NATIONAL TECHNICAL UNIVERSITY OF ATHENS

School of Electrical and Computer Engineering
Computer Science Division

DOCTORAL THESIS

Byzantine Fault Tolerance in Autonomous Robots Evacuation Problems

Author:
Ioannis PAPAIOANNOU

Advisor:
Aris PAGOURTZIS

Computation and Reasoning Laboratory

Athens, June 2024

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογισμών

Ανοχή Βυζαντινών Σφαλμάτων σε Προβλήματα Διαφυγής Αυτόνομων Ρομπότ

Διδακτορική Διατριβή του Ιωάννη Παπαϊωάννου

Τριμελής Συμβουλευτική Επιτροπή: Αριστείδης Παγουρτζής (επιβλέπων)
Παναγιώτης Τσανάκας
Δημήτριος Φωτάκης

Εγκρίθηκε από την Επταμελή Εξεταστική Επιτροπή την 6η Ιουνίου 2024

Αριστείδης Παγουρτζής Καθηγητής ΕΜΠ	Παναγιώτης Τσανάκας Καθηγητής ΕΜΠ	Δημήτριος Φωτάκης Καθηγητής ΕΜΠ
Κωνσταντίνος Γεωργίου Αν. Καθηγητής Toronto Metropolitan University	Ευριπίδης Μάρκου Αν. Καθηγητής Πανεπιστήμιο Θεσσαλίας	Νικόλαος Λεονάρδος Επ. Καθηγητής ΕΜΠ
	Ευάγγελος Κρανάκης Καθηγητής Carleton University	

Αθήνα, Ιούνιος 2024

Ιωάννης Ν. Παπαϊωάννου

Διδάκτωρ Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

στους γονείς μου

Περίληψη

Σε αυτή τη διατριβή μελετάμε προβλήματα αναζήτησης και διαφυγής αυτόνομων ρομπότ, δηλαδή καταστάσεις όπου μια ομάδα ρομπότ πρέπει να βρει έναν ή περισσότερους στόχους που βρίσκονται σε άγνωστα σημεία μιας περιοχής. Στην περίπτωση που μας ενδιαφέρει, ο στόχος είναι μια έξοδος και ο στόχος των ρομπότ είναι είτε να να εντοπίσουν την έξοδο (πρόβλημα αναζήτησης) ή να εγκαταλείψουν την περιοχή (πρόβλημα διαφυγής) όσο το δυνατόν γρηγορότερα. Στην μελέτη αυτή, εξετάζουμε την (n,f)-αναζήτηση και την (n,f)-διαφυγή από έναν κύκλο, όπου n ρομπότ συνεργάζονται για να να εντοπίσουν την έξοδο ή να διαφύγουν μέσω της εξόδου και f από αυτά μπορεί να εμφανίσουν σφάλματα. Για την ανάλυση της χειρότερης περίπτωσης των αλγορίθμων μας, θεωρούμε έναν αντίπαλο που επιλέγει τη θέση της εξόδου και τη συμπεριφορά των εσφαλμένων ρομπότ (τις τροχιές τους καθώς και τα μηνύματα που θα μεταδώσουν) με στόχο την μεγιστοποίηση του χρόνου αναζήτησης και ολοκλήρωσης της διαφυγής. Ο αντίπαλος επιλέγει επίσης ποια ρομπότ θα εμφανίσουν σφάλματα. Διερευνώνται δύο διαφορετικά μοντέλα επικοινωνίας για τη διευκόλυνση των αλληλεπιδράσεων μεταξύ των ρομπότ: το ασύρματο μοντέλο όπου τα ρομπότ μπορούν να επικοινωνούν άμεσα ανεξαρτήτως απόστασης και το μοντέλο Face-to-Face που απαιτεί από τα ρομπότ να συναντηθούν ταυτόχρονα στην ίδια τοποθεσία προκειμένου να ανταλλάξουν πληροφορίες. Παρέχουμε βέλτιστους αλγορίθμους για την (n,f)-αναζήτηση σε έναν κύκλο αντιμετωπίζοντας σενάρια που περιλαμβάνουν f σφάλματα συντριβής ή ένα Βυζαντινό σφάλμα. Επεκτείνουμε τη συζήτηση στην διαφυγή από κύκλο υπό ένα και δύο Βυζαντινά σφάλματα και υπό f Βυζαντινά σφάλματα παρουσιάζοντας λεπτομερείς αλγορίθμους και πραγματοποιώντας μια εις βάθος ανάλυση των χρονικών τους απαιτήσεων.

Λέξεις κλειδιά: Αναζήτηση, Διαφυγή, Αυτόνομα Ρομπότ, Ανοχή σε Σφάλματα, Σφάλματα Συντριβής, Βυζαντινά Σφάλματα, Ασύρματη Επικοινωνία, Μη Ασύρματη Επικοινωνία, Κύκλος.

Abstract

This thesis studies search and evacuation problems involving autonomous robots tasked with locating and reaching an exit positioned at an undisclosed point within a specified territory. The primary focus is on the (n,f)-search and (n,f)-evacuation from a unit circle, where n robots operate collectively to discover or evacuate from an exit, despite the presence of up to f potentially faulty units. The problems are framed to challenge the robots against an adversarial setting that strategically places the exit and manipulates the faulty robots' actions — ranging from their movement trajectories to the dissemination of misleading information —to maximize the time required to complete the search or evacuation.

Two models of communication among the robots are considered: the wireless model, which allows instantaneous communication irrespective of distance, and the face-to-face model, which necessitates physical proximity for information exchange. This study develops optimal algorithms for the (n,f)-search on a circle with scenarios involving f crash faults or a single Byzantine fault, extending to algorithms for complex evacuation scenarios under multiple Byzantine faults. These algorithms are analyzed and lower and upper bounds are provided, particularly focusing on the worst-case completion time that is impacted by the adversarial control of faults.

Keywords: Search; Evacuation; Autonomous Robots; Fault Tolerance; Crash Faults; Byzantine Faults; Wireless Communication; Face-to-Face Communication; Circle.

Acknowledgements

Η παρούσα διδακτορική διατριβή δεν θα μπορούσε να είχε πραγματοποιηθεί χωρίς την πολύτιμη συμβολή του επιβλέποντά μου, κ. Άρη Παγουρτζή. Θα ήθελα να τον ευχαριστήσω θερμά για την εμπιστοσύνη που μου έδειξε, την υπομονή του, την αμέριστη συμπαράστασή του και την ελευθερία που μου παρείχε καθ' όλη τη διάρκεια των διδακτορικών μου σπουδών.

Θα ήθελα επίσης να ευχαριστήσω τους καθηγητές του Corelab, κ. Στάθη Ζάχο και κ. Δημήτρη Φωτάκη, για τη στήριξή τους και το ειλικρινές ενδιαφέρον τους όλα αυτά τα χρόνια. Ακόμη, ήταν μεγάλη τιμή για μένα η συμμετοχή των κ. Παναγιώτη Τσανάκα, Κωνσταντίνου Γεωργίου, Ευριπίδη Μάρκου, Νίκου Λεονάρδου και Ευάγγελου Κρανάκη στην εξεταστική επιτροπή του διδακτορικού μου. Η συνεργασία μας υπήρξε καθοριστική.

Θα ήθελα να εκφράσω την ευγνωμοσύνη μου στα μέλη του Corelab: την Τζέλα, τον Σωτήρη, την Αγγέλα, τον Αντώνη, τον Στρατή, τον Δημήτρη, την Pouran, τον Ορέστη, τη Μαριάννα και τη Δανάη. Υπήρξαν ιδανικοί συνεργάτες και φίλοι, και τους ευχαριστώ για όλες τις στιγμές που περάσαμε μαζί.

Θέλω επίσης να ευχαριστήσω τη Δήμητρα, τον Λάμπρο, τον Χάρη, την Αλεξία, τον Γιώργο, τη Βίκυ, τον Στέλιο, την Αγγελική, τη Χρυσή, τη Μαργαρίτα, τη Ζέτα και τη Μυρτώ για τη στήριξη και την κατανόηση που έδειξαν όλα αυτά τα χρόνια. Η παρουσία και η βοήθειά τους ήταν ανεκτίμητη.

Τέλος θα ήθελα να ευχαριστήσω ολόψυχα την οικογένειά μου, τα αδέρφια μου και τους γονείς μου, που δεν σταμάτησαν ποτέ να με στηρίζουν και να είναι δίπλα μου. Δεν θα μπορούσα να πραγματοποιήσω τίποτα χωρίς αυτούς.

Γιάννης Παπαϊωάννου
Αθήνα, Ιούνιος 2024

The current thesis involves studies that resulted from my cooperation with Pourandokht Behrouz, Konstantinos Georgiou, Orestis Konstantinidis, Evangelos Kranakis, Nikos Leonardos, Aris Pagourtzis, and Marianna Spyrakou.

Contents

List of Figures

Extended Abstract (in Greek)

Εισαγωγή

Σε έναν όλο και πιο αυτοματοποιημένο κόσμο η ανάπτυξη αυτόνομων ρομποτικών συστημάτων σε επιχειρήσεις έρευνας και διάσωσης, επιτήρησης και εξερεύνησης γίνεται όλο και πιο διαδεδομένη. Τα συστήματα αυτά προσφέρουν σημαντικά πλεονεκτήματα σε περιβάλλοντα που είναι είτε πολύ επικίνδυνα είτε απρόσιτα για τους ανθρώπους. Τα παραδείγματα περιλαμβάνουν περιοχές που πλήττονται από καταστροφές, εξερευνήσεις σε μεγάλα θαλάσσια βάθη ή στο διάστημα καθώς και πολύπλοκα αστικά περιβάλλοντα. Τα αυτόνομα ρομπότ μπορούν να εκτελούν καθήκοντα όπως ο εντοπισμός επιζώντων σε συντρίμμια, ο εντοπισμός εξόδων σε φλεγόμενα κτίρια ή η εξερεύνηση άγνωστων περιοχών σε άλλους πλανήτες. Αυτά τα σενάρια συχνά περιλαμβάνουν κρίσιμες αποστολές όπου η ταχύτητα εντοπισμού ενός στόχου συσχετίζεται άμεσα με την επιτυχία της επιχείρησης, είτε για τη διάσωση ζωών και την άμεση διαφυγή από μια επικίνδυνη περιοχή είτε για την αποτελεσματική συλλογή δεδομένων.

Η μελέτη των προβλημάτων αναζήτησης και διαφυγής αντιμετωπίζει αυτές τις προκλήσεις με την ανάπτυξη αλγορίθμων, που ελαχιστοποιούν το χρόνο που απαιτείται για μια ομάδα ρομπότ να εντοπίσει και να προσεγγίσει συγκεκριμένους στόχους (ή εξόδους). Η πολυπλοκότητα αυτών των προβλημάτων μεγεθύνεται από διάφορους ρεαλιστικούς περιορισμούς: τα ρομπότ μπορεί να έχουν περιορισμένες δυνατότητες επικοινωνίας (σε ένα περιορισμένο ή ιδιαίτερα πυκνά δομημένο περιβάλλον, π.χ. σε μια σήραγγα), μπορεί να αντιμετωπίσουν σφάλματα και πρέπει να μπορούν να λειτουργούν σε άγνωστα ή δυναμικά μεταβαλλόμενα περιβάλλοντα. Η κατανόηση του τρόπου σχεδιασμού αποτελεσματικών αλγορίθμων υπό τέτοιες συνθήκες είναι ζωτικής σημασίας για την ενίσχυση της αξιοπιστίας και της αποτελεσματικότητας των αυτόνομων ρομποτικών συστημάτων σε απρόβλεπτα και συχνά επικίνδυνα περιβάλλοντα.

Μια εκτενώς μελετημένη οικογένεια προβλημάτων σε αυτήν την επιστημονική περιοχή αφορά καταστάσεις όπου μια ομάδα ρομπότ πρέπει να βρει έναν ή περισσότερους στόχους που βρίσκονται σε άγνωστα σημεία μιας περιοχής. Το πρόβλημα θεωρείται ιδιαίτερα σημαντικό στη ρομποτική και στην επιστήμη των υπολογιστών ενώ τις τελευταίες δεκαετίες έχουν διατυπωθεί αρκετά αλγοριθμικά αποτελέσματα. Σε μια ενδιαφέρουσα περίπτωση ο στόχος είναι μια έξοδος και ο σκοπός των ρομπότ είναι είτε να εντοπίσουν την έξοδο (πρόβλημα αναζήτησης) είτε να εγκαταλείψουν την περιοχή μέσω της εξόδου (πρόβλημα διαφυγής), όσο το δυνατόν γρηγορότερα.

Θέση και κίνηση των ρομπότ

Στην εργασία μας θεωρούμε ένα σύνολο από n ρομπότ (συχνά αναφερόμαστε σε αυτά ως πράκτορες) που συμβολίζονται ως $a_0, a_1, \ldots, a_{n-1}, f$ εκ των οποίων είναι ελαττωματικά, όλα αρχικά τοποθετημένα στο κέντρο ενός κύκλου μοναδιαίας ακτίνας. Η έξοδος βρίσκεται στον *μοναδιαίο κύκλο*, που είναι η περιφέρεια του δίσκου. Τα ρομπότ έχουν την ικανότητα να αντιλαμβάνονται την περίμετρο του δίσκου και να εντοπίζουν την έξοδο αν τύχει να βρεθούν στην θέση της. Στους αλγορίθμους μας όλοι οι μη εσφαλμένοι (honest) πράκτορες κινούνται με τη μέγιστη ταχύτητα 1, επομένως σε κάθε χρονική στιγμή, όλοι οι πράκτορες γνωρίζουν τη θέση κάθε πράκτορα που ακολουθεί το πρωτόκολλο. Υποθέτουμε ότι τα ρομπότ είναι εξοπλισμένα με αισθητήρες για την ακριβή μέτρηση της απόστασης κατά τη διάρκεια της κίνησης. Έστω ότι το $\vartheta := 2\pi/n$ αντιπροσωπεύει μια γωνία και κάθε ρομπότ a_k κινείται κατά μήκος μιας ακτίνας προς το σημείο $k\vartheta$ στην περίμετρο του μοναδιαίου κύκλου. Το τόξο $[k\vartheta, (k+1)\vartheta)$ ορίζεται ως τομέας S_k. Μετά από 1 χρονική μονάδα, το ρομπότ a_k τοποθετείται στην αρχή του τομέα S_k, ολοκληρώνοντας την αναζήτηση εντός αυτού του τομέα σε χρόνο $1 + \vartheta$ ενώ κινείται αριστερόστροφα (ccw). Συχνά αναφερόμαστε στο χρόνο ϑ, το χρόνο δηλαδή που χρειάζεται ένα ρομπότ για να ψάξει πλήρως έναν τομέα ως γύρο *(round)*.

Υποθέτουμε καθ' όλη τη διάρκεια της εκτέλεσης των αλγορίθμων ότι κάθε φορά που ένας μη εσφαλμένος πράκτορας βρίσκει την έξοδο, το ανακοινώνει και κάθε φορά που αντιλαμβάνεται ότι η ανακοίνωση ενός άλλου πράκτορα είναι εσφαλμένη, ανακοινώνει την διαφωνία του σε όλους.

Μοντέλα επικοινωνίας

Σε αυτή την εργασία διερευνώνται δύο διαφορετικά μοντέλα επικοινωνίας για τη διευκόλυνση των αλληλεπιδράσεων μεταξύ των ρομπότ:

- **Μοντέλο ασύρματης επικοινωνίας (Wireless)**: Στο μοντέλο ασύρματης επικοινωνίας τα ρομπότ μπορούν να επικοινωνούν άμεσα ανεξαρτήτως απόστασης. Τα μηνύματα που ανταλλάσσονται μεταξύ των ρομπότ μεταφέρουν διάφορες πληροφορίες, όπως τοποθεσία, ανακάλυψη εξόδου, διανυθείσες αποστάσεις κ.α. Κάθε μήνυμα έχει ένα μοναδικό αναγνωριστικό αποστολέα (ID) που παραμένει αμετάβλητο. Αναλύοντας αυτά τα μηνύματα, τα ρομπότ μπορούν να προσδιορίσουν τις σχετικές τους θέσεις.

- **Μοντέλο μη ασύρματης επικοινωνίας (Face-to-Face)**: Γνωστό και ως τοπικό μοντέλο, αυτό το μοντέλο επικοινωνίας απαιτεί από τα ρομπότ να βρίσκονται φυσικά στην ίδια τοποθεσία την ίδια χρονική στιγμή, προκειμένου να ανταλλάξουν πληροφορίες.

Δύο άλλα μοντέλα επικοινωνίας που χρησιμοποιούνται ευρέως στην εξερεύνηση γραφημάτων είναι το μοντέλο *βότσαλο* (pebble) και το μοντέλο *πίνακας* (whiteboard). Στο μοντέλο pebble τα ρομπότ είναι εξοπλισμένα με ένα ή περισσότερα βότσαλα, κινητά αντικείμενα που προσδιορίζουν μοναδικά έναν κόμβο ή μια ακμή και περιέχουν ένα μόνο bit πληροφορίας. Τα ρομπότ χρησιμοποιούν τα βότσαλα ως συσκευές επικοινωνίας προκειμένου να εξερευνήσουν το γράφημα [18, 53, 49, 19]. Για την εξερεύνηση γραφημάτων με ρομπότ χωρίς μνήμη εφαρμόζεται το μοντέλο επικοινωνίας του πίνακα. Οι πίνακες οι οποίοι μπορεί να είναι κινητά ή ακίνητα αντικείμενα, έχουν επαρκή μνήμη για την ανταλλαγή πληροφοριών μεταξύ των ρομπότ [52, 25, 44, 51].

Σφάλματα

Στη εργασία αυτή λαμβάνουμε υπόψη δύο τύπους ελαττωματικών συμπεριφορών που παρουσιάζουν τα ρομπότ:

- **Σφάλμα συντριβής (Crash Fault)**: Ένα ρομπότ που αντιμετωπίζει σφάλμα συντριβής σταματάει απότομα τη λειτουργία του. Δεν μετακινείται άλλο και σταματά οποιαδήποτε επικοινωνία.

- **Βυζαντινό σφάλμα (Byzantine Fault)**: Ένα ρομπότ που εμφανίζει Βυζαντινό σφάλμα επιδίδεται σε κακόβουλες δραστηριότητες, συμπεριλαμβανομένων της σκόπιμης αλλαγής της τροχιάς του και της χειραγώγησης πληροφοριών για να μπερδέψει τα μη εσφαλμένα ρομπότ. Επιπλέον, ένα ρομπότ με Βυζαντινό σφάλμα μπορεί να μιμηθεί τη συμπεριφορά ενός ρομπότ με σφάλμα συντριβής.

Αντίπαλος

Για την ανάλυση της χειρότερης περίπτωσης των αλγορίθμων μας θεωρούμε έναν αντίπαλο (adversary) που επιλέγει τη θέση της εξόδου και τη συμπεριφορά των κακόβουλων ρομπότ (τις τροχιές του καθώς και τα μηνύματα που θα μεταδώσουν) ώστε να μεγιστοποιήσει τον προκύπτοντα χρόνο αναζήτησης και ολοκλήρωσης της διαφυγής. Ο αντίπαλος επιλέγει επίσης ποια ρομπότ είναι εσφαλμένα.

Προβλήματα Αναζήτησης και Διαφυγής

Στο πλαίσιο του προβλήματος (n, f)-διαφυγής σε κύκλο με μοναδιαία ακτίνα ορίζονται δύο προβλήματα, λαμβάνοντας υπόψη συνολικά n ρομπότ, με f από αυτά να είναι δυνητικά εσφαλμένα:

- **Πρόβλημα αναζήτησης**
  Στην περίπτωση ενός συνόλου n ρομπότ, με f από αυτά να είναι εσφαλμένα, εισάγουμε τον συμβολισμό $S(n, f)$ για να δηλώσουμε τον χρόνο που απαιτείται για την επιτυχή επίλυση του προβλήματος αναζήτησης. Αυτό αντιπροσωπεύει τη διάρκεια που χρειάζεται για τα μη εσφαλμένα ρομπότ να φτάσουν στην έξοδο και να διασφαλίσουν ότι όλα τα μη εσφαλμένα ρομπότ διαθέτουν αδιαμφισβήτητη γνώση της θέσης της εξόδου. Αυτή η συνεργατική προσπάθεια των ρομπότ να εντοπίσουν και να καθορίσουν την ακριβή θέση της εξόδου αναφέρεται συνήθως ως ομαδική αναζήτηση.

- **Πρόβλημα διαφυγής**
  Συμβολίζεται ως $E(n, f)$ περιλαμβάνει n ρομπότ, συμπεριλαμβανομένων των f εσφαλμένων, και αποσκοπεί στον προσδιορισμό του χρόνου που απαιτείται για μια επιτυχή διαφυγή στην οποία ένα μη ελαττωματικό ρομπότ ανακαλύπτει την έξοδο και όλα τα μη ελαττωματικά ρομπότ πρέπει να φτάσουν με ασφάλεια στη θέση της εξόδου. Είναι σημαντικό να σημειωθεί ότι ο χρόνος διαφυγής $E(n, f)$ είναι εγγενώς μεγαλύτερος ή ίσος με το χρόνο που απαιτείται για την εύρεση της εξόδου $S(n, f)$, καθώς η εύρεση της εξόδου αποτελεί προϋπόθεση για την επιτυχή διαφυγή.

Βιβλιογραφική Ανασκόπηση

Τα προβλήματα που παρουσιάσαμε παραπάνω ξεκίνησαν να μελετώνται αρχικά σε περιβάλλον ευθείας. Το πρόβλημα της αναζήτησης γραμμής, το οποίο επικεντρώνεται σε έναν μεμονωμένο κινητό πράκτορα, που αναζητά μια άγνωστη έξοδο σε μια ευθεία, έχει μελετηθεί εκτενώς στη βιβλιογραφία. Οι πρωτοποριακές εργασίες των Beck και Bellman [11, 17], Baeza-Yates et al. [6], έχουν θέσει τις βάσεις της έρευνας διερευνώντας τόσο στοχαστικά όσο και ντετερμινιστικά περιβάλλοντα. Αξιοσημείωτες είναι οι συνεισφορές των Ahlswede και Wegener [2], Alpern και Gal [5], Stone [84], που έχουν οδηγήσει σε σημαντικές δημοσιεύσεις και βιβλία. Περαιτέρω έρευνες έχουν διερευνήσει διάφορους αλγορίθμους αναζήτησης γραμμής, συμπεριλαμβανομένων τυχαιοποιημένων προσεγγίσεων [68] και εκτιμήσεων του κόστους στροφής (turn cost) [46]. Επιπλέον, το πρόβλημα της αναζήτησης γραμμής επεκτάθηκε, ώστε να εξετάζει και την παρουσία εσφαλμένων ρομπότ, οδηγώντας σε ενδιαφέρουσες έρευνες σχετικά με τα ρομπότ που παρουσιάζουν σφάλματα συντριβής [40] και τα ρομπότ με Βυζαντινά σφάλματα [38]. Αυτές οι μελέτες ρίχνουν φως στις προκλήσεις και τις στρατηγικές που εμπλέκονται στην επίλυση του προβλήματος αναζήτησης γραμμής όταν υπάρχουν ρομποτικά σφάλματα.

Αξίζει να σημειωθεί ότι το πρόβλημα διαφυγής μοιράζεται μια στενή σχέση με το πρόβλημα αναζήτησης, με τα δύο προβλήματα να επικαλύπτονται όταν εμπλέκεται μόνο ένα ρομπότ. Η μελέτη της διαφυγής σε κυκλικές τοπολογίες ξεκίνησε με την πρωτοποριακή εργασία [32], όπου οι συγγραφείς διερεύνησαν τόσο το ασύρματο όσο και το μη ασύρματο μοντέλο επικοινωνίας. Η εργασία αυτή έθεσε τα θεμέλια για τη μετέπειτα έρευνα σχετικά με την διαφυγή σε περιβάλλον κύκλου, περιλαμβάνοντας διάφορα σενάρια. Για μια ολοκληρωμένη επισκόπηση της περιοχής συνιστούμε να ανατρέξετε στο [37].

Η παρούσα διδακτορική διατριβή συνδέεται στενά στο πλαίσιο των προβλημάτων διαφυγής σε κυκλικές τοπολογίες υπό την παρουσία σφαλμάτων με την εργασία [33]. Στη μελέτη τους οι συγγραφείς επικεντρώθηκαν στο πρόβλημα που περιλαμβάνει τρία ρομπότ, εκ των οποίων το ένα είναι εσφαλμένο. Παρείχαν άνω και κάτω φράγματα τόσο για τα σενάρια που περιλαμβάνουν ρομπότ με σφάλματα συντριβής, όσο και για τις περιπτώσεις Βυζαντινού σφάλματος.

Επισκόπηση Κεφαλαίων

Στο Κεφάλαιο 1 περιγράφεται αναλυτικά το μοντέλο των προβλημάτων που μελετούμε καθώς και παρουσιάζεται εκτενώς η σχετική βιβλιογραφία.

Στο Κεφάλαιο 2 εξετάζουμε το πρόβλημα (n,f)-*αναζήτησης σε κύκλο*, ένα πρόβλημα αναζήτησης μιας κρυφής εξόδου σε έναν κύκλο μοναδιαίας ακτίνας για $n > 1$ ρομπότ, f εκ των οποίων είναι εσφαλμένα. Όλα τα ρομπότ ξεκινούν από το κέντρο του κύκλου και μπορούν να κινηθούν οπουδήποτε με μέγιστη ταχύτητα 1. Κατά τη διάρκεια της αναζήτησης, τα ρομπότ μπορούν να επικοινωνούν ασύρματα. Όλα τα μηνύματα που μεταδίδονται από όλα τα ρομπότ επισημαίνονται με τα μοναδικά αναγνωριστικά των ρομπότ, τα οποία δεν μπορούν να αλλοιωθούν. Η αναζήτηση θεωρείται ολοκληρωμένη όταν η έξοδος βρεθεί από ένα μη εσφαλμένο ρομπότ (το οποίο πρέπει να επισκεφθεί τη θέση της εξόδου) και τα υπόλοιπα μη εσφαλμένα ρομπότ γνωρίζουν τη σωστή θέση της εξόδου. Μελετάμε και τα δύο μοντέλα εσφαλμένων ρομπότ. Όταν υπάρχουν μόνο ρομπότ με σφάλμα συντριβής, παρέχουμε βέλτιστους αλγορίθμους για το πρόβλημα αναζήτησης (n,f), με βέλτιστο χρόνο ολοκλήρωσης της αναζήτησης στη χειρότερη περίπτωση $1 + \frac{(f+1)2\pi}{n}$. Η κύρια τεχνική συνεισφορά μας αφορά βέλτιστους αλγορίθμους για την $(n, 1)$-αναζήτηση με ένα ρομπότ με Βυζαντινό σφάλμα, ελαχιστοποιώντας τον χρόνο ολοκλήρωσης της αναζήτησης στη χειρότερη περίπτωση, ο οποίος ισούται με $1 + \frac{4\pi}{n}$. Παρουσιάζουμε επίσης έναν αλγόριθμο για τη μικτή περίπτωση, με ένα Βυζαντινό και $f - 1$ σφάλματα συντριβής, με χρόνο ολοκλήρωσης της αναζήτησης στην χειρότερη περίπτωση $1 + \frac{2\pi}{n}f + 2\sin\frac{2\pi}{n}$.

Στο Κεφάλαιο 3 εξετάζουμε το πρόβλημα (n,f)-*διαφυγής* από ένα κύκλο, ένα πρόβλημα στο οποίο $n > 1$ ρομπότ, f εκ των οποίων είναι εσφαλμένα, προσπαθούν να διαφύγουν μέσω μιας κρυφής εξόδου που βρίσκεται στην περίμετρο ενός μοναδιαίου κύκλου. Για το ασύρματο μοντέλο επικοινωνίας αποδεικνύουμε αρχικά ένα κάτω φράγμα $1 + \frac{4\pi}{n} + 2\sin(\frac{\pi}{2} - \frac{\pi}{n})$ για την περίπτωση ενός εσφαλμένου ρομπότ. Στη συνέχεια μελετάμε την περίπτωση με δύο Βυζαντινά ρομπότ και παρέχουμε έναν αλγόριθμο που επιτυγχάνει διαφυγή σε χρόνο το πολύ $3 + \frac{6\pi}{n} + \delta(n)$, όπου $\delta(n)$ είναι μια φθίνουσα συνάρτηση με μέγιστη τιμή $\delta(4) = 0.5687$, που μηδενίζεται για $n \geq 9$. Για το μη ασύρματο μοντέλο παρέχουμε ένα άνω φράγμα $3+(f+1)\frac{2\pi}{n}$ για την διαφυγή n ρομπότ υπό την παρουσία σφάλματων συντιβής, ένα άνω φράγμα $3 + \frac{4\pi}{n} + 2\sin\frac{\pi}{n}$ για την διαφυγή στην περίπτωση ενός Βυζαντινού ρομπότ και ένα άνω φράγμα $3 + \frac{6\pi}{n} + 2\sin\frac{2\pi}{n}$ στην περίπτωση δύο Βυζαντινών ρομπότ.

Τέλος στο Κεφάλαιο 4 μελετούμε το πρόβλημα της (n, f)-διαφυγής σε κύκλο υπό οποιοδήποτε αριθμό Βυζαντινών σφαλμάτων. Παρουσιάζουμε αλγορίθμους τόσο για το μοντέλο ασύρματης επικοινωνίας όσο και για το μοντέλο μη ασύρματης επικοινωνίας. Αναλύουμε τις χρονικές απαιτήσεις αυτών των αλγορίθμων και προσδιορίζουμε άνω φράγματα για την απόδοσή τους.

Αντιστοιχία Όρων

Μετάφραση	**Αγγλικός όρος**
αναζήτηση	search
αντίπαλος	adversary
άνω φράγμα	upper bound
αριστερόστροφα	ccw
Βυζαντινό σφάλμα	Byzantine fault
βέλτιστος	optimal
γύρος	round
διαφυγή	evacuation
κάτω φράγμα	lower bound
μη εσφαλμένο ρομπότ	honest robot
πράκτορας	agent
σφάλμα	fault
σφάλμα συντριβής	crash fault
τομέας	sector
τόξο	arc
χορδή	chord

Chapter 1

Introduction

In an increasingly automated world, the deployment of autonomous robotic systems in search and rescue, surveillance, and exploration operations is becoming more prevalent. Such systems offer significant advantages in environments that are either too hazardous or inaccessible for humans. Examples include disaster-stricken areas, deep-sea and space explorations, and complex urban settings. Autonomous robots can perform tasks such as locating survivors in rubble, identifying exits in burning buildings, or exploring unknown territories on other planets. These scenarios often involve critical missions where the speed of locating an objective directly correlates with the success of the operation, be it saving lives, efficiently gathering data, or exiting a dangerous area promptly.

The study of search and evacuation problems addresses these challenges by developing algorithms that minimize the time required for a group of robots to find and reach specific targets or exits. The complexity of these problems is magnified by several realistic constraints: robots can have limited communication capabilities (in a confined or a highly structured setting such as a tunnel), they may encounter faults, and they must operate in unknown or dynamically changing environments. Understanding how to design efficient algorithms under such conditions is crucial for enhancing the reliability and effectiveness of autonomous robotic systems in unpredictable and often perilous environments.

1.1 Preliminaries and Notation

An extensively studied family of problems in mobile agent computing concerns situations where a group of robots needs to find one or more targets that are located in unknown points of a territory. The problem is considered particularly important in robotics and computer science and a number of algorithmic and hardness results have been developed over the last few decades. In a particular case of interest, the target is an *exit* and the goal of the robots is either to locate the exit (*search* problem) or to leave the territory (*evacuation* problem), as fast as possible.

1.1.1 Location and Movement (Robot Trajectories)

In our work, we consider a set of n robots denoted as $a_0, a_1, \ldots, a_{n-1}$, f of which are faulty, all initially located on the center of a unit radius circle. The exit is located on the *unit circle*, which is the circumference of the disk. Robots possess the capability to perceive the perimeter of the disk and detect the exit if they happen to be in close proximity to it. In our algorithms, all honest agents move at the maximum speed 1, therefore at each time point, all agents know the location of every agent that follows the protocol. We assume that robots are equipped with pedometers for accurate distance measurement during movement. Let $\partial := 2\pi/n$ represent an angle, and each robot a_k moves along a radius to the point $k\partial$ on the perimeter of the unit circle. The arc $[k\partial, (k+1)\partial)$ is defined as sector S_k. After 1 time unit, robot a_k positions itself at the beginning of sector S_k, completing the search within this sector in time $1 + \partial$ while moving counterclockwise (ccw). We may refer to time ∂, the time that a robot needs to completely search a sector as a *round*.

It is assumed throughout that whenever an honest agent finds the exit it announces this fact, and whenever it realizes that an announcement of another agent is faulty it also announces this to everybody.

1.1.2 Communication Models

In this work, two distinct communication models are explored to facilitate interactions among the robots:

- **Wireless Model**: In the wireless communication model, robots can communicate instantly regardless of distance. Messages exchanged between robots carry various information such as locations, exit discovery, distances traveled, and more. Each message has a unique sender identifier that remains unchanged throughout the communication process. By analyzing these messages, robots can determine their relative positions.

- **Face-to-Face Model**: Also known as the non-wireless or local model, this communication model requires robots to physically gather in the same location simultaneously in order to exchange information. Unlike the wireless model where robots can communicate regardless of their distance, the face-to-face model necessitates direct physical interaction for information exchange.

Two other communication models, widely used in the exploration of graph environments are the *pebble* model and the *whiteboard* model. In the pebble model, robots are equipped with one or more pebbles (tokens), movable objects that uniquely identify a node or an edge and contain a single bit of information. Robots use the pebbles as communication devices in order to explore the graph [18, 53, 49, 19]. To explore graphs with memoryless robots, the whiteboard communication model is applied. Whiteboards, which can be movable or immovable objects, have sufficient memory for robots to exchange information [52, 25, 44, 51].

1.1.3 Fault Types

In our study, we take into account two types of faulty behaviors exhibited by the robots:

- **Crash Faults**: A robot experiencing a crash fault abruptly stops functioning, and becomes unresponsive, resulting in a complete breakdown of message communication.
- **Byzantine Faults**: A robot exhibiting Byzantine behavior engages in malicious activities including deliberately altering its trajectory and manipulating information to confuse the honest (non-faulty) robots. Additionally, a Byzantine robot can mimic the behavior of a crash-faulty robot.

1.1.4 Adversary

For the worst-case analysis of our algorithms, we consider an adversary who selects the location of the exit and the behaviour of the malicious robots (its trajectories as well as the messages they will broadcast) to maximize the resulting search and evacuation completion time. The adversary also chooses which robots are faulty, adding to the challenge.

1.1.5 Search and Evacuation Problems

Within the context of the (n,f)-evacuation problem on a circle with unit radius, two problems are defined, considering a total of n robots, with f of them being potentially faulty:

- **Search Problem**
  In the case of a total of n robots, with f of them being faulty, we introduce the notation $S(n,f)$ to denote the time required to successfully solve the search problem. This represents the duration it takes for the non-faulty robots to reach the exit and ensure that all honest robots possess undeniable knowledge of the location of the exit. This collaborative effort of the robots to locate and establish the precise position of the exit is commonly referred to as group search.

- **Evacuation Problem**
  Denoted as $E(n,f)$ involves n robots, including f faulty ones, and aims to determine the time required for a successful evacuation. In a complete evacuation, a non-faulty robot discovers the exit, and all non-faulty robots must safely reach the exit's location. It is important to note that the evacuation time $E(n,f)$ is inherently greater than or equal to the time required to find the exit $S(n,f)$, as finding the exit is a prerequisite for a successful evacuation.

1.1.6 Symmetric-Persistent algorithms.

As defined by Czyzowicz et al. [33], symmetric-persistent algorithms are a family of natural algorithms that force all robots to immediately go to the disk perimeter and only allow a robot to stop its exploration of the assigned sector if it receives information about the exit. Symmetric-persistent algorithms force all the robots to move in the same direction, either clockwise or counterclockwise.

1.2 Related Work

1.2.1 Search on a Line

There has been extensive literature on line search starting with the seminal papers of Beck and Bellman [11, 17] and Baeza-Yates et.al. [7].

Both cases are concerned with linear search: a single mobile agent searching for an exit placed at an unknown location on an infinite line; a problem also known as the cow-path problem. In the former case, the setting is stochastic, and in the latter deterministic.

Beck et al. furthered their contributions to understanding the linear search problem by examining various assumptions and conditions related to the searcher's strategy and the distribution of the target's location. In their study [12], they extend the basic model by considering different a priori distributions (uniform, triangular, and normal) and the impact these have on the optimal search strategy. In a subsequent work [13], they explore the implications of increasing the cost function associated with distance, suggesting that as search time increases, so does the penalty for continued searching. They also examine how this changes the strategy of the searcher when known distributions are involved. A game-theoretic approach is introduced [14] when the probability distribution of the target's location is unknown. A minmax solution is proposed to determine robust strategies against an adversary who may choose any distribution to maximize the searcher's expected loss. In a later work [15], the authors revisit the problem with a focus on nonlinear cost functions, demonstrating that the general strategies developed under linear assumptions hold even under more complex cost scenarios.

This line of research continued by several authors and culminated with the seminal books by Ahlswede and Wegener [2], Alpern and Gal [5], and Stone [84].

Czyzowicz et al. [42], consider a robot whose speed varies due to factors like travel direction or terrain profile (e.g. when the line is inclined, the robot can accelerate). In this work they design search algorithms that achieve good competitive ratios for the time spent by the robot to complete its search versus the time spent by an omniscient robot that knows the location of the target.

Several other models for line search algorithms were subsequently investigated. Demaine et al. [46], extends the classic linear-search problem by incorporating a directional change cost, d, into the search strategy. The proposed strategy guarantees finding an object on a line, at an unknown distance OPT from the searcher's starting point, with a total cost of no more than $9 \cdot \mathrm{OPT} + 2d$, which has the optimal competitive ratio 9 (as was first shown in [14]) with respect to OPT plus the minimum corresponding additive term. Their work includes solving an infinite linear program through a series of approximating finite programs to derive upper and lower bounds, leading to a proof of optimality for the search cost. This approach is also applied to the "star search" (first solved by Gal [55]), or a variant of the cow-path problem, where an object is hidden along one of several rays emanating from a point. Here, a tight competitive ratio formula involving m, the number of rays, is derived.

Fuchs et al. [54], investigated the online matching problem on a line, where requests must be matched to a set of points on a real line in an online fashion. It disproves a previous conjecture suggesting that a competitive ratio of 9 could be achieved for this problem, similar to the "cow path" problem, where an optimal online algorithm with a competitive ratio of 9 exists. Instead, the paper establishes that no online algorithm can achieve a competitive ratio strictly less than 9.001 for the online matching problem.

Kao et al. [68], introduce the first randomized algorithm for the co-path problem. Here, the cost function is unique as it considers the distance traveled between queries, which is more applicable to real-world problems, particularly in robotics. Previously, the problem was addressed using deterministic algorithms with a known optimal competitive ratio. However, this paper's randomized algorithm shows significant improvement, particularly for the case of two paths ($w = 2$), achieving a competitive ratio of approximately 4.5911, which is nearly twice as efficient as the best possible deterministic algorithms. Their work also discusses the growth of the competitive ratio in relation to the number of paths w. In a subsequent work [67], they extend the classic cow-path problem to the case in which goal locations are selected according to one of a set of possible known probability distributions and present a polynomial-time linear programming algorithm for this problem, with potential applicability to other search problems as well.

Chrobak et al. [27], address the "group search problem" or "evacuation problem," where multiple mobile entities (MEs) begin at a common origin on a line and must locate and simultaneously reach a destination situated at an unknown distance either to the left or right of the origin. The main objective is to minimize the time required for all entities to reach this destination. This problem extends the "cow-path problem," which considers a single entity and has established that the minimum search time in the worst case is $9d - o(d)$ where d is the distance to the target. The authors demonstrate that, contrary to what might be expected, increasing the number of MEs does not reduce the minimum search time needed; it remains at $9d - o(d)$, even for k MEs. They explore scenarios with two MEs moving at different speeds, showing that if the slower ME moves at least 1/3 the speed of the faster, the $9d$ time can still be achieved. The paper situates this problem within the broader context of search and rendezvous problems, highlighting how varying the speed of the MEs and their ability to communicate impacts the strategies and outcomes. Extending on this work, Bambas et al. [9], complement the case when the slower robot's speed is at least one-third that of the faster robot. In cases where the faster robot's speed is 1 and the slower robot's speed is greater than approximately 0.123, this work finds that wireless communication can significantly enhance search efficiency. However, beyond this speed difference, wireless communication offers no advantage over the need for robots to meet to exchange information.

Gal [56], addresses the asymmetric rendezvous problem on a line, initially introduced by Alpern [3]. In this problem, two individuals, placed randomly in a known search region, aim to find each other by moving at unit speed. Gal establishes that in a two-player scenario, it is never optimal for one player to remain stationary, highlighting the importance of both players actively moving to reduce the time to meet. Gal then extends this analysis to consider the meeting time in an n-player scenario, demonstrating an asymptotic behavior of $n/2 + O(\log n)$ in the worst case. A later work by Alpern and Beck [4], shows that the asymmetric rendezvous problem on a line (ARSPL) is strategically equivalent to a new problem they introduce, the double linear search problem (DLSP), where an object is placed equiprobably on one of two lines, and equiprobably at positions $\pm d$. A searcher is placed at the origin of each of these lines. The two searchers move with a combined speed of one, to minimize the expected time before one of them finds the object. The authors solve DLSP (and hence the ARSPL) for the case where the distance d is drawn from a known cumulative probability distribution G, convex on its support. Kan et al. [64], improved the bounds of the symmetric rendezvous search problem on the line using Markov chain theory and mathematical programming theory.

Spieser et al. [83], introduce the "Cow-Path Game," a variant of the competitive vehicle routing problem, exploring the strategic decision-making processes in multi-vehicle systems. Specifically, it focuses on scenarios where self-interested, mobile agents (illustrated as cows in a theoretical model) compete to locate a stationary target distributed on a ring. This model simulates real-world competitive environments, such as taxi drivers searching for fares in urban settings or shipwreck recovery boats seeking treasure. In the game-theoretic approach detailed in the study, each agent bases their search strategy not only on their position and available information but also on the actions of competing agents. This approach leads to the development of strategies where agents may adjust their paths in response to the movements of others, aiming to maximize their own chances of success. The paper extends the analysis from a single-agent scenario to a competitive multi-agent context, highlighting the transition from cooperative search strategies to competitive ones.

1.2.2 Search on a Circle

The circle search model (considered in our work) for n non-faulty robots was introduced as an evacuation problem (completion time with respect to the last finder of the hidden exit) by Czyzowicz et al. [32] and analyzed in both the wireless and face-to-face communication models. This paper addresses the evacuation problem for a team of n mobile robots placed at the center of a circular disk with an unknown exit on its boundary. The robots, which share the same maximum speed, aim to locate and exit through this point, communicating amongst themselves to optimize the evacuation time. The paper presents algorithms and establishes bounds for $n = 2$ and $n = 3$ robots. Additionally, the paper derives nearly tight asymptotic bounds on the relationship between evacuation time and team size for large n. The results in detail appear in Table 1.1. In a later work, Czyzowicz et al. [39] refined the bounds of [32] in the case of two robots in the face-to-face communication model, leveraging a forced meeting strategy to streamline evacuation paths. The new upper bound is ~ 5.628, while the lower bound is now ~ 5.255.

n	Communication	Upper bound	Lower bound
n=2	face-to-face	~ 5.74	~ 5.199
	wireless	~ 4.83	~ 4.83
n=3	face-to-face	~ 5.09	~ 4.519
	wireless	~ 4.22	~ 4.159
large n	face-to-face	$3 + \frac{2\pi}{n}$	$3 + \frac{2\pi}{n} - O(n^{-2})$
	wireless	$3 + \frac{\pi}{n} + O(n^{-4/3}$	$3 + \frac{\pi}{n}$

TABLE 1.1: Results presented in [32]

Pattanayak et al. [82], investigate the evacuation problem involving two robots tasked with locating and exiting through two unidentified exits spaced a distance d apart on the perimeter of a circle and considering wireless and face-to-face communication. They consider both labeled and unlabeled exits, showing that labeled exits consistently result in faster evacuation times.

Brandt et al. [23], further investigate the evacuation problem for two robots, under the face-to-face communication model. This work introduces a new algorithm that omits the forced meeting strategy from [39], which had achieved an evacuation time of 5.628. This revised algorithm improves upon that time, achieving an upper bound of 5.625. For the class of algorithms with exactly one symmetric detour per robot, their numerical simulations suggest that this bound is optimal. Criteria, in order to identify potential worst-case exit placements, are introduced and used to simplify the analysis of evacuation algorithms. This work also discusses how evacuation time for a fixed algorithm and exit placement typically corresponds to complex equations that lack closed-form solutions, complicating analysis. The new criteria help mitigate these difficulties. A later work by Disser et al. [48], introduces a second detour through the interior of the disk, aiming to balance the evacuation time across different exit placements, protecting against the worst-case scenario. The new algorithm avoids forced meetings, allowing for independent movement through the disk's interior. That approach leads to an improved evacuation time of 5.6234.

Lamprou et al. [73], investigate the evacuation problem for two robots, under the wireless communication model. Robots can communicate instantaneously, allowing for coordination once one robot locates the exit. Their work introduces and analyzes strategies for the scenario where the robots have different speeds offering insights into the relationship between evacuation time and the robots' speed ratios.

Chuangpishit et al. [28] present a new framework for studying the evacuation problem of two robots in the face-to-face model from both worst-case and average-case perspectives, introducing new algorithms that balance these metrics for practical applications such as search-and-rescue operations. The paper proposes new algorithms that optimize the average-case evacuation time while ensuring the worst-case time remains bounded. These algorithms offer a continuous Pareto frontier, addressing the multi-objective nature of minimizing both average and worst-case evacuation times. The new algorithms outperform existing strategies in the multi-objective context, particularly improving upon algorithms introduced by Czyzowicz et al. [32].

1.2.3 Faulty Searchers

Fault tolerance in distributed computing has been the subject of extensive research [72, 65, 76]. An interesting variant of the linear search mentioned above involves faulty robots. The two main papers in this line of research are [41] for crash-faulty robots and [38] for Byzantine-faulty robots.

Czyzowicz et al. [41], address the problem of searching for a target on a line using multiple robots, some of which may be faulty. They aim to minimize the competitive ratio, which is the worst-case ratio of the arrival time of the first reliable robot at the target to the distance from the start to the target. They introduce a new class of algorithms called proportional schedule algorithms and provide specific algorithms for any combination of n robots and f faulty units. Their results show that if $n \geq 2f+2$, a simple algorithm achieves a competitive ratio of 1. For cases where $f < n < 2f+2$, they develop algorithms with detailed competitive ratios based on a formula. For the specific case where $n = f+1$, the algorithm is shown to be optimal with a competitive ratio of 9, matching known bounds for a single robot. When $n = 2f + 1$, the algorithm's competitive ratio approaches 3, which they prove to be optimal. This result fills a gap in the existing literature by providing lower bounds for situations where $n \geq 3$, matching the best known upper bounds for these cases.

Czyzowicz et al. [38], focus on fault-tolerant parallel search by n robots on an infinite line, where f of these robots may exhibit Byzantine faults (failing to report a found target or making false claims about its discovery). Despite these challenges, the objective is to develop algorithms that minimize the time to locate a target at a distance d from the origin, ensuring that only non-faulty robots verify the target's discovery. The authors present several algorithms optimized for different ratios of faulty to total robots ($\frac{f}{n}$) and establish corresponding lower bounds on the search time. These algorithms are proven to be optimal for certain densities of faulty robots. For cases where n is greater than or equal to $2f + 2$, a simple algorithm achieves a competitive ratio of 1, signifying immediate discovery of the target by a non-faulty robot at its actual distance. For cases where f is less than n but greater than $2f + 2$, they introduce proportional schedule algorithms. These algorithms offer a competitive ratio that improves as the number of robots n increases, approaching an optimal ratio of 3 as n approaches infinity, closely aligning with theoretical lower bounds. In a later work, Kupavskii et al. [71] improve the bounds for crash-faulty robots (and as a result also for the Byzantine ones).

Directly related to our current work is [33]. In this paper, Czyzowicz et al. investigate the evacuation of three robots in the presence of one faulty, either crash or Byzantine robot. The robots must locate and reach an exit placed at an unknown location on the perimeter, communicating wirelessly throughout the process. The study's primary goal is to minimize the time it takes for the last non-faulty robot to reach the exit, ensuring reliable evacuation despite potential faulty behavior. The authors present two distinct evacuation protocols tailored to the specific types of faults, crash and Byzantine, and evaluate these protocols by establishing both lower and upper time bounds for each scenario. Their findings are summarized in Table 1.2.

Bonato et al. [20], study a variation of the classic cow-path optimization problem where a robot probabilistically fails to detect an item. It is shown that traditional monotone search strategies are not optimal when the search space is a half-line. The researchers introduce and analyze a new class of strategies, termed t-sub-monotone algorithms, which deviate from monotonicity and achieve progressively better performance with increasing parameter t.

Fault	Upper bound	Lower bound
Crash	~ 6.309	~ 5.082
Byzantine	~ 6.921	~ 5.948

TABLE 1.2: Results presented in [33]

There are numerous other research papers on search and evacuation that fall beyond the scope of this work. Examples include variations in the search domain such as in equilateral triangles [43, 29, 8], 2-dimensional [1, 50, 66], in a grid [24], in a d-dimensional grid [30], in m-rays [22], in l_p unit disk [62], on graphs [77, 79], rings [80, 81, 69, 26, 10, 45], torus [70], trees [78], with variation on the termination criteria such as priority evacuation [35, 31, 36] and search-and-fetch [58, 59], with variation on termination costs [47, 21, 34, 63], using robots with asymmetric communication capabilities [57] to name a few. The interested reader could also consult the survey [37] for additional related literature.

1.3 Overview of Chapters

The subsequent chapters delve into specific aspects of the (n,f)-search/evacuation problem. Chapter 2 explores optimal algorithms for (n,f)-search on a circle, addressing scenarios involving f crash-faulty, or 1 Byzantine-faulty robots. A mixed case is also presented [60, 61]. Chapter 3 extends the discussion to circle evacuation under 2 Byzantine faults, taking into account both the Wireless and Face-to-Face communication models [74, 75]. Finally, Chapter 4 introduces Byzantine fault-tolerant protocols tailored for the general case of (n,f)-evacuation on a circle under any number of Byzantine faults using both communication models addressed in this work. This Chapter presents detailed algorithms and conducts an in-depth analysis of their time requirements [16].

Chapter 2

Optimal Circle Search Despite the Presence of Faulty Robots

In this chapter, we consider (n,f)-*search on a circle*, a search problem of a hidden exit on a circle of unit radius for $n > 1$ robots, f of which are faulty. All the robots start at the centre of the circle and can move anywhere with maximum speed 1. During the search, robots may communicate wirelessly. All messages transmitted by all robots are tagged with the robots' unique identifiers which cannot be corrupted. The search is considered complete when the exit is found by a non-faulty robot (which must visit its location) and the remaining non-faulty robots know the correct location of the exit.

We study two models of faulty robots. First, crash-faulty robots may stop operating as instructed, and thereafter they remain nonfunctional. Second, Byzantine-faulty robots may transmit untrue messages at any time during the search so as to mislead the non-faulty robots, e.g., lie about the location of the exit.

When there are only crash fault robots, we provide optimal algorithms for the (n,f)-search problem, with optimal worst-case search completion time $1 + \frac{(f+1)2\pi}{n}$. Our main technical contribution pertains to optimal algorithms for $(n,1)$-search with a Byzantine-faulty robot, minimizing the worst-case search completion time, which equals $1 + \frac{4\pi}{n}$.

We also present an algorithm for the mixed case, with one Byzantine and $f - 1$ crash faulty robots with worst-case search completion time $1 + \frac{2\pi}{n}f + 2\sin\frac{2\pi}{n}$.

2.1 Our Contribution

For $n \geq 2$, we give optimal algorithms for (n,f)-search with only crash failures and for $(n,1)$-search with one Byzantine failure. Our main result is that $(n, 1)$-search on a circle with one Byzantine-faulty robot admits a solution with search completion time $1 + \frac{4\pi}{n}$ and this is worst-case optimal. We also study (n, f, b)-mixed search, where f robots are faulty, b of which are controlled by a Byzantine adversary. In Section 2.2 we prove a lower bound of $1 + \frac{(f+1)2\pi}{n}$ for f crash-faulty robots, hence for Byzantine robots too, and in Section 2.3 we provide an algorithm that matches this bound assuming only crash failures. Then, in Section 2.4 we focus on upper bounds for searching with 1 Byzantine robot. In particular, in Subsection 2.4.1 we analyze the case of 3 robots, in Subsection 2.4.2 the case of 4 robots, and in Subsection 2.4.3 the general case of n robots; we prove that our algorithm matches the aforementioned lower bound. In Section 2.5 we provide an upper bound of $1 + \frac{2\pi}{n}f + 2\sin\frac{2\pi}{n}$ for (n, f, b)-mixed search. Finally, in Section 2.6, we conclude with a brief discussion and open problems.

2.2 Lower Bound

In this section we give a lower bound for our search problem. This result builds on the work in [33]; we extend their arguments to the case of f crash-faulty robots (hence, Byzantine too).

Theorem 2.1 (Lower Bound for (n, f)-Search)**.** *The worst-case search time $S_c(n, f)$ for $n \geq f + 1$ robots exactly f of which are crash-faulty satisfies*

$$S_c(n, f) \geq 1 + (f + 1)\frac{2\pi}{n}.$$

Proof. (Theorem 2.1) Since the maximum speed of the robots is 1, it takes at least time 1 for a robot to reach the perimeter of the disk. Further, every point on the perimeter must be traversed by at least $f + 1$ robots; for if not, the adversary will make the at most f robots visiting this point all faulty in that they remain silent and therefore the non-faulty robots will miss the exit.

Let ℓ_i be the perimeter lengths explored by exactly i robots, where $0 \leq i \leq n$. It is clear from the above discussion that in the worst case $\ell_0 = \ell_1 = \cdots \ell_f = 0$ and $\ell_{f+1} + \ell_{f+2} + \cdots + \ell_n = 2\pi$. The sum of the parts of the perimeter explored by the robots is $(f+1)\ell_{f+1} + (f+2)\ell_{f+2} + \cdots + n\ell_n$. If the robots accomplish this task by exploring the perimeter for time t (after the perimeter of the disk is reached for the first time), then it must be true that

$$\begin{aligned} nt &\geq (f+1)\ell_{f+1} + (f+2)\ell_{f+2} + \cdots + n\ell_n \\ &\geq (f+1)(\ell_{f+1} + \ell_{f+2} + \cdots + \ell_n) \\ &= (f+1)2\pi. \end{aligned}$$

It follows that $t \geq (f+1)2\pi/n$. This completes the proof.

Since $S(n) \geq S_c(n, 1)$, we immediately obtain the following corollary.

Corollary 2.1 (Lower Bound for Byzantine $(n, 1)$-Search). *The worst-case search time $S(n)$ for $n \geq 2$ robots exactly one of which is Byzantine-faulty satisfies $S(n) \geq 1 + \frac{4\pi}{n}$.*

2.3 Search under Crash Failures

In this section we show how to match the lower bound of Theorem 2.1 in the case of crash faults only.

Theorem 2.2 (Upper Bound for (n,f)-Search under Crash Failures). *The worst-case search time $S_c(n,f)$ for $n \geq 2$ robots exactly f of which are prone to crash failures satisfies*

$$S_c(n,f) \leq 1 + (f+1)\frac{2\pi}{n}.$$

Proof. Let $\partial := 2\pi/n$. Our algorithm is as follows. For each $k = 0, \ldots, n-1$, agent a_k moves to the point $k\partial$ of the unit circle and searches ccw for $(f+1)\partial$ radians. When (and if) exit is found, it is reported instantaneously.

Clearly, every sector S_j of the circle would be visited by $f+1$ robots if they all followed the protocol. Since there are at most f faulty robots, there must be at least one honest robot that will visit S_j and announce the correct location. As there can only be crash failures there will not be any contradicting announcements.

2.4 Search under one Byzantine Failure

In this section we analyze upper bounds for our search problem in the presence of a single Byzantine agent. Our main theorem is the following.

Theorem 2.3 (Upper Bound for $(n,1)$-Search under one Byzantine failure)**.** *The worst-case search time $S(n)$ for $n \geq 2$ robots exactly one of which is Byzantine-faulty satisfies*

$$S(n) \leq 1 + \frac{4\pi}{n}.$$

Thus, combining Corollary 2.1 with Theorems 2.3, we conclude that the worst-case search completion time for $(n, 1)$-search satisfies $S(n) = 1 + \frac{4\pi}{n}$.

First observe that it is trivial to prove $S(2) = 1 + 2\pi$, for $(2, 1)$-search since one of the two robots is faulty and the other non-faulty, hence the non-faulty has no other option but to search the entire perimeter.

In the next two Subsections (2.4.1 and 2.4.2) we show the upper bound for the cases $(3, 1)$-search and $(4, 1)$-search. Although the algorithms for these cases can be seen as special cases of the algorithm for the general case (Subsection 2.4.3), this is not the case for their analysis. In addition, presenting them separately allows to better clarify and illustrate the techniques and notions that we employ.

2.4.1 $(3, 1)$-search with a Byzantine-faulty robot

Lemma 2.1 ($(3,1)$-Search)**.** *The worst-case search time for 3 robots exactly one of which is Byzantine-faulty satisfies*

$$S(3) \leq 1 + \frac{4\pi}{3}$$

Proof. We will prove the claim by presenting an algorithm for this case. Consider agents a_0, a_1, a_2 and set $\partial = 2\pi/3$. We describe below the agents' actions in phases (time intervals) $[0, 1)$, $[1, 1 + \partial)$ and $[1 + \partial, 1 + 2\partial)$ and we explain why all agents know the location of the exit at time $1 + 2\partial$. Phase $[0, 1)$: Each agent a_k, $k \in \{0, 1, 2\}$, moves along a radius to the point $k\partial$ of the unit circle.

Phase $[1, 1 + \partial)$: Agent a_k searches ccw the arc $[k\partial, (k + 1)\partial)$.

Phase $[1 + \partial, 1 + 2\partial)$:

(i) If no announcements were made in time interval $[1, 1 + \partial)$ then in time interval $[1 + \partial, 1 + 2\partial)$ either there will be one correct announcement or two announcements. In the latter case the third agent, say

a_k, is honest and the correct announcement is the one by a_{k+1} (otherwise, a_k would have seen in time interval $[1, 1+\partial)$ the exit announced by a_{k-1}).

(ii) If exactly one announcement was made in time interval $[1, 1 + \partial)$, say by agent a_{k-1}, then agent a_k moves directly (along a chord) to the location of the announcement and a_{k+1} searches ccw for another ∂ radians. This takes time at most $2 < \frac{2\pi}{3}$. If a_k or a_{k+1} confirms the announcement then it is correct; otherwise, a_{k+1} in this time interval announces the correct exit point. This case is depicted in Figure 2.1.[1]

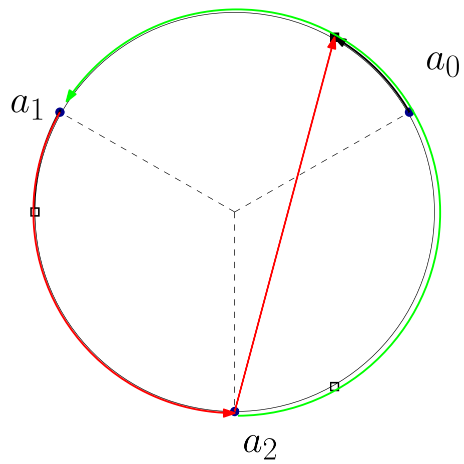

FIGURE 2.1: (3, 1)-search: robot trajectories in case $t < \frac{2\pi}{3}$.

(iii) If two announcements were made in time interval $[1, 1 + \partial)$, then they are in consecutive sectors. The silent agent is certainly non-faulty and will visit one of these sectors in this phase and will thus be able to determine which announcement was the correct one.

This completes the description of the algorithm and the proof.

[1]Figures in this work depict robot trajectories during the execution of our search algorithm. They are restricted to cases where the first announcement is made while robots search their first sector of length $\partial = \frac{2\pi}{n}$, and no other announcement is made until time $1 + \partial$. It is assumed that agent a_0 makes the first announcement. A black square shows the location of the announcement; a white square shows the locations of other agents at that time. A solid dot shows the starting positions of the robots on the unit circle (starting from the center of the circle, they move directly, in time 1, to their starting positions). Recall that the arc length between the starting position of a_0 and the point of the announcement is denoted by t (hence, the announcement takes place in time $1 + t$).

2.4.2 $(4, 1)$-search with a Byzantine-faulty robot

We will first describe an algorithm for this case. Let $\partial = \pi/2$. Each agent a_k moves with speed one to its *starting point* $k\partial$ and then continues ccw. We call the arc from one starting point to the next a *sector*. We think of each agent being responsible for the arc of length π that begins at its starting point and covers at most two consecutive sectors ccw.

Let t denote the length of the arc from the point of the first announcement to the starting point that corresponds to the agent that made the announcement (note, there is always an announcement for some $t \le \pi$). If $t \ge \frac{\pi}{2}$, then each robot checks both sectors that are assigned to it. Otherwise, set $y = \pi - 2$ and suppose an announcement is made by a_0 (w.l.o.g.) at $t < \frac{\pi}{2}$. We consider two cases.

If $t < y$, then a_1 and a_3 will search the two sectors that each is responsible for and a_2 will move along the diameter to check the announcement. This case is depicted in Figure 2.2 below.

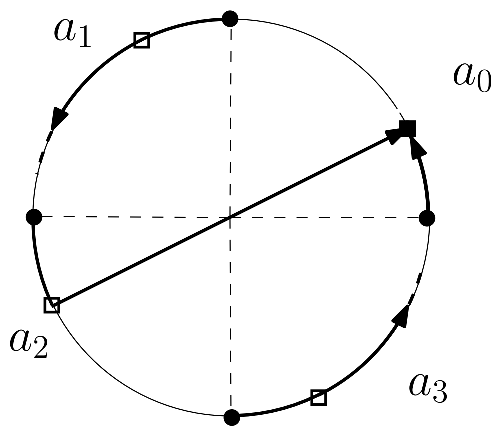

FIGURE 2.2: (4, 1)-search: robot trajectories in case $t < y$.

If $y \le t < \frac{\pi}{2}$, then a_1 continues to cover distance $\sqrt{2}$ (unless $t \ge \sqrt{2}$) and then moves along a chord to check the announcement; a_2 finishes its first sector and then moves back along a chord to its starting point and continues cw to check the arc that a_1 didn't check; a_3 continues searching its two sectors. This case is depicted in Figure 2.3 below.

This completes the description of the algorithm. We will now prove the correctness and the upper bound on the execution time.

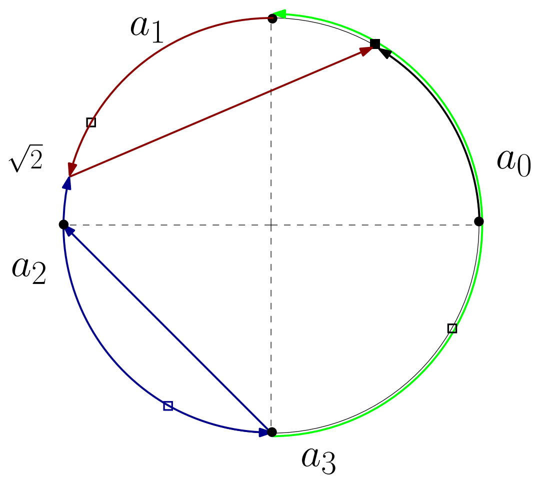

FIGURE 2.3: (4, 1)-search: robot trajectories in case $y \leq t < \frac{\pi}{2}$.

Lemma 2.2 ((4, 1)-Search)**.** *The search time for* 4 *robots exactly one of which is Byzantine-faulty satisfies*

$$S(4) \leq 1 + \pi.$$

Proof. Recall that we denote by t the length of the arc searched on the circle by the agent who made the first announcement, at the time of the announcement.

For $t \geq \frac{\pi}{2}$ we argue that when every robot has checked the sectors it is responsible for (at time $1 + \pi$), all of them know the location of the exit. First, note that if only one announcement is made, then it has to be a valid one. Therefore, assume two announcements are made (note that both are no earlier than $\frac{\pi}{2}$). Observe that they have to come from consecutive sectors: the exit must be at the first sector of the faulty robot, say a_3 since nobody spoke earlier than $\frac{\pi}{2}$, and it is discovered by a_2, while searching its second sector, who makes a correct announcement. The only other announcement can be made by a_3 and is faulty. Therefore, all agents know that the location is at the first of the two sectors in the ccw direction.

For $t < \frac{\pi}{2}$ suppose the first announcement was made by a_0. We claim that in this case the first announcement is checked by two more agents (namely, by a_3 and either a_1 or a_2) and every point of the perimeter is searched by one of the three other agents (unless a second announcement is made in which case it is not necessary to search the whole circle as one of the two must be correct). Assuming this claim, if the first announcement is verified by any other agent, then clearly it is valid. If not, then two agents reject it, thus it must be fake. It follows that another announcement was made which has to be valid. We next verify the claim and the execution time.

Consider the case $t < y$. Note that y was defined so that a_2 reaches the announcement in time less than $1 + y + 2 = 1 + \pi$. Thus, the announcement is checked by a_2 and a_3 in time, while a_1 and a_3 search every point of the perimeter.

Consider now $y \le t < \frac{\pi}{2}$. First, to see that every sector was searched by the first three agents by time $1 + \pi$, we need to argue that a_1 and a_2 covered the first sector. Indeed, a_2 searched an arc of length $\frac{\pi}{2}$ to finish his first sector, a chord of length $\sqrt{2}$ to go back to his starting point, and an arc of length at most $\frac{\pi}{2} - \sqrt{2}$ that was left uncovered by a_1; this sums up to at most $\frac{\pi}{2} + \sqrt{2} + \frac{\pi}{2} - \sqrt{2} = \pi$ as desired. Next, we need to argue that the announcement location was reached by a_1 in time $1+\pi$. This is clear if $t \ge \sqrt{2}$. Otherwise, it is not hard to see that the worst case is $t = y$. In this case, the chord a_1 walks corresponds to an arc of length $\phi = \sqrt{2} + \frac{\pi}{2} - y = 2 + \sqrt{2} - \frac{\pi}{2}$. Thus, the total time it needs is $1 + \sqrt{2} + 2\sin\frac{\phi}{2} < 1 + \pi$.

2.4.3 $(n, 1)$-search with a Byzantine-faulty robot, $n \ge 5$

We will first give the description of the algorithm for this case. For each $k \in Z_n$, agent a_k moves to the *k-th starting point* P_k located at $k\partial$, $\partial = 2\pi/n$, and then continues ccw. We denote the arc of size ∂ from the k-th starting point to the next by S_k and call it the k-th sector. We think of sectors S_k and S_{k+1} as being *assigned* to agent a_k, who is supposed to search them in the ccw direction.

Let t denote the length of the arc from the point of the first announcement to the starting point that corresponds to the agent that made the announcement. We now describe the trajectories of agents for the case that agent a_0 makes the first announcement. We will argue later (in the proof of Theorem 2.3) that the information they exchange is enough for all agents to learn the exit location.

If $t \ge \partial$, then each agent checks both sectors that are assigned to it. Otherwise, set

$$y = 2\partial - 2\sin\partial$$

and suppose an announcement is made by a_0 at $t < \partial$. Consider two cases.

If $t < y$, then each agent a_k with $k \notin \{0, 2\}$ will search its two sectors, while a_2 will start at time $1 + t$ to move along a chord towards the announcement in order to verify it.

If $y \leq t < \partial$, define arc-lengths x_k (in S_k but not to be searched by a_k) recursively as follows.

$$x_{n-2} = 0; \quad x_k = \partial + x_{k+1} - 2\sin\Big(\frac{\partial - x_{k+1}}{2}\Big), \tag{2.1}$$

for $0 < k < n-1$. Agent a_1 continues to cover distance $\partial - x_1$ (unless $t \geq \partial - x_1$) and then moves along a chord towards the announcement in order to verify it; for $1 < k < n-1$, agent a_k continues to cover distance $\partial - x_k$ (unless $t \geq \partial - x_k$), then moves along a chord back to its starting point, and finally searches in the cw direction the arc (of length at most x_{k-1}) that agent a_{k-1} didn't search; agent a_{n-1} continues with its two sectors. This case is depicted in Figure 2.4 below.

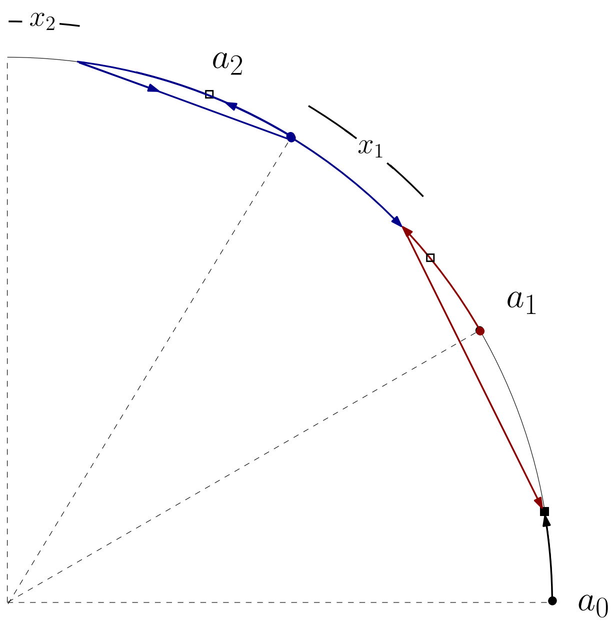

FIGURE 2.4: $(n, 1)$-search: robot trajectories in case $y \leq t < \partial$.

This completes the description of the algorithm. We next show its correctness and the upper bound on its running time.

Lemma 2.3 ((n, 1)-Search, for $n \geq 5$)**.** *The worst-case search time for $n \geq 5$ robots exactly one of which is Byzantine-faulty satisfies*

$$S(n) \leq 1 + \frac{4\pi}{n}.$$

Proof. (Lemma 2.3) We are going to argue about the correctness and the execution time of the algorithm described above.

If $t \geq \partial$, then all agents have searched the sectors assigned to them by time $1 + 2\partial$. We need to show that all of them know the location of the exit. First, note that if only one announcement is made, then it has to be a valid one. Thus, assume two announcements are made. Observe that they have to come from consecutive sectors: one of them is the true one and was discovered by an honest agent, say a_k, while searching sector S_{k+1}. It follows that a_{k+1} is faulty (because it didn't make the announcement) and the other announcement must come from it. Therefore, the agents know that the location is at the first announcement encountered in the ccw direction.

Otherwise ($t < \partial$), suppose the first announcement was made by a_0. We claim the following.

> The first announcement is checked by two more agents and every point of the perimeter is searched by at least one agent different from a_0, unless a second announcement is made.

Note first that if the first announcement is verified by one more agent, then it is proved valid to all. If not, then—assuming the claim—two agents reject it and a_0 is proved faulty to all. Furthermore, every point of the perimeter will be searched by at least one honest agent. It follows—by the second part of the claim—that another announcement will be made and will be recognized by all as valid. We next verify the claim and the execution time for the two cases on t.

Consider the case $t < y$. Note that y was defined so that a_2 reaches the announcement in time less than $1 + y + 2\sin\partial = 1 + 2\partial$. This is because it will spend less than time y on its first sector and then move along the chord that corresponds to two sectors. Thus, the announcement is checked by a_2 and a_{n-1} in time, while the other agents set forth to search every point of the perimeter.

Consider now $y \leq t < \partial$. First, we verify that every sector was searched by one of the agents $a_1, \ldots, a_{n-1}$ by time $1 + 2\partial$. It is clear that a_{n-1} searched sectors S_{n-1} and S_0. Next, we argue that, for

$0 < k < n-1$, agents a_k and a_{k+1} covered sector S_k. Note that x_k is the length of S_k that was not searched by agent a_k. However, x_k is defined so that a_{k+1} has sufficient time to travel back to P_{k+1} and aid a_k. Indeed, the worst case for a_{k+1} is when $t \le \partial - x_k$. (It is not hard to see that when $t > \partial - x_k$ he will have time to spare.) In this case, after reaching point $\partial - x_{k+1}$ of S_{k+1}, it must search a chord corresponding to an arc of $\partial - x_{k+1}$ radians and an arc of length x_k. Since it has $\partial + x_{k+1}$ time left, the definition of x_k is such that he can manage its task. Finally, we need to argue that the announcement was reached by a_1 in time $1 + 2\partial$. This is clear if $t \ge \partial - x_1$. Otherwise, it is not hard to see that the worst case is $t = y$. In this case, the chord a_1 searches corresponds to an arc of length $2\partial - x_1 - y$. Thus, the total time a_1 needs is

$$T = 1 + (\partial - x_1) + 2\sin\Big(\frac{2\partial - x_1 - y}{2}\Big).$$

In the sequel, we will make use of the following simple facts.

Fact 1. For $x \in (0, \frac{\pi}{2})$, $\sin x < x$.

Fact 2. For $x \in (0, \frac{\pi}{2})$, $\sin x < 2\sin\frac{x}{2}$.

Fact 3. For $x \in (0, \frac{\pi}{4})$, $\sin x < x - \frac{x^3}{7}$.

Since, for $n \ge 4$, $2\partial - x_1 - y < \pi$, using Fact 1 (twice) and substituting $y = 2\partial - 2\sin\partial$ we obtain

$$T \le 1 + (\partial - x_1) + (2\partial - x_1 - y) \le 1 + 2\partial - 2x_1 + \sin\partial.$$

To provide a lower on x_1, apply Fact 1 on the recursive definition to obtain

$$x_{n-3} = \partial - 2\sin\frac{\partial}{2}; \quad x_k \ge 2x_{k+1}, \text{for } 0 < k < n-1. \tag{2.2}$$

It follows that

$$x_1 \ge 2^{n-4}\Big(\partial - 2\sin\frac{\partial}{2}\Big).$$

Combining with the upper bound on T, to show $T \le 1 + 2\partial$, it suffices to argue that

$$2^{n-3}\Big(\frac{2\pi}{n} - 2\sin\frac{\pi}{n}\Big) \ge \sin\frac{2\pi}{n}.$$

Using Fact 2, $\sin\frac{2\pi}{n} \le 2\sin\frac{\pi}{n}$. Substituting this and rearranging, it suffices to show that

$$2^{n-3}\cdot\frac{\pi}{n} \ge (2^{n-3} + 1)\sin\frac{\pi}{n}.$$

In view of Fact 3, the sufficient condition simplifies further to

$$2^{n-3} \geq (2^{n-3}+1)\Big(1-\frac{\pi^2}{7n^2}\Big) \iff (2^{n-3}+1)\pi^2 \geq 7n^2,$$

which holds for all $n \geq 9$.

Finally cases $n \in \{5,6,7,8\}$ have been verified computationally as follows. In the table below we list values $y, x_1, \ldots, x_{n-3}$ for $n \in \{5,6,7,8\}$. These values determine the algorithm for these cases. To verify the table, it suffices to verify $y \leq 2\partial - 2\sin\partial$, $T \geq 1+(\partial - x_1)+2\sin(\frac{2\partial - x_1 - y}{2})$, $S(n) \leq 1+2\partial$, and $x_k \leq \partial + x_{k+1} - 2\sin(\frac{\partial - x_{k+1}}{2})$ (for $0 < k < n-2$). With respect to the x_k values, note that those which are double the previous one (marked with an asterisk) need not be verified in view of inequality (2.2).

n	x_5	x_4	x_3	x_2	x_1	y	T	$S(n)$
5				0.0810	0.2285	0.611	3.51327	3.51327
6			0.047	0.135	0.3	0.36	3.07	3.09
7		0.029	0.085	0.17*	0.34*	0.2	2.74	2.79
8	0.02	0.04*	0.08*	0.16*	0.32*	0.1	2.56	2.57

This completes the proof of the lemma.

Now we can complete the rest of the proof of Theorem 2.3.

Proof. (Theorem 2.3) Lemmas 2.1 and 2.2 prove the upper bound for $n = 3, 4$ robots respectively, and cases $n \geq 5$ are covered by Lemma 2.3.

2.5 (n, f, b) - Mixed Search

We define (n, f, b)-mixed search, to mean search for $n > 1$ robots, of which f are faulty so that b among the f are Byzantine and the remaining $f - b$ are crash faulty.

2.5.1 Algorithm for $(n, f, 1)$-mixed search

We will now present an algorithm for $(n, f, 1)$-mixed search and then analyze its time requirements. Consider n robots $a_1, a_2, \ldots, a_n$ and set $\partial := 2\pi/n$. Each robot a_k moves along a radius to the point

$k\partial$ of the perimeter of the unit circle. We call the arc $[k\partial, (k+1)\partial)$ sector S_k; that is, after 1 time unit, agent a_k will be located at the beginning of sector S_k. Robots make announcements if they find the exit and confirm/disprove the announcements of other robots accordingly. Every robot searches one sector in each round, moving counter clockwise (ccw). At any moment, if an announcement is confirmed by another robot, that announcement is correct, and the algorithm terminates as the exit has been found. Details of the main algorithm are as follows.

Algorithm 1 $(n, f, 1)$-Mixed Search

Robot a_k moves along a radius of the circle to the point $k\partial$ of the unit circle.
Robot a_k searches ccw for f rounds and makes an announcement if it finds the exit. It also disproves faulty announcements concerning sectors it has visited.
At time $1 + f\partial$:
if (there is exactly one unconfirmed announcement at the end of round f and no refutations of that announcement) or (there are two unconfirmed announcements at the end of round f in the same sector) **then**
    the robot which is two sectors away from the closest announcement moves through a chord to that announcement (inspector robot). All other robots search one more sector ccw.
else
    all robots search one more sector ccw.

Lemma 2.4. *If there are two announcements in different sectors, the correct one can be determined in time* $1 + \partial(f+1) = 1 + \frac{2\pi}{n}(f+1)$.

Proof. Each sector has been searched by a group of $f + 1$ robots. Let us assume that the sector with the first announcement is searched by a group A of robots, $|A| = f + 1$, and the second announcement is searched by a group B of robots, $|B| = f + 1$. Then $|A \cup B|$ is at least $f + 2$ (since, otherwise we would have that $A = B$), meaning that it contains at least two honest robots. One of them, say h, must be different from the one which made the correct announcement. Thus, h must have searched at least one of the two sectors on which announcements were made, either confirming the correct exit, or disproving the Byzantine announcement. In both cases the correct exit is determined.

Theorem 2.4 ($(n, f, 1)$-Mixed Search)**.** *The worst-case time for* $(n, f, 1)$-*mixed search by Algorithm 1 satisfies*

$$S(n, f) \leq 1 + \frac{2\pi}{n} f + 2\sin\frac{2\pi}{n}.$$

Proof. We consider the following three cases depending on the number of announcements made at the end of round f.

Case 1. No announcement by the end of round f. Then at round $f + 1$ there will be one or two announcements. If there is one, that announcement is correct. If there are two announcements, they are in different sectors. By Lemma 2.4 the correct exit will be found.

Case 2 One announcement made at the end of round f. We consider two subcases depending on whether or not there are any refutations of the announcement.

Subcase 2a. Assume there are no refutations of the announcement:

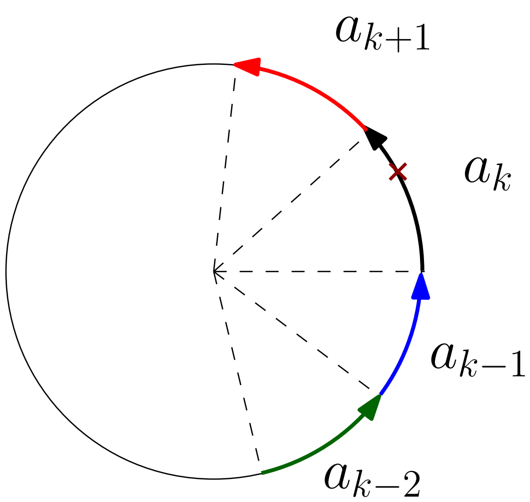

FIGURE 2.5: An announcement is made by a_k

It can be deduced that all other robots that searched the announcement's sector are faulty. As a result at least one of the two next robots (clockwise to the announcement) is honest, resulting in honest majority.

The correct exit will be known in max time $1 + \frac{2\pi}{n} f + 2\sin\frac{2\pi}{n}$ (once the inspector robot visits the announcement location).

For the inspector robot to miss the exit, the next ccw robot must be faulty. In that case, the announcement was made by an honest robot, and the inspector robot will confirm it. If the inspector robot is faulty, the other two robots (the one that made the announcement and the

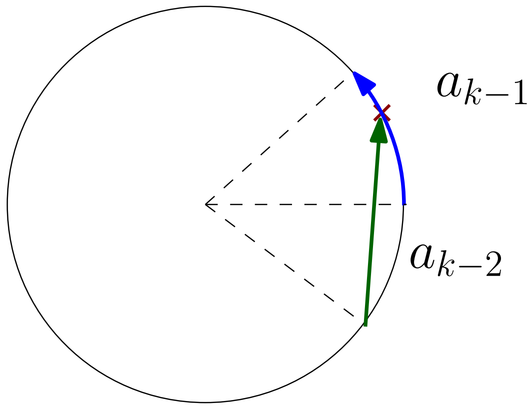

FIGURE 2.6: a_{k-2} will move through a chord to inspect the announcement

one that will visit the announcement at round $f+1$) will confirm the exit.

Subcase 2b. Assume there are refutations of the announcement: If there is no second announcement at round $f+1$, the first announcement is the correct one. If there is a second one, in a different sector, we can determine the correct one by Lemma 2.4. If the second one is in the same sector as the first announcement, then the correct one is the second (the first announcement will have 2 refutations).

Case 3. Two announcements made by the end of round f. We consider two subcases depending on whether or not the announcements were made in the same sectors.

Subcase 3a. Announcements were made in different sectors. We can determine the correct one by Lemma 2.4.

Subcase 3b. Announcements were made in the same sector. In order to have two unconfirmed announcements by round f, one honest, one Byzantine and $f-2$ crash faults have searched that sector. As a result at least one of the two next robots (clockwise to the closest announcement) is honest, resulting in honest majority. The correct exit will be known in max time $1 + \frac{2\pi}{n}f + 2\sin\frac{2\pi}{n}$ (once the inspector robot visits the announcement).

This completes the proof of the claimed time bound.

2.6 Conclusion

In this chapter we considered search on a circle with n robots, where either $f \geq 1$ of them are crash-faulty, or one of them is Byzantine-faulty, and we proved that the optimal worst-case search times are exactly $1 + \frac{(f+1)2\pi}{n}$ and $1 + \frac{4\pi}{n}$, respectively. The optimality for the Byzantine case is quite surprising given that there are very few tight bounds for search on a circle even for the wireless model. We also studied the mixed-case search, where there can be several crash-faulty and one Byzantine-faulty robot, and we provided an upper bound which leaves a small gap compared to the lower bound. Closing this gap is a challenging open question. Extending the results to multiple Byzantine-faulty robots and the evacuation problem are two interesting open directions in the context of circle search.

Chapter 3

Byzantine Fault Tolerant Symmetric-Persistent Circle Evacuation

In this chapter, we consider (n, f)-evacuation from a disk, a problem in which $n > 1$ robots, f of which are faulty, seek to evacuate through a hidden exit which is located on the perimeter of a unit disk.

We focus on symmetric-persistent algorithms, a common natural approach to search and evacuation problems. We consider two communication models: wireless and face-to-face. For the wireless model we first prove a lower bound of $1 + \frac{4\pi}{n} + 2\sin(\frac{\pi}{2} - \frac{\pi}{n})$ for the case of one faulty robot. We also observe an almost matching upper bound obtained by utilizing an earlier search algorithm. We then study the case with two Byzantine robots and we provide an algorithm that achieves evacuation in time at most $3 + \frac{6\pi}{n} + \delta(n)$, where $\delta(n)$ is a decreasing function with maximum value $\delta(4) = 0.5687$, vanishing for $n \geq 9$. For the face-to-face model we provide an upper bound of $3 + (f + 1)\frac{2\pi}{n}$ for evacuation of n robots under crash faults, an upper bound of $3 + \frac{4\pi}{n} + 2\sin\frac{\pi}{n}$ for evacuation in the case of one Byzantine robot and an upper bound of $3 + \frac{6\pi}{n} + 2\sin\frac{2\pi}{n}$ in the case of two Byzantine robots.

3.1 Our Contribution

In Section 3.2 we consider the evacuation problem for n robots, one of which is Byzantine, and we prove a lower bound of $1 + \frac{4\pi}{n} + 2\sin(\frac{\pi}{2} - \frac{\pi}{n})$ for symmetric-persistent algorithms. We also provide an almost matching upper bound of $3 + \frac{4\pi}{n}$. In Section 3.3 we present a symmetric-persistent algorithm for the case of evacuation of n robots with 2 Byzantine faults and we provide an upper bound of $3 + \frac{6\pi}{n}$

for $n \geq 9$ and $3 + \frac{6\pi}{n} + \delta(n)$ for $n < 9$, where $\delta(n) \leq 2\sin(\frac{3\pi}{2n}) + \sqrt{2 - 4\sin(\frac{3\pi}{2n}) + 4\sin^2(\frac{3\pi}{2n})} - 2$. In Section 3.4, we study the face-to-face communication model and provide upper bounds for the problem of evacuating n robots under the presence of faults. We prove an upper bound of $3 + (f+1)\frac{2\pi}{n}$ when f crash faults are present. We also derive an upper bound of $3 + \frac{4\pi}{n} + 2\sin\frac{\pi}{n}$ for evacuation under the presence of one Byzantine fault, and $3 + \frac{6\pi}{n} + 2\sin\frac{2\pi}{n}$ for the case of two Byzantine faults, leaving open the case of $f > 2$ Byzantine faults.

3.2 Evacuation with One Byzantine Fault

We define (n,f)–evacuation, to mean evacuation of $n > 1$ robots, of which f are faulty. In this work, we consider Byzantine faults, which include crash faults as a special case.

3.2.1 A lower bound for symmetric-persistent algorithms

As mentioned earlier, we focus on symmetric-persistent algorithms. In particular, we consider n robots $a_0, a_1, \ldots, a_{n-1}$ with starting position the center of a unit circle and set $\partial := 2\pi/n$. Each robot a_k moves along a radius to the point $k\partial$ of the perimeter of the unit circle.[1] We call the arc $[k\partial, (k+1)\partial)$ sector S_k. After 1 time unit, robot a_k will be located at the beginning of sector S_k and will have searched sector S_k in time $1+\partial$, moving counterclockwise (ccw). Robots make announcements if they find the exit and confirm/disprove the announcements of other robots accordingly.

Theorem 3.1. *Any symmetric-persistent algorithm requires at least time*

$$1 + \frac{4\pi}{n} + 2\sin\Big(\frac{\pi}{2} - \frac{\pi}{n}\Big) \geq 3 + \frac{4\pi}{n} - \frac{\pi^2}{2n^2} \tag{3.1}$$

for the evacuation of n robots, one of which is crash-faulty, from a circle of radius 1.

Proof. Note that if $n = 2$ the result is trivial, so we assume $n \geq 3$. Let us denote by $f(n)$ the left hand side of inequality (3.1).

[1]Note that in fact we represent the circle points in polar coordinates; as the radius is always equal to 1 we give only their angle, for the sake of simplicity.

Let point 0 be the position of robot a_0 on the unit circle at time 1 and denote by x_i the length of the arc between robot a_i and point 0 in the ccw direction. Let ψ_i denote the length of the arc between robots a_i and a_{i+2} at a certain time. Since $\sum_{i=0}^{n-1} \psi_i = 4\pi$, there exists i such that $\psi_i \geq 4\pi/n = 2\partial$. Without loss of generality, let $\psi_0 \geq 2\partial$ be the maximum among ψ_i's and let $\psi = \psi_0$ (note that $\psi = x_2$ as well). For any $\epsilon > 0$, if the adversary places the exit at point $\psi - \epsilon$, and robot a_1 is faulty, then the exit will be discovered by robot a_0 in time $1 + \psi - \epsilon$.

We now consider two cases on ψ.

First, suppose $2\partial \leq \psi < \pi$. By the maximality of ψ, there is at least one robot at distance x from 0 such that $x \in [\pi - \psi/2, \pi + \psi/2]$. The total time this robot will require to reach the exit is at least

$$1 + \psi - \epsilon + 2\sin\Big(\frac{\pi - \psi/2}{2}\Big) \geq 1 + 2\partial + 2\sin\Big(\frac{\pi - \partial}{2}\Big) - \epsilon = f(n) - \epsilon,$$

where the inequality follows because $\psi \geq 2\partial$ and the left-hand side is increasing in ψ.

Next, we consider the case $\pi \leq \psi$. In this case, we will bound the time robot a_2 will need, which is at least

$$1 + \psi - \epsilon + 2\sin(\psi/2)$$

time units. Note that this is increasing in ψ. It follows that it is at least $1 + \pi - \epsilon + 2$, which for $n \geq 4$ is clearly greater than $f(n)$. For $n = 3$, it is at least $1 + 2\partial - \epsilon + 2\sin\partial > f(n) - \epsilon$. The inequality holds since $\sin\partial = \sin(2\pi/3) > \sin(\frac{\pi}{2} - \frac{\pi}{3})$.

Since the above hold for any ϵ, the bound in the left-hand side of inequality 3.1 follows. The right-hand side bound follows from the inequality $\cos(x) \geq 1 - x^2/2$.

To prove our next Theorem, we employ the following upper bound for $S(n, 1)$-Search under one Byzantine failure, proposed in [60]:

Theorem 3.2. *The worst-case search time $S(n)$ for $n \geq 2$ robots exactly one of which is faulty satisfies*

$$S(n) \leq 1 + \frac{4\pi}{n}.$$

Theorem 3.3. *There exists a symmetric-persistent algorithm that requires time at most*

$$3 + \frac{4\pi}{n}$$

for evacuation of n robots, one of which is Byzantine, from a circle of radius 1.

Proof. We utilize Theorem 3.2, which provides a time bound of $1 + \frac{4\pi}{n}$ to find the exit, and add the length of the diameter for the furthest robot to evacuate. Also, that algorithm is symmetric-persistent as it forces all robots to move in the same direction (counterclockwise) and their trajectories change only after receiving information about the exit.

Remark 1. *Note that the above upper bound is within $O(1/n^2)$ from the lower bound of Theorem 3.1.*

3.3 Evacuation with Two Byzantine Faults

3.3.1 Algorithm for $(n, 2)$–evacuation

We will now present an algorithm for the problem of Evacuating $n \geq 5$ robots, 2 of which are Byzantine faulty, and then analyze its time requirements.

Note that in the case of 2 Byzantine robots, if an announcement is confirmed by two other robots, that announcement is correct. Also, an announcement disproved by three other robots is invalidated (announcing a different exit also counts as a disproof). When three robots make different announcements, we can deduce that two of them are Byzantine and as a result the silent ones are honest. After $f + 1 = 3$ rounds, all honest robots move via a chord to the exit to evacuate the circle and the algorithm terminates in time $E(n, 2)$.

Next, we will define disputable announcements and their maximum distance:

Definition 3.1 (Disputable announcement)**.** *An announcement is disputable when neither its validity nor its invalidity is deducible from the available information. An announcement that is not disputable is settled.*

For example, if an announcement has neither $f + 1$ confirmations nor $f + 1$ disproofs it is disputable. Note that in cases where the honesty of a robot can be deduced, its confirmations and/or disproofs result in the corresponding announcements being settled, therefore not disputable. Note also that, if only one announcement is made during the first $f + 1$ rounds, then this announcement is also settled, as it must have come from an honest robot.

Definition 3.2 (Sector distance of two announcements)**.** *We define* $d(S_i, S_j) = \min\{(i - j) \bmod n, (j - i) \bmod n\}$ *to be the distance between sectors* S_i, S_j*. Let also the sector distance of two announcements be the distance between the sectors where the announcements occurred.*

According to this definition, when the announcements are made in the same sector their sector distance is 0, whereas when announcements are in adjacent sectors their sector distance is 1.

Lemma 3.1. *In the case of two or more disputable announcements, the sector distance of any two of them is at most* 1*, in the case where* $f = 2$*.*

Proof. Let us assume that among the disputable announcements, there are two with sector distance of at least 2. Each of the corresponding sectors has been searched by a group of $f + 1 = 3$ robots. Suppose that one of these sectors is searched by a group A of robots, $|A| = f + 1 = 3$, and the other is searched by a group B of robots, $|B| = f + 1 = 3$. Then, since the sector distance is at least 2, $|A \cup B| \geq f + 3 = 5$. With at least 5 different robots searching the sectors with the two announcements, one of them must have at least 3 disproofs (as a reminder, a confirmation of an announcement also provides a disproof of any other announcement) and as a result that announcement would not be disputable, a contradiction.

Lemma 3.2. *In the case where we have three disputable announcements, the maximum sector distance of any two of them is* 0*, that is, they are all in the same sector, in the case where* $f = 2$*.*

Proof. Let us assume that the maximum sector distance of any two of the three disputable announcements is at least 1. Suppose that the first sector is searched by a group A of robots, $|A| = f + 1 = 3$, and the second sector is searched by a group B of robots, $|B| = f + 1 = 3$. Then, $|A \cup B| \geq f + 2 = 4$. Among the (at least) four robots that searched these two sectors, three of them made announcements. Since two of them are faulty, it can be deduced that the robot that did not make any announcement is honest. That honest robot would had confirmed and/or disproved at least one announcement, resulting in fewer than three disputable announcements, a contradiction.

Utilizing Lemma 3.1 and Lemma 3.2 we can deduce that the only possible cases that include disputable announcements after $f + 1 = 3$ rounds, are the following:

- Two disputable announcements in the same sector
- Two disputable announcements in adjacent sectors
- Three disputable announcements in the same sector

In any other case, after $f + 1$ rounds of search, there must be only one settled announcement made by an honest robot. In that case, the search time is $1+3\partial = 1+\frac{6\pi}{n}$ and the evacuation time is $3+3\partial = 3+\frac{6\pi}{n}$, in the worst case.

Definition 3.3 (Inspector robot)**.** *We distinguish between two cases: (a) all disputable announcements are in the same sector, say S_i (b) there are two disputable announcements in two consecutive sectors, say S_i and $S_{i+1 \pmod n}$. In both cases, for $k \geq 1$, the k-th inspector is the robot that is located at the beginning of sector $S_{i-k+1 \pmod n}$ at time $1 + 3\partial$, as shown in Figures 3.1 - 3.2.*

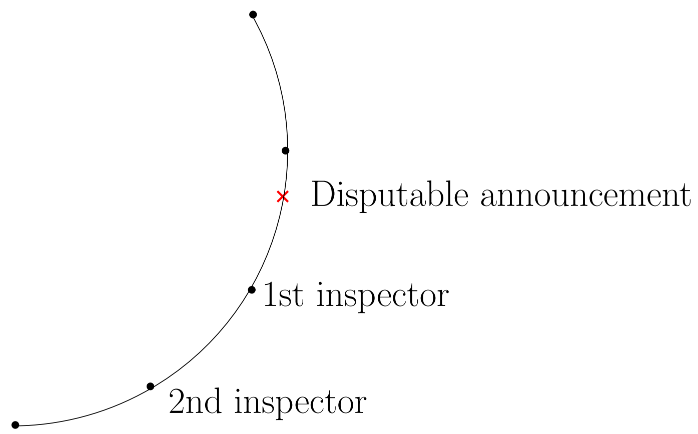

FIGURE 3.1: Inspector robots: one disputable announcement

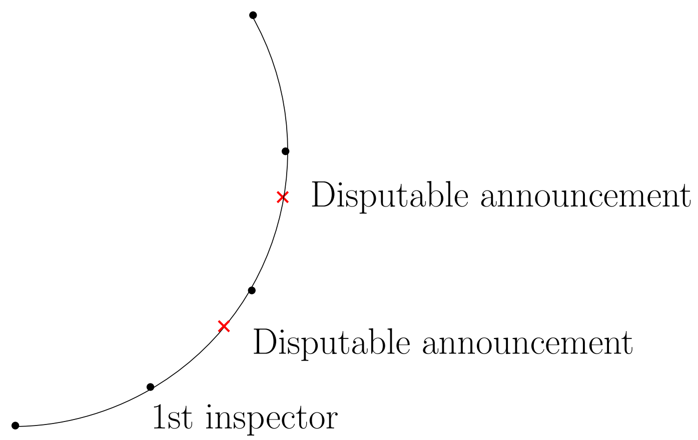

FIGURE 3.2: Inspector robots: two disputable announcements

Details of the main algorithm are as follows.

Algorithm 2 $(n, 2)$-evacuation

```
Set ∂ = 2π/n.
Robot a_k moves along a radius of the circle to the point k∂ of the unit circle.
Until time 1 + 3∂, robot a_k searches ccw and makes an announcement if it finds the exit. It also disproves faulty announcements made at sectors it visits (Staying silent when passing over an announcement's location, counts as disproof).
At time 1 + 3∂:
if there is a consensus regarding the position of the exit (no disputable announcements are present) then all honest robots move via a chord to the exit in order to evacuate.
else
    Inspector(s):
    if two disputable announcements are in the same sector then the first and second inspector robots move via a chord to the location of the nearest announcement. If the exit is not there, they move via a chord to the location of the other announcement.
    else the first inspector robot moves via a chord to the location of the nearest announcement. If the exit is not there, it moves via a chord to the location of the next (ccw) announcement until it evacuates.
    Honest (non-inspector) robots:
    if two disputable announcements exist then all non-inspector honest robots move towards the point between the two announcements. When the exit's location is known according to the inspector's findings, they move there to evacuate.
    else all non-inspector honest robots move via a chord to the location of the farthest announcement in order to evacuate and may alter their trajectory according to the inspector's findings.
```

We define t as the time beyond $1 + 3\partial$ needed to learn the position of the exit. If $t \le 1$, the evacuation time is unaffected and is equal to $3 + 3\partial$. If $t > 1$, the evacuation time is increased by a function $\delta(n)$. The geometric calculation of $\delta(n)$ follows.

Lemma 3.3. *The additional time $\delta(n)$ needed to complete the evacuation process when the position of the exit is known at time $1 + \frac{6\pi}{n} + t$ is defined as follows:*

$$\delta(n) = \begin{cases} 0 & \textit{if } t \le 1 \\ t + \sqrt{t^2 - 2(t-1)(\cos(\pi/n) + 1)} - 2 & \textit{if } t > 1 \end{cases}$$

Proof. As shown in Figure 3.3, suppose at the end of round 3 the exit is not yet known, and possible exits are in points G and E of the circle. Robot a_k placed at point A, moves towards point D, placed exactly between points G and E, which is antipodal to point A (AD=2 as radius r=1). After $t \le r$ the inspector robot moving from point F will determine the correct exit and robot a_k may need to change direction to point E (or G), but the new path that will travel is not larger than the diameter of the circle.

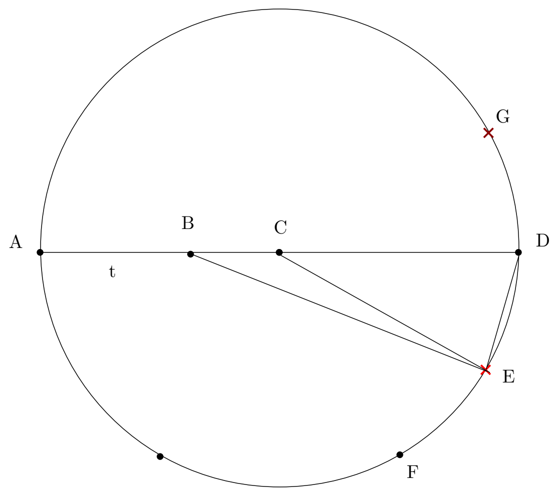

FIGURE 3.3: Evacuation: $A \longrightarrow E$, $t \le 1$

We must show that $BE \le BD$. All the following angles refer to interior angles of triangles. Triangle CED is isosceles ($CE = CD = r = 1$) and angle $D\hat{E}C = C\hat{D}E$. As a result, angle $D\hat{E}B \ge D\hat{E}C$. In triangle BED it holds $D\hat{E}B \ge B\hat{D}E$, therefore $BD \ge BE$.

If $t > 1$, the evacuation time is increased by $\delta(n)$.

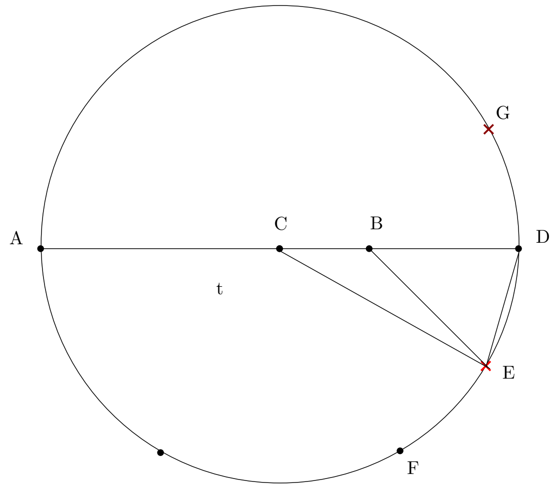

FIGURE 3.4: Evacuation: $A \longrightarrow E$, $t > 1$

As we can see in Figure 3.4, we need to calculate the distance of path ABE. We know that $AB = t$ so we need to determine BE.

In triangle CBE, $CE = 1$, $CB = 1 - BD = 1 - (2 - t) = t - 1$. In the worst case (regarding evacuation), angle $E\hat{C}D = \partial/2$. Now we can calculate BE: $BE^2 = CB^2 + CE^2 - 2 \cdot CB \cdot CE \cdot \cos(\pi/n)$.

Substituting $CB = t - 1$, $CE = 1$ we derive $BE^2 = t^2 - 2(t-1)(\cos(\pi/n) + 1)$.

The total distance the robot will travel in order to evacuate is $AB + BE = t + BE$ and that exceeds the length of the diameter by the quantity $\delta(n)$ defined below:

$$\delta(n) = \begin{cases} 0 & \text{if } t \leq 1 \\ t + \sqrt{t^2 - 2(t-1)(\cos(\pi/n) + 1)} - 2 & \text{if } t > 1 \end{cases}$$

Theorem 3.4 ($(n, 2)$–evacuation)**.** *The worst-case time of Algorithm 2 for* $(n, 2)$*-evacuation satisfies*

$$E(n, 2) \leq 3 + \frac{6\pi}{n}, \ \textit{if } n \geq 9$$

and $E(n, 2) \leq 1 + \frac{6\pi}{n} + 2\sin(\frac{3\pi}{2n}) + \sqrt{2 - 4\sin(\frac{3\pi}{2n}) + 4\sin^2(\frac{3\pi}{2n})}$, *if* $n < 9$.

Proof. If after time $1 + 3\partial$, only one announcement is made, that announcement is correct because in that time, every point of the circle has been searched by at least one honest robot. All robots move via a chord towards the exit, and evacuation is complete in time $3 + 3\partial$ (In this case, search time is $S(n, 2) = 1 + \frac{6\pi}{n}$ and evacuation time is $E(n, 2) = 3 + \frac{6\pi}{n}$).

The same search and evacuation times also hold if no disputable announcements are present. Robots know the position of the exit by time $1 + 3\partial$ and evacuation is complete in time $3 + 3\partial$.

For any other outcome, we consider the following cases (utilizing Lemma 3.1 and Lemma 3.2) depending on the number and location of disputable announcements at time $1 + 3\partial$ (i.e. after 3 rounds):

Case 1: Two disputable announcements in the same sector. Assume that a_0, a_{n-1} made the announcements in the previous rounds (say at first and second round).

- Inspector trajectory: At time $1 + 3\partial$, Algorithm 2 instructs the next two robots in clockwise order (a_{n-3},a_{n-4}) to move via a chord in order to inspect the announcements. The location of the exit will be known when both inspectors visit one of the two announcements. *This gives the worst-case Search time* $S(n, 2) = 1 + \frac{6\pi}{n} + 2sin(3\pi/2n)$*, yielding also the worst-case Evacuation time as explained below.* Once the exit location is known, inspector robots move there via a chord to evacuate.

- Any other honest robot trajectory: At time $1+3\partial$ all other robots move via a chord to the farthest announcement in order to evacuate and may alter their trajectory according to the inspectors' findings.

  If $t = 2\sin(3\pi/2n) \leq 1$ $(n \geq 9)$, then $E(n, 2) = 3 + \frac{6\pi}{n}$.

  If $t = 2\sin(3\pi/2n) > 1$ $(n < 9)$, then $E(n, 2) = 3 + \frac{6\pi}{n} + \delta(n) = 1 + \frac{6\pi}{n} + 2\sin(\frac{3\pi}{2n}) + \sqrt{2 - 4\sin(\frac{3\pi}{2n}) + 4\sin^2(\frac{3\pi}{2n})}$. See Figures 3.5-3.8.

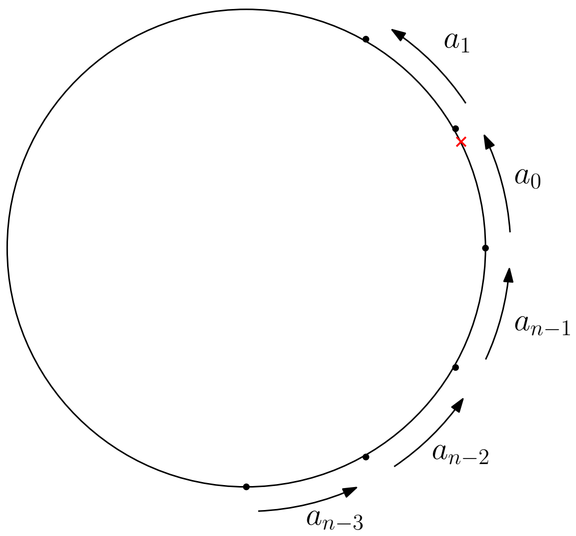

FIGURE 3.5: Robots search the unit circle counter clockwise. One announcement made by a_0 (time $1 + \partial$)

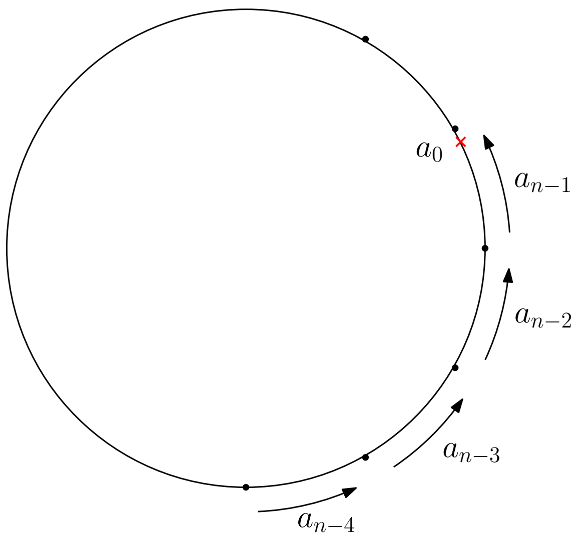

FIGURE 3.6: Robots search the unit circle counter clockwise. a_{n-1} confirms a_0's announcement (time $1 + 2\partial$)

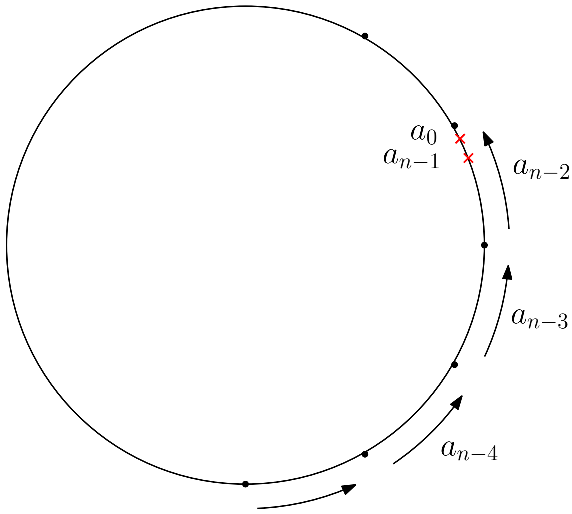

FIGURE 3.7: Robots search the unit circle counter clockwise. Second announcement made in the same sector by a_{n-2} (time $1 + 3\partial$)

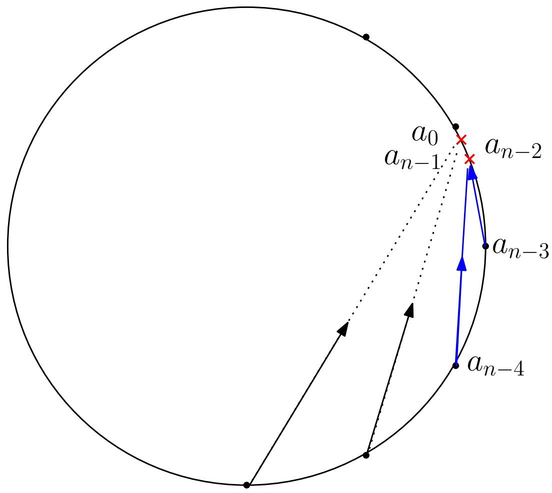

FIGURE 3.8: Inspector robots move to inspect closest announcement. All other non-faulty robots move through a chord to the furthest announcement to evacuate. Their trajectory may alter according to inspector findings

Case 2: Two disputable announcements in adjacent sectors. Assume that a_0, a_{n-1} made the announcements in the previous rounds (say, in the first and second round).

- Inspector trajectory: At time $1 + 3\partial$, Algorithm 2 instructs the next robot in clockwise order (a_{n-3}) to move via a chord to inspect the nearest announcement. The location of the exit will be known when the inspector visits the announcement in time $S(n, 2) = 1 + 3\partial + 2sin(\pi/n)$. When the location of the exit is known, inspector moves there to evacuate.

- Any other honest robot trajectories: At time $1 + 3\partial$ all other robots move via a chord to the farthest announcement to evacuate and may alter their trajectory according to the inspector's findings.

  If $t = 2\sin(\pi/n) \leq 1$ ($n \geq 6$), then $E(n, 2) = 3 + \frac{6\pi}{n}$.

  If $t = 2\sin(\pi/n) > 1$ ($n < 6$), then $E(n, 2) = 3 + \frac{6\pi}{n} + \delta(n)$.

Case 3: Three disputable announcements in the same sector. Assume that a_0, a_{n-1}, a_{n-2} made the announcements in the previous rounds. That means that the Byzantine robot is one of a_0, a_{n-1}, a_{n-2} and all the other robots are honest.

- Inspector trajectory: At time $1 + 3\partial$, Algorithm 2 instructs the next robot in clockwise order (a_{n-3}) to move via a chord to inspect the announcements. The location of the exit will be known when the deducible honest inspector visits two of the three announcements in the worst case ($S(n, 2) = 1 + 3\partial + 4sin(\pi/3n)$). When the location of the exit is known, the inspector move via a chord to evacuate.

- Any other honest robot trajectories: At time $1 + 3\partial$ all other robots move via a chord to the farthest announcement to evacuate and may change trajectory according to the inspector's findings. Because the extra time that the inspectors need to locate the exit is $4sin(\pi/3n) < 1$ for $n \geq 5$, all the robots know the location of the exit before the furthest one (that needs a diameter to evacuate) reaches the center of the circle). As a result, the robots evacuate with no extra delay at time $3 + 3\partial$ ($E(n, 2) = 3 + \frac{6\pi}{n}$). See Figures 3.9-3.12.

This completes the proof of the claimed time bound.

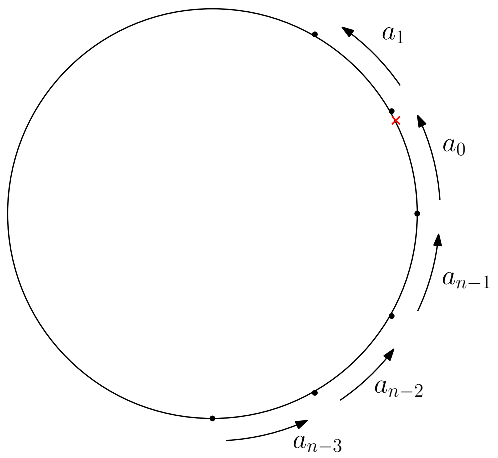

FIGURE 3.9: Robots search the unit circle counter clockwise. One announcement made (time $1 + \partial$)

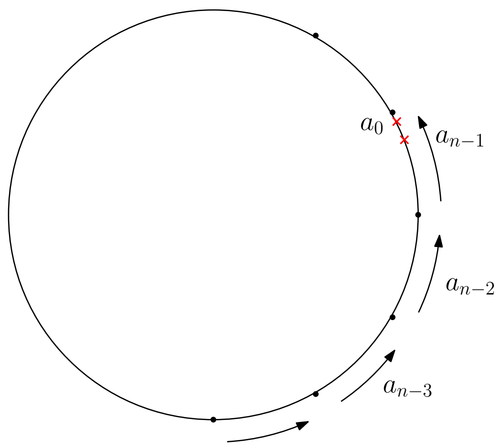

FIGURE 3.10: Robots search the unit circle counter clockwise. Second announcement made in the same sector (time $1 + 2\partial$)

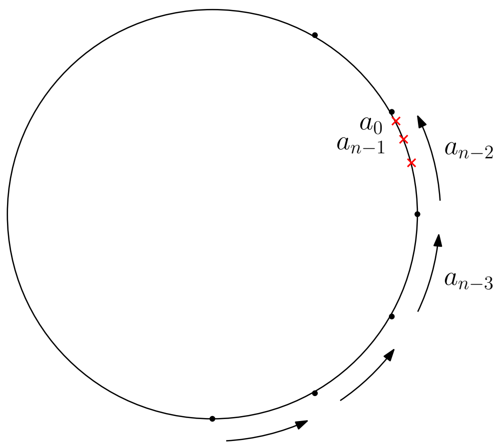

FIGURE 3.11: Robots search the unit circle counter clockwise. Third announcement made in the same sector (time $1 + 3\partial$)

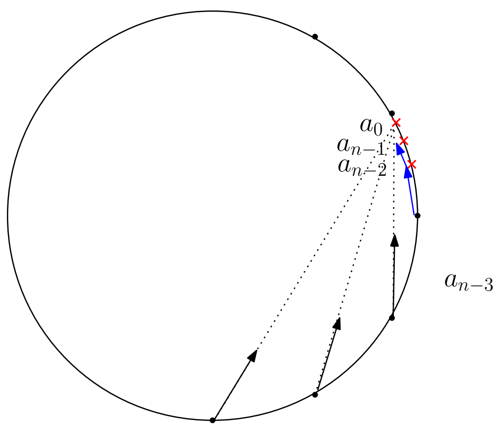

FIGURE 3.12: Inspector robots move to inspect announcements. All other non-faulty robots move through a chord to the furthest announcement to evacuate. Their trajectory may alter according to inspector findings

Some calculations follow that give (for $4 \leq n \leq 9$) the bounds obtained by the above algorithm for the $(n, 2)$ case in comparison with the lower bound obtained for the $(n, 1)$ case (which holds of course for 2 Byzantine robots as well):

n	$(n, 1)$: LB	$(n, 2)$: $3 + 3\partial$	$(n, 2) : \delta(n)$	$(n, 2)$: UB
4	5.5558	7.7124	0.5687	8.2811
5	5.1313	6.7699	0.2361	7.0060
6	4.8264	6.1415	0.0881	6.2297
7	4.5971	5.6927	0.0318	5.7246
8	4.4186	5.3561	0.0095	5.3657
9	4.2756	5.0944	0	5.0944

3.4 Evacuation in the face-to-face communication model

In the following section, we will present evacuation algorithms for n robots in the face-to-face communication model. In this model robots exchange information only when co-located.

3.4.1 Evacuation with crash faults

In the evacuation problem of n robots, f of which are crash faulty ones, we present Algorithm 3 for the face-to-face communication model.

Robots start at the center of the circle and move to the circumference at time 1. They search the circle for time $(f + 1)\partial$ and return to the center to share their findings. At least one robot will know the position of the exit, and all robots will move there to evacuate. See Figures 3.13-3.16.

Algorithm 3 (n, f)-evacuation with crash faults (F2F)

1: Define $\partial = \frac{2\pi}{n}$.
2: Robot a_k moves along a radius of the circle to the point $k\partial$ of the unit circle and start searching ccw.
3: At time $1 + (f + 1)\partial$: all robots return to the center of the circle.
4: At least one robot found the exit and it can inform the rest of the robots
5: All robots move to the exit to evacuate

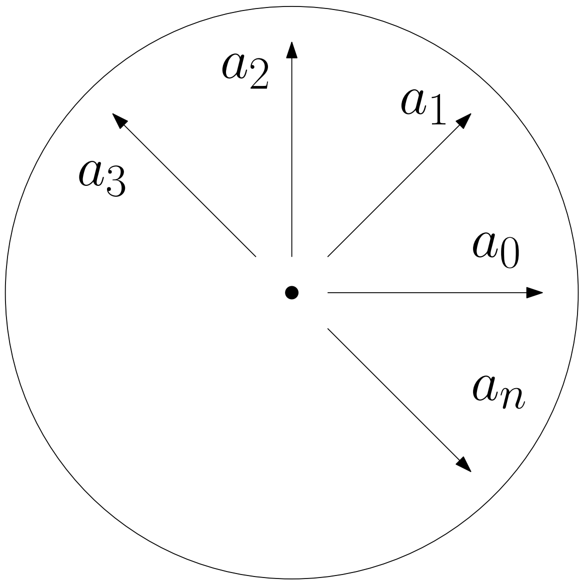

FIGURE 3.13: Each robot a_k move from their starting position (center of the circle) to point $k\partial$ of the circle $(t < 1)$

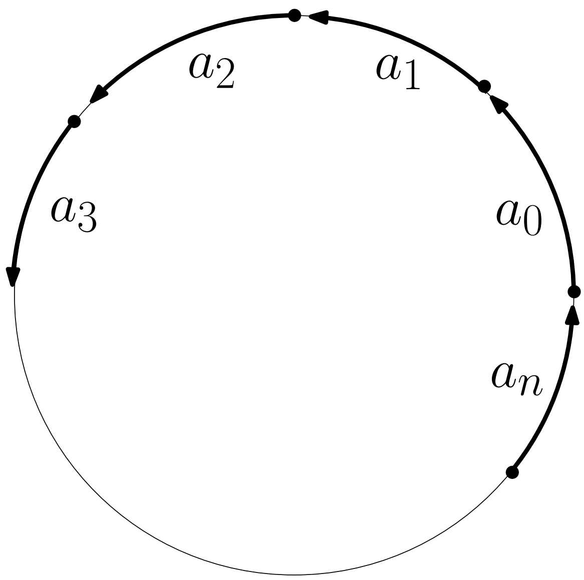

FIGURE 3.14: Robots search the unit circle counter clockwise $(t < 1 + (f + 1)\partial)$

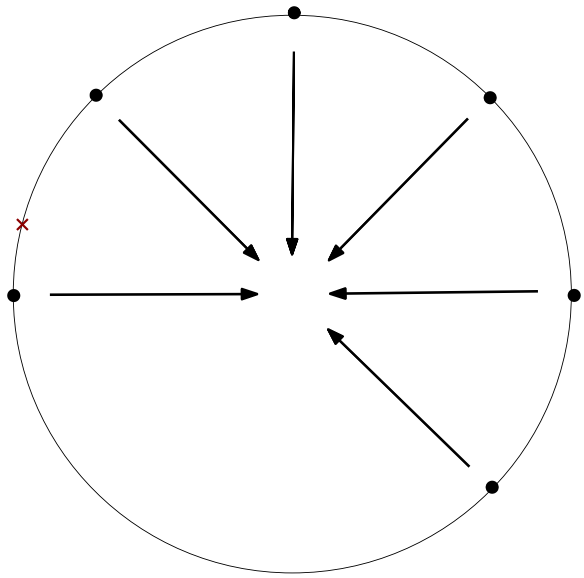

FIGURE 3.15: At least one non-faulty robot found the exit. Robots rendez-vous at the center of the circle to share their findings ($t < 2 + (f + 1)\partial$)

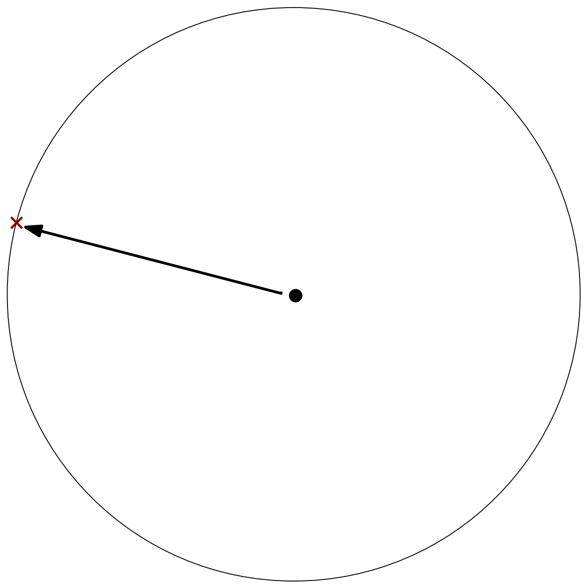

FIGURE 3.16: All non-faulty robots evacuate ($t < 3 + (f + 1)\partial$)

Theorem 3.5 ((n,f)-evacuation with crash faults (F2F)). *The worst-case time for (n,f)-evacuation with f crash faults in the face-to-face model, satisfies*

$$E(n,f) \leq 3 + (f+1)\frac{2\pi}{n}$$

Proof. Since the maximum speed of the robots is 1, it takes at least time 1 for a robot to reach the perimeter of the unit circle. Furthermore, every point on the perimeter must be traversed by at least $f+1$ robots; for if not, the adversary will make the at most f robots visiting this point all faulty (in that they remain silent) and therefore the non-faulty robots will miss the exit.

As shown in [60], the search time for the (n,f) case with f crash faults is tight and equals $(f+1)\frac{2\pi}{n}$. Robots need one additional time unit to rendezvous in the center of the circle (in order to exchange information about the location of the exit) and one more time unit to evacuate. This completes the proof of the claimed time-bound.

3.4.2 Evacuation with One Byzantine Fault

For the evacuation problem of n robots under Byzantine faults, we present Algorithm 4. We use that algorithm to prove an Upper Bound for the evacuation of n robots, one of which is Byzantine.

In a similar manner, robots after searching the circumference of the circle, rendezvous at the center to share their findings. Robots make claims about the location of the exit and check the validity of these claims. If they can deduce the location of the exit, they move there to evacuate. If not, in the worst case all honest robots must visit both the disputable claims to evacuate.

Algorithm 4 (n,f)-evacuation with Byzantine faults (F2F)

1: Robot a_k moves along a radius of the circle to the point $k\partial$ of the unit circle and start searching ccw.
2: At time $1+(f+1)\partial$: all robots return to the center of the circle.
3: Robots that claim they have found the exit inform the rest of the robots
4: If a consensus about the location of the exit have achieved, all robots move to the exit to evacuate. Otherwise, robots must visit all the disputable claims.

Lemma 3.4. *In the* $(n, 1)$ *case, executing Algorithm 4, if there are two claims in different sectors, the correct one can be determined after search time of* $\partial(f+1) = \frac{2\pi}{n}(f+1) = \frac{4\pi}{n}$.

Proof. Each sector has been searched by a group of $f+1$ robots. Let us assume that the sector with the first claim is searched by a group A of robots, $|A| = f+1$, and the second claim is searched by a group B of robots, $|B| = f+1$. Then $|A \cup B|$ is at least $f+2$ (since otherwise we would have that $A = B$), meaning that it contains at least two honest robots. One of them, say h, must be different from the one which made the correct claim. Thus, h must have searched at least one of the two sectors on which claims were made, either confirming the correct exit or disproving the Byzantine claim. In both cases, the correct exit is determined.

Theorem 3.6 ($(n, 1)$–evacuation with one Byzantine fault (F2F))**.** *The worst-case time for* $(n, 1)$*-evacuation with* 1 *Byzantine fault in the face-to-face model, satisfies*

$$E(n,f) \leq 3 + \frac{4\pi}{n} + 2\sin\frac{\pi}{n}$$

Proof. The following cases arise during the execution of Algorithm 4:

1. Only one claim about the location of the exit is made in time $2 + 2\partial$. That claim is correct, and all the robots move through the radius to evacuate. Evacuation is complete in time $3 + \frac{4\pi}{n}$.
2. Two claims about the location of the exit are made.
   - Claims are in different sectors: In that case, we can deduce the location of the exit in time $2 + \frac{4\pi}{n}$ (Lemma 3.4). Robots move through the radius to evacuate in time $3 + \frac{4\pi}{n}$.
   - Claims are in the same sector: In that case, that sector has been searched by one Byzantine and one honest robot. We can't deduce the location of the exit in time $2 + \frac{4\pi}{n}$. In the worst case, the other honest robots must visit both claims, moving via a chord, to find the exit. Evacuation is complete in time $3 + \frac{4\pi}{n} + 2\sin\frac{\pi}{n}$. See Figures 3.17-3.20.

This completes the proof of the claimed time bound.

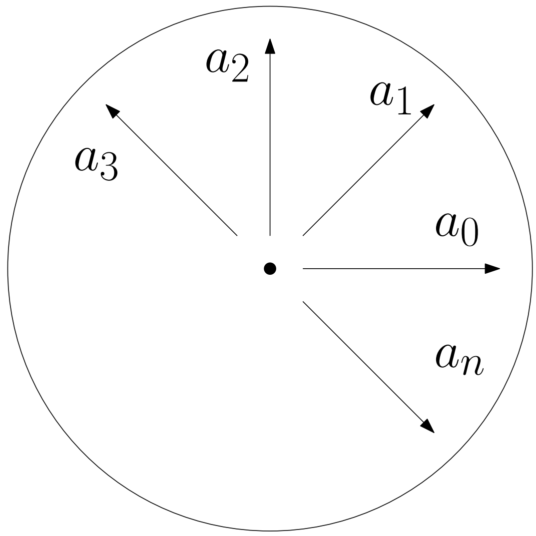

FIGURE 3.17: Each robot a_k move from their starting position (center of the circle) to point $k\partial$ of the circle $(t < 1)$

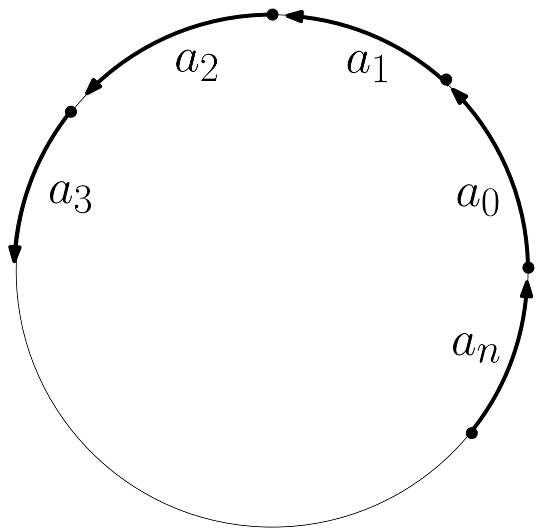

FIGURE 3.18: Robots search the unit circle counter clockwise $(t < 1 + (f + 1)\partial)$

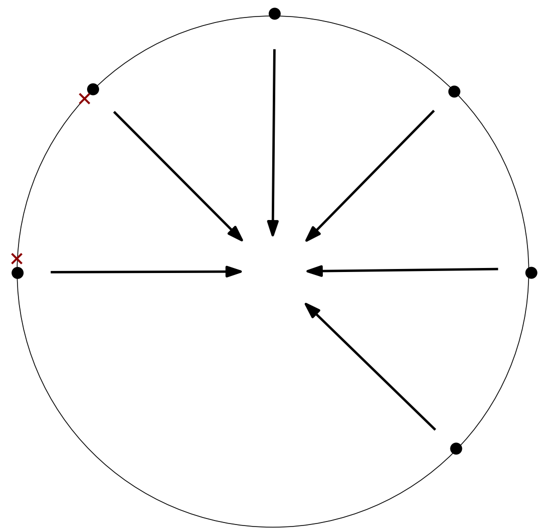

FIGURE 3.19: Two claims in the same sector were made in the previous step. Robots rendez-vous at the center of the circle to share their findings ($t < 2 + (f + 1)\partial$)

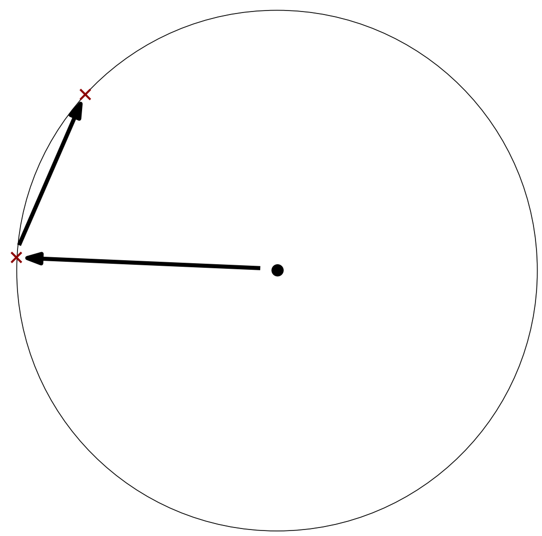

FIGURE 3.20: After the information exchange about the exit, robots must visit both claims, in the worst case, to evacuate ($t < 3 + (f + 1)\partial + 2\sin\frac{\pi}{n}$)

3.4.3 Evacuation with Two Byzantine Faults

Continuing our work in the evacuation problem, we consider the $(n, 2)$ case, where we also use Algorithm 4 to prove an Upper Bound when two Byzantine robots are involved.

Before the main analysis, we prove the following lemma:

Lemma 3.5. *In the* $(n, 2)$ *case,* $n \geq 5$, *executing Algorithm 4, if there are claims in different non consecutive sectors, the sector with the exit can be determined after search time of* $\partial(f+1) = \frac{2\pi}{n}(f+1) = \frac{6\pi}{n}$.

Proof. In a similar way to Lemma 3.4, each sector has been searched by a group of $f+1$ robots. Assume that one sector is searched by a group of robots A, $|A| = f+1$, and the second sector is searched by a group C of robots, $|C| = f+1$. Then $|A \cup C|$ is at least $f+3$ (as non consecutive sectors), meaning that it contains at least three honest robots. Two of them must be different from the one which made the claim in the correct sector. Each of these two robots must have searched at least one of these two sectors, either confirming the exit or disproving the Byzantine claim(s) (even indirectly). As a result, one Byzantine claim will have at least three disproofs (direct or indirect) and the sector with the exit will be determined.

Corollary 3.1. *In the* $(n, 2)$ *case,* $n \geq 5$, *executing Algorithm 4, if there are three claims in three different sectors, the exit can be determined after search time of* $\partial(f+1) = \frac{2\pi}{n}(f+1) = \frac{6\pi}{n}$.

Theorem 3.7 ($(n, 2)$–evacuation with Two Byzantine faults (F2F))**.** *The worst-case time for* $(n, 2)$*-evacuation with* 2 *Byzantine faults in the face-to-face model, satisfies*

$$E(n, f) \leq 3 + \frac{6\pi}{n} + 2\sin\frac{2\pi}{n}.$$

Proof. The following three cases arise during the execution of Algorithm 4:

1. All claims in the same sector: In the worst case, honest robots must visit all disputable announcements to evacuate. Evacuation is complete in time $3 + \frac{6\pi}{n} + 6\sin\frac{\pi}{3n}$ for 3 claims and in time $3 + \frac{6\pi}{n} + 4\sin\frac{\pi}{2n}$ for 2 claims.

2. Claims made in consecutive sectors:

- 3 claims. Using Corollary 3.1 we know the location of the exit in time $2 + \frac{6\pi}{n}$. Evacuation is complete in time $3 + \frac{6\pi}{n}$.
- 2 claims. In the worst case robots must visit both disputable claims in the two consecutive sectors to evacuate. Evacuation is complete in time $3+\frac{6\pi}{n}+2\sin\frac{2\pi}{n}$. See Figures 3.21-3.24.

3. Claims made in non consecutive sectors: By Lemma 3.5, we can deduce the sector with the exit in time $2 + \frac{6\pi}{n}$. If the sector contains two disputable claims, in the worst case, robots must visit both of them moving via a chord. Evacuation is complete in time $3 + \frac{6\pi}{n} + 2\sin\frac{\pi}{n}$.

This completes the proof of the claimed time bound.

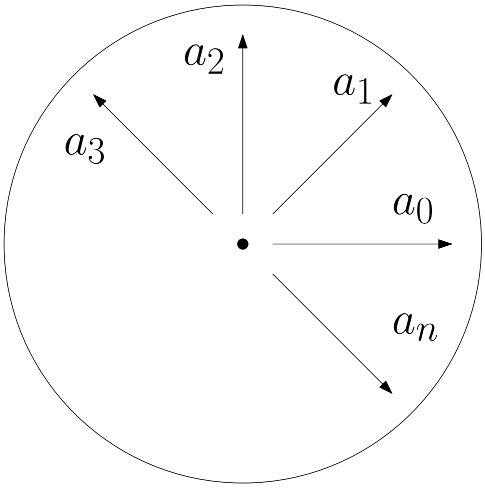

FIGURE 3.21: Each robot a_k move from their starting position (center of the circle) to point $k\partial$ of the circle ($t < 1$)

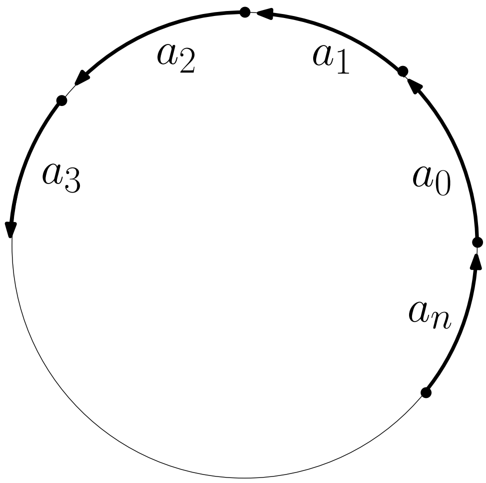

FIGURE 3.22: Robots search the unit circle counter clockwise ($t < 1 + (f + 1)\partial$)

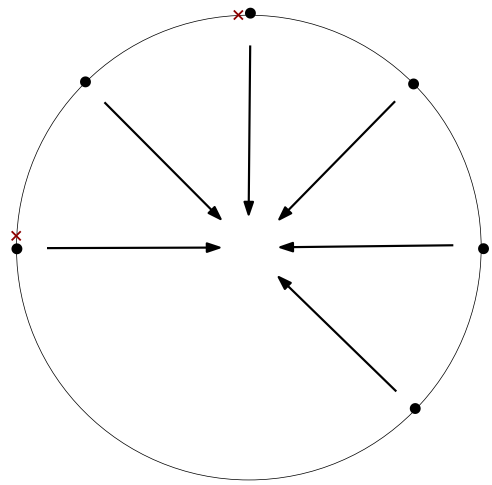

FIGURE 3.23: Two claims were made in the previous step. Robots rendez-vous at the center of the circle to share their findings ($t < 2 + (f + 1)\partial$)

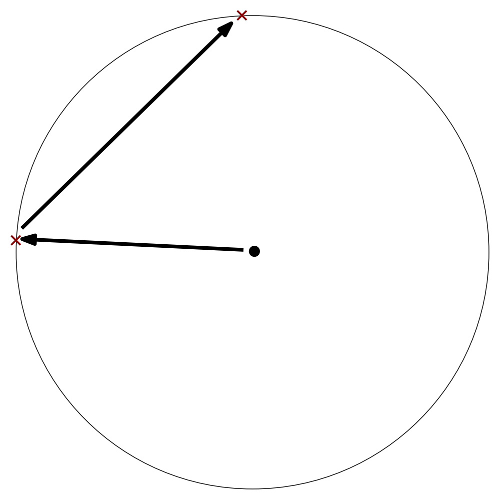

FIGURE 3.24: After the information exchange about the exit, robots must visit both claims, in the worst case, to evacuate ($t < 3 + (f + 1)\partial + 2 \sin \frac{2\pi}{n}$)

3.5 Conclusion

In this chapter, we studied the evacuation problem of n robots with one or two Byzantine faults in the wireless model and provided a lower bound for the $(n, 1)$-evacuation case and an upper bound for the $(n, 2)$-evacuation case. An interesting possible direction after that would be to tighten our bounds or generalize for f Byzantine robots. In particular, we conjecture that $3 + 3\partial$ is a lower bound for the $(n, 2)$ evacuation problem for infinitely many n. We also provided algorithms and upper bounds on the evacuation time in the face-to-face communication model. In particular, under the presence of crash faults and one and two Byzantine faults.

Chapter 4

Byzantine Fault-Tolerant Protocols for (n, f)-evacuation from a Circle

In this chapter, we address the problem of (n, f)-evacuation on a circle, which involves evacuating n robots, with f of them being faulty, from a hidden exit located on the perimeter of a unit radius circle. The robots commence at the center of the circle and possess a speed of 1.

We introduce algorithms for both the Wireless and Face-to-Face communication models under any number of Byzantine faults. We analyze the time requirements of these algorithms and we establish upper bounds on their performance.

4.1 Our Contribution

In Section 4.2 we consider the evacuation problem for n robots f of which are Byzantine faulty in the wireless communication model, and we propose an algorithm for that case, proving the following upper bound

$$E(n, f) \leq 1 + (f + 1) \cdot \frac{2\pi}{n} + \max\{G_e(k^*), H_e(k^*)\}$$

where $G_e(k^*)$ and $H_e(k^*)$ is the time needed to evacuate two crucial groups of robots, during the execution of our algorithm. For a more detailed analysis please refer to Theorem 4.1.

In Section 4.3 we propose an algorithm for the Face-to-Face communication model and in Theorem 4.2 we prove an upper bound of

$$E(n,f) \leq 3 + (f+1) \cdot \frac{2\pi}{n} + \max_{2 \leq k \leq n} \left\{ 2(k-1) \cdot \sin\left(\frac{f-k+2}{k-1} \cdot \frac{\pi}{n} \right) \right\}$$

We must note that our analysis and experimental results show that our proposed algorithm performs better than the trivial algorithm in cases detailed by Lemma 4.7.

4.2 Evacuating under Wireless Communication

We define (n,f)-evacuation, to mean evacuation of $n > 1$ robots, $f < \frac{n}{2}$ of which are faulty. In this work, we study Byzantine faults.

We consider n robots $a_0, a_1, \ldots, a_{n-1}$ with a starting position at the center of a unit circle and set $\partial := 2\pi/n$. Each robot a_i moves along a radius to the point $i\partial$ of the perimeter of the unit circle.[1] We call the arc $[i\partial, (i+1)\partial)$ sector s_i. After one time unit, robot a_i will be located at the beginning of sector s_i and will have searched sector s_i in time $1 + \partial$, moving counterclockwise (ccw). Each sector search counts as a round. Each robot is tasked to search $(f+1)$ consecutive sectors. Robots make announcements if they find the exit and approve/disprove the announcements of other robots accordingly.

In our analysis, it is important to know the announcements' distance because in that way we can eliminate the number of unsettled announcements. We extend Definition 3.2:

Definition 4.1 (Sector distance of a set of announcements)**.** *We define $d(s_i, s_j) = \min\{(i-j) \bmod n, (j-i) \bmod n\}$ to be the distance between sectors s_i, s_j. Let the distance of a set of announcements X be the length of the shortest arc containing all announcements in X; let this arc be called $arc(X)$. Finally, let the sector distance of X be the distance between the sectors where the two endpoints of $arc(X)$ fall.*

Since faulty robots are present, it is difficult for honest robots to differentiate between these announcements. To help our analysis, we will use disputable announcements (Definition 3.1) and the group of robots responsible for resolving them.

[1]Note that in fact we represent the circle points in polar coordinates; as the radius is always equal to 1 we give only their angle, for the sake of simplicity.

If there are k disputable announcements we will denote them as $X_1, \ldots X_k$, where X_j is before X_{j+1} in counterclockwise order (ccw), $j \in \{1, \ldots, k-1\}$ and X_1 is the announcement with the maximum sector distance from its previous announcement ccw.

In case consensus is not reached after $f+1$ rounds (i.e. disputable announcements are present), we need more robots to visit and settle them. As also mentioned in Chapter 2, we call these robots *inspector robots* and we extend Definition 3.3 for k disputable announcements as follows:

Definition 4.2 (Inspector robots - k disputable announcements)**.** *Assume that there are k disputable announcements, $X_1, \ldots X_k$. Let the first X_1 be in sector s_j. Then the i-th inspector is the robot that is located at the beginning of sector $s_{j-i+1 \pmod n}$ at time $1 + (f+1)\partial$.*

Based on the number of disputable announcements and their maximum distance, we will determine the number of inspector robots that are sufficient to settle the disputable announcements.

Lemma 4.1. *Assume that after executing $f+1$ rounds of the algorithm there are $k = 2$ disputable announcements with sector distance $d \leq f-1$. Then $f-d$ inspectors are sufficient for all the honest robots to learn where the exit is.*

Proof. At the $f+1$ rounds of the algorithm in total $f+1+d$ different robots searched the area of the 2 disputable announcements. Assume that among them there are exactly h honest robots, $1 \leq h \leq f+1+d-(k-1)$.

Therefore each false announcement has at least $k-2+h$ disproofs, since each announcement is a disproof of any other announcement and all of the h robots have visited at least one of the two announcements. Since there are $f-d$ inspectors, there are at most $f-(f+1+d-h) = h-d-1$ faulty robots and at least $f-d-(h-d-1) = f+1-h$ honest robots among them.

Hence each false announcement will have at least $k-2+h+f+1-h = f+1$ disproofs.

As a result of Lemma 4.1, when we have $k = 2$ announcements made with sector distance d, the number of inspectors needed is $f-d$. Because inspectors should be robots that have not previously visited any of the announcements, $n \geq f+1+d+f-d = 2f+1$.

Lemma 4.2. *Assume that after executing $f+1$ rounds of the algorithm there are k disputable announcements with sector distance d. Then $f+2-k$ inspectors are sufficient for all the honest robots to learn where the exit is.*

Proof. At the $f+1$ rounds of the algorithm in total $f+1+d$ different robots searched the area of the k announcements. We assume that the robots that searched the sector containing the first announcement, after the end of the $f+1$ rounds, will visit the rest of the announcements.

Therefore, $f+1$ robots know the location of the correct exit. Let f_1, $0 \leq f_1 \leq f$ of them be faulty and $f+1-f_1$ be honest. From the rest $f+1+d-(f+1) = d$ robots that searched the area of the k announcements, let f_2, $0 \leq f_2 \leq d$, $f_1+f_2 \leq f$ be faulty and $d-f_2$ be honest.

Since each announcement is a disproof for any other announcement, each false announcement has $k-2$ disproofs from the other $k-2$ false announcements, plus $f+1-f_1$ disproofs from the honest robots that search all the sectors with announcements.

Since there are $f+2-k$ inspectors then at most $f-(f_1+f_2)$ are faulty and at least $f+2-k-(f-(f_1+f_2)) = 2-k+f_1+f_2$ are honest.

Therefore the number of disproofs that each false announcement has is at least:

$$k-2+f+1-f_1+2-k+f_1+f_2 = f+1+f_2 \geq f+1$$

and hence each false announcement is settled.

We now present Algorithm 5, for the problem of Evacuating n robots, $f < \frac{n}{2}$ of which are Byzantine faulty, in the wireless communication model and then analyze its time requirements. Figures 4.1 - 4.4 helps visualizing the steps of Algorithm 5.

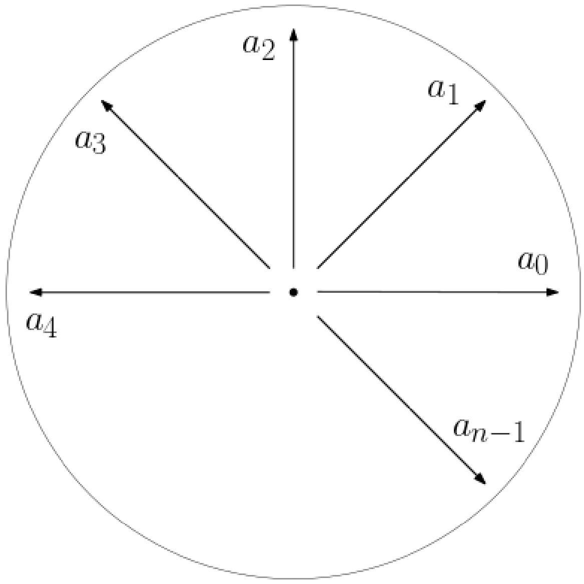

FIGURE 4.1: Each robot a_i moves from their starting position (center of the circle) to point $i\partial$ of the circle $(t < 1)$

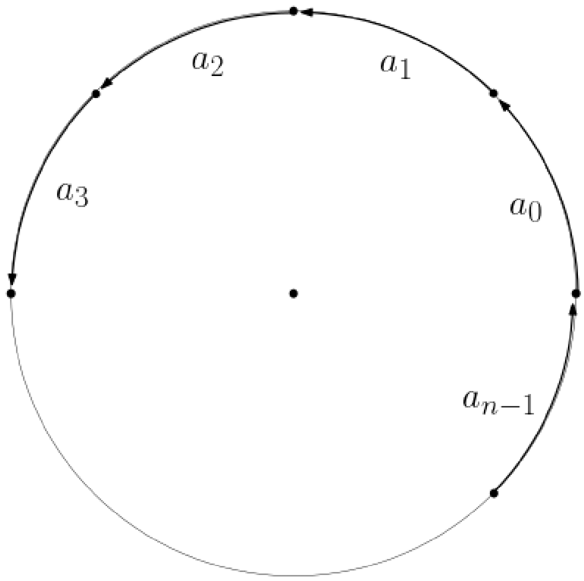

FIGURE 4.2: Robots search the unit circle counter-clockwise $(t < 1 + (f + 1)\partial)$

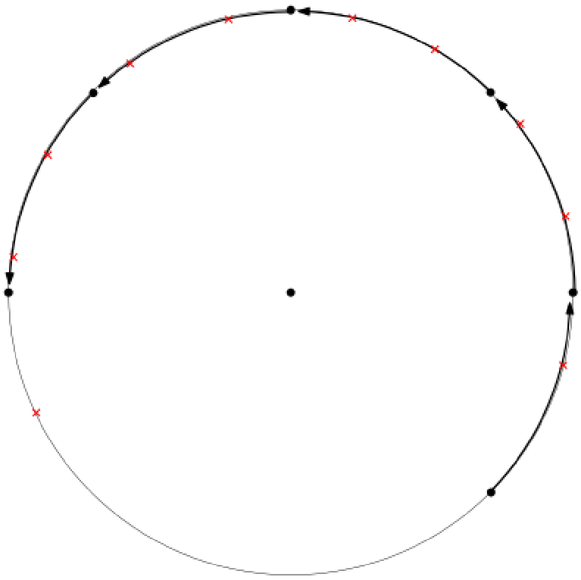

FIGURE 4.3: After f+1 rounds of search, announcements are present ($t = 1 + (f + 1)\partial$)

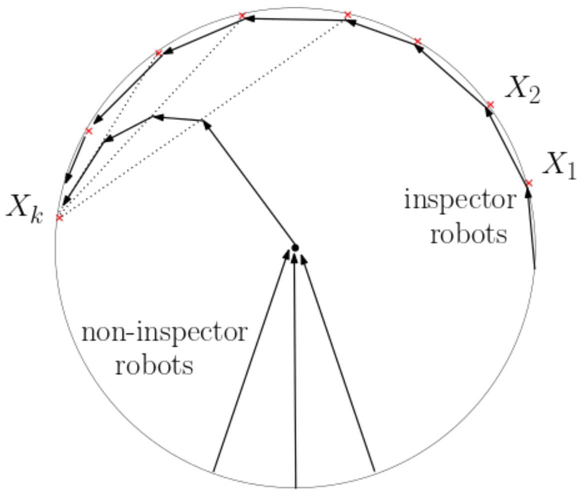

FIGURE 4.4: Evacuation paths of inspector robots and non-inspector robots, ($t > 1 + (f + 1)\partial$)

Algorithm 5 (n, f)-evacuation

Set $\partial = 2\pi/n$.

Robot a_i moves along a radius of the circle to the point $i\partial$ of the unit circle.

Until time $1 + (f + 1)\partial$, robot a_i searches ccw and makes an announcement if it finds the exit. It also disproves faulty announcements made at sectors it visits (Staying silent when passing over an announcement's location, counts as disproof). Every search of a sector ∂ counts as a round.

At time $1 + (f + 1)\partial$:

if there is a consensus regarding the position of the exit (no disputable announcements are present) **then** all honest robots move via a chord to the exit in order to evacuate.

else if there are $k \geq 2$ disputable announcements and their distance is d **then**

Inspector(s):

The $f + 2 - k$ inspectors move via a chord to the location of the nearest announcement (X_1). If the exit is not there, they move via a chord to the location of the next nearest announcement (X_2). They continue until they find the exit and they evacuate.

Honest (non-inspector) robots:

The honest robots gather to the center of the circle. By the time they arrive, c announcements have been approved or disproved by the inspectors. Then they move towards the middle M_{c+1} of the chord that connects the announcements X_{c+1}, X_k, and wait until X_{c+1} is approved or disproved by the inspectors, then move to the middle M_{c+2} of the chord that connects the announcements X_{c+2}, X_k, and continue this process iteratively. If, at any point, the exit is discovered, the robots head toward it to evacuate.

After executing Algorithm 5 for $f + 1$ rounds, it is expected that a range of 1 to $f + 1$ robots will have made announcements. Some of these announcements will be settled, and some of them will be disputable. Regarding the distance of disputable announcements, we get the following lemmas:

Lemma 4.3. *If $2 \leq k \leq f + 1$ announcements are made, the maximum sector distance of any two of them, in order for all k of them to remain disputable is $f + 1 - k$.*

Proof. Suppose there are k disputable announcements and consider the two of them at maximum sector distance D. For the sets A and B

of robots that were supposed to pass over each one, we have $|A \cup B| = f + 1 + d$ where $d = f + 1$ if $D > f + 1$ and $d = D$ otherwise.

Any silent robot in this union casts at least one disproof to one of these announcements. Suppose z is the total number of robots that spoke, w of those confirming one and y the other. Since the $z - w - y$ announcements count as a disproof for both, the sum of disproofs for these two announcements is at least $f + 1 + d + z - w - y$. If this is greater than $2f$, then one of them would have at least $f + 1$ disproofs. Thus, $f + 1 + d + z - w - y \leq 2f$, which implies $d \leq f - (z - w - y) - 1$. The bound follows because $z - w - y \geq k - 2$.

In order to calculate the worst placement of disputable announcements by the Adversary, we prove the following lemma.

Lemma 4.4 (Maximum Robot Trajectory)**.** *Assume that we have $k + 1$ points on the circle that can lie in an arc of angle $a < 2\pi$. The maximum distance that a robot will traverse in order to visit all $k + 1$ points is if the points are placed in equal distances in the arc of angle a.*

Proof. Assume that $\partial_1, \partial_2, \ldots, \partial_k$ are the interior angles that are formed with the placement of the points, as shown in Figure 4.5. Then we must have that $\partial_1 + \partial_2 + \cdots + \partial_k = a$ and that $\partial_i < \pi$, $i = 1, \ldots, n$

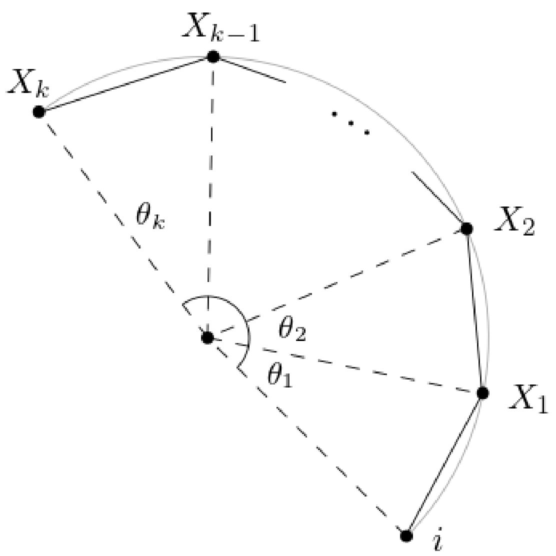

FIGURE 4.5: Maximum trajectory scenario

Then the distance that a robot will traverse in order to visit all points is:

$$\begin{aligned} D(\partial_1, \partial_2, \ldots, \partial_k) &= \sum_{i=1}^{k} x_i \\ &= \sum_{i=1}^{k} 2 \cdot sin\Big(\frac{\partial_i}{2}\Big) \end{aligned}$$

Let $f(\partial) = sin(\frac{\partial}{2})$, $f'(\partial) = \frac{1}{2} \cdot cos(\frac{\partial}{2})$, $f''(\partial) = -\frac{1}{4} sin(\frac{\partial}{2}) < 0$ when $\partial \in (0, 2\pi)$. Hence f is concave. We wish to find the angles ∂_i such that D is maximized. We will work on the maximization of the quantity

$$\frac{D(\partial_1, \partial_2, \ldots, \partial_k)}{2k} = \sum_{i=1}^{k} \frac{1}{k} sin\Big(\frac{\partial_i}{2}\Big), \quad w.r.t.\ \partial_1, \partial_2, \ldots, \partial_k$$

Now by Jensen's inequality, since f is concave, we have that:

$$\begin{aligned} \sum_{i=1}^{k} \frac{1}{k} \cdot sin\Big(\frac{\partial_i}{2}\Big) &\leq f\Big(\frac{\sum_{i=1}^{k} \partial_i}{k}\Big) \\ &= sin\Big(\frac{\sum_{i=1}^{k} \partial_i}{2k}\Big) \\ &= sin\Big(\frac{a}{2k}\Big) \end{aligned}$$

We note that the equality holds ($\frac{D(\partial_1, \partial_2, \ldots, \partial_k)}{2k}$ is maximised) if $\partial_i = \frac{a}{k}$, $i = 1, \ldots, k$, since

$$\begin{aligned} \sum_{i=1}^{k} \frac{1}{k} \cdot sin\Big(\frac{\partial_i}{2}\Big) &= \sum_{i=1}^{k} \frac{1}{k} \cdot sin\Big(\frac{a}{2k}\Big) \\ &= \frac{k}{k} \cdot sin\Big(\frac{a}{2k}\Big) \\ &= sin\Big(\frac{a}{2k}\Big) \end{aligned}$$

Therefore, we have that:

$$\max_{\partial_1, \partial_2, \ldots, \partial_k} D(\partial_1, \partial_2, \ldots, \partial_k) = 2 \cdot k \cdot \sin\left(\frac{a}{2 \cdot k}\right)$$

Next, we will calculate the chord between two consecutive disputable announcements, in their maximum distance.

Lemma 4.5. *Let n be the total number of robots, f be the total number of faulty robots and $2 \leq k \leq f + 1$ be the number of disputable announcements of algorithm 5. Then the worst case maximum distance of two consecutive disputable announcements X_j, X_{j+1} is:*

$$\chi = 2 \cdot \sin\left(\frac{d+1}{k-1} \cdot \frac{\pi}{n}\right)$$

where d is the sector distance of X_1, X_k.

Proof. By Lemma 4.4, the worst placement of the announcements by the Adversary is when all the consecutive announcements are equidistant $(d(X_j, X_{j+1}) = d(X_j, X_{j-1}), j \in [2, k-1])$.

The arc distance of X_1, X_k is $(d + 1) \cdot \frac{2\pi}{n}$. Therefore the chord that connects X_j, X_{j+1} has length $\chi = 2 \cdot \sin\left(\frac{d+1}{k-1} \cdot \frac{\pi}{n}\right)$

Corollary 4.1 (Inspector Search Time)**.** *Inspectors need to check $k-1$ announcements in order to know the location of the exit, in the worst case. The time that inspectors need to search is*

$$G_s(k) = 2(k-1) \cdot \sin\left(\frac{f-k+d+2}{k} \cdot \frac{\pi}{n}\right)$$

Proof. By Lemma 4.4, the worst placement of the announcements by the Adversary is when the last inspector and the disputable announcements are equidistant. In that way, the Adversary maximizes the total required search time (and as a result the required evacuation time). By Lemma 4.2 the arc distance of the last inspector and X_k is $(d + (f - k + 2)) \cdot \frac{2\pi}{n} = (f - k + d + 2)\frac{2\pi}{n}$

We immediately gain the following corollary:

Corollary 4.2 (Inspector Evacuation Time)**.** *The time that inspectors need to evacuate is*

$$G_e(k) = 2k \cdot \sin\left(\frac{f-k+d+2}{k} \cdot \frac{\pi}{n}\right)$$

Theorem 4.1 ((n,f) - Evacuation with Byzantine faults (Wireless)). *The worst-case time of algorithm 5 for (n,f) - Evacuation with $n > 2f$ robots and f Byzantine faults in the Wireless model, satisfies:*

$$E(n,f) \leq 1 + (f+1) \cdot \frac{2\pi}{n} + \max\{G_e(k^*), H_e(k^*)\}$$

where,

$$\begin{aligned} H_e(k^*) = 1 + &\sqrt{1 - \sin^2\left((k^* - c - 1)\frac{f - k^* + d + 2}{k^*} \cdot \frac{\pi}{n}\right)} \\ &+ (k^* - c - 1) \cdot \sin\left(\frac{k^* - f + d + 2}{k^*} \cdot \frac{\pi}{n}\right), \end{aligned}$$

$$k^* = \operatorname*{arg\,max}_{k \in \{2, f+1\}} \left(2(k-1)\sin\left(\frac{f - k + d + 2}{k} \cdot \frac{\pi}{n}\right)\right)$$

Proof. First we prove the correctness of Algorithm 5, and then its time complexity:

Correctness: For the correctness of Algorithm 5 it suffices to prove that all non-faulty robots will eventually evacuate. Since every sector of the circle is searched by $(f+1)$ different robots, by the end of round $(f+1)$, the exit is among the disputable announcements. By Lemma 4.2, $f + 2 - k$ inspectors are sufficient to settle all the disputable announcements, hence all non-faulty robots will learn the location of the exit and evacuate.

Time Complexity: The worst case time of Algorithm 5 is analyzed as follows:

All robots move from the center to the perimeter of the circle in 1 time unit and conduct search for $(f+1) \cdot \partial$ time units. Then the inspector robots search for the exit among the disputable announcements and at the same time the honest robots move closer to the candidate exits in order to evacuate. The *inspector* robots by Corollary 4.2 need $G_e(k^*)$ time to evacuate.

The *honest* robots need at worst case H_e time which is analysed as:

- One time unit to get to the middle of the circle. By the time the robot arrives at the center $c = \left\lfloor \frac{1}{2 \cdot \sin \frac{f-k+d+2}{k} \cdot \frac{\pi}{n}} \right\rfloor$ announcements have been approved or disproved.

- The robot moves to the middle point M_{c+1} of the line segment between

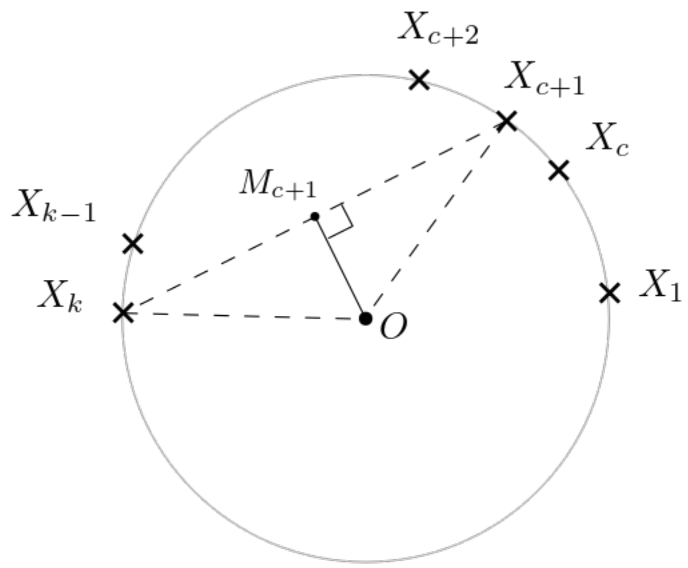

FIGURE 4.6: Honest robots move from the center of the circle to the first middle point M_{c+1}

X_{c+1}, X_k which has distance $\overline{OM_{c+1}} = \sqrt{1 - \sin^2\left((k^* - c - 1)\frac{f-k+d+2}{k} \cdot \frac{\pi}{n}\right)}$ (see Figure 4.6).

- The time needed to move through the middle point M_j of the line segments of X_j, X_k, $j \in \{c+2, k\}$ is $(k^* - c - 1) \cdot \sin\left(\frac{k^*-f+d+2}{k^*} \cdot \frac{\pi}{n}\right)$. We note that the triangles $X_k \overset{\triangle}{X_c} X_{c+1}$ and $X_k \overset{\triangle}{M_c} M_{c+1}$ are similar, since $X_{c+1}\widehat{X_k}X_{c+2} = M_{c+1}\widehat{X_k}M_{c+2}$ and $\frac{\overline{X_kX_{c+1}}}{\overline{X_kM_{c+1}}} = \frac{\overline{X_kX_{c+2}}}{\overline{X_kM_{c+2}}}$ (see Figure 4.7). Therefore, by the similarity of the triangles, it holds that:

$$\frac{\overline{M_{c+1}M_{c+2}}}{\overline{X_{c+1}X_{c+2}}} = \frac{\overline{X_kX_{c+2}}}{\overline{X_kM_{c+2}}} \Rightarrow \overline{M_{c+1}M_{c+2}} = \frac{1}{2}\overline{X_{c+1}X_{c+2}} = \sin\left(\frac{k^* - f + d + 2}{k^*} \cdot \frac{\pi}{n}\right)$$

Similarly, in order to move through the middle points $M_j, j \in \{c+2, k\}$ (Figure 4.7) the time required is:

$$\overline{M_{c+1}M_{c+2}\ldots M_kX_k} = \frac{1}{2}\overline{X_{c+1}X_{c+2}\ldots X_k} = (k^* - c - 1) \cdot \sin\left(\frac{k^* - f + d + 2}{k^*} \cdot \frac{\pi}{n}\right)$$

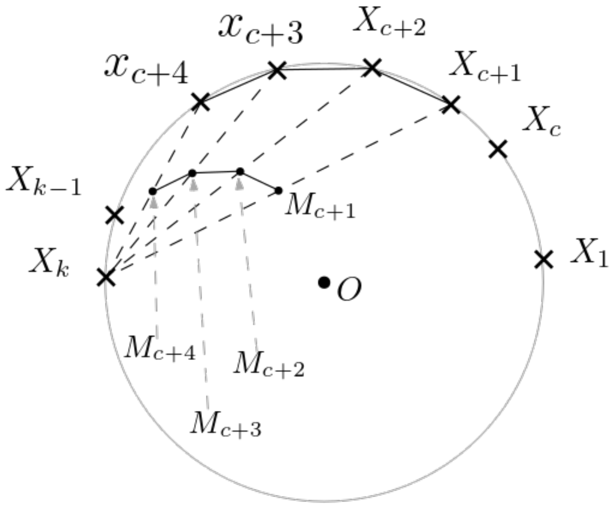

FIGURE 4.7: Final evacuation trajectory of the honest robots, moving through the middle points.

4.2.1 Simulation Results for the Wireless model

To complement our analysis about the performance of Algorithm 5 we simulated the running time as illustrated in Figure 4.8. Its effectiveness is demonstrated under varying ratios of n and f and is further compared with the trivial algorithm.

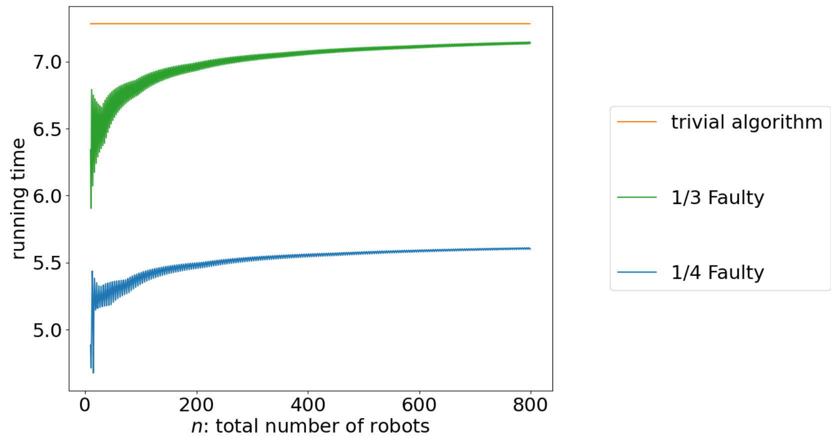

FIGURE 4.8: Total evacuation time using Algorithm 5, in different values of n and f

Notably, our algorithm demonstrates its best performance as the percentage of faulty robots f decreases.

4.3 Evacuating under Face-to-Face Communication

For the evacuation problem of n robots under f Byzantine faults in the Face-to-Face communication model, we present Algorithm 6. We use that algorithm to prove an Upper Bound for the evacuation of n robots, f of which are Byzantine.

In Algorithm 6, robots start at the center of the circle and after searching the circumference of the circle for $f+1$ rounds (Figures 4.9 - 4.10), rendezvous at the center to share their findings (Figure 4.11). Robots make claims about the location of the exit and check the validity of these claims. If they can deduce the location of the exit, they move there to evacuate. If not, in the worst case all honest robots must visit all the disputable claims to evacuate (Figure 4.12). Algorithm 6 is illustrated in Figures 4.9 - 4.12.

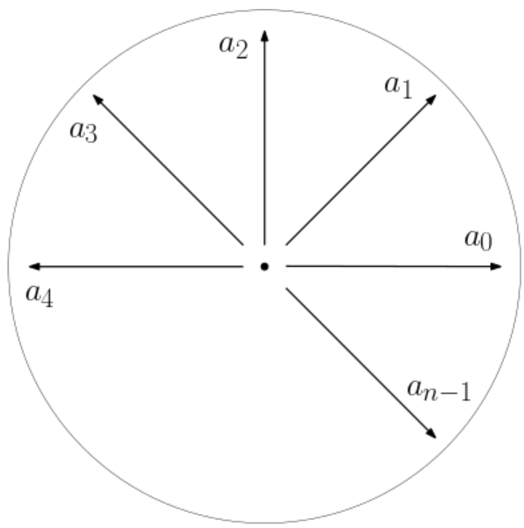

FIGURE 4.9: Each robot a_i move from their starting position (center of the circle) to point $i\partial$ of the circle ($t < 1$)

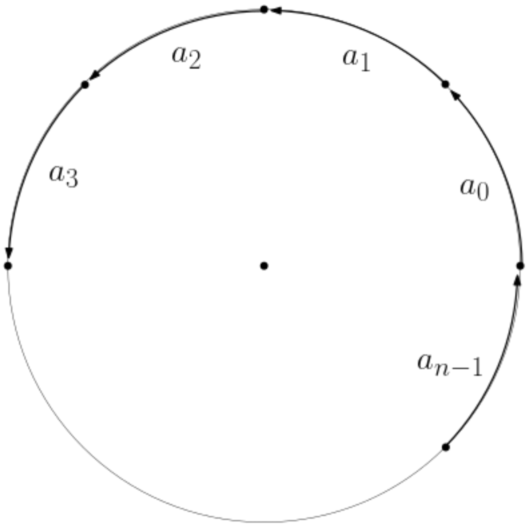

FIGURE 4.10: Robots search the unit circle counter clockwise ($t < 1 + (f + 1)\partial$)

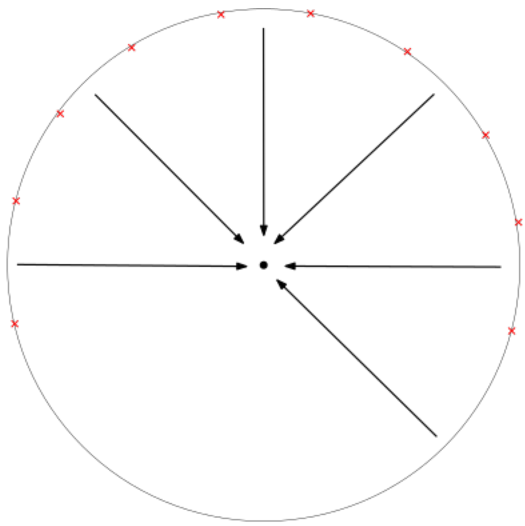

FIGURE 4.11: After f+1 rounds of search, announcements are present. Robots move to the center of the circle do discuss their findings ($t = 1 + (f + 1)\partial$)

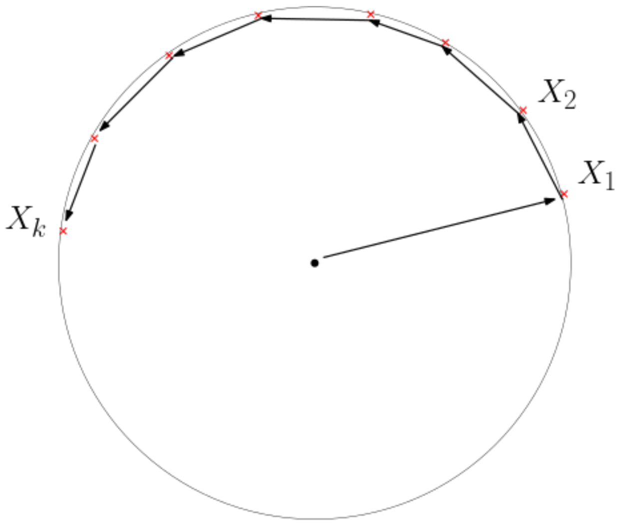

FIGURE 4.12: Evacuation path of non-faulty robots ($t > 1 + (f + 1)\partial$)

Algorithm 6 (n, f)-evacuation with Byzantine faults (F2F)

1: Define $\partial = \frac{2\pi}{n}$.
2: Robot a_i moves along a radius of the circle to the point $i\partial$ of the unit circle and start searching ccw.
3: At time $1 + (f + 1)\partial$: all robots return to the center of the circle.
4: Robots that claim they have found the exit inform the rest of the robots
5: If a consensus about the location of the exit have achieved, all robots move to the exit to evacuate. Otherwise, robots must visit all the disputable claims.

The correctness and time complexity of Algorithm 6 is analyzed in the following theorem.

Theorem 4.2 ((n, f) - Evacuation with Byzantine faults (F2F))**.** *The worst-case time for (n, f) - Evacuation with n robots and f Byzantine faults in the Face-to-Face model, satisfies*

$$E(n, f) \leq 3 + (f + 1) \cdot \frac{2\pi}{n} + \max_{2 \leq k \leq n} \left\{ 2(k - 1) \cdot \sin\left(\frac{f - k + 2}{k - 1} \cdot \frac{\pi}{n} \right) \right\}$$

Proof. We will prove the correctness and the time complexity of Algorithm 6

Correctness: It suffices to prove that all honest robots will eventually evacuate. Since in Algorithm 6 every sector is searched by $(f + 1)$

different robots then, when the robots meet at the center of the circle after the $f + 1$ rounds, the exit is among the disputable announcements. Therefore, by searching all the disputable announcements the robots will eventually find the exit and evacuate.

Time Complexity: The worst case time of Algorithm 6 is analyzed as follows:

- All robots move from the center to the perimeter of the circle in one time unit and conduct search for $(f + 1) \cdot \partial$ time units.
- Then robots return to the center of the circle in one time unit to exchange information about the exit.
- If there is consensus on the exit, the robots move to the exit in one time unit.
- If there are $k \geq 2$ disputable claims, then the robots move to the perimeter of the circle to search all the disputable claims. By Lemmas 4.3, 4.5 the worst case time in order to visit all the claims is

$$(k-1)\chi = 2(k-1) \cdot \sin\left(\frac{f-k+2}{k-1} \cdot \frac{\pi}{n}\right)$$

The upper bound follows.

4.3.1 Comparison with the trivial algorithm

The trivial algorithm for the face to face evacuation requires that every robot searches the perimeter of the circle until they find the exit, and therefore the worse time complexity is $\mathcal{T}(n,f) = 1 + 2\pi$. In this section, we will compare the worse time complexity of Algorithm 6 and the trivial algorithm.

In Lemma 4.6 we prove that if $f \geq 0.384209 \cdot n$ then the trivial algorithm has better performance than algorithm 6 and in Lemma 4.7 we prove that when $n > \frac{2\pi}{\pi-1-2\pi\beta}$ and $0 < \beta < \frac{\pi-1}{2\pi}$ if $f \leq \beta \cdot n$ then Algorithm 6 has better performance than the trivial algorithm.

Lemma 4.6. *If* $0.384209 \cdot n \leq f < n$ *, then the trivial algorithm has better worst time complexity than Algorithm 6.*

Proof. Let $f = \beta \cdot n$, for some $\beta \in \{0, 1\}$. We will prove that if $\beta \geq 0.384209$ the trivial algorithm has better time complexity than

Algorithm 6. In case there are two announcements ($k = 2$) their maximum arc distance, by Lemma 4.3 is:

$$(f - k + 2)\frac{2\pi}{n} = n \cdot \beta \cdot \frac{2\pi}{n} = 2\pi \cdot \beta \tag{4.1}$$

Hence the adversary can place the two announcements with arc distance $2\pi{\cdot}\beta$ so the time needed for the robots to visit the two disputable announcements is $2 \cdot r \cdot \sin\left(\frac{2\beta\pi}{2}\right) = 2 \cdot \sin(\beta \cdot \pi)$

Therefore, in this instance the time complexity of Algorithm 6 is:

$$\begin{aligned} E(n,f) &\geq 3 + 2 \cdot \sin(\beta \cdot \pi) + (f+1) \cdot \frac{2\pi}{n} \\ &= 3 + 2 \cdot \sin(\beta \cdot \pi) + 2\pi \cdot \beta + \frac{2\pi}{n} \\ &> 3 + 2 \cdot \sin(\beta \cdot \pi) + 2\pi \cdot \beta \end{aligned} \tag{4.2}$$

Now we calculate the values of β for which the trivial algorithm is better than Algorithm 6, using the lower bound of Equation 4.2.

$$\mathcal{T}(n,f) \leq 3 + 2 \cdot \sin(\beta \cdot \pi) + 2\pi \cdot \beta \tag{4.3}$$

By solving the above inequality we get that:

$$\beta \geq 0.384209$$

Lemma 4.7. *If $f \leq \beta \cdot n$, for some $0 < \beta < \frac{\pi-1}{2\pi} \approx 0.34$ and $n > \frac{2\pi}{\pi-1-2\pi\beta}$ then Algorithm 6 has better worst time complexity than the trivial algorithm.*

Proof. Since $sin(x) \leq x, \forall \beta \in \mathbb{R}$ we can bound $\max_{2\leq k\leq n}\left\{2(k-1) \cdot \sin\left(\frac{f-k+2}{k-1} \cdot \frac{\pi}{n}\right)\right\}$ in the following way:

$$\begin{aligned} &\max_{2\leq k\leq f+1, f<n} (k-1) \cdot 2 \cdot \sin\frac{f-k+2}{k-1} \cdot \frac{\pi}{n} \\ &\leq \max_{2\leq k\leq f+1, f<n} (k-1) \cdot 2 \cdot \frac{f-k+2}{k-1} \cdot \frac{\pi}{n} \\ &= 2 \cdot \frac{f\pi}{n} \\ &\leq 2 \cdot \beta \cdot \pi \end{aligned}$$

since the maximum is achieved for $k = 2$.

Hence,

$$E(n,f) \leq 3 + (\beta \cdot n + 1) \cdot \frac{2\pi}{n} + 2\beta\pi \tag{4.4}$$

Now we calculate the values of β and n for which the trivial algorithm is worst than the upper bound of Equation 4.4:

$$\begin{aligned} 3 + (\beta \cdot n + 1) \cdot \frac{2\pi}{n} + 2\beta\pi &< \mathcal{T}(n,f) \\ 3 + (\beta \cdot n + 1) \cdot \frac{2\pi}{n} + 2\pi\beta &< 2\pi + 1 \\ \frac{\pi}{n} &< \pi - 1 - 2\pi \cdot \beta \end{aligned}$$

Therefore it should hold that

$$\pi - 1 - 2\pi \cdot \beta > 0 \Rightarrow \beta < \frac{\pi - 1}{2\pi}$$

and that

$$n > \frac{\pi}{\pi - 1 - 2\pi\beta}$$

4.3.2 Simulation Results for the F2F model

In this section, we present the simulation results obtained from our experiments, which aim to evaluate the performance of the proposed Algorithm 6 and compare it with the trivial algorithm. These simulations helped us to evaluate the gap between $f > 0.384209 \cdot n$ and $f < \frac{\pi-1}{2\pi} \approx 0.34$ to complement our analytical results (Lemmas 4.6, 4.7) where we proved that our algorithm outperforms the trivial algorithm.

By analyzing these results, we gain insight that our algorithm has a better performance than the trivial with regard to evacuation time when the number of faulty robots is bounded by one-third of the total n robots.

Figure 4.13 demonstrates a summary of these simulations, where Algorithm 6 is compared against the trivial algorithm, depicted in red. It also depicts in blue, orange and green the performance of our algorithm under different ratios of faulty robots. Note that these

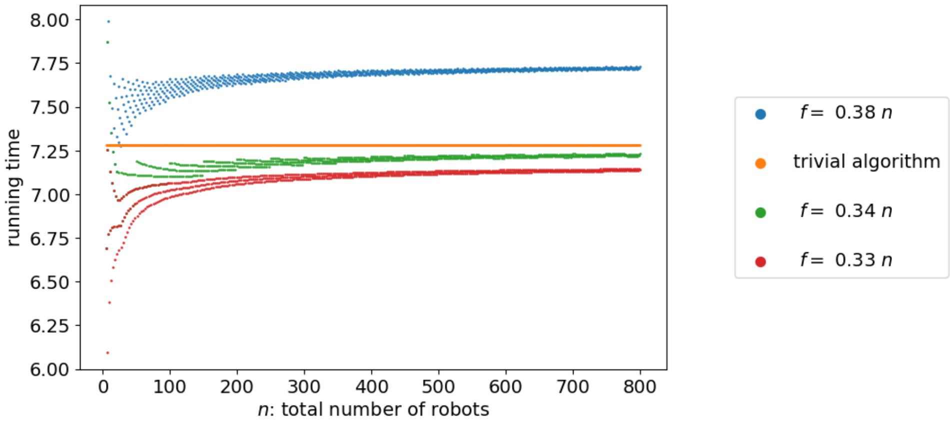

FIGURE 4.13: Total evacuation time of Algorithm 6 for different values of n and f and comparison with the trivial algorithm

ratios are all set close to one-third in order to provide a more precise picture on when exactly our algorithm performs better.

4.4 Conclusion

The study presented in this chapter enhances our understanding of evacuation problems on circular topology and highlights the significance of addressing faulty robots in evacuation algorithms.

We introduce algorithms that cater to the general case of having f Byzantine faults among the n robots. These algorithms are designed for both the wireless and face-to-face communication models, considering the movement capabilities of the robots, which allows them to move anywhere on the platform with a speed of 1. Our proposed algorithms contribute to the field by providing upper bounds in both communication models. Finding a lower bound for these cases and tightening the gap between them is a challenging open question.

Chapter 5

Conclusion

Throughout this thesis, we have explored search and evacuation problems involving autonomous robots on a circular topology, under various fault conditions. Our work covered two main areas: search problems and evacuation problems, each presenting unique challenges.

Considering search problems, our focus was on scenarios with n robots, where up to f robots could be crash-faulty or one robot could exhibit Byzantine behavior. We determined the optimal worst-case search time for f crash faults or a single Byzantine fault in the wireless communication model. We also studied a mixed-case scenario, combining several crash-faulty with one Byzantine-faulty robot, and established an upper bound that slightly deviates from the lower bound.

In our study of evacuation problems, we first addressed scenarios where n robots had to evacuate under the presence of up to two Byzantine faults. We provided algorithms and analyzed their time requirements leading to a lower and an upper bound for the $(n, 1)$-evacuation scenario and an upper bound for the $(n, 2)$-evacuation scenario in both the wireless and the face-to-face communication models. After that, we studied the generalized case of (n, f)-evacuation with n robots, f of which are Byzantine faulty and provided upper bounds also under wireless and the face-to-face communication.

The family of symmetric-persistent algorithms that we explored in our work, can be investigated further, particularly in scenarios that incorporate variable robot capabilities, additional environmental constraints, or optimized performance metrics. Addressing these factors could lead to more robust and efficient search and evacuation strategies, potentially transforming how autonomous systems are deployed in complex and unpredictable environments.

Bibliography

[1] S. Acharjee, K. Georgiou, S. Kundu, and A. Srinivasan. "Lower Bounds for Shoreline Searching With 2 or More Robots". In: *23rd International Conference on Principles of Distributed Systems (OPODIS 2019)*. Ed. by Pascal Felber, Roy Friedman, Seth Gilbert, and Avery Miller. Vol. 153. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020, 26:1–26:11.

[2] R. Ahlswede and I. Wegener. *Search problems*. Wiley-Interscience, 1987.

[3] S. Alpern. "The Rendezvous Search Problem". In: *SIAM Journal on Control and Optimization* 33.3 (1995), pp. 673–683.

[4] S. Alpern and A. Beck. "Asymmetric rendezvous on the line is a double linear search problem". In: *Mathematics of Operations Research* 24.3 (1999), pp. 604–618.

[5] S. Alpern and S. Gal. *The theory of search games and rendezvous*. Vol. 55. Springer, 2003.

[6] R. Baeza Yates, J. Culberson, and G. Rawlins. "Searching in the plane". In: *Information and Computation* 106.2 (1993), pp. 234–252.

[7] R. Baeza-Yates, J. Culberson, and G. Rawlins. "Searching in the Plane". In: *Inf. Comput.* 106.2 (Oct. 1993), pp. 234–252.

[8] I. Bagheri, L. Narayanan, and J. Opatrny. "Evacuation of Equilateral Triangles by Mobile Agents of Limited Communication Range". In: *Algorithms for Sensor Systems*. Ed. by Falko Dressler and Christian Scheideler. Springer, 2019, pp. 3–22.

[9] E. Bampas, J. Czyzowicz, L. Gasieniec, D. Ilcinkas, R. Klasing, T. Kociumaka, and D. Pajak. "Linear Search by a Pair of Distinct-Speed Robots". In: *Algorithmica* 81.1 (2019), pp. 317–342.

[10] E. Bampas, N. Leonardos, E. Markou, A. Pagourtzis, and M. Petrolia. "Improved periodic data retrieval in asynchronous rings with a faulty host". In: *Theoretical Computer Science* 608 (2015). Structural Information and Communication Complexity, pp. 231–254.

[11] A. Beck. "On the linear search problem". In: *Israel Journal of Mathematics* 2.4 (1964), pp. 221–228.

[12] A. Beck and M. Beck. "Son of the linear search problem". In: *Israel Journal of Mathematics* 48.2-3 (1984), pp. 109–122.

[13] A. Beck and M. Beck. "The revenge of the linear search problem". In: *SIAM journal on control and optimization* 30.1 (1992), pp. 112–122.

[14] A. Beck and D. Newman. "Yet more on the linear search problem". In: *Israel Journal of Mathematics* 8.4 (1970), pp. 419–429.

[15] A. Beck and P. Warren. "The return of the linear search problem". In: *Israel Journal of Mathematics* 14.2 (1973), pp. 169–183.

[16] P. Behrouz, O. Konstantinidis, N. Leonardos, A. Pagourtzis, I. Papaioannou, and M. Spyrakou. "Byzantine Fault-Tolerant Protocols for (n, f)-Evacuation from a Circle". In: *Algorithmics of Wireless Networks*. Ed. by Konstantinos Georgiou and Evangelos Kranakis. Cham: Springer Nature Switzerland, 2023, pp. 87–100. ISBN: 978-3-031-48882-5.

[17] R. Bellman. "An optimal search". In: *Siam Review* 5.3 (1963), pp. 274–274.

[18] M. A. Bender, A. Fernández, D. Ron, A. Sahai, and S. Vadhan. "The Power of a Pebble: Exploring and Mapping Directed Graphs". In: *Information and Computation* 176.1 (2002), pp. 1–21.

[19] S. Bhatt, S. Even, D. Greenberg, and R. Tayar. "Traversing Directed Eulerian Mazes." In: *J. Graph Algorithms Appl.* 6 (Jan. 2002), pp. 157–173.

[20] A. Bonato, K. Georgiou, C. MacRury, and P. Pralat. "Probabilistically Faulty Searching on a Half-Line". In: *14th Latin American Theoretical Informatics Symposium, Lecture Notes in Computer Science* (2020).

[21] P. Bose and J.-L. De Carufel. "A general framework for searching on a line". In: *Theoretical Computer Science* 703 (2017), pp. 1–17.

[22] S. Brandt, K.-T. Foerster, B. Richner, and R. Wattenhofer. "Wireless evacuation on m rays with k searchers". In: *Theoretical Computer Science* 811 (2020). Special issue on Structural Information and Communication Complexit, pp. 56–69.

[23] S. Brandt, F. Laufenberg, Y. Lv, D. Stolz, and R. Wattenhofer. "Collaboration without communication: Evacuating two robots from a disk". In: *International Conference on Algorithms and Complexity* (2017), pp. 104–115.

[24] S. Brandt, J. Uitto, and R. Wattenhofer. "A Tight Lower Bound for Semi-Synchronous Collaborative Grid Exploration". In: *32nd International Symposium on Distributed Computing (DISC 2018)*. Ed. by Ulrich Schmid and Josef Widder. Vol. 121. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018, 13:1–13:17.

[25] P. Brass, F. Cabrera-Mora, A. Gasparri, and J. Xiao. "Multi-robot tree and graph exploration". In: *IEEE Transactions on Robotics* 27.4 (2011), pp. 707–717.

[26] J. Chalopin, S. Das, A. Labourel, and E. Markou. "Tight bounds for black hole search with scattered agents in synchronous rings". In: *Theoretical Computer Science* 509 (2013). Structural Information and Communication Complexity, pp. 70–85.

[27] M. Chrobak, L. Gasieniec, D. Krizanc, T. Gorry, and R. Martin. "Linear search with terrain-dependent speeds". In: *Current trends in theory and practice of computer science, SOFSEM*. Vol. 8939. Lecture Notes in Computer Science. Springer, 2015, pp. 164–176.

[28] H. Chuangpishit, K. Georgiou, and P. Sharma. "A Multi-Objective Optimization Problem on Evacuating 2 Robots from the Disk in the Face-to-Face Model; Trade-Offs between Worst-Case and Average-Case Analysis". In: *Information* 11.11 (2020).

[29] H. Chuangpishit, S. Mehrabi, L. Narayanan, and J. Opatrny. "Evacuating equilateral triangles and squares in the face-to-face model". In: *Computational Geometry* 89 (2020), p. 101624.

[30] L. Cohen, Y. Emek, O. Louidor, and J. Uitto. "Exploring an infinite space with finite memory scouts". In: *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '17. Barcelona, Spain: Society for Industrial and Applied Mathematics, 2017, pp. 207–224.

[31] J Czyzowicz, K. Georgiou, R. Killick, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. Shende. “Priority evacuation from a disk: The case of $n = 1, 2, 3$”. In: *Theoretical Computer Science* 806 (2020), pp. 595–616.

[32] J. Czyzowicz, L. Gasieniec, T. Gorry, E. Kranakis, R. Martin, and D. Pajak. “Evacuating Robots via Unknown Exit in a Disk”. In: *Distributed Computing - 28th International Symposium, DISC 2014, Proceedings*. Ed. by Fabian Kuhn. Vol. 8784. Lecture Notes in Computer Science. Springer, 2014, pp. 122–136.

[33] J. Czyzowicz, K. Georgiou, M. Godon, E. Kranakis, D. Krizanc, W. Rytter, and M. Włodarczyk. “Evacuation from a disc in the presence of a faulty robot”. In: *International Colloquium on Structural Information and Communication Complexity*. Springer. 2017, pp. 158–173.

[34] J. Czyzowicz, K. Georgiou, R. Killick, E. Kranakis, D. Krizanc, M. Lafond, L. Narayanan, J. Opatrny, and S. Shende. “Energy Consumption of Group Search on a Line”. In: *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*. Ed. by Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi. Vol. 132. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2019, 137:1–137:15.

[35] J. Czyzowicz, K. Georgiou, R. Killick, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. Shende. “God Save the Queen”. In: *9th International Conference on Fun with Algorithms (FUN 2018)*. Ed. by Hiro Ito, Stefano Leonardi, Linda Pagli, and Giuseppe Prencipe. Vol. 100. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018, 16:1–16:20.

[36] J. Czyzowicz, K. Georgiou, R. Killick, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. Shende. “Priority evacuation from a disk: The case of $n \geq 4$”. In: *Theoretical Computer Science* 846 (2020), pp. 91–102.

[37] J. Czyzowicz, K. Georgiou, and E. Kranakis. “Group Search and Evacuation”. In: *Distributed Computing by Mobile Entities; Current Research in Moving and Computing*. Ed. by P. Flocchini, G. Prencipe, and N. Santoro. Springer, 2019. Chap. 14, pp. 335–370.

[38] J. Czyzowicz, K. Georgiou, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. Shende. “Search on a Line by Byzantine Robots”. In: *ISAAC*. 2016, 27:1–27:12.

[39] J. Czyzowicz, K. Georgiou, E. Kranakis, L. Narayanan, J. Opatrny, and B. Vogtenhuber. “Evacuating Robots from a Disk Using Face-to-Face Communication”. In: *Discrete Mathematics & Theoretical Computer Science*. Vol. 22 (4). 2020.

[40] J. Czyzowicz, E. Kranakis, D. Krizanc, L. Narayanan, and J. Opatrny. “Search on a line with faulty robots”. In: *Distributed Computing* 32.6 (2019), pp. 493–504.

[41] J. Czyzowicz, E. Kranakis, D. Krizanc, L. Narayanan, and J. Opatrny. “Search on a line with faulty robots”. In: *Distributed Computing* 32.6 (2019), pp. 493–504.

[42] J. Czyzowicz, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, and S. M. Shende. “Linear search with terrain-dependent speeds”. In: *Algorithms and Complexity - 10th International Conference, CIAC 2017, Athens, Greece, May 23-26*. Ed. by Aris Pagourtzis Dimitris Fotakis and Vangelis Th. Paschos. Vol. 10236. Lecture Notes in Computer Science. Springer, 2017, pp. 430–441.

[43] J. Czyzowicz, E. Kranakis, K. Krizanc, L. Narayanan, J. Opatrny, and S. Shende. “Wireless Autonomous Robot Evacuation from Equilateral Triangles and Squares”. In: *ADHOCNOW*. Springer, 2015, pp. 181–194.

[44] S. Das, P. Flocchini, S. Kutten, A. Nayak, and N. Santoro. “Map construction of unknown graphs by multiple agents”. In: *Theoretical Computer Science* 385.1 (2007), pp. 34–48.

[45] S. Das, R. Focardi, F. L. Luccio, E. Markou, and M. Squarcina. “Gathering of robots in a ring with mobile faults”. In: *Theoretical Computer Science* 764 (2019). Selected papers of ICTCS 2016 (The Italian Conference on Theoretical Computer Science (ICTCS), pp. 42–60.

[46] E. D. Demaine, S. P. Fekete, and S. Gal. “Online searching with turn cost”. In: *Theoretical Computer Science* 361.2 (2006), pp. 342–355.

[47] E. D. Demaine, S. P. Fekete, and S. Gal. “Online searching with turn cost”. In: *Theoretical Computer Science* 361.2 (2006). Approximation and Online Algorithms, pp. 342–355.

[48] Y. Disser and S. Schmitt. "Evacuating two robots from a disc: a second cut". In: *International Colloquium on Structural Information and Communication Complexity* (2019), pp. 200–214.

[49] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. "Robotic exploration as graph construction". In: *IEEE Transactions on Robotics and Automation* 7.6 (1991), pp. 859–865.

[50] O. Feinerman, A. Korman, Z. Lotker, and J.-S. Sereni. "Collaborative search on the plane without communication". In: *Proceedings of the 2012 ACM Symposium on Principles of Distributed Computing*. PODC '12. Madeira, Portugal: Association for Computing Machinery, 2012, pp. 77–86.

[51] P. Fraigniaud, L. Gasieniec, D. R. Kowalski, and A. Pelc. "Collective tree exploration". In: *Networks* 48.3 (2006), pp. 166–177.

[52] P. Fraigniaud and D. Ilcinkas. "Digraphs Exploration with Little Memory". In: *STACS 2004*. Ed. by Michel habib Volker Diekert. Vol. 2996. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2004, pp. 246–257.

[53] P. Fraigniaud, D. Ilcinkas, S. Rajsbaum, and S. Tixeuil. "Space Lower Bounds for Graph Exploration via Reduced Automata". In: *SIROCCO 2005*. Ed. by Michel Raynal Andrzej Pelc. Vol. 3499. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2005, pp. 140–154.

[54] B. Fuchs, W. Hochstattler, and W. Kern. "Online matching on a line". In: *Theoretical Computer Science* 332 (2005).

[55] S. Gal. "Minimax solutions for linear search problems". In: *SIAM Journal on Applied Mathematics* 27 (1974), pp. 17–30.

[56] S. Gal. "Rendezvous Search on the Line". In: *Operations Research* 47.6 (1999), pp. 974–976.

[57] K. Georgiou, N. Giachoudis, and E. Kranakis. "Evacuation from a Disk for Robots with Asymmetric Communication". In: *33rd International Symposium on Algorithms and Computation (ISAAC 2022)*. Vol. 248. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022, 19:1–19:16.

[58] K. Georgiou, G. Karakostas, and E. Kranakis. "Search-and-Fetch with 2 Robots on a Disk: Wireless and Face-to-Face Communication Models". In: *Discrete Mathematics Theoretical Computer Science* 21.3 (2019).

[59] K. Georgiou, G. Karakostas, and E. Kranakis. "Treasure evacuation with one robot on a disk". In: *Theoretical Computer Science* 852 (2021), pp. 18–28.

[60] K. Georgiou, E. Kranakis, N. Leonardos, A. Pagourtzis, and I. Papaioannou. "Optimal Circle Search Despite the Presence of Faulty Robots". In: *Algorithms for Sensor Systems - 15th International Symposium on Algorithms and Experiments for Wireless Sensor Networks, ALGOSENSORS 2019, Revised Selected Papers*. Ed. by Falko Dressler and Christian Scheideler. Vol. 11931. Lecture Notes in Computer Science. Springer, 2019, pp. 192–205.

[61] K. Georgiou, E. Kranakis, N. Leonardos, A. Pagourtzis, and I. Papaioannou. "Optimal Circle Search Despite the Presence of Faulty Robots". In: *Information Processing Letters* 182 (2023), p. 106391.

[62] K. Georgiou, S. Leizerovich, J. Lucier, and S. Kundu. "Evacuating from l_p Unit Disks in the Wireless Model". In: *Theoretical Computer Science* (2022).

[63] K. Georgiou and J. Lucier. "Weighted Group Search on a Line: (Extended Abstract)". In: *Algorithms for Sensor Systems: 16th International Symposium on Algorithms and Experiments for Wireless Sensor Networks, ALGOSENSORS 2020, Pisa, Italy, September 9-10, 2020, Revised Selected Papers*. Pisa, Italy: Springer-Verlag, 2020, pp. 124–139.

[64] Q. Han, D. Du, J. Vera, and L. F. Zuluaga. "Improved Bounds for the Symmetric Rendezvous Value on the Line". In: *Operations Research* 56.3 (2008), pp. 772–782.

[65] J. Hromkovič, R. Klasing, B. Monien, and R. Peine. "Dissemination of Information in Interconnection Networks (Broadcasting & Gossiping)". In: *Combinatorial Network Theory*. Ed. by Ding-Zhu Du and D. Frank Hsu. Boston, MA: Springer US, 1996, pp. 125–212.

[66] A. Jeż and J. Łopuszański. "On the two-dimensional cow search problem". In: *Information Processing Letters* 109.11 (2009), pp. 543–547.

[67] M.-Y. Kao and M. L. Littman. "Algorithms for informed cows". In: *Proceedings of the AAAI 1997 Workshop on Online Search*. 1997.

[68] M.-Y. Kao, J. H. Reif, and S. R. Tate. “Searching in an unknown environment: An optimal randomized algorithm for the cow-path problem”. In: *Information and Computation* 131.1 (1996), pp. 63–79.

[69] R. Klasing, E. Markou, and A. Pelc. “Gathering asynchronous oblivious mobile robots in a ring”. In: *Theoretical Computer Science* 390.1 (2008), pp. 27–39.

[70] E. Kranakis, D. Krizanc, and E. Markou. “Deterministic symmetric rendezvous with tokens in a synchronous torus”. In: *Discrete Applied Mathematics* 159.9 (2011), pp. 896–923.

[71] A. Kupavskii and E. Welzl. “Lower Bounds for Searching Robots, some Faulty”. In: July 2018, pp. 447–453.

[72] L. Lamport, R. Shostak, and M. Pease. “The Byzantine Generals Problem”. In: *ACM Trans. Program. Lang. Syst.* 4.3 (July 1982), pp. 382–401.

[73] I. Lamprou, R. Martin, and S. Schewe. “Fast two-robot disk evacuation with wireless communication”. In: *International Symposium on Distributed Computing* (2016), pp. 1–15.

[74] N. Leonardos, A. Pagourtzis, and I. Papaioannou. “Byzantine fault tolerant symmetric-persistent circle evacuation”. In: *Algorithms for Sensor Systems: 17th International Symposium on Algorithms and Experiments for Wireless Sensor Networks, ALGOSENSORS 2021, Lisbon, Portugal, September 9–10, 2021, Proceedings*. Lisbon, Portugal: Springer-Verlag, 2021. ISBN: 978-3-030-89239-5.

[75] N. Leonardos, A. Pagourtzis, and I. Papaioannou. “Byzantine fault tolerant symmetric-persistent circle evacuation”. In: *Theoretical Computer Science* 956 (2023), p. 113839.

[76] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers Inc., 1996.

[77] E. Markou, N. Nisse, and S. Pérennes. “Exclusive graph searching vs. pathwidth”. In: *Information and Computation* 252 (2017), pp. 243–260.

[78] E. Markou and A. Pelc. “Efficient Exploration of Faulty Trees”. In: *Theory Comput Syst* 40 (2007), pp. 225–247.

[79] A. R. Molla, K. Mondal, and W. K. Moses. “Byzantine Dispersion on Graphs”. In: *2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 2021, pp. 942–951.

[80] A. R. Molla, K. Mondal, and W. K. Moses. “Efficient Dispersion on an Anonymous Ring in the Presence of Weak Byzantine Robots”. In: *Algorithms for Sensor Systems*. Ed. by Cristina M. Pinotti, Alfredo Navarra, and Amitabha Bagchi. Cham: Springer International Publishing, 2020, pp. 154–169.

[81] A. R. Molla, K. Mondal, and W. K. Moses. “Optimal dispersion on an anonymous ring in the presence of weak Byzantine robots”. In: *Theoretical Computer Science* 887 (2021), pp. 111–121.

[82] D. Pattanayak, H. Ramesh, P. S. Mandal, and Schmid S. “Evacuating two robots from two unknown exits on the perimeter of a disk with wireless communication”. In: *Proceedings of the 19th International Conference on Distributed Computing and Networking* (2018), pp. 1–4.

[83] K. Spieser and E. Frazzoli. “The Cow-Path Game: A Competitive Vehicle Routing Problem”. In: *51st IEEE Conference on Decision and Control (CDC)* (2012), pp. 6513–6520.

[84] L. Stone. *Theory of optimal search*. Academic Press New York, 1975.