύπαρξη του όμως από μόνη της δεν εξασφαλίζει την ύπαρξη και ενός αλγορίθμου για την εύρεση ενός τέτοιου διανύσματος. Στην πραγματικότητα δεν υπάρχει μέχρι στιγμής κανένας γνωστός αποτελεσματικός αλγόριθμος ο οποίος βρίσκει τέτοια βραχέα διανύσματα.

Closest Vector Problem (CVP) : Δοθέντος ενός δικτυώματος $L \subset \mathbb{R}^n$, ενός σημείου-στόχου $t \in \mathbb{R}^n$ και ενός φράγματος απόστασης d, το πρόβλημα του εγγύτατου διανύσματος αναζητά ένα σημείο $v \in L$ σε απόσταση $\|t - v\| \le d$ από τον στόχο, εφόσον ένα τέτοιο σημείο είναι στοιχείο του δικτυώματος L.

Αξίζει να αναφέρουμε ένα ακόμη πρόβλημα στο οποίο στηρίζεται το σχήμα του Gentry που θα αναπτύξουμε παρακάτω.

γ-Bounded Distance Decoding Problem (γ-BDDP): Δοθέντος βάσης B ενός δικτυώματος L διάστασης n και ενός διανύσματος $\vec{t} \in \mathbb{R}^n$ έτσι ώστε $dist(L, \vec{t}) * \gamma \le \lambda_1(L)$, να βρεθεί το πλησιέστερο μη μηδενικό διάνυσμα $\vec{v} \in L$ στο $\vec{t}$ δηλαδή $\|\vec{t} - \vec{v}\| \le \gamma * dist(L, \vec{t})$ όπου $dist(L, \vec{t}) = min_{\vec{v} \in L}\{\|\vec{t} - \vec{v}\|\}$

2.8. Μέτρο Πιθανότητας

Σε αυτή την ενότητα θα δωθεί μια σύντομη εισαγωγή σε κάποιες έννοιες της θεωρίας μέτρου [9] που θα μας βοηθήσουν να ορίσουμε το μέτρο πιθανότητας. Η θεωρία μέτρου στα μαθηματικά περιλαμβάνει την αυστηρή αξιωματική θεμελίωση και επίσης τη γενίκευση των εννοιών του μήκους, του εμβαδού και του όγκου. Οι εφαρμογές επεκτείνονται και σε άλλες, πέρα από τις γεωμετρικές, έννοιες του μέτρου. Σε απλουστευμένη αλλά αδόκιμη ορολογία, για δεδομένο σύνολο S ονομάζουμε "μέτρο" οποιαδήποτε διαδικασία ή κανόνα αποδίδει ένα "μέγεθος" σε κάθε ένα υποσύνολο του S κατά συνεπή (δηλ. χωρίς αντιφάσεις) τρόπο.

Ορισμός 2.36. Έστω F μια συλλογή από υποσύνολα ενός βασικού συνόλου Ω, $F \subset 2^{\Omega}$. Η F καλείται **άλγεβρα** εάν $\Omega \in F$ και το F είναι κλειστό ως προς τις συνολοθεωρητικές πράξεις του συμπληρώματος και της πεπερασμένης τομής.

Πιο φορμαλιστικά:

- $\Omega \in F$
- Για κάθε $A \in F$ τότε $A^c \in F$

- Για κάθε ακολουθία $A_{1,}A_{2,}...,A_n \in F$ τότε $\overset{n}{\underset{i=1}{U}} A_i \in F$

Εάν επιπλέον απαιτήσουμε να μην είναι κλειστή ως προς πεπερασμένες αλλά και ως προς αριθμήσιμα άπειρες ενώσεις δηλαδή:

- Για κάθε ακολουθία $A_{1,}A_{2,..},A_n,...\in F$ τότε $\overset{\infty}{\underset{i=1}{U}} A_i \in F$

Τότε η συλλογή F καλείται **σ-άλγεβρα**.

Εάν F είναι μια σ-άλγεβρα επί του Ω τότε η δυάδα (Ω, F) καλείται **μετρήσιμος χώρος** και τα στοιχεία της F καλούνται **μετρήσιμα σύνολα**.

Παραδείγματα.

1. Έστω $F=\{\varnothing,\Omega\}$. Η F είναι σ-άλγβερα επί του Ω.
2. Το δυναμοσύνολο του Ω αποτελεί μια σ-άλγεβρα επί του Ω, $F=\{2^{\Omega}\}$.
3. Η οικογένεια συνόλων $F=\{A\in 2^{\Omega} | A \text{ ή } A^c \text{ είναι αριθμήσιμο}\}$ είναι μια σ-άλγεβρα επί του Ω.

Ορισμός 2.37. Έστω (Ω, F) ένας μετρήσιμος χώρος. **Μέτρο** του χώρου καλείται μια απεικόνιση $\mu: F \rightarrow [0,\infty]$ με τις ιδιότητες:

- $\mu(\varnothing)=0$
- Εάν $A_i \in F \Rightarrow \mu(\overset{\infty}{\underset{n=1}{U}} A_i)=\sum_{i=1}^{\infty}\mu(A_i)$

Η τριάδα (Ω, F, μ) καλείται **χώρος μέτρου**. Εάν μ(Ω)=1 τότε το μέτρο καλείται μέτρο πιθανότητας και συχνά συμβολίζεται με P και η αντίστοιχα τριάδα (Ω, F, P) καλείται χώρος πιθανότητας με:

$$P: F \rightarrow [0,1]$$

Το Ω καλείται σε αυτή την περίπτωση **δειγματικός χώρος** και τα στοιχεία του F **ενδεχόμενα**.
Αν F είναι μία οικογένεια υποσυνόλων του Ω τότε αποδεικνύεται εύκολα ότι υπάρχει μοναδική ελάχιστη σ-άλγεβρα που την περιέχει. Αυτή συμβολίζεται με σ(F). Αν ο Ω είναι το σύνολο R των πραγματικών και T η οικογένεια των ανοιχτών συνόλων του, τότε το σύνολο σ(T) συμβολίζεται με B(R) και τα στοιχεία του καλούνται **σύνολα Borel** [10] του R (πρόκειται δηλαδή για την ελάχιστη σ-άλγεβρα που περιέχει τα ανοιχτά σύνολα. Αποδεικνύεται ότι περιέχει και όλα τα κλειστά σύνολα και όλα τα διαστήματα του R).

2.9. Θεωρία Πολυπλοκότητας

Πριν προχωρήσουμε στο υπόλοιπο κομμάτι της συγκεκριμένης διατριβής είναι σημαντικό να παρουσιάσουμε κάποιες πτυχές της θεωρίας πολυπλοκότητας. Σκοπός της συγκεκριμένης θεωρίας είναι να υπολογίσει τους υπολογιστικούς πόρους που απαιτούνται για την επίλυση των διαφόρων υπολογιστικών προβλημάτων και να τα ταξινομήσει σε κλάσεις πολυπλοκότητας με βάση την υπολογιστική “δυσκολία” επίλυσης τους. Όταν αναφερόμαστε σε πόρους υπολογιστικούς μπορεί κάθε φορά να εννοούμε διαφορετικά πράγματα. Ένας είδος υπολογιστικού πόρου για παράδειγμα είναι η μνήμη (space) που απαιτείται για την εκτέλεση του αλγορίθμου ενώ σε άλλες περιπτώσεις είναι ο χρόνος (time) που απαιτείται για να ολοκληρωθεί ο αλγόριθμος που επιλύει το πρόβλημα. Στην Κρυπτογραφία οι αλγόριθμοι που αναπτύσσονται και εκτελούνται είναι εξαρτώμενοι από τον χρόνο άρα ως υπολογιστικό πόρο θα εννοούμε τον χρόνο. Με τον όρο υπολογιστικό πρόβλημα εννοούμε μια μαθηματική μοντελοποίηση ενός φυσικού προβλήματος που μπορεί να επιλυθεί από μια υπολογιστική μηχανή. Στην κατηγορία των υπολογιστικών προβλημάτων υπάρχουν δυο πεδία με το μεγαλύτερο επιστημονικό ενδιαφέρον. Το πρώτο αφορά την έρευνα πάνω στην εύρεση αλγορίθμων που επιλύουν συγκεκριμένα προβλήματα. Το δεύτερο πεδίο αφορά την θεωρία πολυπλοκότητας [11] που μελετά πότε ένα πρόβλημα είναι επιλύσιμο ή όχι από μια μηχανή ακόμη και αν δοθούν άπειροι υπολογιστικοί πόροι.

Στην επιστήμη των υπολογιστών όταν αναφερόμαστε σε μια τέτοια μηχανή θα εννοούμε την μηχανή που ορίζει η σχολή *Church-Turing*. Αναφερόμαστε στην μηχανή Turing η οποία είναι μια υποθετική συσκευή η οποία χειρίζεται σύμβολα σύμφωνα με ένα σύνολο κανόνων. Παρά την απλότητά της, μια μηχανή Turing μπορεί να προσαρμοστεί ώστε να προσομοιώνει την λογική οποιουδήποτε αλγορίθμου, και είναι ιδιαίτερα χρήσιμη στο να εξηγεί τις λειτουργίες μιας κεντρικής μονάδας επεξεργασίας στο εσωτερικό του υπολογιστή.

Η μηχανή του Turing εφευρέθηκε το 1936 από τον Alan Turing. Η **μηχανή Turing** δεν προορίζεται σαν μια τεχνολογία υπολογιστών αλλά κυρίως σαν μια υποθετική κατασκευή που αντιπροσωπεύει μια υπολογιστική μηχανή. Οι μηχανές Turing μας βοηθούν να καταλάβουμε τα όρια του μηχανικού υπολογισμού.

Πιο συγκεκριμένα, μια μηχανή Turing αποτελείται από:

1. Μία ταινία που χωρίζεται σε κελιά, το ένα δίπλα στο άλλο. Κάθε κελί περιέχει ένα σύμβολο από ένα πεπερασμένο αλφάβητο. Το αλφάβητο περιέχει ένα ειδικό κενό σύμβολο (το οποίο εδώ συμβολίζουμε με "0") και ένα ή παραπάνω άλλα σύμβολα. Υποθέτουμε πώς αυτή η ταινία είναι απείρως επεκτάσιμη προς τα αριστερά και προς τα δεξιά, δηλαδή μία μηχανή Turing είναι πάντα προμηθευμένη με όση ταινία της χρειάζεται για τους υπολογισμούς. Κελιά τα οποία δεν έχουν συμπληρωθεί, υποθέτουμε πως είναι εξοπλισμένα με το κενό γράμμα. Σε κάποια μοντέλα, η ταινία έχει αριστερό άκρο, το οποίο είναι εξοπλισμένο με ένα ειδικό σύμβολο, όμως είναι απείρως επεκτάσιμη προς τα δεξιά.

2. Μία κεφαλή που μπορεί να διαβάζει και να γράφει σύμβολα πάνω στην ταινία και που μπορεί να μετακινεί την ταινία κατά ένα (και μόνο ένα) κελί κάθε φορά. Σε μερικά μοντέλα κινείται μόνο η κεφαλή, ενώ η ταινία παραμένει σταθερή.

3. Ένα μητρώο καταστάσεων που αποθηκεύει μία κατάσταση της μηχανής, μία κάθε φορά από ένα πεπερασμένο πλήθος. Ανάμεσα σε αυτές είναι η ειδική αρχική κατάσταση με την οποία ξεκινά το μητρώο. Αυτές οι καταστάσεις, γράφει ο Turing, αντικαθιστούν την "πνευματική κατάσταση" στην οποία αρχικά θα ήταν το άτομο που θα έκανε τους υπολογισμούς.

4. Ένας πεπερασμένος πίνακας (ενίοτε ονομαζόμενος ως διαδραστικός πίνακας ή συνάρτηση μετάβασης) από οδηγίες και ορίζει στη μηχανή να κάνει τα ακόλουθα σε σειρά (για τα πενταπλά μοντέλα):

- Είτε απαλείφει, ή καταγράφει ένα σύμβολο και έπειτα,
- Μετακινεί την κεφαλή και έπειτα,
- Θεωρεί την ίδια ή μια καινούργια κατάσταση όπως ορίζεται.

Παρατηρήστε ότι κάθε κομμάτι της μηχανής (δηλαδή οι καταστάσεις και τα σύμβολα) και οι πράξεις της (όπως η εκτύπωση, η διαγραφή και η κίνηση της ταινίας) είναι πεπερασμένα, ευδιάκριτα και διακεκριμένα. Το πιθανώς απεριόριστο μήκος της ταινίας είναι που δίνει στη μηχανή μια απεριόριστη χωρητικότητα.

Φορμαλιστικά, μια μηχανή Turing ορίζεται ως η επτάδα $M=(Q,\Gamma,b,\Sigma,\delta,q_0,F)$, όπου

- Q είναι το πεπερασμένο σύνολο καταστάσεων.
- Γ είναι το πεπερασμένο σύνολο των επιτρεπτών συμβόλων της ταινίας.
- $\beta\in\Gamma$ και είναι το κενό σύμβολο (το μοναδικό σύμβολο που επιτρέπεται να εμφανιστεί άπειρες φορές στην ταινία σε οποιοδήποτε στάδιο του υπολογισμού).
- $\Sigma\subset\Gamma-\{b\}$ είναι το σύνολο των συμβόλων εισόδου.
- $\delta:(Q-F)\,x\,\Gamma\rightarrow Q\,x\,\Gamma\,x\,\{L,R\}$, είναι η συνάρτηση μετάβασης όπου το L σημαίνει αριστερή ολίσθηση και το R δεξία ολίσθηση.
- $q_0\in Q$, είναι η αρχική κατάσταση της μηχανής.
- $F\subset Q$ είναι το σύνολο των τελικών καταστάσεων.

Για τον ορισμό των δυο σπουδαιότερων κλάσεων πολυπλοκότητας **N** και **NP** είναι αναγκαία η διαφοροποίηση μεταξύ μιας **ντετερμινιστικής (DTM)** και μιας **μη ντετερμινιστικής μηχανής (NTM)** Turing. Χωρίς να ακολουθήσουμε αυστηρό φορμαλισμό γιατί δεν αφορά το αντικείμενο της παρούσης διπλωματικής μπορούμε να πούμε ότι μια μη ντετερμινιστική μηχανή Turing (NTM) είναι μια μηχανή Turing που μπορεί να εκτελέσει οποιαδήποτε εντολή-δράση από ένα σύνολο συγκεκριμένων εντολών-δράσεων ανεξάρτητα από την κατάσταση στην οποία βρίσκεται. Αντίθετα μια ντετερμινιστική μηχανή Turing (DTM), μπορεί να εκτελέσει μία προκαθορισμένη ενέργεια για κάθε κατάσταση που βρίσκεται.

Επιστρέφοντας στα προηγούμενα η **χρονική πολυπλοκότητα** ενός προβλήματος ορίζεται ως ο αριθμός των βημάτων που απαιτούνται για να επιλυθεί ένα υπολογιστικό πρόβλημα χρησιμοποιώντας έναν αλγόριθμο.

Ορισμός 2.38. (Big-O Notation). Έστω f(x) και g(x) δύο συναρτήσεις με πεδίο ορισμού ένα σύνολο $M\subset\mathbb{R}$. Τότε γράφουμε $f(x)=O\,g(x)$ καθώς το $x\rightarrow\infty$ αν και μόνο $\exists x_0,k\in\mathbb{R}$ τέτοιο ώστε $|f(x)|\leq k\,g(x)\,\forall\,x>x_0,x\in\mathbb{R}$

Κλάση πολυπλοκότητας είναι ένα σύνολο από συναρτήσεις που μπορούν να υπολογιστούν εντός συγκεκριμένου αλγοριθμικού χρόνου (time complexity). Ορίζονται απο τρείς παραμέτρους:

1. Το **υπολογιστικό πρόβλημα.** Αναφερόμαστε συνήθως σε προβλήματα απόφασης, αναζήτησης, βελτιστοποίησης κ.α.

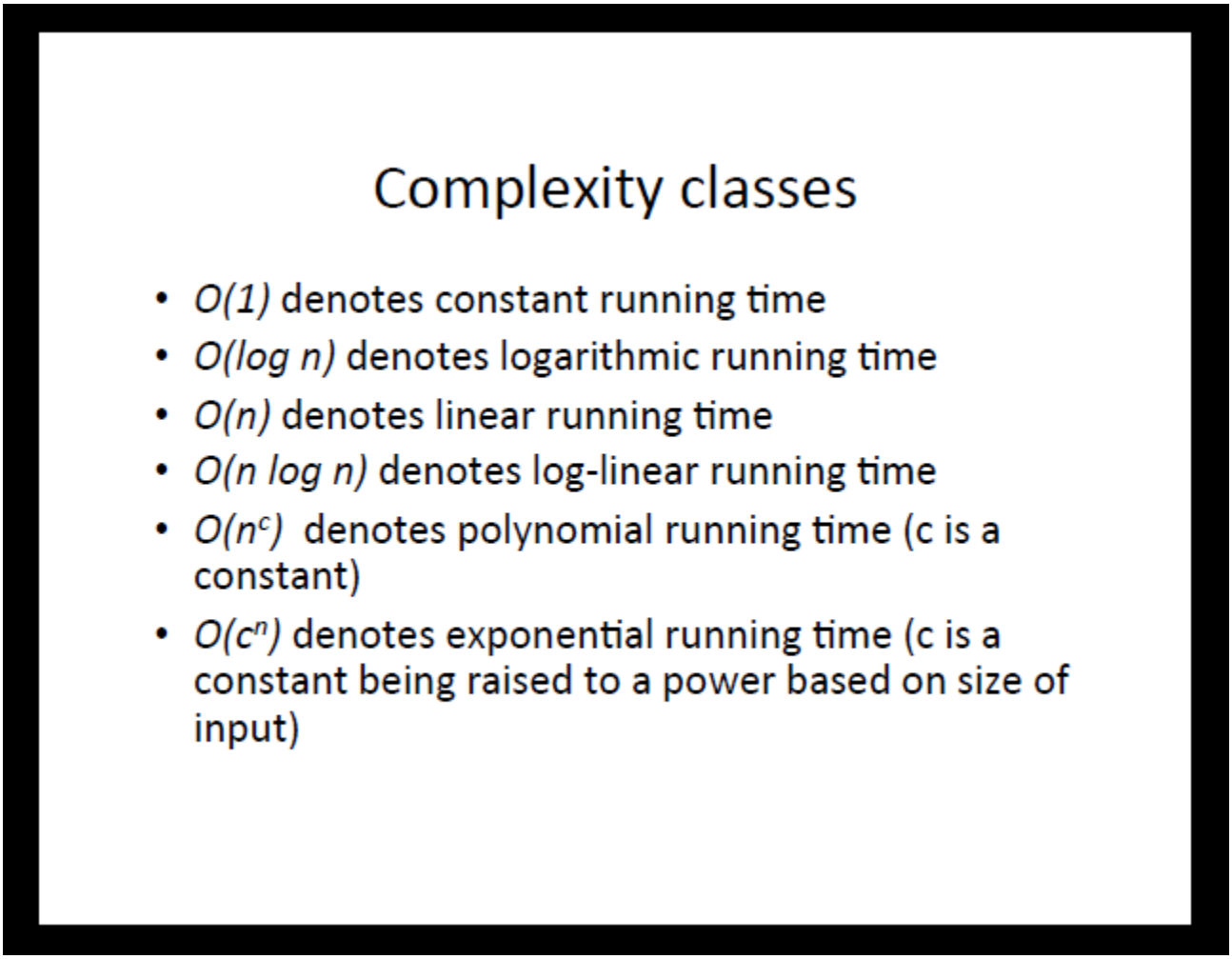
2. **Το υπολογιστικό μοντέλο**. Όπως αναφέρθηκε συχνά το υπολογιστικό μοντέλο είναι μια μηχανή Turing αν και υπάρχουν διάφορες κλάσεις που αναφέρονται σε διαφορετικά υπολογιστικά μοντέλα.

3. Οι **υπολογιστικοί πόροι**. Όπως αναφέρθηκε και στα προηγούμενα συνήθως αναφερόμαστε στον χώρο και στον χρόνο.

Παρακάτω παρουσιάζονται κάποιες κλάσεις πολυπλοκότητας για προβλήματα απόφασης (desicion problems):

Complexity class	Model of Computation	Time constraint
$\mathrm{DTIME}(f(n))$	Deterministic Turing machine	$f(n)$
$\mathrm{NTIME}(f(n))$	Non-deterministic Turing machine	$f(n)$
$\mathbf{P} = \mathrm{DTIME}(n^{O(1)})$	Deterministic Turing machine	n^k, polynomial time
$\mathbf{NP} = \mathrm{NTIME}(n^{O(1)})$	Non-deterministic Turing machine	n^k
$\mathbf{EXP} = \mathrm{DTIME}(2^{n^{O(1)}})$	Deterministic Turing machine	$2^{(n^k)}$, exponential time

Εικόνα 2.38. Σημαντικές κλάσεις πολυπλοκότητας προβλημάτων απόφασης με βάση τον χρόνο ως υπολογιστικό πόρο



Εικόνα 2.38. Κλάσεις πολυπλοκότητας

Η **κλάση P** αποτελείται από όλα τα προβλήματα απόφασης που επιλύονται από μια ντετερμινιστική μηχανή σε πολυωνυμικό χρόνο.

Η **κλάση NP** αποτελείται από όλα τα προβλήματα απόφασης που η λύση "εξακριβώνεται" σε πολυωνυμικό χρόνο με δεδομένο τις σωστές πληροφορίες, ή ισοδύναμα, των οποίων οι λύσεις μπορούν να βρεθούν σε πολυωνυμικό χρόνο σε μια μη ντετερμινιστική μηχανή.
Είναι προφανές πως $P \subset NP$ όμως παραμένει ανοιχτό το ερώτημα αν $NP \subset P$ δηλαδή αν $P = NP$

Το πρόβλημα **P** vs **NP** είναι ένα σημαντικό άλυτο πρόβλημα στην επιστήμη των υπολογιστών. Στην απλή διατύπωση του το ερώτημα που θέτει είναι, εάν κάθε πρόβλημα του οποίου η ύπαρξη λύσης μπορεί να επιβεβαιωθεί γρήγορα από έναν υπολογιστή μπορεί επίσης και να επιλυθεί γρήγορα από τον υπολογιστή.

Μια απάντηση στο ερώτημα **P** = **NP** θα καθόριζε αν προβλήματα που μπορούν να επιβεβαιωθούν σε πολυωνυμικό χρόνο, όπως το πρόβλημα αθροίσματος υποσυνόλου, μπορούν και να λυθούν σε πολυωνυμικό χρόνο. Αν αποδειχθεί ότι **P** ≠ **NP**, θα σημαίνει ότι υπάρχουν προβλήματα στην **NP** (όπως τα **NP-complete** προβλήματα) τα οποία είναι δυσκολότερο να υπολογιστούν από το να επιβεβαιωθούν. Δεν θα μπορούν δηλαδή να υπολογιστούν σε πολυωνυμικό χρόνο αλλά η απάντηση μπορεί να επιβεβαιωθεί σε πολυωνυμικό χρόνο.

Εκτός από το να είναι ένα σημαντικό πρόβλημα στην Θεωρία Υπολογισμού, μια απόδειξη του θα εχει σημαντικές επιρροές στα Μαθηματικά, την Κρυπτογραφία, την μελέτη Αλγορίθμων, την Τεχνητή Νοημοσύνη, την Θεωρία Παιγνίων, την Φιλοσοφία, τα Οικονομικά και πολλά άλλα πεδία.

Η Κρυπτογραφία, για παράδειγμα, βασίζεται σε ορισμένα προβλήματα που είναι δύσκολα υπολογιστικά. Το πόσο μια εποικοδομητική και αποτελεσματική λύση θα πρέπει να αποτελέσει απειλή για την κρυπτογραφία εξαρτάται από τις λεπτομέρειες. Μια λύση τάξης $O(N^2)$ ή καλύτερη και ένας εύλογος σταθερός όρος θα ήταν καταστροφική.

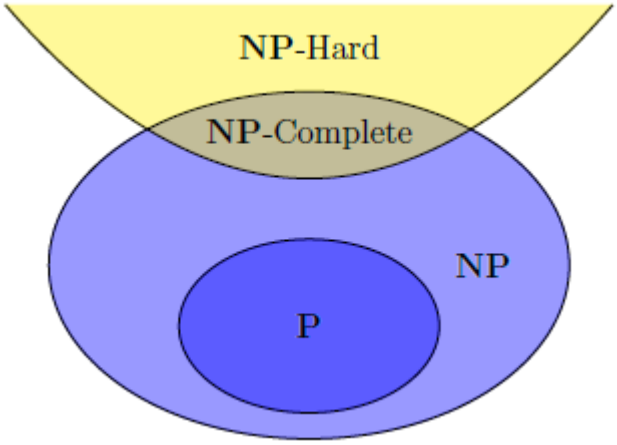
Ειδικά όταν αναφερόμαστε στην ασφάλεια κρυπτογραφικών συστημάτων πρέπει να αναφερθούμε και σε δυο ακόμα ειδικές κλάσεις πολυπλοκότητας τις **NP-hard** και **NP-complete**.

Ορισμός 2.39. Έστω A , B δύο προβλήματα απόφασης. Μια **αναγωγή** από το A στο B είναι μια πολυωνυμικού χρόνου υπολογιστική συνάρτηση $f:\Sigma^*\to\Sigma^*$ όπου Σ* είναι το σύνολο όλων των εισόδων τέτοιων ώστε $x\in A\Leftrightarrow f(x)\in B$. Αν το A ανάγεται στο B και το B επιλύεται σε πολυωνυμικό χρόνο τότε και το A επιλύεται σε πολυωνυμικό χρόνο.

Ένα πρόβλημα απόφασης A είναι κλάσης **NP-hard** αν και μόνο αν κάθε άλλο πρόβλημα B ανάγεται στο A. Εάν το A τυγχάνει να είναι και κλάσης **NP** τότε λέμε πως είναι κλάσης **NP-complete.**
Με βάση τα ανωτέρω είναι προφανές πως εάν ένα πρόβλημα A είναι **NP-hard** τότε δεν μπορεί αυτό να επιλυθεί σε πολυωνυμικό χρόνο εκτός εάν αποδειχθεί πως **P=NP**. Υπάρχουν προβλήματα που είναι NP-hard αλλά όχι NP-complete όπως είναι το **πρόβλημα τερματισμού** (halting problem).

Πιο συγκεκριμένα, δοθέντος προγράμματος και μιας εισόδου, το πρόβλημα έγκειται στον προσδιορισμό του τερματισμού ή μη τερματισμού του προγράμματος. Σε αυτό το αφηρημένο πλαίσιο δεν υπάρχει περιορισμός πόρων που αφορούν τη μνήμη ή το χρόνο εκτέλεσης του προγράμματος, δηλαδή μπορεί να είναι αυθαίρετα μακροσκελές και να καταλάβει αυθαίρετα μεγάλο αποθηκευτικό χώρο πριν σταματήσει. Το ερώτημα είναι αν το πρόγραμμα που μας δόθηκε θα σταματήσει ποτέ με μια συγκεκριμένη είσοδο.

Ο Turing απέδειξε ότι, αν υπήρχε αλγόριθμος να προσδιορίζει αν ένα ζευγάρι προγράμματος-εισόδου πέφτει σε ατέρμονα βρόχο ή όχι, τότε η δομή του αλγορίθμου θα έπρεπε να είναι τέτοια ώστε να έρχεται σε αντίφαση με τον εαυτό του. Έτσι αποδείχτηκε η μη ύπαρξη τέτοιου αλγορίθμου.



Εικόνα 2.38. Κλάσεις πολυπλοκότητας: P, NP, NP- hard, NP-complete

3. Κρυπτογραφία Δημοσίου Κλειδιού

Σε αυτή την ενότητα παρουσιάζονται σημαντικοί ορισμοί των κρυπτογραφικών συστημάτων δημοσίου κλειδιού. Ένας εναλλακτικός όρος της κρυπτογραφίας δημοσίου κλειδιού είναι η ασύμμετρη κρυπτογραφία όπως ήδη έχει αναφερθεί στο πρώτο κεφάλαιο. Σε αυτό το είδος κρυπτογράφησης η μια οντότητα κατέχει ένα μυστικό κλειδί ενώ οι άλλες οντότητες έχουν πρόσβαση σε ένα δημόσιο κλειδί που σχετίζεται με το μυστικό κλειδί της πρώτης. Σε αντίθεση με την συμμετρική κρυπτογραφία όπου υπάρχει μόνο ένα κλειδί το οποίο θα πρέπει να κρατείται με απόλυτη μυστικότητα ανάμεσα στις οντότητες που θέλουν να επικοινωνήσουν. Γενικά, η κρυπτογραφία δημοσίου κλειδιού περιλαμβάνει πιο πολύπλοκους αλγορίθμους ως προς την υπολογιστική πολυπλοκότητα και γι αυτό γενικά θεωρούνται λιγότερα αποδοτικά συστήματα ως προς τον χρόνο εκτέλεσης κι αυτός είναι ένας από τους λόγους που υπάρχουν ακόμα τα συμμετρικά συστήματα παρ'όλη την μικρότερη αντίσταση που παρουσιάζουν σε επιθέσεις.

Η ιδέα για ένα σχήμα δημοσίου κλειδιού προτάθηκε για πρώτη φορά από τους Diffie – Hellman [12] το 1976 στο πανεπιστήμιο του Stanford. Η βασική ιδέα προήλθε από την ανάγκη ασφαλούς ανταλλαγής μηνυμάτων μεταξύ οντοτήτων μέσα από ένα μη ασφαλή δίαυλο επικοινωνίας.

Ένας **αλγόριθμος πολυωνυμικού χρόνου** (Polynomial-Time PT) είναι ένας αλγόριθμος που τρέχει σε πολυωνυμικό χρόνο δηλαδή ο χρόνος των υπολογιστικών βημάτων που απαιτούνται για να ολοκληρωθεί ο αλγόριθμος για δεδομένη είσοδο είναι της τάξης $O(n^k)$ όπου n είναι το μέγεθος της εισόδου και $k \in \mathbb{N}$. Γενικά αν ένας αλγόριθμος τρέχει σε πολυωνυμικό χρόνο θεωρείται *αποδοτικός αλγόριθμος*.

3.1. Σύγχρονη Κρυπτογράφηση

Γενικά στην σύγχρονη κρυπτογραφία μια από τις βασικές απαιτήσεις είναι η απόδειξη της ασφάλειας των χρησιμοποιούμενων κρυπτογραφικών αλγορίθμων. Ένας από τους βασικότερους τρόπους είναι να το να βασίσουμε την κρυπτογραφία σε γνωστά δύσκολα μαθηματικά προβλήματα, επιδιώκοντας να φτιάξουμε κρυπτογραφικούς αλγόριθμους οι οποίοι να μπορούν να υπολογιστούν από τον καθένα αλλά είναι πρακτικά αδύνατη η αντιστροφή τους εκτός από συγκεκριμένες οντότητες.
Έτσι ορίζουμε τις **μονόδρομες συναρτήσεις (one-way function)** και θα αναφέρουμε μερικές υποψήφιες.

Ορισμός 3.1. Μια συνάρτηση $f : X \to Y$ είναι **αμελητέα** (negligible) εάν για κάθε πολυώνυμο $p(x) \in \mathbb{Z}[x]$, $\exists \varepsilon \in \mathbb{Z} \ : \ \forall n > \varepsilon \Rightarrow f(n) < \frac{1}{p(n)}$.

Ορισμός 3.2. Μονόδρομη συνάρτηση είναι μια συνάρτηση $f : X \to Y$ εάν ισχύουν τα εξής:

1. **(Είναι εύκολο να υπολογιστεί:)** Υπάρχει πολυωνυμικός αλγόριθμος A τέτοιος ώστε $A(x) = f(x) = y, \forall x \in X$
2. **(Είναι δύσκολο να αντιστραφεί:)** Για κάθε πολυωνυμικό αλγόριθμο B υπάρχει αμελητέα συνάρτηση $v_B(k)$ τέτοια ώστε για αρκετά μεγάλο k ισχύει: $P[B(f(x)) = x] \le v_B(k)$ (negligible probability)

Υπάρχει μια κατηγορία μονόδρομων συναρτήσεων που η αντιστροφή τους είναι επίσης εύκολη αρκεί να γνωρίζει κάποιος την "πληροφορία" που απαιτείται για να γίνει εφικτή η αντιστροφή. Τότε λέμε πως η μονόδρομη συνάρτηση περιέχει μια *"κερκόπορτα"* ή μια καταπακτή όπως συχνά αποκαλείται και έτσι η αντίστοιχη μονόδρομη συνάρτηση καλείται **συνάρτηση καταπακτής (trapdoor function)**.

Ωστόσο παραμένει άγνωστο αν υπάρχουν τελικά μονόδρομες συναρτήσεις μιας και δεν υπάρχει καμία απόδειξη για την ύπαρξη τέτοιων συναρτήσεων. Η απάντηση σε αυτό το ερώτημα οδηγεί κατευθείαν στο πρόβλημα $P = NP$ και μια πιθανή απόδειξη περί ύπαρξης τέτοιων συναρτήσεων θα σημάνει ότι τελικά $P \neq NP$.

Ωστόσο υπάρχουν κάποιες πολύ “καλές” υποψήφιες μονόδρομες συναρτήσεις που πηγάζουν από μερικά δύσκολα προβλήματα των Μαθηματικών. Ο αναγνώστης μπορεί να δει [22], [24] για περισσότερες υποψήφιες μονόδρομες συναρτήσεις.

Μια λίστα από υποψήφιες μονόδρομες συναρτήσεις

Όπως ήδη αναφέρθηκε είναι ζωτικής σημασίας για ένα ασφαλές σύστημα δημοσίου κλειδιού να αποτελείται από συναρτήσεις οι οποίες είναι εύκολο να υπολογιστούν αλλά δύσκολο να αντιστραφούν. “Εύκολο” σημαίνει πως είναι δυνατός ο υπολογισμός από κάποιον πιθανοθεωρητικό αλγόριθμο πολυωνυμικού χρόνου τον οποίο συμβολίζουμε με **PPT** (probabilistic polynomial time algorithm). “Δύσκολο” σημαίνει πως η πιθανότητα της αντιστροφής από κάποιον PPT αλγόριθμο είναι αμελητέα. Οι ακόλουθες συναρτήσεις κατέχουν αυτές τις επιθυμητές ιδιότητες και συνεπώς αποτελούν καλές υποψήφιες μονόδρομες συναρτήσεις.

1. **Παραγοντοποίηση σύνθετου αριθμού σε γινόμενο ακεραίων.** Η συνάρτηση $f:(x,y)\rightarrow xy, x,y\in\mathbb{Z}$ θεωρείται μονόδρομη.

2. **Το πρόβλημα του διακριτού λογαρίθμου.** Έστω p πρώτος αριθμός. Τότε $\mathbb{Z}_p^*$ είναι κυκλική ομάδα που σημαίνει πως $\mathbb{Z}_P^*=\{g^i \bmod p : 1\le i\le p-1\}$ για κάποιον γεννήτορα $g\in\mathbb{Z}_p^*$.
   Η συνάρτηση $f:(p,g,x)\rightarrow(g^x \bmod p)$ θεωρείται μονόδρομη συνάρτηση.

3. **Πρόβλημα της παραγοντοποίησης μεγάλων πρώτων αριθμών.** Έστω $n=pq$, p,q *πρώτοι* . Η παραγοντοποίηση ενός τέτοιου n είναι δύσκολο να υπολογιστεί.

3.2 Ορισμός ενός συστήματος δημοσίου κλειδιού

Ο χρόνος εκτέλεσης των αλγόριθμων κρυπτογράφησης και αποκρυπτογράφησης υπολογίζεται ως συνάρτηση μιας παραμέτρου ασφαλείας k. Όταν αναφερόμαστε πως οι κρυπτογραφικοί αλγόριθμοι του συστήματος τρέχουν σε πολυωνυμικό χρόνο θα εννοούμε ότι ο χρόνος εκτέλεσης φράσσεται από μια πολυωνυμική συνάρτηση της παραμέτρου k.

Ορισμός 3.2. Ένα σχήμα Ε κρυπτογράφησης δημοσίου κλειδιού είναι μια πλειάδα (KeyGen, Enc, Dec) από πιθανοθεωρητικούς αλγορίθμους PPT.

(1) **Ο αλγόριθμος παραγωγής των κλειδιών κρυπτογράφησης** (KeyGen) δέχεται σαν είσοδο την παράμετρο ασφαλείας k και παράγει ένα ζεύγος κλειδιών (pk, sk). Το pk είναι το **δημόσιο κλειδί** ενώ το sk είναι το **ιδιωτικό** κλειδί. Τα pk και sk έχουν μήκος το λιγότερο το μήκος του k.

(2) **Ο αλγόριθμος κρυπτογράφησης** (Enc) δέχεται σαν είσοδο το δημόσιο κλειδί pk και ένα κείμενο m που καλείται μήνυμα το οποίο αντλείται από ένα σύνολο M, το χώρο μηνυμάτων. Παράγει ένα κρυπτογραφημένο μήνυμα c από τον χώρο των κρυπτοκειμένων C. Συμβολίζουμε $Enc_{pk}(m)\rightarrow c$

(3) **Ο αλγόριθμος αποκρυπτογράφησης** (Dec) δέχεται σαν είσοδο το ιδιωτικό κλειδί sk και ένα κρυπτοκείμενο c . Παράγει το αρχικό μήνυμα. Χωρίς βλάβη της γενικότητας υποθέτουμε ότι ο αλγόριθμος Dec είναι ντετερμινιστικός και γράφουμε $Dec_{sk}(c)\rightarrow m$.

Ορισμός 3.3. Ένα σχήμα είναι **ασφαλές** εάν κάθε PPT αλγόριθμος αντιπάλου έχει αμελητέα πιθανότητα να παραβιάσει το σύστημα.

Η ασφάλεια ενός κρυπτοσυστήματος συμμετρικού ή ασύμμετρου αντανακλά την ανικανότητα ενός αντιπάλου να αντλήσει οποιαδήποτε μη τετριμένη πληροφορία για τα αρχικά μηνύματα. Στην βιβλιογραφία υπάρχουν διάφορες επιθυμητές

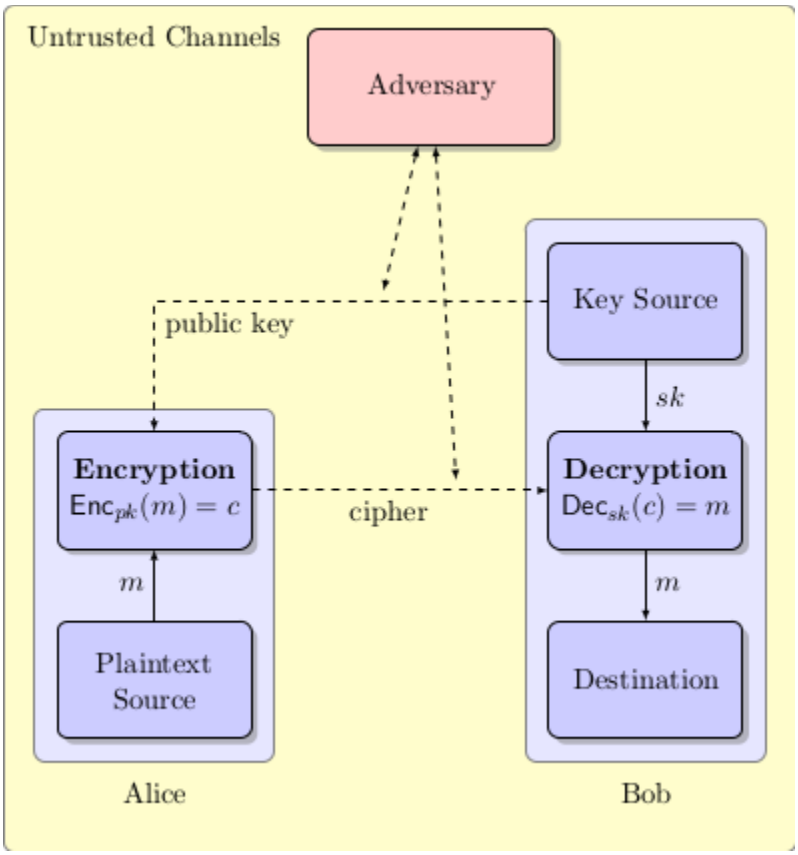
ιδιότητες που οφείλει να έχει ένα κρυπτοσύστημα προκειμένου να θεωρείται ασφαλές [13]. Στην παρούσα διπλωματική αναφέρουμε τις 3 πιο συνηθισμένες.

1. Το ιδιωτικό κλειδί δεν πρέπει να μπορεί να εξαχθεί από το δημόσιο κλειδί.

2. Η γνώση των κρυπτοκειμένων ή μέρους αυτών δεν οδηγεί σε καμία σημαντική πληροφορία για τα αρχικά μηνύματα

3. Θα πρέπει να είναι δύσκολο να εξαχθεί κάποια πληροφορία σχετικά με την κίνηση των αρχικών μηνυμάτων όπως πχ η πληροφορία ότι ένα μήνυμα στάλθηκε δυο φορές.

Με βάση τα ανωτέρω μια επικοινωνία μεταξύ δυο οντοτήτων του Bob και της Alice σχηματοποιείται ως εξής:



Εικόνα 3.2. : Διάγραμμα επικοινωνίας οντοτήτων με χρήση τεχνικών δημοσίου κλειδιού

4. Ομομορφική Κρυπτογράφηση

Οι απαιτήσεις για ασφάλεια των δεδομένων έχουν αυξηθεί κατά πολύ τα τελευταία χρόνια. Εξαιτίας της ραγδαίας εξέλιξης της τεχνολογίας, η συχνότητα των επιθέσεων σε ψηφιακά αγαθά και σε τεχνολογικές συσκευές έχει αυξηθεί σημαντικά. Για την ασφαλή αποθήκευση και ανταλλαγή των δεδομένων υπάρχουν πολλές λύσεις όπως η κρυπτογράφηση τους. Τα πράγματα περιπλέκονται όμως αν προστεθεί η απαίτηση για υπολογισμούς που εφαρμόζονται στα κρυπτογραφημένα δεδομένα μας χωρίς όμως να απαιτείται η αποκρυπτογράφηση τους πρώτα. Σε αυτό το θεωρητικό πλαίσιο αναπτύχθηκε ένας πολύ ειδικός κλάδος της κρυπτογραφίας, η **ομομορφική κρυπτογραφία** και τα αντίστοιχα συστήματα που την υλοποιούν ονομάζονται **ομομορφικά κρυπτοσυστήματα**.

Η ιδέα για την κατασκευή πλήρως ομομορφικών συστημάτων εμπνεύστηκε για πρώτη φορά από τους Rivest, Adleman και Dertouzos [15] λίγο μετά την ανακάλυψη του RSA [16]. Τότε τέθηκε για πρώτη φορά το ερώτημα αν υπάρχει κρυπτοσύστημα που επιτρέπει την εκτέλεση πράξεων πάνω στα κρυπτογραφημένα δεδομένα που παράγει, χωρίς να είναι αναγκαία η αποκρυπτογράφηση τους πρώτα. Τα σχήματα όπως RSA, Paillier, ElGamal κ.α έχουν ομομορφικές ιδιότητες αλλά μόνο για συγκεκριμένες πράξεις γι αυτό και ονομάζονται μερικώς ομομορφικά σχήματα.

4.1 Ορισμός ενός Ομομορφικού Σχήματος

Στην προηγούμενη ενότητα δόθηκε ο ορισμός ενός συστήματος δημοσίου κλειδιού. Για τα ομομορφικά κρυπτοσυστήματα θα πρέπει να ενισχύσουμε το ήδη υπάρχων σχήμα με έναν επιπλέον αλγόριθμο [17].

Ορισμός 4.1. Ένα κρυπτοσύστημα δημοσίου κλειδιού $E=(KeyGen, Enc, Dec)$ είναι ομομορφικό εάν για κάθε k και όλα τα ζεύγη (pk, sk) εξόδου από τον KeyGen(k) είναι δυνατόν να οριστούν ομάδες C, M αντίστοιχα ώστε να ισχύει:

- Ο χώρος μηνυμάτων M και το σύνολο των κρυπτοκειμένων που παράγονται από τον Enc_{pk} είναι στοιχεία του C.
- Για κάθε $m_1, m_2 \in M$ και $c_1, c_2 \in C$ με $m_1 = Dec_{sk}(c_1)$ και $m_2 = Dec_{sk}(c_2)$ έχουμε:

$Dec_{sk}(c_1 * c_2) = m_1 * m_2$ όπου * είναι η πράξη των ομάδων C, M. Να σημειωθεί ότι μπορεί η πράξη της ομάδας M να είναι διαφορετική από την πράξη της ομάδας C.

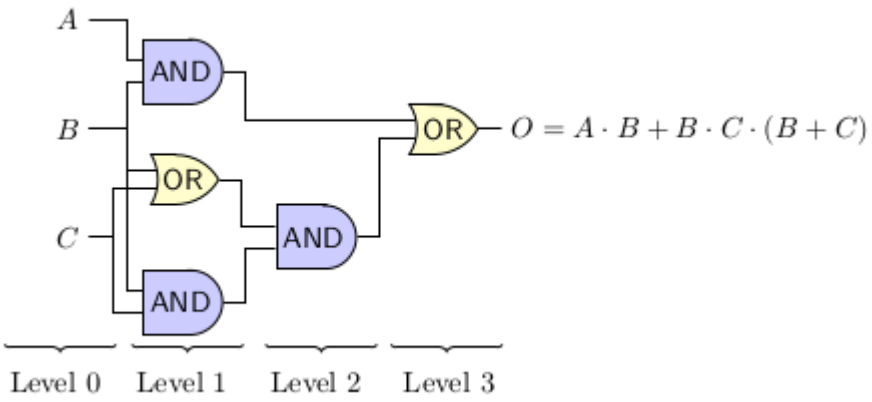
Ένα ομομορφικό σύστημα είναι ένα PKS που μεταφέρει την αλγεβρική δομή του πεδίου ορισμού στην αλγεβρική δομή του πεδίου τιμών. Είναι εφοδιασμένο με έναν **αποδοτικό αλγόριθμο (Eval)** που μπορεί να υπολογίσει το άθροισμα ή το γινόμενο δύο μηνυμάτων των οποίων το δημόσιο κλειδί είναι γνωστό όπως και τα κρυπτογραφήματα τους χωρίς να είναι γνωστά όμως τα ίδια τα μηνύματα.

Επιπλέον ένα πλήρως ομομορφικό σύστημα μπορεί να αποτιμήσει την τιμή μιας συνάρτησης μηνυμάτων $f(m_1, \ldots, m_t)$ όπου f είναι μια οποιαδήποτε επιθυμητή συνάρτηση εφόσον όμως είναι αποδοτική. Ωστόσο καμία απολύτως χρήσιμη πληροφορία σχετικά με τα μηνύματα $m_1, \ldots, m_t$ ή για την $f(m_1, \ldots, m_t)$ ή για ενδιάμεσες τιμές της f μπορεί να εξαχθεί. Οι είσοδοι, οι έξοδοι ή οι ενδιάμεσες τιμές παραμένουν πάντα κρυπτογραφημένες. Πριν προχωρήσουμε στην κατασκευή τέτοιων σχημάτων θα ορίσουμε μια σημαντική έννοια.

Κυκλώματα

Γενικά **κύκλωμα** (circuit) καλείται ένα κατευθυνόμενο άκυκλο γράφημα όπου οι κόμβοι του καλούνται **πύλες (gates)** και οι ακμές του **συνδέσεις (wires)** . Ανάλογα την φύση του κυκλώματος οι τιμές εισόδου του κυκλώματος μπορεί να είναι ακέραιοι, boolean τιμές κ.α ενώ οι αντίστοιχες πύλες μπορεί να είναι συνολοθεωρητικές ή αριθμητικές πράξεις ή λογικές πύλες (AND, OR, NOR, XOR, NAND,...). Με την βοήθεια κυκλωμάτων αναπαραστήσουμε τις συναρτήσεις αποτίμησης.

Παράδειγμα. Έστω η συνάρτηση f που αναπαριστά την έκφραση A·B + B·C·(B+C) με εισόδους (A, B, C). Τότε η f μπορεί να αναπαρασταθεί από το παρακάτω κύκλωμα με τις λογικές πύλες AND και OR.

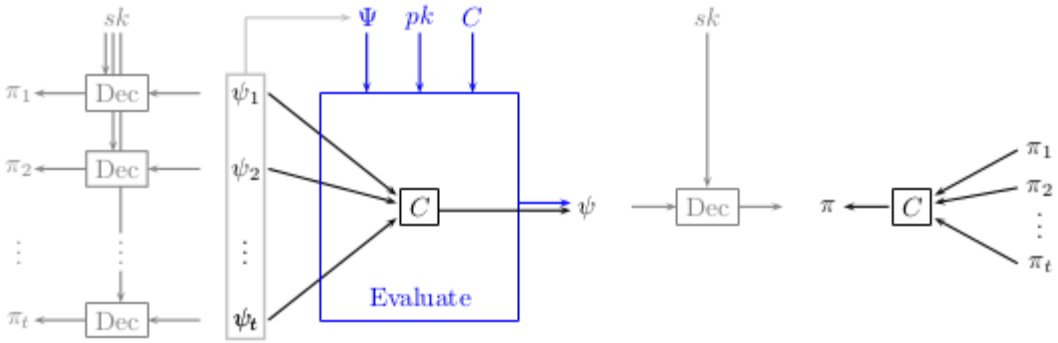


Εικόνα 4.1: Παράδειγμα αναπαράστασης ενός κυκλώματος

Δυο πολύ σημαντικά μέτρα της πολυπλοκότητας ενός κυκλώματος είναι το **μέγεθος** (size) και το **βάθος** (depth) .

Ορισμός 4.2. Μέγεθος ενός κυκλώματος ορίζεται ο αριθμός των πυλών που δεν αποτελούν είσοδοι του κυκλώματος. **Βάθος** ενός κυκλώματος καλείται το μήκος του μεγαλύτερου μονοπατιού μεταξύ μιας πύλης εισόδου και μιας πύλης εξόδου.

Ορισμός 4.3. (Πλήρης Ομομορφική Κρυπτογράφηση). Ένα κρυπτοσύστημα δημοσίου κλειδιού (KeyGen, Enc, Dec, Eval) είναι **πλήρως ομομορφικό** εάν υπάρχει αποδοτικός (efficient) αλγόριθμος αποτίμησης (**Eval)** τέτοιος ώστε, για ένα έγκυρο δημόσιο κλειδί pk, ένα επιτρεπτό κύκλωμα C και ένα σύνολο κρυπτοκειμένων $\Psi = \{c_1, c_2, \ldots, c_n\}$ όπου $Enc_{pk}(m_i) \rightarrow c_i$, παράγει $Eval_{pk}(C, \Psi) \rightarrow c$.



Εικόνα 4.3.: Σχηματική Παρουσίαση ενός Evaluation-αλγορίθμου

Παρακάτω θα παρουσιάσουμε έναν διαφορετικό τρόπο κατασκευής πλήρως ομομορφικών σχημάτων και για να γίνει κατανοητό χρειαζόμαστε κάποιους σημαντικούς ορισμούς.

Ορισμός 4.4. Ένα ομομορφικό σχήμα Ε λέμε πως είναι **ορθό** (correct) για μια οικογένεια C_E κυκλωμάτων εάν για κάθε ζευγάρι (sk, pk) που παράγονται από τον αλγόριθμο $KeyGen_E(\lambda)$, για κάθε $m_1, \ldots, m_t$ και για κάθε $\Psi = \langle c_1, \ldots, c_t \rangle$ με $Enc_{pk}(m_i) \rightarrow c_i$ ισχύει ότι:

Εάν $Eval_E(pk, C, \Psi) \rightarrow c$ τότε $Dec_E(sk, c) \rightarrow C(m_{1,}, \ldots, m_t)$

Ορισμός 4.5. Ένα ομομορφικό σχήμα Ε λέμε πως είναι **συμπαγές (compact)** , εάν υπάρχει πολυώνυμο f τέτοιο ώστε για κάθε τιμή παραμέτρου ασφάλειας λ, ο αλγόριθμος αποκρυπτογράφησης Dec_E μπορεί να αναπαρασταθεί από ένα κύκλωμα D_E μεγέθους το πολύ f(λ).
Ένα ομομορφικό σχήμα Ε λέμε ότι έχει την ιδιότητα της **συμπαγούς αποτίμησης (compactly evaluation)** για την οικογένεια C_E εάν είναι **συμπαγές και ορθό** για κάθε κύκλωμα που ανήκει στην οικογένεια C_E .

Πόρισμα 4.6. Ένα ομομορφικό σχήμα Ε είναι πλήρως ομομορφικό εάν έχει την ιδότητα της συμπαγούς αποτίμησης.

Ορισμός 4.7. Μια οικογένεια ομομορφικών κρυπτογραφικών σχημάτων $\{E^{(d)} : d \in \mathbb{N}\}$ λέμε πως είναι **οριακά πλήρως ομομορφική (leveled fully homomorphic)** εάν:

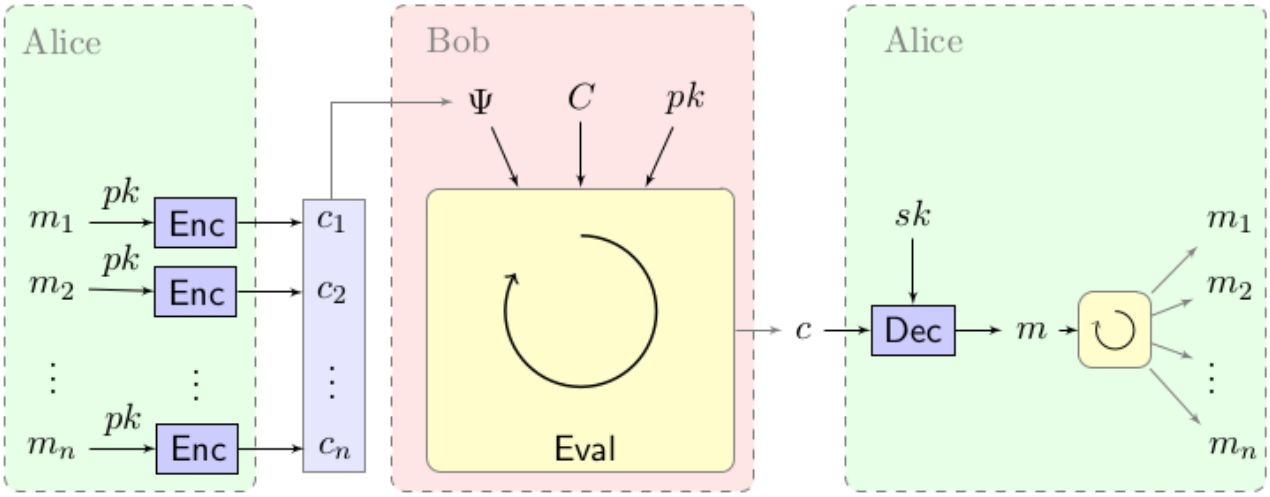
- $\forall d \in \mathbb{N}$, όλα τα σχήματα έχουν το ίδιο κύκλωμα (C) αποκρυπτογράφησης.
- Έχει την ιδιότητα της συμπαγούς αποτίμησης για όλα τα κυκλώματα βάθους το πολύ d.
- Και η αλγοριθμική πολυπλοκότητα των κρυπτογραφικών αλγορίθμων της οικογένειας $E^{(d)}$ είναι πολυώνυμο των λ, d και του μεγέθους του κυκλώματος C.

Ένα κρυπτοσύστημα που υποστηρίζει ταυτόχρονα και την πράξη της πρόσθεσης και του πολλαπλασιασμού (δηλαδή ένα πλήρως ομομορφικό) διατηρεί την δομή δακτυλίου που ορίζει τον χώρο μηνυμάτων (plaintext space). Χρησιμοποιώντας ένα τέτοιο σχήμα όπως προαναφέρθηκε επιτρέπει σε μια μη εξουσιοδοτημένη οντότητα να εκτελέσει πράξεις με τα κρυπτογραφημένα δεδομένα χωρίς πρώτα να χρειαστεί να τα αποκρυπτογραφήσει διατηρώντας έτσι την ιδιωτικότητα των δεδομένων.

Μια ευρεία εφαρμογή ενός τέτοιου συστήματος είναι η υπολογιστική νέφους (cloud computing). Καθώς ο όγκος των δεδομένων που αποθηκεύουμε διαρκώς αυξάνει έχει σαν αποτέλεσμα να απαιτούνται ολοένα και περισσότεροι πόροι για την επεξεργασία τους. Έτσι είναι χρήσιμο να υπάρχει ένας μηχανισμός που επιτρέπει να γίνει αυτή η επεξεργασία από κάποιον τρίτο χωρίς να παραβιάσουμε την ιδιωτικότητα τους.

Για παράδειγμα ας υποθέσουμε ότι η Alice επιθυμεί να αποθηκεύσει ένα ευαίσθητο αρχείο $m \in \{0,1\}^n$ στον cloud server του Bob. Οπότε στέλνει στον Bob $Enc(m_1), \ldots, Enc(m_n)$.

Ας υποθέσουμε ότι το αρχείο είναι μια βάση δεδομένων που περιέχει μια λίστα ατόμων με προσωπικές πληροφορίες και η Alice επιθυμεί να αναζητήσει άτομα που είναι 22 ετών. Αντί να κάνει λήψη του αρχείου από τον Bob, να το αποκρυπτογραφήσει και έπειτα να κάνει την αναζήτηση της ζητούμενης πληροφορίας, μπορεί να ζητήσει από τον Bob να ψάξει για εκείνη χωρίς ο Bob να γνωρίζει τι είναι ακριβώς αυτό το οποίο ψάχνει ούτε καν το περιεχόμενο του συνολικού αρχείου μιας και έχει κρυπτογραφηθεί από την Alice. Η απάντηση που θα λάβει η Alice από τον Bob θα είναι σε κρυπτογραφημένη μορφή και με το ιδωτικό κλειδί που μόνο εκείνη γνωρίζει μπορεί να το αποκρυπτογραφήσει και να μάθει τον αριθμό των ατόμων με ηλικία 22 έτη.



Εικόνα 4.3.: Διάγραμμα ενός ομομορφικού σχήματος κρυπτογράφησης

4.2 Μερικώς Ομομορφικά Σχήματα (Partial Homomorphic Encryption)

Χρειάστηκαν περίπου 30 χρόνια έρευνας για να φτάσουμε σε ένα ασφαλές συμπέρασμα σχετικά με την ύπαρξη ενός ομομορφικού σχήματος. Ωστόσο στο ενδιάμεσο αναπτύχθηκαν πολλά μερικώς ομομορφικά σχήματα. Με τον όρο μερικώς ομομορφικά σχήματα εννοούμε κρυπτοσυστήματα που δεν είναι πλήρως ομομορφικά δηλαδή δεν έχουν την ομομορφική ιδιότητα για κάθε κύκλωμα αλλά για τουλάχιστον για ένα. Σε αυτή την ενότητα θα παρουσιάσουμε κάποια τέτοια συστήματα τα οποία είμαι ομομορφικά ως προς την πρόσθεση ή τον πολλαπλασιασμό. Πρώτα θα περιγράψουμε το κρυπτοσύστημα του **RSA** το οποίο είναι πολλαπλασιαστικά ομομορφικό αλλά σε αυτή την έκδοση δεν παρέχει σημασιολογική ασφάλεια (semantic security). Έπειτα θα δείξουμε το κρυπτοσύστημα **Goldwasser–Micali** που είναι το πρώτο ομομορφικό σχήμα που παρέχει συγχρόνως σημασιολογική ασφάλεια. Στην συνέχεια θα παρουσιάσουμε το σχήμα **ElGammal** το οποίο είναι επίσης σημασιολογικά ασφαλές και πολλαπλασιαστικά ομομορφικό. Τέλος θα δείξουμε το σχήμα **Paillier** που είναι προσθετικά ομομορφικό.

Σε αυτό το σημείο πρέπει να αναφερθεί πως ένα κρυπτογραφικό σύστημα θα λέμε ότι παρέχει **σημασιολογική ασφάλεια,** αν για οποιοδήποτε πιθανοτικό αλγόριθμο πολυωνυμικού χρόνου, η πιθανότητα να εξάγει οποιαδήποτε πληροφορία για το αρχικό μήνυμα είναι αμελητέα, με δεδομένο ότι είναι γνωστό το κρυπτοκείμενο του μηνύματος ή το μήκος του.

4.2.1. RSA

Το 1978 οι Rivest, Shamir, Adleman σχεδίασαν το πολύ γνωστό κρυπτοσύστημα RSA. Αρχικά θα ορίσουμε τις συναρτήσεις $KeygGen_{RSA}$, $Encrypt_{RSA}$, $Decrypt_{RSA}$.

Σχήμα (Unpadded RSA).

$KeygGen_{RSA}$

1. Επιλέγουμε δύο τυχαίους και μεγάλους πρώτους p ,q.
2. Υπολογίζουμε $n = p \bullet q$.
3. Υπολογίζουμε την τιμή της συνάρτησης φ του Euler: $\varphi(p \bullet q) = (p-1)(q-1)$.

4. Επιλέγουμε έναν ακέραιο e τέτοιο ώστε $1<e<\varphi(p\bullet q)$ και $gcd(e,\varphi(n))=1$ δηλαδή $e^{\varphi(n)}\equiv 1 \bmod n$.
5. Θέτουμε $d\equiv e^{-1} mod(p-1)(q-1)$.
6. Επιστρέφουμε το δημόσιο κλειδί $pk=\{n,e\}$ και το ιδιωτικό κλειδί $sk=\{d\}$.

$Encrypt_{RSA}$

1. Δίνουμε ως είσοδο ένα μήνυμα $m\in M$.
2. Επιστρέφουμε $c=m^e \bmod n$.

$Decrypt_{RSA}$

1. Δίνουμε ως είσοδο ένα κρυπτοκείμενο $c\in C$.
2. Επιστρέφουμε το αρχικό κείμενο $m=c^d \bmod n$.

Έστω λοιπόν $pk=\{n,e\}$, $sk=\{d\}$ το δημόσιο και ιδιωτικό κλειδί αντίστοιχα. Αν $m_{1,}m_2\in M$ έχουμε $c_1=Encrypt_{RSA}(pk,m_1)=m_1^e \bmod n$ και $c_2=Encrypt_{RSA}(pk,m_2)=m_2^e \bmod n$.
Τότε $c_1\bullet c_2 \bmod n=m_1^e\bullet m_2^e \bmod n=(m_1\bullet m_2)^e \bmod n$. Έτσι αποδείξαμε πως:
$Encrypt_{RSA}(pk,m_1)\bullet Encrypt_{RSA}(pk,m_2)=Encrypt_{RSA}(pk,m_1\bullet m_2)$.

Αυτό σημαίνει πως ο RSA είναι ένα πολλαπλασιαστικό ομομορφικό σχήμα. Αξίζει να αναφερθεί πως ο RSA είναι το πρώτο μερικώς ομομορφικό σχήμα έτσι μπορεί να θεωρηθεί πως η ανάπτυξη της ομομορφικής κρυπτογραφίας οφείλεται σε αυτό το κρυπτοσύστημα.

4.2.2. Goldwasser–Micali

Το κρυπτοσύστημα Goldwasser–Micali είναι το πρώτο που είναι σημασιολογικά ασφαλές και αναπτύχθηκε από τους Shafi Goldwasser και Silvio Micali το 1982. Το σχήμα αυτό είναι το πρώτο που χρησιμοποίησε έναν πιθανοτικό αλγόριθμο κρυπτογράφησης. Η ασφάλεια του βασίζεται στο γεγονός ότι μόνο δεδομένης της παραγοντοποίησης ενός αριθμού σε γινόμενο πρώτων μπορει να καταλάβει αν κάποιος αριθμός είναι τετραγωνικό υπόλοιπο.

Ορίζεται το σύμβολο του Legendre ως: $\left(\frac{x}{p}\right)$ ώστε:

$$\left(\frac{x}{p}\right)=\begin{Bmatrix} 0, x\equiv 0 \bmod p \\ 1, x\neq 0 \bmod p \text{ και } x\equiv y^2 \bmod p,\ y\in\mathbb{Z} \\ -1, x\neq 0 \bmod p \text{ και δεν υπάρχει τέτοιος } y \end{Bmatrix}$$

Σχήμα (Goldwasser–Micali)

$KeyGen_{G-M}$

1. Επιλέγουμε δύο τυχαίους και μεγάλους πρώτους p ,q.
2. Υπολογίζουμε $N=p\bullet q$.
3. Βρίσκουμε $x\in\mathbb{Z}$ τέτοιο ώστε $\left(\frac{x}{p}\right)=\left(\frac{x}{q}\right)=-1$.
4. Επιστρέφουμε το δημόσιο κλειδί $pk=\{N,x\}$ και το ιδιωτικό κλειδί $sk=\{p,q\}$.

$Encryption_{G-M}$

1. Επιλέγουμε ένα τυχαίο $y \in \{0,1,\ldots,N-1\}$.
2. Έστω $m=(m_1,\ldots,m_n)$.Υπολογίζουμε $c_i = y^2 x^{m_i} mod N, i=1,\ldots,n$
3. Επιστρέφουμε $\{c_1,\ldots,c_n\}$.

$Decryption_{G-M}$

1. Για κάθε c_i ελέγχουμε εάν είναι τετραγωνικό υπόλοιπο δηλαδή εάν υπάρχει x ώστε $x^2 \equiv c_i mod N$.
2. Εάν c_i είναι τετραγωνικό υπόλοιπο τότε $m_i = 0$ αλλιώς $m_i = 1$.

Για τον έλεγχο αν ο c_i είναι τετραγωνικό υπόλοιπο mod N ακολουθείται η εξής διαδικασία:

1. Υπολόγισε $c_{ip} = c_i mod\, p$, $c_{iq} = c_i mod\, q$.
2. Εάν $c_{ip}^{(\frac{p-1}{2})} \equiv 1 mod\, p$ και $c_{iq}^{(\frac{q-1}{2})} \equiv 1 mod\, q$ τότε c_i είναι τετραγωνικό υπόλοιπο.

Το σχήμα Goldwasser–Micali είναι ομομορφικό ως προς την πράξη $\oplus$. Αυτό αποδεικνύεται ως εξής:
Έστω b_1, b_2 τυχαία bits και c_1, c_2 τα αντίστοιχα κρυπτογραφήματα τους. Τότε έχουμε:

$Encrypt_{G-M}(b_1) \bullet Encrypt_{G-M}(b_2) = y_1^2 x^{b_1} y_2^2 x^{b_2} mod N = (y_1 y_2)^2 x^{b_1+b_2} mod\, N$ =
$Encrypt_{G-M}(b_1) \oplus Encrypt_{G-M}(b_2)$.

4.2.3. ElGammal

Έπειτα από επτά χρόνια από την δημοσίευση του RSA ένα νέο μερικώς ομομορφικό σχήμα εμφανίστηκε στο πεδίο της έρευνας. Ο Taher ElGammal κατασκεύασε ένα σχήμα εμπνευσμένο από το πρωτόκολλο ανταλλαγής κλειδιών Diffie-Hellmann. Το σχήμα ElGammal είναι ένα μη-ντετερμινιστικό σχήμα κρυπτογράφησης όπου σε κάθε μήνυμα αντιστοιχούν πολλά διαφορετικά κρυπτογραφήματα και αυτός είναι ο λόγος που το σχήμα είναι σημασιολογικά ασφαλές (semantically secure).

Σχήμα (ElGammal)

$KeyGen_{ElG}$

1. Παράγουμε μια πολλαπλασιαστική κυκλική ομάδα G τάξης p και γεννήτορα g, όπου p κάποιος πολύ μεγάλος πρώτος και $g \in \mathbb{Z}$.
2. Επιλέγουμε τυχαίο $w \in \{0,1,\ldots,p-1\}$.
3. Υπολογίζουμε $h = g^w$.
4. Επιστρέφουμε το δημόσιο κλειδί $pk = \{G, p, g, h\}$ και το ιδιωτικό κλειδί $sk = \{w\}$.

$Encrypt_{ElG}$

1. Επιλέγουμε τυχαίο $\alpha \in \{0,1,\ldots,p-1\}$ και ένα μήνυμα m. Υπολογίζουμε $c_1 = g^y$.
2. Υπολογίζουμε $c_1 = g^y, s = h^y, c_2 = ms$.
3. Επιστρέφουμε το κρυπτοκείμενο $c = \{c_1, c_2\}$.

$Decrypt_{ElG}$

1. Υπολογίζουμε $s = c_1^w$.
2. Επιστρέφουμε $m = c_2 s^{-1}$.

Έστω δύο μηνύματα $m_{1,} m_2$ και τα αντίστοιχα κρυπτογραφήματα τους $c_1 = \{g^{y_1}, m_1 h^{y_1}\}$ και $c_2 = \{g^{y_2}, m_2 h^{y_2}\}$. Έχουμε $c_1 c_2 = \{g^{y_1+y_2}, m_1 m_2 h^{y_1+y_2}\}$. Για την αποκρυπτογράφηση έχουμε: $s = g^{(y_1+y_2)w} = h^{y_1} h^{y_2}$. Επομένως $m = m_1 m_2 h^{y_1+y_2} (h^{y_1} h^{y_2})^{-1} = m_1 m_2$.

Οπότε $Encrypt_{ElG}(pk_{1,} m_1) Encrypt_{ElG}(pk_{2,} m_2) = Encrypt_{ElG}(pk_1 + pk_{2,} m_1 m_2)$.

4.2.4. Paillier

Ο Paillier το 1999, παρουσίασε ένα νέο αλγόριθμο δημοσίου κλειδιού, ο οποίος έχει πολλές σημαντικές αλγεβρικές ιδιότητες και αποτελεί ουσιαστικά επέκταση του αλγορίθμου Okamoto – Uchiyama.

$KeyGen_{Pail}$

1. Επιλέγουμε δύο τυχαίους μεγάλους πρώτους p, q τέτοιους ώστε $gcd(pq, (p-1)(q-1)) = 1$.
   Η συνθήκη αυτή ικανοποιείται εάν και οι δύο πρώτοι έχουν ίσο μήκος.
2. Υπολογίζουμε $n = pq$ και $\lambda = lcm(p-1, q-1)$.
3. Επιλέγουμε τυχαίο $g \in \mathbb{Z}^*_{n^2}$ ώστε η τάξη του να διαιρείται από το n.
4. Υπολογίζουμε $\mu = (L(g^\lambda \bmod n^2))^{-1} \bmod n$, όπου $L(u) = \frac{u-1}{n}$. Να σημειωθεί ότι ο συμβολισμός $\frac{a}{b}$ δεν σημαίνει $\alpha * \frac{1}{b}$ αλλά το πηλίκο της διαίρεσης του α με τον b.
5. Εξάγουμε το δημόσιο κλειδί $pk = (n, g)$ και το ιδιωτικό κλειδί $sk = (\lambda, \mu)$.

$Encrypt_{Pail}$

1. Δίνουμε ένα μήνυμα m ως είσοδο όπου $m \in \mathbb{Z}_n$.
2. Επιλέγουμε τυχαίο $r \in z_n^*$.
3. Υπολογίζουμε $c = g^m r^n \bmod n^2$.

$Decrypt_{Pail}$

1. Δίνουμε σαν είσοδο ένα κρυπτοκείμενο c όπου $c \in \mathbb{Z}_n^*$.
2. Υπολογίζουμε $m = L(c^\lambda \bmod n^2) \mu \bmod n$.

Το σχήμα του Paillier είναι ομομορφικό ως προς την πράξη της πρόσθεσης και χρησιμοποιείται στα συστήματα ηλεκτρονικής ψηφοφορίας.

Συγκεκριμένα παρουσιάζει τις εξής αλγεβρικές ιδιότητες:

Για την πρόσθεση

- $D(E(m_{1,} r_1) * E(m_{2,} r_2) \bmod n^2) = (m_1 + m_2) \bmod n$,
- $D(E(m_{1,} r_1) * g^{m_2} \bmod n^2) = (m_1 + m_2) \bmod n$

Για τον πολλαπλασιασμό

- $D\left(E\left(m_{1,}r_1\right)^{m_2} mod\, n^2\right)=m_1 m_2 mod\, n$,
- $D\left(E\left(m_{2,}r_2\right)^{m_1} mod\, n^2\right)=m_1 m_2 mod\, n$.

Πιο γενικά:

- $D\left(E\left(m_{1,}r_1\right)^{k} mod\, n^2\right)=km_1 mod\, n$ όπου k μια σταθερά.

Να σημειωθεί ότι αν και ο Paillier έχει ομομορφικές ιδιότητες και για την πρόσθεση και για τον πολλαπλασιασμό, δεν αποτελεί ένα πλήρως ομομορφικό σχήμα γιατί για την πράξη του πολλαπλασιασμού με δοσμένα $E\left(m_1\right)$ και $E\left(m_2\right)$ δεν μπορούμε να υπολογίσουμε το $E\left(m_1 * m_2\right)$. Μπορούμε μόνο να υπολογίσουμε το $E\left(m_1 + m_2\right)$. Το $E\left(m_1 * m_2\right)$ μπορεί να υπολογιστεί μόνο με δεδομένο $E\left(m_1\right)$ και m_2 . Δηλαδή το ένα μήνυμα από τα δυο δεν είναι κρυπτογραφημένο και παραβιάζει τον ορισμό ενός πλήρως ομομορφικού σχήματος.

5. Gentry – Ένα πλήρες ομομορφικό σχήμα

Για δεκαετίες πριν ο Craig Gentry το 2009 ανακαλύψει την επαναστατική του μέθοδο ώστε να δημιουργήσει ένα πλήρες ομομορφικό σχήμα, πολλοί ερευνητές σε παγκόσμιο επίπεδο προσπαθούσαν να σχεδιάσουν ένα τέτοιο σύστημα. Εδώ και 30 χρόνια η πλήρης ομομορφική κρυπτογράφηση παρέμενε ένα ανοιχτό πρόβλημα πριν ο Gentry την καταστήσει εφικτή. Κατά την διάρκεια όλων αυτών των ετών το καλύτερο κρυπτοσύστημα με τις επιθυμητές ιδιότητες της πλήρους ομομορφικής κρυπτογράφησης ήταν αυτό των Boneh–Goh-Nissim [17]. Το βασικό του μειονέκτημα ήταν ότι ενώ επέτρεπε τον υπολογισμό ενός απεριόριστου αριθμού πράξεων ως προς την πρόσθεση, επέτρεπε το πολύ μιας πράξης ως προς τον πολλαπλασιασμό. Ένας βασικός λόγος που ένα τέτοιο σχήμα δεν μπορεί να υπολογίσει κυκλώματα πέρα από ένα ορισμένο βάθος είναι πως κατά την εκτέλεση των υπολογισμών σταδιακά έχουμε συσσώρευση σφαλμάτων που καλούμε **θόρυβο** και έτσι κατά την αποκρυπτογράφηση καταλήγουμε σε λανθασμένη τιμή.

Αυτό που σκέφτηκε ο Gentry σε αντίθεση με τους υπόλοιπους και οδηγήθηκε στην λύση είναι να χρησιμοποιήσει μια τεχνική που κανείς στον κόσμο πριν δεν είχε σκεφτεί και στηρίζεται στα δικτυώματα. Αντί να προσπαθήσει λοιπόν να κατασκευάσει ένα σχήμα που έχει από την αρχή όλες τις ιδιότητες που οφείλει να έχει ένα ομομορφικό σχήμα, έκανε κάτι διαφορετικό. Κατασκεύασε ένα **“κάπως” ομομορφικό σχήμα (somewhat homomorphic scheme)** από το οποίο καταλήγει με κάποιες ειδικές τεχνικές που θα αναπτύξουμε παρακάτω, σε ένα **πλήρως ομομορφικό (full homomorphic scheme)**.

Πριν προχωρήσουμε στην αναλυτική περιγραφή του σχήματος του Gentry θα περιγράψουμε την κατασκευή από μια πιο γενική σκοπιά, ώστε να βοηθήσουμε τον αναγνώστη να εξοικειωθεί πρώτα με το σχήμα διαισθητικά.

5.1. Συνοπτική Παρουσίαση του σχήματος Gentry

Η κατασκευή ξεκινά από ένα “**somewhat**” ομομορφικό σχήμα το οποίο περιορίζεται στον υπολογισμό κάποιων πολυωνύμων μικρού βαθμού κρυπτογραφημένων δεδομένων. Ο περιορισμός αυτός υφίσταται γιατί όπως αναφέραμε κάθε κρυπτοκείμενο εμπεριέχει θόρυβο και αυτός ο θόρυβος πολλαπλασιάζεται καθώς εκτελούνται αριθμητικές πράξεις μεταξύ διαφορετικών κρυπτοκειμένων οπότε τελικά καταλήγουμε με ένα αποτέλεσμα σε κρυπτογραφημένη μεν μορφή, αλλά αδύνατο να μας οδηγήσει στο αναμενόμενο αποτέλεσμα κατά την αποκρυπτογράφηση του.

Στην συνέχεια ο Gentry αποδεικνύει στην εργασία του πως ένα τέτοιο σύστημα μπορεί να τροποποιηθεί σε ένα σχήμα που ονομάζεται **bootstrappable** μέσω μιας διαδικασίας που ονομάζεται “**squashing**”**.** Σε αυτό το στάδιο το κρυπτοσύστημα του Gentry είναι ικανό να υπολογίσει μια ελαφρά επαυξημένη έκδοση του κυκλώματος αποκρυπτογράφησης του με μια επιπλέον πράξη.

Στο τέλος ο Gentry αποδεικνύει πως κάθε “somewhat” ομομορφικό σχήμα που είναι bootstrappable μπορεί να μετατραπεί σε full ομομορφικό σχήμα μέσω της τεχνικής **bootstrapping** η οποία περιλαμβάνει μια αναδρομική διαδικασία που ονομάζεται refreshing. Κατά το refreshing το “θορυβώδες” σχήμα του Gentry ανανεώνει περιοδικά τα κρυπτογραφήματα εφαρμόζοντας την διαδικασία αποκρυπτογράφησης τους ομομορφικά αποκτώντας έτσι ένα νέο κρυπτογράφημα το οποίο κρυπτογραφεί την ίδια τιμή με πριν εμπεριέχοντας όμως μικρότερο θόρυβο. Μέσω αυτής της περιοδικής “ανανέωσης” (refreshing) των κρυπτοκειμένων που λαμβάνει χώρα κάθε φορά που ο θόρυβος τους αυξάνει πολύ, καθίσταται επιτρεπτός ο υπολογισμός αυθαίρετα μεγάλου αριθμού προσθέσεων και πολλαπλασιασμών.

Το μεγάλο μειονέκτημα του σχήματος μέχρι και αυτή την στιγμή που συγγράφεται αυτή η διατριβή είναι πως αν και επιλύει σε θεωρητικό επίπεδο το πρόβλημα της ύπαρξης ενός σχήματος πλήρους ομομορφικής κρυπτογράφησης, δεν είναι καθόλου πρακτικό. Το μέγεθος των κρυπτοκειμένων αυξάνει κατά πολύ όσο αυξάνουμε το επίπεδο ασφάλειας και έτσι καθιστά τον χρόνο υπολογισμού υπερβολικά μεγάλο για να μπορεί να στηρίξει πρακτικές εφαρμογές. Ωστόσο έχουν προταθεί πολλές βελτιώσεις και τροποποιήσεις του αρχικού σχήματος Gentry κάποιες από τις οποίες δεν στηρίζονται πλέον καν σε ιδεώδη δικτυωμάτων.

Η μέθοδος του Gentry λοιπόν μπορεί να θεωρηθεί ότι περιλαμβάνει τα εξής 3 βήματα:

Βήμα 1: Κατασκευή ενός σχήματος κρυπτογράφησης χρησιμοποιώντας ιδεώδη δικτυώματα, το οποίο είναι **somewhat homomorphic,** δηλαδή περιορίζεται στον υπολογισμό πολυωνύμων μικρού βαθμού σε κρυπτογραφημένα δεδομένα. Το σχήμα αυτό στηρίζεται στο σχήμα Goldreich-Goldwasser-Halevi [18] που δημοσιεύθηκε το 1997 και στηρίζεται σε προβλήματα δικτυωμάτων.

Βήμα 2: Εφαρμογή της μεθόδου **squashing** στο κύκλωμα αποκρυπτογράφησης του σχήματος του προηγούμενου βήματος ώστε να μετατραπεί σε **bootstrappable.**

Βήμα 3: Εφαρμογή της μεθόδου **bootstrapping** στο σχήμα που δημιουργήθηκε στο βήμα 2 με σκοπό να μετατραπεί σε ένα πλήρως ομομορφικό σχήμα. Η μέθοδος περιλαμβάνει μια αναδρομική διαδικασία "ανανέωσης".

Η καινοτόμος ιδέα της μεθόδου Gentry για τη δημιουργία ενός πλήρως ομομορφικού συστήματος (full homomorphic) από ένα κάπως ομομορφικό (somewhat homomorphic) είναι οι μέθοδοι **squashing** και **bootstrapping**. Ωστόσο από μαθηματική άποψη το πιο ελκυστικό βήμα είναι το πρώτο βήμα. Πρέπει να τονίσουμε πως πολλοί από τους αλγορίθμους που αναφέρθηκαν ανωτέρω είναι σε αφηρημένη (θεωρητική) μορφή.

5.2. Μαθηματική κατασκευή του σχήματος Gentry

5.2.1. Somewhat Homomorphic Scheme

Όπως αναφέρθηκε ήδη ο στόχος της κατασκευής πρώτα ενός κάπως ομομορφικού σχήματος (SHS) είναι για να κατασκευάσουμε τελικά ένα σχήμα κρυπτογράφησης που να είναι bootstrappable σε σχέση με ένα γενικό σύνολο πυλών (πράξεων). Για την κατασκευή αυτή θα χρησιμοποιήσουμε μαθηματικά αντικείμενα της θεωρίας δικτυωμάτων. Υπενθυμίζουμε ότι στα πλήρως ομομορφικά σχήματα χρειαζόμαστε και έναν τέταρτο αλγόριθμο που συμβολίζουμε με $Eval_{Gentry}$.

Αρχικά ο αλγόριθμος $KeyGen_{Gentry}$ παίρνει ως είσοδο έναν δακτύλιο R και μια βάση B_I ενός ιδεώδους $I \lhd R$ (υπενθυμίζουμε πως υπάρχει ένας φυσικός ισομορφισμός μεταξύ δικτυωμάτων και ιδεώδων δακτυλίων) και χρησιμοποιείται για να ενσωματώσουμε το αρχικό μήνυμα σε ένα διάνυσμα σφάλματος (error vector). Επιπλέον ένας άλλος αλγόριθμος $IdealGen(R, B_I)$ χρησιμοποιείται για την παραγωγή του *δημόσιου* και *ιδιωτικού* κλειδιού. Το ιδιωτικό κλειδί αποτελείται από μια "καλή" βάση ενός ιδεώδους δικτυώματος J που θα συμβολίσουμε με B_{sk} . Το ιδεώδες δικτύωμα J επιλέγεται με τέτοιον τρόπο ώστε να είναι σχετικά πρώτο με το I δηλαδή $I+J=R$.Το δημόσιο κλειδί αποτελείται από μια "κακή" βάση B_{pk} του J. Για παράδειγμα μια τέτοια κακή βάση είναι η HNF της B_{sk} . Ο αλγόριθμος $Encrypt_{Gentry}$ παίρνει ως είσοδο ένα μήνυμα $\vec{m}$ και το δημόσιο κλειδί B_{pk} . Ο χώρος των μηνυμάτων θα τον συμβολίζουμε με P (plaintext space) και είναι ένα υποσύνολο του $R \bmod B_I$. Περιλαμβάνει έναν επιπλέον αλγόριθμο που συμβολίζουμε με $Samp(\vec{m}, B_I)$ και παράγει ένα διάνυσμα $\vec{e}$ από το σύμπλοκο $\vec{m}+I$. Στην συνέχεια κρυπτογραφεί το μήνυμα και παράγει ένα κρυπτοκείμενο $\vec{c} = \underbrace{\vec{m}+\vec{i}}_{\vec{e}} \bmod B_{pk}$. Η ομομορφική ιδιότητα του σχήματος αντικατοπτρίζεται από τον αλγόριθμο $Eval_{Gentry}$. Ο συγκεκριμένος αλγόριθμος δέχεται ως είσοδο ένα κύκλωμα C το οποίο λαμβάνεται από μια κλάση C_E επιτρεπτών κυκλωμάτων του οποίου οι πύλες περιλαμβάνουν πράξεις modulo B_I . Επίσης λαμβάνει ως είσοδο το δημόσιο κλειδί B_{pk} και ένα σύνολο κρυπτοκειμένων $\Psi = \{\vec{c}_1, \ldots, \vec{c}_m\}$.

Ο αλγόριθμος εκτελεί τις πράξεις σε κατάλληλη σειρά ώστε να υπολογίσει τελικά το κρυπτοκείμενο $\vec{c}$.

1. $Add(B_{pk}, \vec{c}_1, \vec{c}_2) = \vec{c}_1 + \vec{c}_2 \bmod B_{pk}$
2. $Mult(B_{pk}, \vec{c}_1, \vec{c}_2) = \vec{c}_1 * \vec{c}_2 \bmod B_{pk}$

Ο λόγος που το σχήμα είναι "κάπως" ομομορφικό είναι εξαιτίας του θορύβου που αυξάνεται μετά από κάθε πράξη. Όπως έχει ήδη αναφερθεί το πρόβλημα αυτό αντιμετωπίζεται μέσω του "refreshing" που τελικά μετατρέπει το σχήμα μας στην πολύ ισχυρή κατασκευή ενός πλήρως ομομορφικού σχήματος.

Ο αλγόριθμος $Decrypt_{Gentry}$ δέχεται ως είσοδο ένα κρυπτοκείμενο και το ιδιωτικό κλειδί B_{sk} και παράγει το αρχικό μήνυμα:

$$\vec{m} = (\vec{c} \bmod B_{sk}) \bmod B_I \quad .$$

Συγκετρωτικά έχουμε λοιπόν για το πρώτο στάδιο του σχήματος του Gentry:

$KeyGen_{Gentry}$

1. Δίνουμε ως είσοδο έναν δακτύλιο $R=\mathbb{Z}[X]/f(x)$ όπου f μονικό πολυώνυμο και μια βάση B_I όπου $I \lhd R$.
2. Εκτελούμε τον αλγόριθμο $IdealGen(R, B_I)$ και παράγουμε την δυάδα (B_{pk}, B_{sk}) , όπου (B_{pk}, B_{sk}) είναι βάσεις ενός ιδεώδους J με την ιδιότητα I + J = R.
3. Επιστρέφουμε το ζεύγος κλειδιών (pk, sk) όπου:
4. Το δημόσιο κλειδί ειναι το pk και είναι $pk=(R, B_I, B_{pk}, Samp)$ και το ιδιωτικό κλειδί είναι το sk και είναι $sk=B_{sk}$.

$Encrypt_{Gentry}$

1. Εισάγουμε το αρχικό μήνυμα $\vec{m} \in P$ και το δημόσιο κλειδί pk.
2. Εκτελούμε τον αλγόριθμο $Samp(\vec{m}, B_I) \rightarrow \vec{e} \in \vec{m} + I$.
3. Υπολογίζουμε $\vec{c} = \vec{e} \bmod B_{pk}$, $\vec{c} \in R \bmod B_{pk}$.

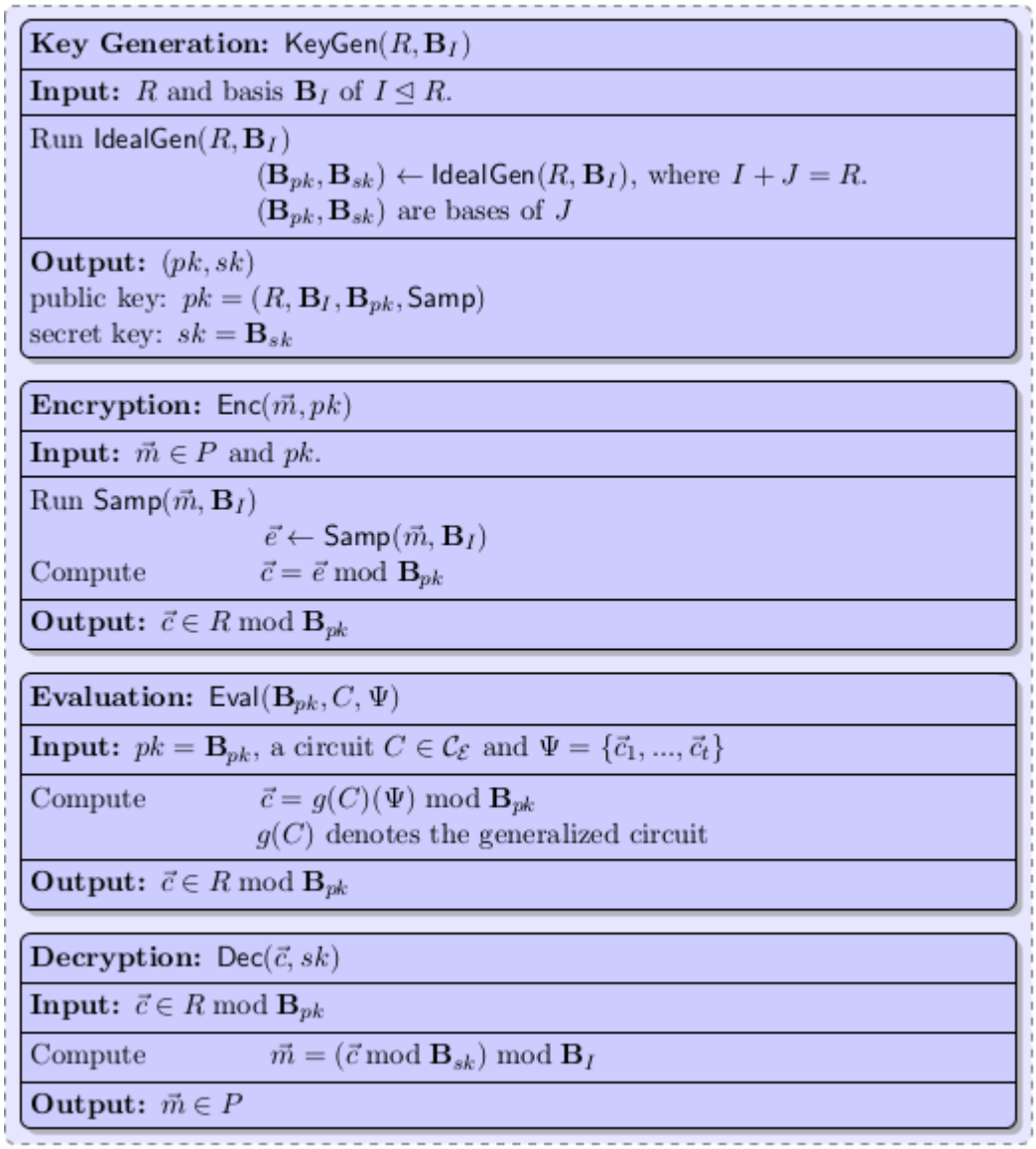
$Eval_{Gentry}$

1. Εισάγουμε το δημόσιο κλειδί pk, ένα κύκλωμα $C \in C_E$ από μια επιτρεπτή οικογένεια κυκλωμάτων και ένα σύνολο κρυπτοκειμένων $\Psi = \{\vec{c}_1, \ldots, \vec{c}_t\}$.
2. Υπολογίζουμε $\vec{c} = g(C)(\Psi) \bmod B_{pk}$, $g(C)$ συμβολίζει το γενικευμένο κύκλωμα (generalized circuit).
3. Επιστρέφουμε την τιμή της αποτίμησης του κυκλώματος σε κρυπτογραφημένη μορφή όπου $\vec{c} \in R \bmod B_{pk}$.

$Decrypt_{Gentry}$

1. Εισάγουμε $\vec{c} \in R \bmod B_{pk}$.
2. Αποκρυπτογραφούμε το κρυπτογραφημένο αποτέλεσμα υπολογίζοντας $\vec{m} = (\vec{c} \bmod B_{sk}) \bmod B_I$
3. Επιστρέφουμε το $\vec{m} \in P$.

Παρατηρήσεις. Ο Gentry περιγράφει στην εργασία του [19] πως γίνεται η κατασκευή ενός “επιτρεπτού” κυκλώματος C και αποδεικνύει πως το ανωτέρω σχήμα έχει την ιδιότητα της ορθότητας για ένα τέτοιο κύκλωμα. Επιπλέον αποδεικνύει ότι το παραπάνω σχήμα παρέχει σημασιολογική ασφάλεια.
Η ασφάλεια του κρυπτοσυστήματος αυτού στηρίζεται στο πρόβλημα (**γ-BDDP**) .



Εικόνα 5.2.1.: Το somewhat homomorphic σχήμα του Gentry χρησιμοποιώντας ιδεώδη δικτυώματα.

5.2.1.1. Ορθότητα (correctness) του SHS

Η απόδειξη της ορθότητας του σχήματος αποτελείται από δύο φάσεις. Στην πρώτη αποδεικνύεται πως ο αλγόριθμος $Decrypt_{Gentry}$ πράγματι παράγει το αρχικό μήνυμα για ένα κρυπτογράφημα που προέκυψε από το SHS. Στο δεύτερο κομμάτι αποδεικνύεται πως ο αλγόριθμος $Eval_{Gentry}$ έχει την ιδιότητα της ορθότητας (correctness).

1η φάση ($Decrypt_{Gentry}$ **)**

Έστω c ένα κρυπτοκείμενο που προέκυψε από το ανωτέρω SHS. Σύμφωνα με τα παραπάνω $\vec{c}=\vec{e}+\vec{j}$ για κάποιο $\vec{j}\in J$. Εφόσον B_{sk} είναι βάση του ιδεώδους δικτυώματος (ideal lattice) J, μπορούμε να γράψουμε $\vec{j}=B_{sk}\vec{a}$, οπότε $\vec{c}=B_{sk}\vec{a}+\vec{e}$. Σύμφωνα με την διαδικασία αποκρυπτογράφησης που περιγράφηκε πρέπει να υπολογίσουμε το $\vec{c}\,mod\,B_{sk}$. Έχουμε $\vec{c}\,mod\,B_{sk}=\vec{c}-B_{sk}\lfloor B_{sk}^{-1}\vec{c}\,\rfloor=B_{sk}[B_{sk}^{-1}\vec{c}\,]=B_{sk}[B_{sk}^{-1}(B_{sk}\vec{a}+\vec{e})]=B_{sk}[\vec{a}+B_{sk}^{-1}\vec{e}]$. Επίσης $B_{sk}[\vec{a}+B_{sk}^{-1}\vec{e}]=B_{sk}[B_{sk}^{-1}\vec{e}]$. Αν απαιτήσουμε πως οι στήλες του B_{sk}^{-1} έχουν ευκλείδεια νόρμα μικρότερη από $1/2\|\vec{e}\|$ τότε $[B_{sk}^{-1}\vec{e}]=B_{sk}^{-1}\vec{e}$. Οπότε $\vec{c}\,mod\,B_{sk}=B_{sk}*B_{sk}^{-1}\vec{e}=\vec{e}$. Λόγω του ότι $\vec{e}=\vec{m}+\vec{i}\,,i\in I$ μπορούμε να εξάγουμε το $\vec{m}$ υπολογίζοντας το $\vec{c}\,modB_I$.

2η φάση ($Eval_{Gentry}$ **)**

Για την απόδειξη της ορθότητας του αλγορίθμου θα χρειαστεί να δωθούν κάποιοι ορισμοί.

Ορισμός 5.1. (Γενικευμένο κύκλωμα – generalized Circuit). Έστω C είναι ένα $(mod\,B_I)$ κύκλωμα. **Γενικευμένο κύκλωμα** g(C) του C είναι ένα κύκλωμα που σχηματίζεται αντικαθιστώντας τις πράξεις Add_{B_I} , $Mult_{B_I}$ του C με τις πράξεις του + και * του δακτυλίου R.

Ορισμός 5.2. (X_{Enc},X_{Dec}) **.** Με X_{Enc} θα συμβολίζουμε την εικόνα της $Samp$. Τότε όλα τα κρυπτοκείμενα που εξάγονται από τον Enc είναι της μορφής $X_{Enc}+J$. Με X_{Dec} συμβολίζουμε το σύνολο που αποτελείται από στοιχεία της μορφής $R\,mod\,B_{sk}$.

Ορισμός 5.3. (επιτρεπτό κύκλωμα - permitted Circuit).

Έστω $C_E{}'=\{C\,|\,\forall(\vec{x}_{1,}\vec{x}_{2,\ldots},\vec{x}_t)\in X_{Enc}{}^t:g(C)(\vec{x}_{1,}\vec{x}_{2,\ldots},\vec{x}_t)\in X_{Dec}\}$. Μπορούμε να πούμε πως ένα **επιτρεπτό κύκλωμα** είναι ένα κύκλωμα που όταν γενικευτεί η έξοδος του είναι X_{Dec} αν η είσοδος είναι X_{Enc} . Αυτό συμβαίνει όταν το σφάλμα $g(C)(\vec{e}_{1,},\ldots,\vec{e}_t)$ των εξαγόμενων κρυπτοκειμένων είναι κάτω από ένα όριο (μικρό). Εάν $C_E\subseteq C_E{}'$ τότε καλούμε την οικογένεια C_E **οικογένεια επιτρεπτών κυκλωμάτων.**

Ορισμός 5.4. (έγκυρο κρυπτοκείμενο - valid ciphertext). Ένα κρυπτοκείμενο $\vec{c}$ καλείται **έγκυρο** κρυπτοκείμενο σε σχέση με το E, το δημόσιο κλειδί pk, και την οικογένεια επιτρεπτών κυκλωμάτων C_E εάν ισούται με $Eval_{gentry}(pk,C,\Psi)$ για κάποιο $C\in C_E$, και για κάθε $\vec{c}\in X_{Enc}$.

Επίσης ορίζεται το **ταυτοτικό** κύκλωμα C με την ιδιότητα η έξοδος του αλγορίθμου $Eval_{Gentry}$ ισούται με την έξοδο του αλγορίθμου $Encrypt_{Gentry}$.

Θεώρημα 5.5. Έστω C_E ένα σύνολο επιτρεπτών κυκλωμάτων το οποίο περιέχει και το μοναδιαίο κύκλωμα. Τότε για έγκυρα κρυπτοκειμενα $\vec{c}$ τότε $Decrypt_{Gentry}(sk,\vec{c})=C(\vec{m}_{1,}\ldots,\vec{m}_t)$ δηλαδή το σχήμα E είναι ορθό.

Απόδειξη. Έστω $\Psi=\{\vec{c}_{1,}\vec{c}_{2,}\ldots,\vec{c}_t\}$ όπου $\vec{c}_k=\vec{m}_k+\vec{i}_k+\vec{j}_k$ όπου $\vec{m}_k\in P,\vec{i}_k\in I,\vec{j}_k\in J$ και $\vec{m}_k+\vec{i}_k\in X_{Enc}$. Οπότε έχουμε $\vec{c}=Eval_{Gentry}(B_{pk},C,\Psi)=g(C)(\Psi)\,mod\,B_{pk}\in g(C)(\vec{m}_1+\vec{i}_{1,\ldots},\vec{m}_t+\vec{i}_t)+J$.

Εφόσον το κύκλωμα είναι επιτρεπτό κύκλωμα έχουμε:

$$\begin{aligned}&Dec_{Gentry}(\vec{c},B_{sk})=g(C)(\vec{m}_1+\vec{i}_{1,\ldots},\vec{m}_t+\vec{i}_t)\,mod\,B_I\\&=(g)(C)(\vec{m}_1+\vec{i}_{1,\ldots},\vec{m}_t+\vec{i}_t+J\,mod\,B_{sk})\,mod\,B_I\\&=C(\vec{m}_{1,}\ldots,\vec{m}_t)\end{aligned}$$

Έτσι αποδείξαμε ότι το σχήμα SHS ειναι ορθό (correct) για επιτρεπτά κυκλώματα. Το επόμενο βήμα είναι να αυξήσουμε τον πληθάριθμο του συνόλου των επιτρεπτών κυκλωμάτων.

5.2.1.2 Μεγιστοποιήση του βάθους του κυκλώματος (Maximizing Circuit Depth)

Όπως ειπώθηκε ανωτέρω το παραπάνω σχήμα εκτιμά σωστά τις πύλες $Add_{B_I}, Mult_{B_I}$ εάν $X_{Enc}+X_{Enc}\subseteq X_{Dec}$ και $X_{Enc}*X_{Enc}\subseteq X_{Dec}$. Εξαιτίας του ότι χρησιμοποιούμε ιδεώδη δικτυώματα είναι δυνατή μια γεωμετρική αναπαράσταση των συνόλων X_{Enc} και X_{Dec} ως υποσύνολα του $\mathbb{Z}_n$.

Ορισμός 5.6. (r_{Enc}, r_{Dec}) Με r_{Enc} συμβολίζουμε την μικρότερη ακτίνα ώστε $X_{Enc}\subseteq B(r_{Enc})$.
Επίσης με r_{Dec} συμβολίζουμε την μεγαλύτερη ακτίνα ώστε $X_{Dec}\supseteq B(r_{Dec})$.

Υπό αυτή την έννοια το σύνολο των επιτρεπτών κυκλωμάτων μπορεί να οριστεί ως εξής:
$C_E=\{C\,|\,\forall\{\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_t\}\in B(r_{Enc})^t : g(C)(\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_t)\in B(r_{Dec})\}$
Προκειμένου να αυξήσουμε λοιπόν το σύνολο C_E είναι απαραίτητο να βρούμε ένα φράγμα για την ευκλείδεια νόρμα $\|g(C)(\vec{e}_1,\ldots,\vec{e}_t)\|$.
Ο Gentry απέδειξε πως το SHS μπορεί να αποτιμήσει κυκλώματα βάθους το πολύ:
$loglogr_{Dec}-loglog(\gamma_{Mult}(R)r_{Enc})$ όπου $\gamma_{Mult}(R)$ είναι ο παράγοντας διαστολής (expansion factor) του πολλαπλασιασμού.

5.2.2 Μετατρέποντας το SHS σε ένα bootstrappable σχήμα (Squashing)

Πριν ο Gentry προχωρήσει στην τεχνική του **squashing** του κυκλώματος αποκρυπτογράφησης, παρουσίασε μέσω κάποιων μαθηματικών αποδείξεων οι οποίες παραλείπονται γιατί η ανάλυση τους ξεφεύγει του σκοπού της συγκεκριμένης διπλωματικής ένα ελαφρά βελτιωμένο σχήμα σε σχέση με το αρχικό το οποίο παρουσιάζει μειωμένη πολυπλοκότητα του κυκλώματος αποκρυπτογράφησης.

Έτσι αν συμβολίσουμε το νέο σχήμα με $E^*=(KeyGen^*, Enc^*, Dec^*, Eval^*)$ έχουμε το νέο βελτιωμένο σχήμα:

$KeyGen^*$

1. Δίνουμε ως είσοδο έναν δακτύλιο $R=\mathbb{Z}[X]/f(x)$ όπου f μονικό πολυώνυμο και μια βάση B_I όπου $I\lhd R$.
2. Εκτελούμε τον αλγόριθμο $IdealGen(R, B_I)$ και παράγουμε την δυάδα (B_{pk}, B_{sk}) , όπου (B_{pk}, B_{sk}) είναι βάσεις ενός ιδεώδους J με την ιδιότητα I + J = R.
3. Υπολογίζουμε ένα διάνυσμα $\vec{v}_{sk}\in J^{-1} : P(rot(\vec{v}_{sk}^{-1}))\supset B(r_{Dec}/(2*n^{2.5}\|f\|\|B_I\|))$, όπου $rot(\vec{v}_{sk}^{-1})=\{\vec{v}_{sk}^{-1}\,\chi\, x^i \bmod f(x)\}$.
4. Επιστρέφουμε το ζεύγος κλειδιών (pk, sk) όπου:
5. Το δημόσιο κλειδί ειναι το pk και είναι $pk=(R, B_I, B_{pk}, Samp)$ και το ιδιωτικό κλειδί είναι το sk και είναι $sk=\vec{v}_{sk}$.

Encrypt[*]

1. Εισάγουμε το αρχικό μήνυμα $\vec{m} \in P$ και το δημόσιο κλειδί pk.
2. Εκτελούμε τον αλγόριθμο $Samp(\vec{m}, B_I) \rightarrow \vec{e} \in \vec{m} + I$.
3. Υπολογίζουμε $\vec{c} = \vec{e} \bmod B_{pk}$, $\vec{c} \in R \bmod B_{pk}$.

Eval[*]

1. Εισάγουμε το δημόσιο κλειδί pk, ένα κύκλωμα $C \in C_E$ και μια ένα σύνολο κρυπτοκειμένων $\Psi = \{\vec{c}_1, \ldots, \vec{c}_t\}$.
2. Υπολογίζουμε $\vec{c} = g(C)(\Psi) \bmod B_{pk}$, $g(C)$ συμβολίζει το **γενικευμένο κύκλωμα (generalized circuit)**.
3. Επιστρέφουμε την τιμή της αποτίμησης του κυκλώματος σε κρυπτογραφημένη μορφή όπου $\vec{c} \in R \bmod B_{pk}$.

Decrypt[*]

1. Εισάγουμε $\vec{c} \in R \bmod B_{pk}$.
2. Αποκρυπτογραφούμε το κρυπτογραφημένο αποτέλεσμα υπολογίζοντας $\vec{m} = (\vec{c} - \lceil \vec{v}_{sk} \times \vec{c} \rfloor) \bmod B_I$
3. Επιστρέφουμε το $\vec{m} \in P$.

Key Generation: $\mathsf{KeyGen}^*(R, \mathbf{B}_I)$

Input: R and basis $\mathbf{B}_I$ of $I \trianglelefteq R$.

Run $\mathsf{IdealGen}(R, \mathbf{B}_I)$

Compute $(\mathbf{B}_{pk}, \mathbf{B}_{sk}) \leftarrow \mathsf{IdealGen}(R, \mathbf{B}_I)$
$\vec{v}_{sk} \in J^{-1}$ such that
$\mathcal{P}(\mathrm{rot}(\vec{v}_{sk}^{-1})) \supset \mathcal{B}(r_{\mathsf{Dec}}/(2 \cdot n^{2.5} \|f\| \cdot \|\mathbf{B_I}\|))$

Output: (pk, sk)
public key: $pk = (R, \mathbf{B}_I, \mathbf{B}_{pk}, \mathsf{Samp})$
secret key: $sk = \vec{v}_{sk}$

Encryption: $\mathsf{Enc}^*(\vec{m}, pk) = \mathsf{Enc}(\vec{m}, pk)$

Evaluation: $\mathsf{Eval}^*(\mathbf{B}_{pk}, C, \Psi) = \mathsf{Eval}(\mathbf{B}_{pk}, C, \Psi)$

Decryption: $\mathsf{Dec}^*(\vec{c}, sk)$

Input: $\vec{c} \in R \bmod \mathbf{B}_{pk}$

Compute $\vec{m} = (\vec{c} - \lfloor \vec{v}_{sk} \times \vec{c} \rceil) \bmod \mathbf{B}_I$

Output: $\vec{m} \in P$

Εικόνα 5.2.2.: Το βελτιωμένο somewhat homomorphic scheme

Προκειμένου να μετατρέψει ο Gentry το σχήμα του σε ένα σχήμα που να είναι **bootstrappable** εφάρμοσε την τεχνική του **squashing**. Η βασική ιδέα πίσω από αυτόν τον μετασχηματισμό είναι να ενσωματώσει μια υπόδειξη (hint) για το μυστικό κλειδί μέσα στο δημόσιο. Αυτή η "υπόδειξη" αποτελείται από ένα σύνολο $S=\{\vec{t}_i : i=1,2,\ldots,s\}$ το οποίο περιέχει ένα σύνολο διανυσμάτων t που το άθροισμά τους είναι το ιδιωτικό κλειδί $\vec{v}_{sk}$.

Κάνοντας κάτι τέτοιο μείωσε όπως είναι προφανές την ασφάλεια του κρυπτοσυστήματος και για να την ενισχύσει ξανά χρησιμοποίησε ένα νέο υπολογιστικό πρόβλημα που είναι δύσκολο να επιλυθεί και ονομάζεται **Sparse SubSet Sum Problem (SSSP).**

Έτσι το νέο σχήμα εφοδιάστηκε με δυο επιπλέον αλγορίθμους , $SplitKey$, $ExpandCT$ και όπως είναι προφανές ένα νέο αλγόριθμο Dec .

SplitKey

Ο αλγόριθμος $SplitKey$ εισάγει μια "κρυφή" πληροφορία για το ιδιωτικό κλειδί μέσα στο δημόσιο κλειδί. Είναι μέρος του αλγορίθμου $KeyGen^*$. Το hint αποτελείται από ένα σύνολο τυχαίων διανυσμάτων $\tau=\{\vec{t}_1,\ldots,\vec{t}_r\}\in J^{-1}$ και ένα μυστικό υποσύνολο διανυσμάτων που αθροίζεται στο ιδιωτικό κλειδί δηλαδή : $\vec{v}_{sk}=\sum_{i\in S}\vec{t}_i \bmod I$.

Οι είσοδοι του είναι το δημόσιο και ιδιωτικό κλειδί που παρήχθησαν από τον $Encrypt^*$ και εξάγει μια δυάδα (sk,τ) όπου $sk=\boldsymbol{SK}, sk_{ij}=\begin{cases}1, j\ \text{είναι το}\ i^{th}\ \text{στοιχείο του}\ S\\ 0, \text{διαφορετικά}\end{cases}$

ExpandCT

Ο αλγόριθμος αυτός υπολογίζει τα γινόμενα $\vec{x}_i=\vec{t}_i x\, \vec{c} \bmod B_I, i=1,\ldots,r$, όπου $\vec{c}$ είναι η έξοδος του αλγορίθμου $Encrypt^*$. Το νέο κρυπτοκείμενο είναι $\psi=\{\vec{c},\vec{x}_1,\ldots,\vec{x}_r\}$.

Decrypt

Ο αλγόριθμος δέχεται σαν είσοδο το νέο ιδιωτικό κλειδί $\boldsymbol{SK}$ και το νέο κρυπτοκείμενο ψ . Η γνώση του $\boldsymbol{SK}$ επιτρέπει την εξαγωγή του σχετικού $\{\vec{x}_i\}$ και στην συνέχεια την αποκρυπτογράφηση σύμφωνα με την σχέση $\vec{m}=\vec{c}-\lceil\sum_{i\in S}\vec{x}_i\rfloor \bmod B_I$.

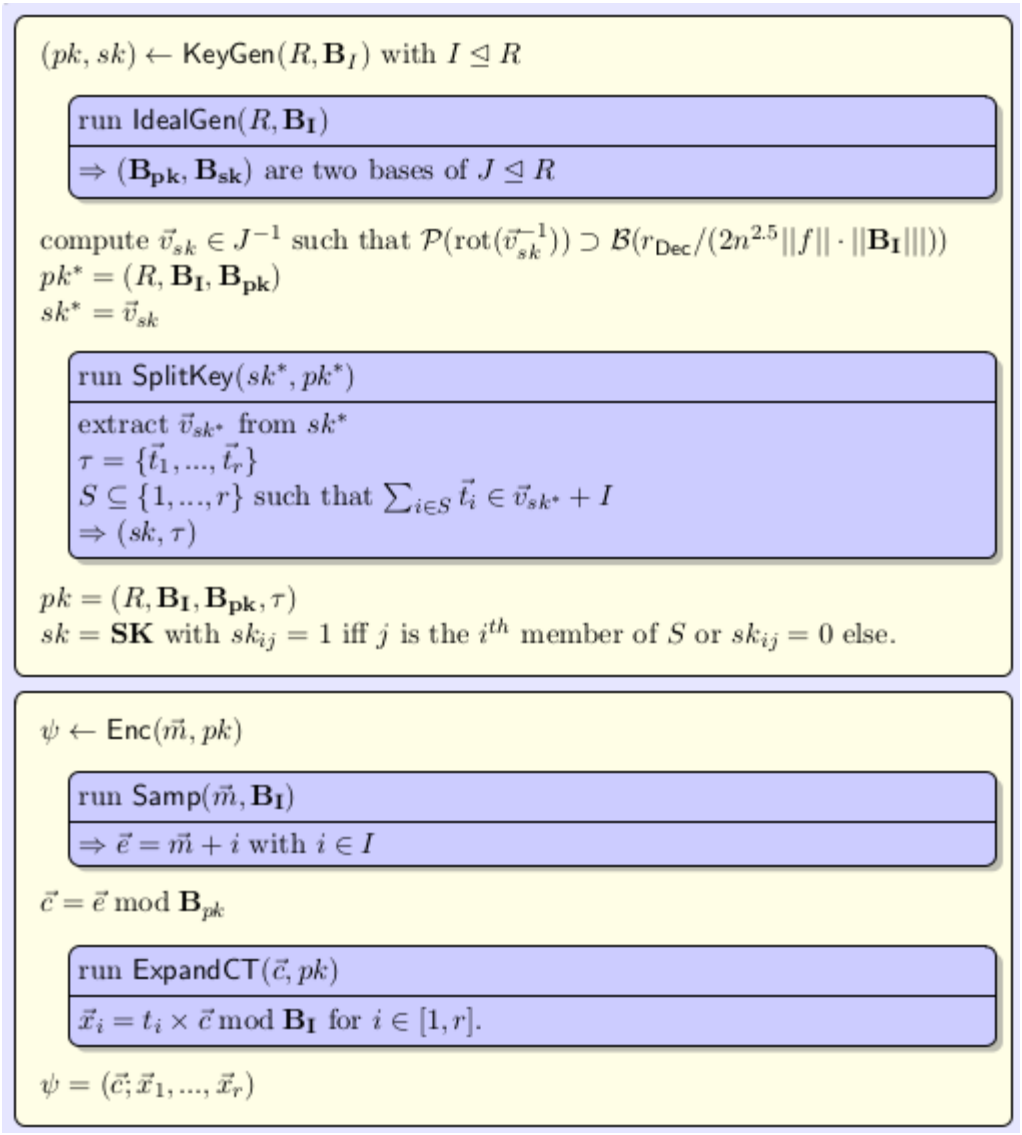
Ο Gentry απέδειξε πως το σχήμα αυτό μπορεί να αποτιμήσει ομομορφικά το κύκλωμα αποκρυπτογράφησης του είναι δηλαδή bootstrappable [19], [παραγ. 10.3].

Ασφάλεια του σχήματος

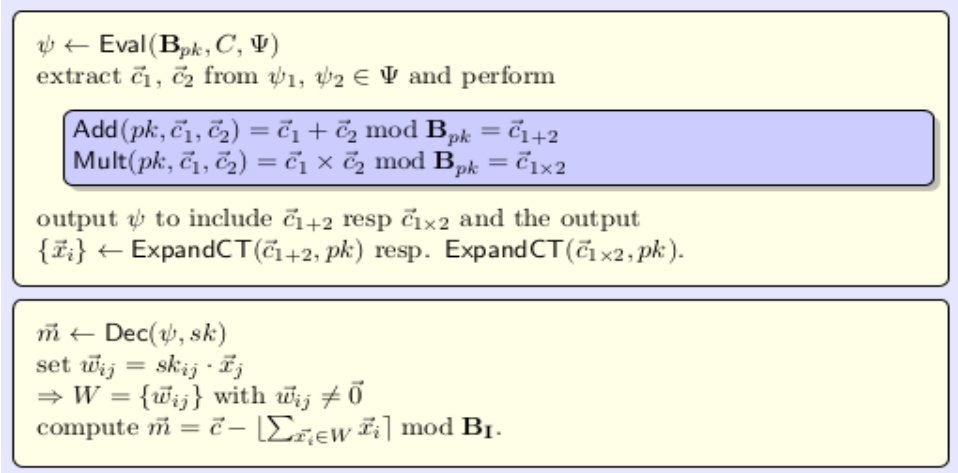
Όπως αναφέρθηκε πριν η εισαγωγή του τ στο δημόσιο κλειδί απαιτεί την ενίσχυση της ασφάλειας του συστήματος η οποία επιτυγχάνεται με την εισαγωγή ενός δεύτερου δύσκολου προβλήματος εκτός του **BDDP.** Το νέο πρόβλημα καλείται ***sparse vector subset sum problem*** **(SVSSP)** το οποίο σχετίζεται στενά με το ήδη γνωστό πρόβλημα ***sparse subset sum problem*** (**SSSP**) το οποίο είναι **NP-complete**.

Ορισμός 5.7. Ένα **στιγμιότυπο(instance)** του **SVSSP** είναι ένα ζευγάρι $(S,\vec{t})$ όπου $S=\{\vec{x}_1,\vec{x}_2,\ldots,\vec{x}_n\}$. Το πρόβλημα που ανάγεται είναι η ύπαρξη ενός υποσυνόλου $T\subset S$ που αθροίζεται στο διάνυσμα στόχο $\vec{t}$.

Είναι προφανές πως απαιτούμε να είναι αρκετά μεγάλο το S καθώς αν είναι μικρό, τότε το σύστημα είναι ευπαθές σε επιθέσεις τύπου **brute force**.



Εικόνα 5.2.2.1.: The Keygen and Encrypt algorithms for the Squashed scheme



Εικόνα 5.2.2.2.: The Evaluation and Decrypt algorithms of the Squashed Scheme

5.2.3. Το πλήρως Ομομορφικό Σχήμα Κρυπτογράφησης

Η ενότητα αυτή περιγράφει το τελικό βήμα της κατασκευής ενός πλήρως ομομορφικού σχήματος. Η σπουδαία ιδιότητα (***bootstrappability***) που έχει το προηγούμενο σχήμα συνεπάγεται **LFH** κρυπτογράφηση από την οποία παράγεται ένα **FHE** σχήμα κρυπτογράφησης.

Συμβολισμός: Στο υπόλοιπο της ενότητας έχουμε τον ακόλουθο συμβολισμό: a_{i_j} συμβολίζει το j^{th} bit του a_i Εάν ένα bit a_{i_j} κρυπτογραφείται με ένα δημόσιο κλειδί pk_z θα συμβολίζεται με $^{z}\lceil a_{i_j} \rfloor$.

5.2.3.1 Η τεχνική Bootstrapping και το Refreshing

Ύστερα από τόσα που έχουν αναφερθεί για την "μαγική" ιδιότητα bootstrappability είναι φυσικό το ερώτημα: Γιατί είναι τόσο σημαντικό αυτό το χαρακτηριστικό? Γενικά μιλώντας, ο κυριότερος λόγος είναι πως η **bootstrappability** επιτρέπει σε ένα σύστημα να ανανεώσει τα κρυπτοκείμενα περιοδικά ώστε να χειριστεί το πρόβλημα της σταδιακής αύξησης του θορύβου. Έτσι εάν είναι δυνατή η αποτίμηση κυκλωμάτων d, μετά η ανανέωση των κρυπτοκειμένων και η εκ νέου αποτίμηση του κυκλώματος αποκρυπτογράφησης βάθους d και ούτω καθεξής, τότε μπορούμε να αποτιμήσουμε κυκλώματα οποιουδήποτε βάθους.

Refreshing. Για να ανανεώσουμε ένα κρυπτοκείμενο c που κρυπτογραφεί ένα αρχικό μήνυμα m με ένα δημόσιο κλειδί ακολουθούμε την εξής διαδικασία:

1. Επανακρυπτογραφούμε με ένα άλλο δημόσιο κλειδί pk_2 .
2. Εφαρμόζουμε ομομορφικά το κύκλωμα αποκρυπτογράφησης D_E στο αποτέλεσμα χρησιμοποιώντας την κρυπτογράφηση του πρώτου ιδιωτικού κλειδιού sk_1 με το δημόσιο κλειδί pk_2 .

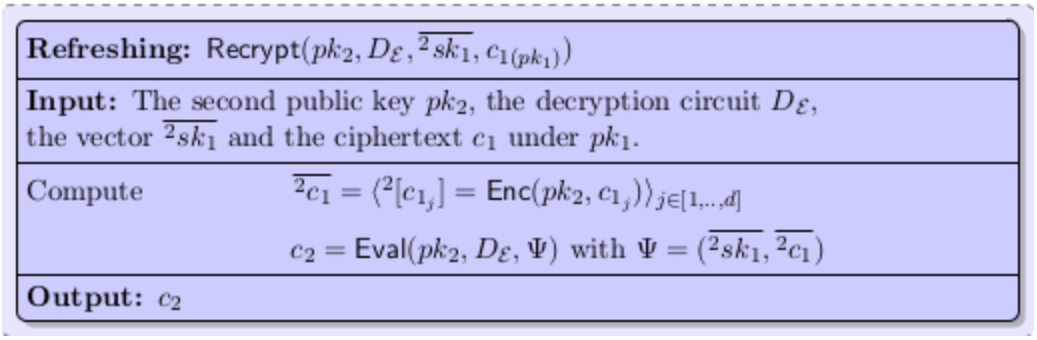
Ως εκ τούτου πετυχαίνουμε μια κρυπτογράφηση του m με χρήση του δημόσιου κλειδιού pk_2 που αντιστοιχεί στο ιδιωτικό κλειδί sk_2 .

Με τον όρο ομομορφική εφαρμογή του κυκλώματος D_E εννούμε την αποκρυπτογράφηση του κρυπτοκειμένου που παράχθηκε με το pk_1 και την ομομορφική κρυπτογράφηση του με το pk_2 .

Πιο φορμαλιστικά έχουμε:
Έστω ένα bootstrappable σχήμα Ε με χώρο μηνυμάτων $P=\{0,1\}$. Έστω επίσης $(pk_{1,}sk_1)$ και $(pk_{2,}sk_2)$ δύο ζεύγη κλειδιών που δημιουργήθηκαν από το Ε. Έστω επίσης $c_{1_{(pk_1)}}=Enc(pk_{1,}m)$, $^2[sk_{1_j}]=Enc(pk_{2,}[sk_{1_j}])$.

Το διάνυσμα που αποτελείται από όλα τα κρυπτογραφημένα bits (με το κλειδί pk_2) συμβολίζεται με $\overline{^2sk_1}=\langle^{2,}[sk_{1_1}]^2[sk_{1_2}]...^2[sk_{1_d}]\rangle$

Με βάση τα ανωτέρω ο αλγόριθμος $Recrypt_E$ φαίνεται στο παρακάτω σχήμα:



Εικόνα 5.2.3.1.: Παράδειγμα αλγορίθμου Recrypt

Ο αλγόριθμος αυτός όπως έχει ήδη αναφερθεί χρησιμοποιείται για να μειώσει τον θόρυβο που σχετίζεται με την πρώτη κρυπτογράφηση c_1 υπό το κλειδί pk_1 . Αυτό επιτυγχάνεται διότι η διαδικασία της αποκρυπτογράφησης που εμπεριέχει ο αλγόριθμος αφαιρεί θόρυβο. Συγχρόνως όμως ο Eval αλγόριθμος επαναδημιουργεί θόρυβο αφού κρυπτογραφεί εκ νέου με το κλειδί pk_2 . Ο θόρυβος όμως αυτός είναι χαμηλότερος του αρχικού πράγμα που είναι επιθυμητό.

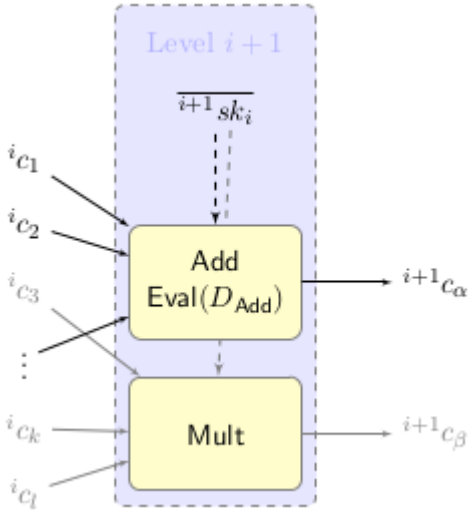
Η παραπάνω τεχνική είναι προφανές πως δεν έχει καμία αξία αν εφαρμόζεται σε ένα μόνο μήνυμα δηλαδή εφαρμόζεται μόνο στο κύκλωμα αποκρυπτογράφησης D_E . Σκοπός είναι να μπορέσουμε να πραγματοποιήσουμε αυθαίρετο αριθμό πράξεων.

Έστω λοιπόν το σχήμα Ε του οποίου το κύκλωμα αποκρυπτογράφησης είναι επαυξημένο με μια επιπλέον πύλη Add. Αυτό το κύκλωμα συμβολίζεται με D_{Add} . Για δυο κρυπτοκείμενα $c_{1,}c_2$ ισχύει: $c_{1_{(pk_1)}}=Enc(m_{1,}pk_1)$, $c_{2_{(pk_1)}}=Enc(m_{2,}pk_2)$

Επίσης σύμφωνα με τα όσα αναφέρθηκαν πριν, υπολογίζουμε τα εξής διανύσματα: $\overline{^2c_{1_{(pk_1)}}}=\langle[^2c_{1_{(pk_j)}}]\rangle, j\in[1,...,d]$ όπου $[^2c_{1_{(pk_j)}}]=Enc(pk_{2,}c_{1_{(pk_1)}})$.

Τότε προκύπτει: $c\leftarrow Eval(pk_{2,}D_{Add},\overline{^2sk_1},\overline{^2c_{1(pk1)}},\overline{^2c_{2(pk2)}})$ και είναι η κρυπτογράφηση του (m_1+m_2) υπό το pk_2 .

Παράδειγμα. Έστω μια πύλη Add στο επίπεδο $i+1$. Δέχεται σαν είσοδο ένα κρυπτογραφημένο ιδιωτικό κλειδί $^{i+1}sk_i$ και μια πλειάδα κρυπτοκειμένων τα οποία είναι κρυπτογραφημένα με το κλειδί pk_i . Η παραπάνω διαδικασία αποτιμά ομομορφικά την πύλη D_{Add} για να λάβει ένα κρυπτοκείμενο υπό το pk_{i+1} στο επίπεδο $i+1$.



Εικόνα 5.2.3.2.: Μια Add πύλη στο επίπεδο i+1

Όπως έχει ήδη αναφερθεί για να είναι το σχήμα bootstrappable πρέπει να μπορεί συμπαγώς να αποτιμήσει όχι μόνο το δικό του κύκλωμα αποκρυπτογράφησης αλλά επίσης ελαφρά επαυξημένες εκδόσεις του.

Ορισμός 5.8. (Επαυξημένο Κύκλωμα Αποκρυπτογράφησης – Augmented Decryption Circuit)
Έστω D_E το κύκλωμα αποκρυπτογράφησης του E το οποίο δέχεται ένα ιδιωτικό κλειδί sk και ένα κρυπτοκείμενο c ως είσοδο . Έστω P ο χώρος των μηνυμάτων (plaintext space). Έστω Γ ένα σύνολο από πύλες με εισόδους και εξόδους από το P συμπεριλαμβανομένης της τετριμένης πύλης (η πύλη που η είσοδος και η έξοδος είναι οι ίδιες). Ένα κύκλωμα που αποτελείται από πολλαπλά αντίγραφα του D_E που συνδέονται με μία μονή πύλη (των πλήθος των αντιγράφων ισούται με τον αριθμό των εισόδων στην g καλείται **g-επαυξημένο κύκλωμα αποκρυπτογράφησης (g-augmented decryption circuit) .** Συμβολίζουμε το σύνολο των g-επαυξημένων κυκλωμάτων αποκρυπτογράφησης με $D_E(\Gamma), g \in \Gamma$.

Ορισμός 5.9. (Bootstrappapble Κρυπτογράφηση)
Έστω C_E το σύνολο των κυκλωμάτων του E που είναι συμπαγή. Το σχήμα E λέμε ότι είναι bootstrappable ως προς το Γ εάν $D_E(\Gamma) \subset C_E$.
Η παραπάνω διαδικασία μας παράγει ένα οριακά πλήρως ομομορφικό σχήμα. Το δημόσιο κλειδί του αποτελείται από μια ακολουθία δημόσιων κλειδιών μαζί με μια ακολουθία κρυπτογραφημένων ιδιωτικών κλειδιών.
$pk^{(d)} = \{pk_{0,}\ldots, pk_d; \overline{^1 sk_0}, \ldots, \overline{^d sk_{d-1}}\}$, όπου sk_i είναι κρυπτογραφημένα με το pk_{i+1} .
Το ιδιωτικό κλειδί είναι $sk^{(d)} = sk_d$. Η ασφάλεια του νέου αυτού σχήματος μπορεί να θεωρηθεί ίδια με την ασφάλεια του αρχικού σχήματος E [4], (θεωρ. 4.2.3.) . Δυστυχώς όμως δεν μπορούμε να ισχυριστούμε το ίδιο και για το μήκος του κλειδιού καθώς αυξάνει ανάλογα με το βάθος d του κυκλώματος. Το ακόλουθο θεώρημα αναφέρεται σε αυτό. Πιο συγκεκριμένα:

Θεώρημα 5.10. Εστω E ένα boostrappable σχήμα. Τότε για κάθε ακέραιο d, είναι δυνατή η κατασκευή $E^{(d)}$ το οποίο μπορεί να εκτιμήσει κάθε κύκλωμα βάθους d. Το κύκλωμα αποκρυπτογράφησης για το σχήμα $E^{(d)}$ είναι το ίδιο με το E καθώς και η πολυπλοκότητα του αλγορίθμου κρυπτογράφησης. Ωστόσο το μήκος του κλειδιού για το σχήμα $E^{(d)}$ είναι $O(d)$ φορές μεγαλύτερο από αυτό του σχήματος E. Εξαιτίας αυτής της εξάρτησης με το d το σχήμα αυτό καλείται **οριακά πλήρως ομομορφικό σχήμα**.

5.2.3.2 Από ένα Leveled Fully Homomorphic σε ένα Full Homomorphic σχήμα

Για να κατασκευάσουμε ένα πλήρως ομομορφικό σχήμα από ένα LFH σχήμα πρέπει να αφαιρέσουμε την εξάρτηση από το βάθος του κυκλώματος. Έτσι το πρόβλημα ανάγεται στην κατασκευή ενός συστήματος E^+ από το $E^{(d)}$, όπου το δημόσιο κλειδί είναι ανεξάρτητο του βάθους d του κυκλώματος.

Ο προφανής τρόπος να μειώσουμε το μέγεθος του δημόσιου κλειδιού είναι να χρησιμοποιήσουμε ένα μόνο όπως για παράδειγμα ένα ζευγάρι (pk, sk) που εξάγεται από το σχήμα E ώστε: $pk^+ = \{pk; \overline{sk}\}$ και $sk^+ = \{sk\}$ όπου $\overline{sk}$ είναι ένα διάνυσμα που αποτελείται από τα κρυπτογραφημένα bits του sk υπό το pk. Το σχήμα E^+ είναι ορθό εφόσον ο αλγόριθμος Recrypt λειτουργεί όπως αναφέρθηκε στα προηγούμενα με την διαφορά όμως πως τα ανανεωμένα κρυπτοκείμενα προέρχονται από το ίδιο δημόσιο κλειδί και όχι από d + 1 διαφορετικά. Αυτό ωστόσο έχει ένα μειονέκτημα.

Για να είναι ασφαλές ένα τέτοιο σχήμα απαιτείται μια νέα απαίτηση ασφαλείας για το σχήμα που ονομάζεται **KDM (key-dependent message)** . Δυστυχώς δεν έχει αποδειχθεί μέχρι σήμερα αν ένα σημασιολογικά ασφαλές σύστημα παρέχει παράλληλα ασφάλεια KDM. Παρακάτω παραθέτουμε αυτούσια τα λόγια του Gentry μέσα από την εργασία του:

> *"Absent proof of KDM-security in the plain model, one way to obtain fully homomorphic encryption from bootstrappable encryption is simply to assume that the underlying bootstrappable encryption scheme is also KDM-secure. This assumption, though unsatisfying, does not seem completely outlandish. While an encrypted secret key is very useful in a bootstrappable encryption scheme - indeed, one may view this as the essence of bootstrappability - we do not see any actual attack on a bootstrappable encryption scheme that provides a self-encrypted key."*

Παρόλο λοιπόν που παραμένει ένα ανοιχτό ερώτημα για την επιστήμη πόσο ασφαλές είναι ένα τέτοιο σύστημα η εργασία του Gentry αποτελεί είναι πραγματικά πρωτοποριακή για όλους τους κρυπτογράφους επιστήμονες στον κόσμο.

6. Συμπεράσματα

Στην παρούσα διατριβή μελετήσαμε το σχήμα του Gentry το πρώτο πλήρες ομομορφικό σχήμα που αναπτύχθηκε ποτέ στην ιστορία της επιστήμης των υπολογιστών και παρουσιάσθηκε το 2009. Αξίζει να σημειωθεί πως το σχήμα που κατασκεύασε ο Gentry σε πρώτη φάση είναι καθαρά θεωρητικό και γι αυτό η μεγάλη του συνεισφορά έγκειται στην απόδειξη της ύπαρξης ενός τέτοιου συστήματος καθώς μέχρι την στιγμή που προτάθηκε δεν ήμασταν καν σίγουροι για την ύπαρξη μιας τέτοιας κατασκευής. Εν συνεχεία ο ίδιος ο Gentry, λίγο αργότερα παρουσίασε μια πρώτη υλοποίηση του σχήματος του. Τα συμπεράσματα που προέκυψαν σχετικά με την αποδοτικότητα μιας τέτοιας υλοποίησης δεν ήταν σε πρώτη φάση ενθαρρυντικά. Ας αναλογιστεί κανείς πως για την εκτέλεση των αλγορίθμων χρησιμοποιήθηκε ένας ισχυρός υπολογιστής IBM System x3500 server με έναν επεξεργαστή 64-bit quad-core Intel Xeon E5450 και 24 GB Ram. Ενδεικτικά του πολύ αργού χρόνου εκτέλεσης να αναφέρουμε πως για τη δημιουργία των κλειδιών μόνο, χρειάστηκαν σε μερικές περιπτώσεις ώρες για διάφορες τιμές των παραμέτρων. Μέχρι και την στιγμή που γράφεται η συγκεκριμένη διατριβή έχουν αναπτυχθεί νεότερα σχήματα κάποια από τα οποία είναι ελαφρώς πιο αποδοτικά. Ωστόσο, τα περισσότερα τα περισσότερα έχουν κάποια πολλά κοινά στοιχεία:

1. Ένα αποδοτικό latticed-based κρυπτοσύστημα, όπου η ασφάλεια του στηρίζεται σε γνωστά δυσεπίλυτα προβλήματα της *θεωρίας δικτυωμάτων*.
2. Ένας επιπλέον αλγόριθμος *Evaluation* που υλοποιεί "καλά" ορισμένες συναρτήσεις c_{add}, c_{mult} ώστε να μην αυξάνουν τον θόρυβο πάνω από επιτρεπτά όρια.
3. Τεχνικές που κάνουν το σχήμα πλήρως ομομορφικό με χρήση του *Evaluation* αλγορίθμου.

Όλα τα ανωτέρω σχήματα δεν θα είχαν δημιουργηθεί χωρίς την επαναστατική ιδέα του Gentry να κατασκευάσει ένα *full homomorphic* σχήμα από ένα *somewhat homomorphic* σχήμα. Θα μπορούσαμε να πούμε χωρίς υπερβολή πως ο Gentry εκτός του ότι απέδειξε ένα άλυτο πρόβλημα 30 ετών έφτιαξε και το προσχέδιο για την κατασκευή πλήρως ομομορφικών σχημάτων:

1. Κατασκευή αρχικά ενός *somewhat homomorphic* scheme.

2. Απλοποίηση του αλγορίθμου αποκρυπτογράφησης όσο το δυνατόν περισσότερο με την τεχνική του *squashing*.

3. Εφαρμογή την τεχνική *bootstrapping* στο σχήμα δηλαδή την περιοδική ανανέωση των κρυπτοκειμένων εφαρμόζοντας την διαδικασία αποκρυπτογράφησης ομομορφικά αποκτώντας έτσι ένα νέο κρυπτοκείμενο που κρυπτογραφεί την ίδια τιμή όπως και πριν, αλλά έχει χαμηλότερο θόρυβο.

Η δυνατότητα πραγματοποίησης οποιουδήποτε υπολογισμού πάνω σε κρυπτογραφημένα δεδομένα είναι φυσικό ότι έχει πληθώρα εφαρμογών. Οι Rivest, Adleman και Dertouzos πρότειναν την υλοποίηση αναζήτησης πάνω σε κρυπτογραφημένα δεδομένα τα οποίο είναι αποθηκευμένα σε έναν (ενδεχομένως) κακόβουλο εξυπηρετητή. Ο χρήστης κωδικοποιεί το ερώτημα αναζήτησης με τέτοιο τρόπο ώστε όταν ο εξυπηρετητής το αποτιμήσει, θα λάβει ως απάντηση ένα κρυπτογραφημένο αποτέλεσμα.

Τέτοιες εφαρμογές έχουν τεράστια σημασία στη σύγχρονη εποχή με την δυνατότητα υπολογισμού στο νέφος (cloud computing), όπου διάφοροι πάροχοι υπηρεσιών έχουν συγκεντρώσει τεράστια υπολογιστική ισχύ (επεξεργαστική ή αποθηκευτική). Τυπικά τα διάφορα δεδομένα διατηρούνται κρυπτογραφημένα, αποκρυπτογραφούνται όμως όταν οι χρήστες θέλουν να τα επεξεργαστούν. Αυτό θέτει τεράστια προβλήματα ιδιωτικότητας, καθώς ο πάροχος μπορεί να είναι κακόβουλος ή να χρειαστεί να παρέχει τα διάφορα δεδομένα σε κυβερνητικούς φορείς. Αυτό που θα θέλαμε ιδανικά, θα ήταν να εκμεταλλευτούμε την υπολογιστική ισχύ του νέφους, χωρίς όμως να θυσιάσουμε την ιδιωτικότητα των δεδομένων μας. Με άλλα λόγια, θέλουμε να επιτρέψουμε την πλήρη επεξεργασία των δεδομένων, δηλαδή να πραγματοποιήσουμε οποιεσδήποτε λειτουργίες σε αυτά, χωρίς όμως να δώσουμε πρόσβαση σε αυτά, ώστε για παράδειγμα, να μπορούμε να ελέγξουμε αν ένα email είναι ανεπιθύμητο (spam), χωρίς όμως να εξετάσουμε τα περιεχόμενα του. Κάτι τέτοιο είναι πλέον θεωρητικά δυνατό με την Πλήρη Ομομορφική Κρυπτογραφία - Fully Homomorphic Encryption (FHE).

Άλλες εφαρμογές της πλήρους ομομορφικής κρυπτογραφίας, αφορούν όλα τα είδη ασφαλούς υπολογισμού μεταξύ δύο ή περισσότερων οντοτήτων όπως για παράδειγμα οι έξυπνοι μετρητές ηλεκτρικής ενέργειας στα έξυπνα δίκτυα ή στα συστήματα ηλεκτρονικής ψηφοφορίας.

Η ομομορφική κρυπτογράφηση είναι σε πρώιμο στάδιο και η παγκόσμια επιστημονική κοινότητα προσανατολίζεται στην δημιουργία πιο αποδοτικών συστημάτων ώστε να την καταστήσουν πρακτική το συντομότερο δυνατό. Όταν αυτό συμβεί η ασφάλεια και η ιδιωτικότητα των δεδομένων μας θα αλλάξει ριζικά προς όφελος μας.

Βιβλιογραφία

[1] Raj Samani, Jim Reavis, Brian Honan, CSA Guide to Cloud Computing: Implementing Cloud Privacy and Security, Syngress, 2014

[2] Garrett Birkhoff, Lattice Theory, American Mathematical Soc., 1967

[3] Kenneth Ireland and Michael Rosen, A Classical Introduction to Modern Number Theory, Springer, 1998

[4] John B. Fraleigh, First Course in Abstract Algebra, Pearson , 2003

[5] Saunders MacLane, Categories for the Working Mathematician, Springer, 1998

[6] P. Shor., Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer., 1994

[7] H. Minkowski, Geometrie der Zahlen, Teubner, 1910

[8] Hanrot, G., Pujol, X. and Stehlé, D., Algorithms for the Shortestand Closest Lattice Vector Problems, In Proceedings of the Thirdinternational conference on Coding and cryptology, 2011

[9] Heinz Bauer, Measure and Integration Theory, de Gruyter, 2001

[10] S.M. Srivastava, A Course on Borel Sets, Springer, 1998

[11] A. Sanjeev and B. Boaz, Computational Complexity, Cambridge University Press, 2009

[12] W. Diffie and M. Hellman, New directions in cryptography, IEEE Transactionson Information Theory, 1976

[13] S. Goldwasser and M. Bellare, Lecture Notes on Cryptography, Massachusetts Institute of Technology, 2008

[14] J. Katz and Y. Lindell, Introduction to Modern Cryptography, Chapman &Hall/CRC Press, 2008

[15] R. Rivest, L. Adleman, and M. Dertouzos, On data banks and privacy homomorphisms, In Foundations of Secure Computation, 1978

[16] R. L. Rivest, A. Shamir, and L. Adleman, A method for obtaining digital signatures and public key cryptosystems, Communications of the ACM, 1978

[17] D. Boneh, E. Goh, and K. Nissim, Evaluating 2-dnf formulas on ciphertexts, In Proceedings of Theory of Cryptography, 2005

[18] O. Goldreich, S. Goldwasser, and S. Halevi. Public-key cryptosystems from lattice reduction problems, In Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology, 1997

[19] C. Gentry, A fully homomorphic encryption scheme. PhD thesis, Stanford University, 2009

[20] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, Introduction to Algorithms., MIT Press, 2001