UNIVERSITY OF SOUTHAMPTON
Faculty of Engineering and Physical Sciences
Electronics and Computer Science

Efficiency and Fairness of Resource Utilisation under Uncertainty

by

Jan Buermann
ORCID ID: 0000-0002-4981-6137

A thesis submitted in partial fulfilment for
the degree of Doctor of Philosophy

Supervisors: Doctor Jie Zhang and Doctor Enrico Gerding
and Doctor Baharak Rastegari
Examiners: Professor Carmine Ventre and Doctor Enrico
Marchioni

June 2022

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
ELECTRONICS AND COMPUTER SCIENCE

A thesis submitted in partial fulfilment for the degree of Doctor of Philosophy

EFFICIENCY AND FAIRNESS OF RESOURCE UTILISATION UNDER UNCERTAINTY

by Jan Buermann

The efficient use of resources is a crucial problem of our time. Besides the constraints of efficient usage of scarce resources, in real-world problems the ubiquitous constraint of uncertainty further affects the use or distribution of most resources. Solution approaches are problem-dependent and have various benefits and difficulties. In this work we examine these benefits and difficulties in two different settings of uncertainty, both with their own benefits and difficulties. Moreover, we address the two problems using different techniques applicable to other settings.

In the first problem the uncertainty is with respect to the resource itself. In the well-studied problem of fair multi-agent resource allocation it is generally assumed that the quantity of each resource is known a priori. However, in many real-world problems, such as the production of renewable energy which is typically weather-dependent, the exact amount of each resource may not be known at the time of decision making. This work investigates the fair division of a homogeneous, divisible resource where the available amount is given by a probability distribution. Specifically, the notion of ex-ante envy-freeness, where, in expectation, agents weakly prefer their allocation over every other agent's allocation is considered. This work shows how uncertainty changes the relationship of fairness and efficiency, how the solution space is affected, how difficult the problem becomes, and gives algorithms for the case of two agents with utility function that are linear up to a maximal value. This is achieved by showing how in expectation a higher efficiency can be attained; the worst case might still affect the results; and that the problem is strongly NP-hard. Additionally, we provide two variants of a greedy algorithm for the case of two agents. One variant is optimal for the case of uniform probability distributions over the events. For the case of arbitrary probability distributions, we show that this problems is also NP-hard. Accordingly, we address the possibility of approximation. We show that one variant is not able to approximate all instances. Nevertheless, we show empirically that for realistic instances both variants of the algorithm can approximate the optimal solution. Hence the work lays the foundation for further research into homogeneous resources under uncertainty.

In the second problem, the uncertainty comes from the behaviour of an agent and this behaviour is countered by random strategies. In full-knowledge multi-robot adversarial patrolling, a group of robots have to detect an adversary who knows the robots' strategy. The adversary can easily take advantage of any deterministic patrolling strategy, which necessitates the employment of a randomised strategy. Previous algorithms have to be repeated to calculate the solution for different instances and lack insight into the strategy space. In comparison, this work shows how

enumerative combinatorics can be used to provide the closed formulae of the probabilities of detecting the adversary. Hence, it facilitates characterising optimal random defence strategies in comparison to formerly used iterative black-box models. We provide the probability functions for four cases based on open and closed polylines using two different robot movement patterns. Moreover, we show how analysing the structure of the strategy space can further reduce the runtime. Hence, the work introduces a new technique into adversarial patrolling that can be used to improve runtime and foster further research.

In conclusion, the work provides progress in two established research areas, and highlights the potential and importance of the consideration of the effects of uncertainty. Foremost, including uncertainty opens up research which is more attuned to real-world problems. Additionally, addressing these problems, including with novel approaches, allows finding (computationally) more efficient solutions.

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

General

$\lvert\cdot\rvert$	The cardinality of a set
$\mathrm{E}_{\mathcal{D}}[X]$	The expected value of random variable X over probability distribution $\mathcal{D}$
$f : X \to Y$	Definition of a function with domain X and codomain Y
$f : x \mapsto y$	Definition of the value of a function with $y = f(x)$
$\underset{x\in X}{\mathrm{mean}}\{x\}$	The average value of a set, i.e. $\underset{x\in X}{\mathrm{mean}}\{x\} := \frac{\sum_{x\in X} x}{\lvert X\rvert}$ for finite $X \subsetneq \mathbb{R}$
$[n]$	The set $\{1, 2, \dots, n\}$
$\mathbb{R}_+$	The positive real number including zero, i.e. $\mathbb{R}_+ = [0, +\infty)$
X_{-i}	A set of sets without the i-th set, i.e. $X_{-i} = \{X_1, X_2, \dots, X_n\}\backslash\{X_i\}$ with $X_i \subset \mathbb{R}^{m_i}$, $X = \{X_1, X_2, \dots, X_n\}$ with $n \in \mathbb{N}$ and $m_i \in \mathbb{N}$ for all $i \in [n]$
$(x_i)_{i\in I}$	The vector of $x_i \in Y$ for any set Y and $i \in I$ with any countable set I
Y^X	The set of all possible functions f with $f : X \to Y$
Y^n	The Cartesian product of n times the set Y, i.e. $Y^n = \underbrace{Y \times \dots \times Y}_{n\ times}$

Chapter 3

$A = (a_1, \dots, a_n)$	Allocation to all n agents
$a_i : \Omega \to [0, 1]$	Allocation function for agent i
$\mathcal{F}$	The set of all valid allocation functions, $\mathcal{F} \subset [0, 1]^{\Omega}$
Λ	The set of all valid allocations, $\Lambda = \mathcal{F}^n$
m	The number of events, $m = \lvert\Omega\rvert$
n	The number of agents
Ω	The available events
ω	Overloaded notation: one event $\omega \in \Omega$ as well as the available resource, i.e. $\omega := X(\omega)$
Θ	The set of all valid utility functions, $\Theta \subset \mathbb{R}^{[0,1]}$
$v_i : [0, 1] \to \mathbb{R}$	The utility function of agent i
$V_i : \mathcal{F} \to \mathbb{R}$	The utility of agent i
$W(A)$	The (ex-ante) social welfare of allocation A, $W(A) := \sum_{i\in[n]} V_i(a_i)$
$X : \Omega \to [0, 1]$	The random variable representing the amount of the events

Chapter 3 - Complexity

$\hat{s}_i$	Saturation amount of agent i
$a_i^p : [m] \to [0, 1]$	The function determining the size of a piece for an allocation by pieces
$\Phi : [n] \times \Omega \to [m]$	The mapping determining in which event the agents' pieces are allocated

	for an allocation by pieces

Chapter 3 - Two Agents Linear Satiable Utility Functions

$a_i^j(\omega)$	The allocation of agent i in event ω restricted to the saturation amount of agent j, i.e. $a_i^j(\omega) = \min\{a_i(\omega), q_j\}$, for all events $\omega \in \Omega$
$EF(i,k)$	The envy-freeness of agent $i \in [n]$ with respect to agent $k \in [n]$
k_ω^A	The indifferent amount in ω for allocation A
$\mathcal{L}_\mathcal{M}$	The set of all linear satiable utility function
Ω_1, Ω_1^{opt}	The events preferentially given to the second agent for the allocation produced by GreedyTake-Amt and the optimal allocation, respectively
Ω_2, Ω_2^{opt}	The events preferentially given to the first agent for the allocation produced by GreedyTake-Amt and the optimal allocation, respectively
Ω_1^F, Ω_2^F	The flip set of Ω_1/Ω_2^{opt} and Ω_2/Ω_1^{opt}, respectively
q_i	The saturation amount of agent i
u_i	The maximal value of agent i

Chapter 4

$C(i,j)$	The entry of Catalan's triangle [79] determined by $i, j \in \mathbb{N}$
$dist(j,j')$	The distance between segment j and j' (depends on the underlying graph)
$dist$	The distance to a specified j segment, i.e. $dist := dist(1,j)$, in the case of a circle
$Pr(j',j,t,d)$	The probability that a robot in segment j' detects the adversary in segment j within t steps on a polyline with d segments.
P_W	The probability of a robot walking path ω, i.e. $P_W = P_W(\omega)$ where ω is clear from the context

Acronyms

AMT	GreedyTake-Amt
ES	Equal Share
EXP	GreedyTake-Exp
PMF	Probability Mass Function
PoEF	Price of Envy-Freeness
RR	Research Requirement

Declaration of Authorship

I, Jan Buermann, declare that this thesis entitled Efficiency and Fairness of Resource Utilisation under Uncertainty and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as: Jan Buermann, Enrico H. Gerding, and Baharak Rastegari. 2020. Fair Allocation of Resources with Uncertain Availability. In Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9-13, 2020, IFAAMAS, 9 pages.,
   Jan Buermann and Jie Zhang. 2020. Multi-Robot Adversarial Patrolling Strategies via Lattice Paths. In Proc. of the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020), Yokohama, Japan, July 11-17, IJCAI, 7 pages. .

Signed: Jan Buermann

Date: 17 June 2022

Acknowledgements

Firstly, I would like to thank all of my supervisors, Jie Zhang, Enrico H. Gerding and Baharak Rastegari, for all their support and feedback that they willingly provided during my PhD. I would like to thank Jie Zhang for his time and guidance, particularly in the exploration of problems and developing solutions. I want to thank Enrico H. Gerding for his patience when giving feedback, as well as facilitating my independence and supporting my career progression. I also want to thank Baharak Rastegari for willingly joining my supervisory panel and supporting my growth in communicating my findings with clarity and specificity. I also would like to thank all of my examiners/assessors for taking the time to read my reports and thesis, and for their input to improve these.

Secondly, I want to express my thanks to my PhD peers who were with me throughout this journey, particularly Schuh and Fan. Thank you for always being around and keeping the work atmosphere light, moaning about upcoming mandatory milestones and making days in the office (and Teams meetings!) enjoyable.

Drittens möchte ich mich bei meinen Eltern bedanken. An meine Mutter: Danke, dass du immer für mich da bist und mich ermutigt hast mein Bestes zu geben. An meinen Vater: Danke, dass du mich immer ermutigt hast, meine Identität zu finden und meine Interessen zu verfolgen. Danke, dass ihr mir geholfen habt, die beste Version von mir selbst zu sein und daran zu glauben, dass ich alles schaffen kann, selbst wenn alles was ich gesagt habe vermutlich wie Unsinn geklungen hat!

Fourthly, I also want to thank my parents-in-law for supporting me in applying and getting onto the PhD, foremost by wholeheartedly taking me into their family and allowing me to settle into the journey. I really appreciated it.

Finally, I want to thank my perfect wife for being in my life and for her enduring support. Integral to my life and my PhD journey - the most extraordinarily gracious person. Always encouraging me but also critiquing my work to help me grow and become a better person every day. Thank you for helping me through the PhD valley and for making the joyous days exceptional. Thank you for always supporting and encouraging me even when you were stressed with your own PhD work. I love you very much.

Chapter 1

Introduction

Every day we encounter countless situations in which we have to make complex decisions without having all the relevant information [57, 64]. This lack of information or *uncertainty* is a prevalent feature of many real-world problems. Some problems have limited consequences like, 'is a coat required because it will rain later?', 'will the next dice roll be favourable to win the game?' or 'will the journey to work take longer today because the roads are congested?'. Others have more serious consequences like 'how much does a potential buyer of a car value the car?' or 'are burglars going to break into someone's house?'. Finally, some problems affect wider circles and have a much higher impact like 'will particular shares increase or decrease?' or 'who will win the next election?'. As can be seen, uncertainty comes from different sources. Two common sources are the environment, (e.g. the weather) and the behaviour of other people (e.g. their opinions or values) [1, 36, 60, 64]. Both of these require different strategies to deal with or mitigate the effects of uncertainty in order to make effective decisions. In this thesis we explore the different aspects of taking uncertainty into consideration in decision making settings, and examine the benefits and difficulties in two example settings. We choose these examples since they deal with fundamentally different aspects of uncertainty, and the tools to address these have applications in many other settings.

In the first setting, a group of neighbours want to contribute to the efforts to save the environment by communally sharing electricity from a photovoltaic system fairly among themselves. This is closely linked to the development of a smarter energy grid which is less centralised and better able to shift and react to demands. The decentralised energy production requires an increase and adoption of renewable energy systems, and local energy markets [56]. Specifically, this problem focuses on the local component of such a system, i.e. a local energy exchange in comparison to an exchange via the wider grid. Moreover, local electricity markets are more communal which increases the prominence of considerations like fairness [55]. More generally, this is the problem of dividing a homogeneous resource in a fair manner. Hence, besides the allocation of electricity this problem has a wide range of applications such as the division of estate, storage space, bandwidth or processing time [36, 63, 82] and the techniques we discuss in this thesis can be applied to other important areas including emission permits for greenhouse gases [5], fair load shedding [62], and uncertain computational resources [9, 45].

With regards to the specific setting, the neighbours are faced with two immediate problems. The first one is about allocating energy in a fair manner. There are many ways to define fairness, but a common approach is to ensure that no participant would prefer another participant's allocation over their own. This is a common fairness notion called *envy-freeness.* The second problem is that energy generation from renewable energy sources is less reliable and predictable than that from conventional sources like fossil and nuclear fuels. In comparison to those, the photovoltaic system will not always produce enough energy to satisfy the needs of all neighbours [52, 80].

In response to the uncertainty we could decide on the allocation at production time or *ex-post.* However, an *ex-post* envy-free allocation often necessitates that everyone gets the same amount of energy, irrespective of valuations [36]. The reason is that for a homogeneous resource only the amount is of importance and as long as an agent is not satisfied a higher allocation is always better. Moreover, an ex-post allocation gives the neighbours no opportunity to plan ahead and arrange shifting their activity or buy electricity in advance from the market as a possible reaction to shortcomings. Another approach is to take the worst-case assumption [65]. Although, this is an appropriate approach for emergency situations it might not be very efficient if the actual amount of energy is substantially different from this worst case. Instead we utilise that there are accurate methods of predicting the production of renewable energies [48], e.g. possible solar energy output based on weather forecasts. This allows us to decide on the allocation beforehand or *ex-ante* conditional on the actually produced amount. As we will show in this thesis, an allocation conditional on the available amounts allows for a more flexible approach to fairness which we refer to as *ex-ante envy-freeness.* Therefore, the overall aim for this problem is to ensure a sensible (i.e. efficient) allocation whilst also meeting this fairness condition.

In the second setting, a defender is trying to use a number of robots to defend a perimeter or fence against an *adversary* trying pass the perimeter or fence; a setting known as *multi-robot adversarial patrolling.* This problem is well established and has numerous security applications including crime prevention [6], stopping piracy [61], defending critical infrastructure [61], and protecting animals, natural reserves or the environment [14]. In these settings robots have the benefit of providing a cheaper and mobile addition to assist humans in monitoring vast open areas as well as being more reliable [1, 6, 67].

In this setting, the defender is confronted with uncertainty over the location where the adversary tries to pass the perimeter or fence. Ultimately, taking the defender's point of view, we lack the knowledge of the adversary's reasoning for choosing a location neither do we know the adversary's capabilities. This knowledge of the adversary's reasoning and capabilities is an important issue of adversarial patrolling which forms the basis of determining the right strategy, and further informs on the possibility to detect penetration attempts in general. Addressing this, we contribute to the investigation on what is possible against the strongest adversary and take the frequently considered assumption that the adversary has learned, e.g. via observation, the robots' strategies. However, to have a chance of detecting such a so called *full-knowledge adversary* we have to resort to random strategies [1, 68]. Hence, the goal is to find an optimal random strategy which is one that minimises the risk of the adversary successfully passing the perimeter or fence.

Regarding the environment, polylines represent a real-world setting and, at the same time, allow the calculation of optimal strategies in polynomial time. Polyline graphs, where the vertices of the graph form either a line (open polyline) or a circle (closed polyline), model the real-world environment of patrolling either a fence or a perimeter, respectively[1, 68]. In this polyline model the robots walk on the vertices of the graph and the adversary attacks one vertex. For the random walk strategies informing the walk of the robot, an often considered sub-class are those where the next step of a robot depends only on the position of the robot and not on any previous locations. Those strategies are said to satisfy the *Markov property* and are also called *memoryless*. The significance of this setting is that it performs well in comparison to longer histories [1, 68]. Moreover, in comparison to the usual intractable problems of security settings in general, and patrolling in particular, the considered setting allows tractable results [1, 44, 74].

However, the current approach to determine optimal strategies has an unnecessary runtime which also hinders further improvement. In more detail, the current algorithms to determine an optimal strategy use Markov chain modelling to determine the probability functions that express the possible strategies [1, 73, 74, 78]. Subsequently, this system of equations can be solved for an optimal solution [1]. However, the black box approach where the system of equations is only used internally and the algorithm simply returns an optimal random strategy has two drawbacks. First, the algorithm has to be repeated for every instance of the setting. Second, without knowing the functions we cannot analyse them to improve the process even further. This is despite the facts that polyline instances differ only in length and that paths on them can be described concisely. Therefore, the overall aim for this problem is to determine the probability functions analytically to reduce runtime and utilise them to further improve the calculation process.

In summary, this thesis addresses finding good or optimal solutions efficiently in the presence of uncertainty for the two highlighted problems: uncertain resource allocation and adversarial patrolling. Both are concerned with managing uncertainty and both show how uncertainty can be beneficial. Namely, in the resource allocation problem, using the exogenous probabilities of events allows overcoming inefficiency, and in the patrolling problem, random strategies allow the possibility of detecting the adversary. Finally, while different, the two problems are connected by more than merely the existence of uncertainty. Both problems consider entities that act on their own or on instruction. Additionally, in both cases a resource is at the centre of the problem (the electricity and the robots). Fundamentally, both will be addressed within the design and analysis of algorithms as formal procedures to solve mathematically modelled problems mostly within theoretical computer science. With regards to the entities (including neighbours, people, parties, etc.), all problems are considered in a computational context. This means that the interest in the resource or the use thereof by the entities are considered to be represented by autonomous software agents. These agents acting on behalf of entities are considered the main actors in this context. Henceforth, in this thesis the term *agent/s* is used to refer to all entities that act for themselves or on behalf thereof.

1.1 Research Requirements

Implicitly, the preceding introduction of the two problems highlights three research requirements which are pursued for both problems. These requirements are: integrating and managing uncertainty appropriate to the setting, achieving high efficiency, and exploring computational efficiency and providing efficient algorithms.

Research Requirements 1. Integrated Uncertainty Uncertainty is a prevalent constraint of real-world problems. Nevertheless, when considering problems it is frequently assumed that either there is no uncertainty or it is limited to a specific case, e.g. the worst case. However, under these assumptions even the best solutions can be far of the actually best solutions. Hence, uncertainty has to be included in any realistic consideration of problems. We do this by including it in the formal problem models as well as designing algorithms that are informed by uncertainty and manage or use it.

Research Requirements 2. Efficient Results Having ensured that uncertainty is considered, the overall aim is to design algorithms that find solutions with high efficiency or maintain the level of efficiency of algorithms by others. This efficiency is expressed in terms of mathematical *optimisation functions*. Any solution that attains the highest optimisation function value, subject to constraints, is called an optimal solution. Both of the problems have their own constraints that limit the achievable efficiency.

Research Requirements 3. Computationally Tractable While finding optimal solutions is important for practical use it is crucial to find solutions as quick as possible using as little storage as necessary. Sometimes that means designing algorithms with a short runtime. In other cases, the problem is provably hard to solve, e.g. NP-complete, and we relax the problem in favour of finding solutions in polynomial time. Two relaxations considered here are either sacrificing generality or optimality for computational efficiency. This is achieved by restricting the problem to a specific sub-problem that can be solved efficiently or by calculating solutions that are feasible and guarantee a fraction of the optimal efficiency.

1.2 Research Challenges

The introduced research requirements, while broad, provide a frame for the challenges of the two problems of multi-agent resource allocation and adversarial patrolling. Set within their own research areas both problems have their own *research challenges* which, nevertheless, can be linked back to at least one of the research requirements. Where we do not mention them explicitly they are referenced by their respective number with the challenges' titles. We next present the research challenges of both problems in turn.

Problem Domain A. Establish and Provide Algorithms for Fair Allocation of Homogeneous Resources with Uncertain Availability

The multi-agent resource allocation problem fits into the area of fair division [19]. While it covers all homogeneous resources, electricity shows the importance of the research well since tackling climate change needs continuous development of the energy sector. Moreover, the example of electricity allocation helps us to highlight the problems we are facing.

Research Challenge A.1. Model Resource Uncertainty *(Research Requirements: 1, 2)* The first requirement is to find the right modelling of uncertainty in order to allow comparing ex-ante against ex-post allocations. The need for that and the potential to achieve better results (Research Requirement 2) is clearly highlighted by the photovoltaic example. Moreover, while the literature on fair division of resources is extensive and contains models that include uncertainty (see Section 2.3 in Chapter 2), the research in the area mainly focuses on the allocation of bundles of items or heterogeneous resources [19, 31]. These allocations are altogether different from dividing homogeneous resources.

Research Challenge A.2. Establish Ex-Ante Envy-Freeness *(Research Requirement: 1)* After having modelled uncertainty, we can use this to define ex-ante envy-freeness. This concept is more flexible yet closely related to the common notion of (ex-post) envy-freeness [13, 19, 26, 38]. We will show that this is a meaningful measure that allows useful results beyond egalitarian solutions and thus opens up more research.

Together, Research Challenge A.1 and A.2 seek to establish this work as the first that considers fairness distribution of homogeneous resources when the amount of available resource is ex-ante uncertain.

Research Challenge A.3. Establish the Difference Between the Ex-Post and the Ex-Ante Settings *(Research Requirements: 1, 2)* One of the purposes of ex-ante envy-freeness is to allow more flexibility in comparison to ex-post envy-freeness and consequently to achieve more efficiency (Research Requirement 2); herein, efficiency is the sum of valuations, also called utilities, of individual agents' own allocation. However, the ex-ante setting changes the dynamic of the problem in comparison to ex-post. Hence, to better understand where the space of allocations provides flexibility and where not, it is necessary to address the similarities and differences between the ex-ante and ex-post efficiency and fairness constraint efficiency. This includes considering the worst case of fairness constraint efficiency.

Research Challenge A.4. Determine the Computational Complexity *(Research Requirement: 3)* Irrespective of the worst case, the increased flexibility of ex-ante envy-freeness increase the number of possible allocations. Among those allocations, we are interested in computing efficient allocations (see next research challenge), especially, which surpass the ex-post setting. Therefore, as a precursor to determining procedures that produce allocations with high efficiency, we have to determine which cases are computationally intractable.

Research Challenge A.5. Design Algorithms that Produce Computationally Efficient Solutions *(Research Requirements: 2, 3)* Having intractability results allows us to pursue the ultimate goal which is to determine algorithms which produce efficient and fair solutions with acceptable computational efficiency. In tractable settings we intend to present algorithms that achieve optimality. However, where complexity does not allow this in polynomial time, we seek polynomially calculable solutions via relaxations as specified in Research Requirement 3.

Problem Domain B. Improving Results for Full-Knowledge Multi-Robot Adversarial Patrolling

In comparison to the previous problem, the adversarial patrolling problem, including the uncertainty, has already been addressed in other works [1, 73, 74, 78]. Hence, the challenges under the focus of the research requirements are on improving upon the previous approach. As outlined above in the introduction, finding an optimal strategy is a two-step process. In the first step the probability functions that describe the random strategies are determined. The main challenge is to improve the runtime of this step. Following the first step, the probability functions are used to determine an optimal strategy. This step is not only affected by improvements to the first step but the improvement of this second step is also a part of our work.

Research Challenge B.1. Uphold Efficiency *(Research Requirements: 2, 3)* First off, any improvement to the procedure of finding a strategy has to produce an optimal strategy. While we mainly aim to reduce the runtime, this does not justify producing worse results than those produced by the currently used algorithms.

Research Challenge B.2. Reduce Computational Requirements of the First Step *(Research Requirement: 3)* Our main challenge for this problem is to completely remove the computational requirements of the first step and present analytical expressions. For that we rely on the fact that on a polyline every vertex has at most two adjacent vertices and therefore only two directions exists. This allows concise description of all paths. Moreover, since different polyline instances differ only in length, the resulting expressions are valid for all instances. Consequently, finding an optimal strategy would only require to determine the optimal strategy based on the probability functions, i.e. the second step of the procedure, and remove the runtime of the first step altogether.

Research Challenge B.3. Use Insight to Reduce Computational Requirements of the Second Step *(Research Requirement: 3)* Finally, the analytical expressions allow us to gain insight into the strategies which the previous black-box algorithms did not allow. We want to use this insight to reduce the runtime of the second step as well. There is no guarantee of an analytical expression for the solution to the probability functions. However, solving the system of equations depends on the number of equations. Hence, we shall investigate the structure of the strategy space to reduce the number of equations.

1.3 Research Contribution

Addressing the research challenges (see Section 1.2) of the two problems (see Section 1.2) the work in this thesis provides the following contributions.

Problem Domain A The focus is particular on efficiency (Research Challenge A.5) and fairness (Research Challenge A.2) with respect to the uncertainty (Research Challenge A.1). In more detail, it is shown that:

A.I. Ex-ante allocations are ex-ante efficient if and only if the allocations are ex-post efficient. This means that efficient allocations can be calculated easily for reasonable utility functions (Research Challenge A.5).

A.II. Ex-post envy-free allocations are also ex-ante envy-free. However, the opposite is not necessarily true. There are settings where ex-ante envy-free allocations can achieve a higher ex-ante efficiency than ex-post envy-free allocations (Research Challenge A.3).

A.III. The social welfare of the ex-ante efficient allocation under ex-ante envy-freeness can be substantially smaller than the welfare of the (unconstrained) ex-ante efficient allocation (Research Challenge A.5). To be precise, the price of envy-freeness has a lower bound of $\Omega(n)$ which is asymptotically tight for concave utility functions.

A.IV. The problem of maximising the ex-ante social welfare (Research Challenge A.5) under ex-ante envy-freeness (Research Challenge A.2) is strongly NP-hard even for continuous and concave utility functions, and uniform probabilities (Research Challenge A.4).

A.V. For two agents with satiable utility functions and events with uniform probabilities, the problem becomes polynomially solvable (Research Challenge A.4 and A.5).

A.VI. For two agents with satiable utility functions and events with non-uniform probabilities, the problem is NP-hard (Research Challenge A.4).

A.VII. For two agents with satiable utility functions and events with non-uniform probabilities, two variants of a greedy algorithm perform empirically well (Research Challenge A.5).

Problem Domain B The focus in this thesis is on the optimal random walk strategy of one robot on a circular or line graph and the probability of detecting a full-knowledge adversary with known penetration time (the time required for the adversary to successfully pass the perimeter or fence) under the Markov property. The work presents a novel approach to model all possible paths of a robot's random strategy as lattice paths.

B.I. For four different settings resulting from two movement patterns on open/closed polylines, lattice path modelling is used to describe the number of possible paths the robot can walk for a given number of steps (Research Challenge B.2).

B.II. From this, analytic expressions for the probabilities of detecting the adversary in the vertices of the graph are presented (Research Challenge B.2).

B.III. This removes the previously necessary step of calculating the functions including its time and space complexity from the overall procedure (Research Challenge B.2).

B.IV. Finally, connections between the segments and the penetration time, and the resulting probabilities are highlighted (Research Challenge B.3).

B.IV.a) For the case of a robot that can walk clockwise or anticlockwise on a perimeter (closed polyline) we divide the probability functions into three sets. We then show that two out of three sets can be excluded when solving the system for an optimal strategy (Research Challenge B.3 and B.2).

B.IV.b) For the same case, it is shown empirically that the optimal solution can be obtained by finding the intersection of only two probability functions. This reduces the runtime from linear to constant (Research Challenge B.3 and B.2).

For Problem Domain A on the allocation of resources with uncertain availability, the results of Contributions A.I. to A.IV. have been published at the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020).

Jan Buermann, Enrico H. Gerding, and Baharak Rastegari. 2020. Fair Allocation of Resources with Uncertain Availability. In Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9-13, 2020, IFAAMAS, 9 pages.

Moreover, for Problem Domain B on adversarial patrolling, the results of Contribution B.I., one setting of Contribution B.II., Contribution B.III. and Contribution B.IV.b) have been published at the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020).

Jan Buermann and Jie Zhang. 2020. Multi-Robot Adversarial Patrolling Strategies via Lattice Paths. In Proc. of the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020), Yokohama, Japan, July 11-17, IJCAI, 7 pages.

1.4 Thesis Structure

The rest of the thesis is structured as follows:

- Chapter 2 provides the necessary background and places this research into the wider research areas. The background addresses the relevant methodologies including social choice, lattice paths and optimisation.
- In Chapter 3 the resource allocation problem (Problem Domain A) is addressed, and research and results are presented. This includes, the presentation of a model for the problem as well as formal statements of Contributions A.I. to A.VII. and proofs.
- In Chapter 4 the adversarial patrolling problem (Problem Domain B) is considered. The chapter introduces the modelling and presents the formal statements of Contributions B.I. to B.IV. and proofs thereof.
- Chapter 5 concludes the work by summarising the results and linking them back to the challenges. Additionally, the chapter presents directions for future research.

Chapter 2

Background and Related Work

Since the two problems of resource allocation and adversarial patrolling are within distinct research fields, we place them into their wider context separately. We show how the research challenges (see Section 1.2) fit into those fields in Sections 2.3 and 2.4 for the resource allocation problem and adversarial patrolling problem, respectively. Prior to this, we provide a brief background on computational social choice (Section 2.1) and optimisation (Section 2.2). These two sections provide a brief overview and further references to aid the subsequent literature reviews.

2.1 Computational Social Choice

We give a brief introduction into the main concepts used in *computational social choice*. A more detailed introduction can be found in 'Handbook of Computational Social Choice' by Brandt et al. [19]. Our introduction in this section provides additional background and context for the resource allocation problem. Specific definitions for the resource allocation problem within the methodology of computational social choice are in the respective chapter (see Chapter 3).

Computational social choice extends a research area called social choice by applying formal computer science methods like algorithm design, algorithm analysis and complexity to social choice problems. These problems are mainly concerned with determining collective decisions, including, but not limited to, problems like the resource allocation problem discussed in this thesis [19, 54]. Social choice is an increasingly interdisciplinary field with foundations in economic research. Much more recent, computational social choice has gained momentum in the beginning of this century and while there is no clear seminal work that forms the beginning of social choice, it was established as a field within computer science in the early 2000s [19].

In the methodology of computational social choice we consider problems in a mathematical framework with axioms and conclusions. Every problem has a number of agents interested in a collective decision, e.g. whom to elect into an office or how a resources should be divided. The agents have preferences that affect the decision. One manifestation of this is that every agent has a preference relation $\preceq$ over possible outcomes. This is an ordering of the possible outcomes where $A \succ_i B$ indicates that agent i prefers outcome A over outcome B. For example, in an election an agent prefers candidates in a specific order and $A \succ_i B$ indicates that agent i prefers candidate A over candidate B. Otherwise, preferences often take the form of a valuation or *utility function* $u_i(o) : O \to \mathbb{R}$ assigning, for an agent i, the value $u_i(o)$ to outcome o out of all outcomes O. For example, when dividing a resource, an agent's utility function reflects the agent's value for every possible amount of the resource *allocated* to them. Finally, the outcome to a social choice problem is calculated by a *social choice function* f that maps from the preferences of the agents to an outcome. If we imagine a voting setting, the social choice function $f : (\preceq_i)_{i \in [n]} \mapsto o$ determines a winner o; if we imagine a resource allocation problem, the social choice function $f : (u_i)_{i \in [n]} \mapsto o$ determines the allocation o of resources to the agents.

2.2 Mathematical Optimisation

Secondly, we briefly introduce mathematical optimisation. The reason for this is that, like algorithm design generally, multi-agent or social choice problems require finding the best possible outcome subject to constraints. This makes mathematical optimisation or mathematical programming a possible tool for finding a solution. Moreover, it is a commonly used tool in algorithm design. However, mathematical optimisation is far too extensive to give a comprehensive overview here. Hence, we intend to give a brief overview of general concepts and important notions. For a more extensive introductions, see e.g. 'Introduction to Mathematical Optimization' by R. C. Robinson [29].

A mathematical program consists of functions and variables the describe a problem. A solution to the problem are values for the $n \in \mathbb{N}$ variables $x \in \mathbb{R}^n$ which maximise or minimise, depending on the problem, an objective function and satisfy the constraints. The objective function $f : x \to \mathbb{R}$ determines the value of any solution. The $m \in \mathbb{N}$ constraints restrict the possible solutions. Each constraints $i \in [m]$ is a functions over the variables $g_i : x \to \mathbb{R}$ limited to a value $y_i \in \mathbb{R}$. Optimisation Problem 1 is the notation of a mathematical program as used in this thesis. The first line (Line 2.1) shows the objective function and if the aim is maximisation (max) or minimisation (min). Any other line (Line 2.2 in this instance) shows the constraints.

Optimisation Problem 1 A general maximisation problem.

$$\max \quad f(x) \tag{2.1}$$

$$\text{s.t.} \quad g_i(x) \le y_i \qquad \forall i \in [m] \tag{2.2}$$

Based on the type of functions involved in the problem, optimisation problems can be classified into different groups. These types include *linear programming* and *convex programming* in which case the involved functions are linear or satisfy convexity, respectively. Linear programming is a very important tool and problems in this category can be solved in polynomial time. Convex optimisation is a broader class but many problems in this class can still be solved in polynomial time.

While there are different ways for finding optimal solutions we want to highlight one here in particular. These are the Karush-Kuhn-Tucker conditions which are well-known first order derivative tests. For non-linear higher order programming these conditions are, under certain conditions, necessary and sufficient for optimal solutions. For a formal definition of these see Robinson [29].

Finally, a further important class of optimisation problems are *integer programming* problems. These problems are characterised by having at least one of the variables x being limited to integer values. Solving these problems is generally NP-complete even if the variables are restricted to be binary, i.e. zero or one.

2.3 Fair Allocation of Resources

Our resource allocation problem is firstly related to the area of fair division including its subfields of indivisible items [77], cake cutting [66, 70], and estate or land division [63]. Moreover, we have motivated our model by the requirements and constraints of local energy markets with renewable energy sources but this model can also be applied to other important areas including emission permits for greenhouse gases [5], fair load shedding [62], and uncertain computational resources [9, 45]. We are going to focus here on the closely related works and areas.

Fair division is foremost divided along two dimensions: divisible resources against indivisible items; and homogeneous against heterogeneous resources. The first one describes if the resource itself (or in case of several resources, any of them) is worth only in its entirety (like a football) or can be arbitrarily split (like a cake). For the second dimension, every part of a homogeneous resource has the same value (e.g. marginal utility for energy), whereas different parts of a heterogeneous resource can have different values (e.g. different contents of a cake, cream versus a cherry). We consider these problem areas to highlight parallels, how they inform our research and contrast non-applicable approaches.

2.3.1 Indivisible Resources and Relaxed Envy-Freeness

Starting with the area of indivisible goods, this is an extensive field of ongoing work that requires solutions different to divisible resources but which highlights the importance to consider reasonable relaxations of envy-freeness. Indivisible items require that all allocations assign whole items [77]. This means that techniques in this area cannot apply continuous solutions possible in divisible resources. Hence, those techniques cannot bring the efficiency improvements we seek. Moreover, having to assign whole items also means that envy-free allocations do not have to exist, and determining their existence is already hard [19]. This has lead to the definition of fairness measures that relax envy-freeness [12, 13]. Budish [22] started a line of research with an envy-freeness relaxation where certain items can be ignored when determining envy-freeness. This simple modification where for each agent's envy-freeness any one item can be ignored is a notion that already guarantees the existence of allocations satisfying it [22].

Though not applicable to a homogeneous resource, this encourages our relaxation of envy-freeness to overcome the lack of meaningful allocations in order to increase efficiency. In comparison to indivisible items, in our case of divisible homogeneous resources envy-freeness is attainable and there is a trivial solution. However, clearly this is not always a desirable outcome since it diminishes efficiency. Hence, similar to indivisible items envy-freeness might be too harsh of a condition. Realising this, Feige and Tennenholtz [36] showed how to use a relaxation of fairness that holds in expectation to improve the ex-ante social welfare. However, in contrast to our consideration of envy-freeness, their fairness criterion adapts the concept of proportionality. Moreover, their mechanisms are uniform lotteries over equally sized allocations of the same resource amount, while in our setting the probability distribution is given exogenously and the events signify different amounts of available resource. Furthermore, in contrast to their lotteries, we use deterministic (conditional) allocations to improve ex-ante social welfare under ex-ante envy-freeness. Besides Feige and Tennenholtz's work, there are other works on divisible goods, where these usually consider several divisible items [19, 47, 63] or auction the divisible resource [46, 50]. Both are distinctly different to our setting.

Focusing on exogenous probabilities and fairness in these settings, Gajdos and Tallon [39] study the relationship of ex-ante and ex-post envy-freeness under ex-ante efficiency when agents have different perceptions of availability. They focus on two very simple cases where the amount of resource is equal for all events. They show that, for their setting, ex-ante optimal allocations are included in ex-post optimal ones and that ex-post envy-free allocations are a subset of ex-ante envy-free solutions. In contrast, we show that ex-post and ex-ante efficiency are the same when we do not require envy-freeness, and quantify the degradation of ex-ante efficiency from ex-ante envy-freeness as well as the complexity of calculating ex-ante envy-free efficient allocations.

This degradation as a consequence of requiring fairness is a worst case which is important to identify. We use the price of envy-freeness to quantify the loss in social welfare of envy-free allocations. The measure was introduced by Bertsimas et al. [18] for resource allocation problems. Furthermore, it has been used to quantify the degradation of efficiency in cake cutting by, for example, Aumann and Dombb [8], and Caragianniset al. [25].

2.3.2 Heterogeneous Resources and Complexity

The other large area of fair division, cake cutting, cannot represent our setting because of the structure of utilities, but its hardness results are more similar on a technical level. In cake cutting the resource is one connected heterogeneous resources (often the unit interval in one [19] or several dimensions [63, 70]). Thus, the utility functions are dependent on the exact location in the interval [19] and assigning a different interval of equal size might increase or decrease the utility. This is already the case for cake cutting with linear utility functions as Bei et al. [17] consider. Their functions would value the left side of the resource more than the right side of the resource or vice versa. This structure of the cake cutting problem persists even when multiple cakes [71] or multidimensional cakes [70] are considered.

The difference in resource space also means that the algorithmic approaches differ. Cake cutting is focused on the number of cuts, the way of cutting the resource, and then allocating the 'slices' to the agents [11]. This is the case even for piecewise uniform utility functions as considered by Chen et al. [26]. Their procedure first performs a cutting step and then an allocation step. These algorithms are fundamentally different to our problem where any bit of the resource principally has the same value to the agents.

Nevertheless, complexity results in cake cutting are more in line with our work given a more diverse allocation space. In cake cutting fairness in conjunction with efficiency is hard to achieve.

Bei et al. [17] consider the fairness measure of proportionality which ordinarily means that each agent gets at least one n-th of their utility of the entire cake. They show that, under proportionality, efficiency is NP-hard to approximate for general piecewise constant utility functions. Their proof is based on the principal idea of giving the agents very specific parts of the resource and adjusting the agents' utility accordingly. However, their construction relies on the heterogeneity of the resource which is not applicable to our case of homogeneous resources.

2.3.3 Similarities to Scheduling and Packing Problems

Finally, for completeness we want to mention that the problem appears similar to scheduling and packing problems; however, the solution approaches applied in these two areas are not applicable. Firstly, in scheduling, especially with divisible tasks, jobs have to be assigned to machines which are similar to the event's resources being assigned to agents [28, 37, 41, 58]. However, the setting differs in a number of points. Beginning with classic scheduling, this setting considers indivisible jobs [20] and is thus closer to indivisible goods. Besides this, among those works that consider the cost (negative valuation) of jobs, scheduling does not consider the breadth of valuations we allow. A common approach is a cost matrix summarising the cost for each job-machine combinations. Such a cost matrix is for example used by Fiat and Levavi [37], who consider envy-freeness among the machines. Moreover, even for the case of fractional or divisible tasks as in the works of Christodoulou et al. [28] or Dereniowski and Kubiak [32], respectively, costs are scaled linearly. However, this differs to our setting where agents can value the resource non-linearly. Moreover, we allow utility functions that have a maximal value which is incompatible with scheduling models where jobs have to be completed in full.

Secondly, our resource allocation problem is also close to packing problems which often consider indivisible items, equal sized containers and do not consider fairness for individual agents. Packing problems are mostly variants of the knapsack and the bin packing problem [81]. However, the goal of packing problems to reduce the number of containers to pack all items is incompatible with our problem. Equally are the fixed size containers incompatible with our utility functions. Fixed size containers would correspond to a cap on resource for every agent which would reduce the impact of having different events. Additionally, even for the less frequently considered case of divisible items [51] these incompatibilities do not change.

Irrespective of these observations, packing problems are usually not considered in an agent context and, if they consider fairness, have very different fairness measure. For example, the fairness measure of Shachnai and Tamir [72] requires only that at least a certainty number of items from each item type is chosen and the fairness measure of Azar et al. [10] only requires that items cannot be rejected if they can be packed. Equally different, the log based α-fair utility [7] recently considered by Diakonikolas et al. [33] is independent of the parties and more closely related to proportionality.

2.4 Adversarial Patrolling

Multi-robot patrolling is an area of ongoing research with substantial attention in the past decade [43]. The area's four main modelling dimensions are: the environment, the type of adversary, the objective or evaluation, and the approach or method. In terms of the overall goal, the area is divided between regular patrolling and adversarial patrolling [43]. In regular patrolling the aims are to optimise a frequency related goal; for example, minimising travel time or maximising the visits to important points [34]. Yet, the aim of adversarial patrolling is to defend against an attacker, by detecting [15] or handling [74] his or her penetration attempts.

Generally, approaches from regular patrolling are not applicable to adversarial patrolling. The adversarial model is mostly concerned about the adversary's knowledge about the robot's strategy. This ranges from the adversary having no information on the robots behaviour (random strategies) over being able to learn the robot's behaviour to having full knowledge of the robot's behaviour [2, 15, 43]. For example, the learning adversary in the work of Sak et al. [68] collects information on the robot's visits and uses this data to predict safe attack times. In comparison, we assume that the adversary has full knowledge of the defenders' strategy which can also be interpreted as the adversary has had enough time to learn the strategy. It is well established that a full-knowledge adversary can only be caught or detected using random strategies [1]. Which also implies that a strategy that optimises frequency goals is insufficient to ensure detecting or catching a full-knowledge adversary.

The environment in patrolling settings are generally assumed to be discrete [14] or the underlying continuous environment is discretised. For example, Elmaliach et al. [34] consider a 2D environment which is divided into cells. Similarly, the vertices in Sea et al. [69] represent coordinates in a plane. Discrete environments might be general graphs [15, 27], which represent important points, or polylines [1, 35], which represent defending a perimeter or fence. Important for our work, Agmon et al. [1] show that any continuous fence or perimeter can be represented as a graph with vertices that represent sections of equal travel time. Hence, any result for discrete polylines can be applied to cover continuous environments as well.

Multi-robot adversarial patrolling also overlaps with the game-theoretic area of security games [15, 74]. This area is dominated by Stackelberg games [74]. In our case the defender is the leader that commits to a strategy first and then the attacker responds with a penetration attempt. However, determining optimal strategies on general graphs and in game-theoretic models is mostly either NP-hard [44] or, while not proven, due to non-linear constrains and the application of non-linear optimisation, assumed to be NP-hard [74]. For example, Chevaleyre [27] and Basilico et al. [15] prove that, in their respective cases, determining the leader's optimal strategy is NP-hard. Consequentially, a great number of works use experimental evaluation of heuristics [15, 16, 27, 30, 35, 43, 69]. These approximate heuristic approaches are in contrast to our work of finding specific optimal solutions in polynomial time for specific graphs and movement patterns.

A more closely related game is the patrolling game introduced by Alpern et al. [4] which, nonetheless, has a different focus and considers different strategies. They assume the defender (patroller) and attacker are playing a zero-sum game (a setting were if one side wins and the other side looses) which, similar to our work, is played on a graph and the attacker takes a specified number of steps to attack one of the vertices. In comparison to our work, their aim is to determine or establish a bound on the value of the game which is the probability that the attacker is intercepted. Moreover, in contrast to our assumption of a full-knowledge adversary, they consider different strategies of the attacker, including uniform probability over the vertices as well as strategies that consider vertices based on specific features of the graph or the walk of the robots.

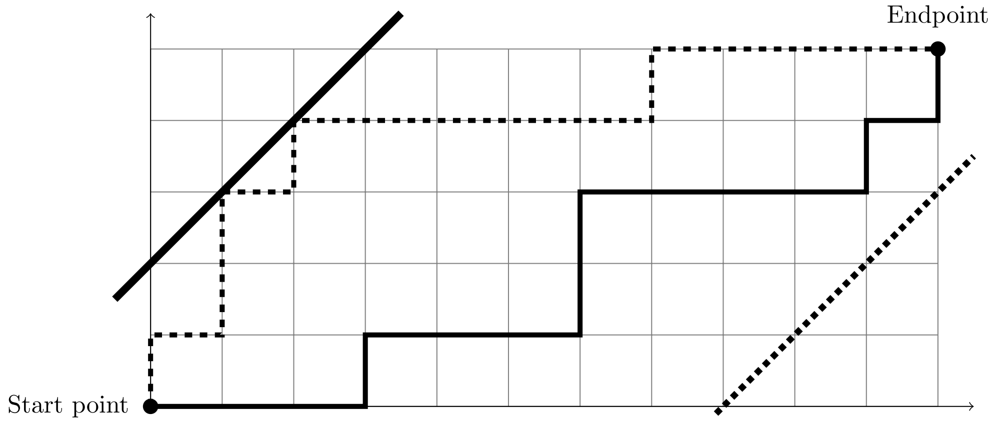


FIGURE 2.1: Examples of a $\mathbb{Z}^2$ lattice and two lattice paths. Both paths start in $(0,0)$ and end in $(12,6)$. Moreover, both paths use only horizontal and vertical unit steps. The two diagonal lines represent path restrictions. The lower right dotted line might be a line paths are not allowed to touch or cross. The diagonal line to the left might be line a path has to touch. For example, the dashed line touches the line twice.

More recently, Alpern et al. [3] determined a patrolling strategy for the patrolling game on a line graph and a penetration time of 2 time steps. Their setting is different to ours in the types of strategies and the knowledge of the adversary. Their defender chooses randomisation over all paths, where we consider memoryless walks, and their attacker chooses a strategy first, whereas in our case the defender sets a strategy first and the attacker knows that strategy.

The work most closely related to our work is the works of Agmon et al. [1] and subsequent papers [73, 74, 78] addressing different aspects like coordinated attacks [73], sequential attacks [74] and deception [78]. Their general aim is to find optimal polynomial-time patrolling strategies (maximising the minimal probability of the adversary being detected) for specific graphs and specific goals. They use algorithmic approaches, mainly the construction and evaluation of Markov transition matrices, to find the probabilities of the robots detecting the adversary [1] or the number of paths [74]. In contrast, we give explicit terms and functions for the number of paths and the probability of detecting the adversary which removes the necessity for the respective algorithms.

We do this by utilising lattice paths from within the area of combinatorics. A lattice path is a path along points in an Euclidean space (see also Figure 2.1). Lattice path problems aim to count the number of paths given a start and end point, the allowed step directions, coordinates restricting the paths, and further path features. Lattice paths have been studied for a long time and are, despite apparent simplicity, powerful. Moreover, they have application to many problems including physical systems, encoding, probability and statistics [49]. We will formally define lattice paths for our case in the respective chapter (see Chapter 4).

2.5 Summary

We have highlighted research related to our two problems including what previous research has not addressed. In conclusion, we summarise the unaddressed points and link them back to the research requirements. Beginning with the resource allocation problem (Problem Domain A), it is clear that research on homogeneous resources is lacking attention with more focus in the area of fair division on heterogeneous divisible and indivisible items. Additionally, the research area is lacking consideration of uncertainty. However, homogeneous items deserve attention and uncertainty should be considered (Research Requirement A.1). So far solutions to the fair division of homogeneous resources is limited to equal share solutions which do not provide a lot of efficiency. This is the first reason homogeneous resources deserve attention (Research Requirement A.5). In order to progress in pursuing these two goals it is sensible to mirror the research on indivisible items and relax envy-freeness. Similar to Feige and Tennenholtz's [36] relaxation of proportionality the relaxation of envy-freeness has to be appropriate for our setting and take the uncertainty into consideration (Research Requirement A.2). Additionally, our work has to extend beyond a modelling perspective since related areas do not readily provide algorithms to solve this new problem (Research Requirement A.5). Finally, as with any new modelling of a problem we have to establish the difference to previous models (Research Requirements A.3) and determine the computational requirements (Research Requirements A.4).

The previous research of the adversarial patrolling problem (Problem Domain B) is lacking in computational efficiency. This is directly because of the approach of using Markov chains. First of all, the approach has a significant runtime, even for simple graphs. This should be reduced (Research Requirement B.2). Moreover, the black box nature of the Markov chain approach does not allow us to gain structural insights into the random strategy solution space. Gaining structural insights would make it easier to investigate the random strategies and help to improve the overall process in terms of runtime (Research Requirement B.3). However, improving the runtime and understanding cannot come at the cost of optimality as provided by Agmon et al.'s [1] approach. Hence, the optimality has to be uphold by our approach (Research Requirement B.1).

Chapter 3

Fair Allocation of Resources with Uncertain Availability

In this chapter we address Problem A (see Section 1.2), i.e. the allocation of resources with uncertain availability, and present our contributions as presented in Section 1.3. Specifically, we argue that considering ex-ante envy-freeness provides more flexibility and results in better solutions in terms of social welfare compared to ex-post envy-freeness. We start by introducing the formal model in Section 3.1. Afterwards, we illustrate the social welfare benefits of ex-ante envy-freeness in a numerical example for the introductory setting of neighbours sharing a photovoltaic system in Section 3.2. While the potential benefit of the setting are highlighted in the example, in the worst case the ex-ante envy-freeness might not allow better allocations than the ex-post envy-freeness. This degradation is measured by the *price of envy-freeness* and we discuss this result and other smaller contributions that highlight similarities between the ex-ante and ex-post case in Section 3.3. Following this, we show that the problem of maximising ex-ante social welfare under ex-ante envy-freeness is NP-hard (see Section 3.4). Considering this hardness and the instances for which this already holds, we turn to the case of two agents in pursuit of Research Challenge A.5 in Section 3.5. For this case we show that depending on the probability distribution over the events, uniform or arbitrary, the problem is optimally solvable in polynomial time or is also NP-hard, respectively. For both settings (uniform or arbitrary probabilities) we devise a greedy algorithm in Section 3.5.3. This algorithm produces different allocations based on the ordering of the events. An ordering by amounts gives us the optimal solution in the case of uniform event probabilities (see Section 3.5.4). For arbitrary probability distributions over the events the same ordering as well as the ordering by expected amounts provide different empirical (see Section 3.5.7) performance guarantees. Theoretically, ordering by the expected amount might not approximate an optimal allocation but ordering by the amount indicates to guarantee a better approximation (see Section 3.5.5.2).

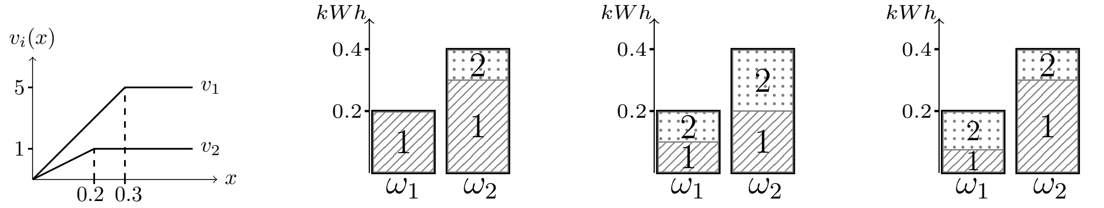
3.1 Preliminaries

There are $n \in \mathbb{N}$ agents which are interested in a resource whose actual amount is uncertain. For the available amount there are m possible outcomes $\Omega = \{\omega_1, \ldots, \omega_m\}$ with $\omega_i \in [0,1]$ for all $i \in [m]$. Each outcome $\omega \in \Omega$ has an associated probability $f(\omega)$ with $f(\omega) > 0$ and $\sum_{\omega \in \Omega} f(\omega) = 1$. This defines a Bernoulli probability measure μ on the space Ω which we define for completeness but to which we will not refer back to. Note that despite considering only events of one outcome we use the term event in this chapter when referring to the result of the random experiment. The *allocation* of the resource to the agents is represented by the vector $A = (a_1, a_2, \ldots, a_n)$ where the allocation to an agent i is a function $a_i : \Omega \to [0,1]$. An allocation A is *valid* if it satisfies these two validity constraints: positivity that implies $a_i(\omega) \geq 0$ for all $i \in [n]$ and $\omega \in \Omega$, and respecting the available amount, i.e. $\sum_{i \in [n]} a_i(\omega) \leq \omega$ for all $\omega \in \Omega$. It is important to notice that allocation functions are conditional on the events. Let $\mathcal{F} \subset [0,1]^{\Omega}$ be the set of all valid allocation functions and let $\Lambda = \mathcal{F}^n$ be the set of all valid allocations.

An agent $i \in [n]$ values the amount of received resource according to a monotonically increasing *utility function* $v_i : [0,1] \to \mathbb{R}$ which satisfies non-negativity ($v(x) \geq 0 \; \forall x \in [0,1]$) and no valuation for zero ($v(0) = 0$). The monotonicity reflects free disposal. Let $\Theta \subset \mathbb{R}^{[0,1]}$ be the set of all valid utility functions. Additionally, let $V_i : \mathcal{F} \to \mathbb{R}$ denote an agent i's *expected utility*. Given an agent's allocation function their expected utility of an allocation is equal to the expected utility of the allocation function. That is, $V_i(a_j) := \sum_{\omega \in \Omega} v_i(a_j(\omega)) f(\omega)$ for any $i, j \in [n]$. We note here, we deliberately do not scale utilities into the range $[0,1]$ to allow agents being weighted differently. Moreover, the fact that we do not normalise is not the source of the price of envy-freeness or the hardness since we can always assume to have events with smaller amounts and adjust the utility functions accordingly.

Allocations are measured in terms of ex-ante social welfare and require ex-ante envy-freeness, both notations naturally extend their ex-post definitions. Ex-ante social welfare, $W(A) := \sum_{i \in [n]} V_i(a_i)$, measures the sum of expected utilities. This is in contrast to ex-post social welfare $\sum_{i \in [n]} v_i(a_i(\omega))$ which measures the sum of utilities with respect to one event $\omega \in \Omega$. Ex-ante envy-freeness requires that every agent i weakly prefers their own allocation, in terms of expected utility, over every other agent j's allocation, i.e. $V_i(a_i) \geq V_i(a_j)$ with $i, j \in [n]$. Again in contrast, ex-post envy-freeness requires preference in terms of utilities, i.e. $v_i(a_i(\omega)) \geq v_i(a_j(\omega))$ for all $i, j \in [n]$. Overall, the goal is, given the agents' utility functions, to find a valid ex-ante envy-free allocation A such that A maximises ex-ante social welfare. We note that, henceforth, any reference to envy-freeness or social welfare without preposition refers to the respective ex-ante notion. Furthermore, an allocation which maximises social welfare is called *efficient* and an allocation that maximises social welfare under envy-freeness is called *optimal*.

Finally, the price of envy-freeness is defined as the ratio $\max_{\theta \in \Theta} \frac{W(A_E(\theta))}{W(A_{EF}(\theta))}$ where A_E is an unrestricted (i.e. without considering envy-freeness) ex-ante efficient allocation and A_{EF} is an optimal (,i.e. ex-ante envy-free and efficient) allocation. The price of envy-freeness expresses the degradation of efficiency due to the enforcement of ex-ante envy-freeness and a higher value indicates a higher efficiency loss.



(A) The agents' utility functions in the example.

(B) The efficient allocation unconstrained by envy-freeness.

(C) The ex-post envy-free allocation.

(D) An ex-ante but not ex-post envy-free allocation.

FIGURE 3.1: The allocations of the example. In each case, the bars are the two events, ω_1 and ω_2 with an available amount of $0.2\,kWh$ and $0.4\,kWh$, respectively. The patterns and numbers inside the events refer to the allocations to the agents. The diagonal lines indicate an allocation to agent 1 and the dots indicate an allocation to agent 2.

3.2 Example - Renewable Energy Setting

Before we present our results and contributions we start with illustrating the potential social welfare benefits of ex-ante envy-freeness. Using our motivational renewable energy example from the introduction, consider a setting where two agents share a photovoltaic system. Both would like to know how much electricity they can expect for the next day so that they can plan ahead and would be able to acquire further electricity from other sources if necessary. The weather forecast for the next day states the weather as one of two events; event ω_1 is that the day is cloudy which occurs with a probability of $2/3$, whereas event ω_2 is that the day is sunny which occurs with a probability of $1/3$. Based on these weather predictions, the photovoltaic system produces an amount of $0.2\,kWh$ and $0.4\,kWh$ of energy, respectively. The two agents are interested in electricity and value it according to the utility functions $v_1(x) = \frac{5}{0.3} \cdot x$ for $0 \le x < 0.3$, $v_1(x) = 5$ for $x \ge 0.3$, and $v_2(x) = \frac{1}{0.2} \cdot x$ for $0 \le x < 0.2$ and $v_2(x) = 1$ for $x \ge 0.2$ (see Figure 3.1A). The objective is to find an allocation. We present three different allocations: the social welfare maximising allocation, the ex-post envy-free allocation and an ex-ante but not ex-post envy-free allocation. In all three allocations, the next day the agents get deterministically the allocation associated with the occurring event (the actual weather).

In the social welfare maximising allocation, agent 1 would get everything in the first event and $0.3\,kWh$ in the second, while agent 2 gets the remaining $0.1\,kWh$ (see Figure 3.1B), resulting in an expected social welfare of $4\frac{1}{18}$:

$$\begin{aligned} &f(\omega_1) \cdot v_1(0.2) + f(\omega_2) \cdot v_1(0.3) + f(\omega_1) \cdot v_2(0) + f(\omega_2) \cdot v_2(0.1) \\ =& \frac{2}{3} \cdot \frac{5}{0.3} \cdot 0.2 + \frac{1}{3} \cdot \frac{5}{0.3} \cdot 0.3 + \frac{2}{3} \cdot \frac{1}{0.2} \cdot 0 + \frac{1}{3} \cdot \frac{1}{0.2} \cdot 0.1 = 4\frac{1}{18}. \end{aligned}$$

In contrast, giving equal amounts to both agents in both cases (see Figure 3.1C) achieves an expected social welfare of $2\frac{8}{9}$:

$$\begin{aligned} &f(\omega_1) \cdot v_1(0.1) + f(\omega_2) \cdot v_1(0.2) + f(\omega_1) \cdot v_2(0.1) + f(\omega_2) \cdot v_2(0.2) \\ =& \frac{2}{3} \cdot \frac{5}{0.3} \cdot 0.1 + \frac{1}{3} \cdot \frac{5}{0.3} \cdot 0.2 + \frac{2}{3} \cdot \frac{1}{0.2} \cdot 0.1 + \frac{1}{3} \cdot \frac{1}{0.2} \cdot 0.2 = 2\frac{8}{9}. \end{aligned}$$

Since this allocation gives both agents the same in both events, it is both ex-ante and ex-post envy-free. Now, consider the allocation where the first agent gets $0.075kWh$ in the first event and $0.3kWh$ in the second event, and the second agent gets the remaining energy each time (see Figure 3.1D). For the first event agent 1 values their and the other agent's allocation at $v_1(0.075) = \frac{5}{0.3} \cdot 0.075 = \frac{5}{4} = 1\frac{1}{4}$ and $v_1(0.125) = \frac{5}{0.3} \cdot 0.125 = \frac{25}{12} = 2\frac{1}{12}$ respectively. For the second event, agent 2 values their and the other agent's allocation at $v_2(0.1) = \frac{1}{0.2} \cdot 0.1 = \frac{1}{2}$ and $v_2(0.3) = \frac{1}{0.2} \cdot 0.2 = 1$. Hence, the allocation for neither event is ex-post envy-free. In comparison, considering expected utilities, agent 1 is ex-ante envy free since they value their own and the other agent's allocation at $2\frac{1}{2}$ and $1\frac{17}{18}$, respectively:

$$\frac{2}{3} \cdot v_1(0.075) + \frac{1}{3} \cdot v_1(0.3) = \frac{2}{3} \cdot \frac{5}{4} + \frac{1}{3} \cdot \frac{5}{0.3} \cdot 0.3 = 2\frac{1}{2}$$
$$\frac{2}{3} \cdot v_1(0.125) + \frac{1}{3} \cdot v_1(0.1) = \frac{2}{3} \cdot \frac{25}{12} + \frac{1}{3} \cdot \frac{5}{0.3} \cdot 0.1 = 1\frac{17}{18}$$

Similarly, agent 2 is ex-ante envy-free since they value their own allocation as well as the other agent's allocation both at $\frac{7}{12}$:

$$\frac{2}{3} \cdot v_2(0.125) + \frac{1}{3} \cdot v_2(0.1) = \frac{2}{3} \cdot \frac{1}{0.2} \cdot 0.125 + \frac{1}{3} \cdot \frac{1}{2} = \frac{7}{12}$$
$$\frac{2}{3} \cdot v_2(0.075) + \frac{1}{3} \cdot v_2(0.3) = \frac{2}{3} \cdot \frac{1}{0.2} \cdot 0.075 + \frac{1}{3} \cdot 1 = \frac{7}{12}$$

Hence, the allocation is ex-ante envy-free and has with $\frac{30}{12} + \frac{7}{12} = 3\frac{1}{12}$ an ex-ante social welfare higher than that of the ex-post envy-free allocation.

The example highlights how the allocation illustrated in Figure 3.1D utilises the fact that the allocation can vary between the events. This allows increased ex-ante social welfare in comparison to the allocation illustrated in Figure 3.1C.

3.3 Ex-Post vs. Ex-Ante Settings

Adding to the example in the previous section, we present a number of smaller results to highlight how the ex-post and ex-ante efficient and envy-free allocations are related. Moreover, we show how, in the worst case (under linear utility functions), ex-ante allocations have the same degradation of efficiency as ex-post giving all agents the same allocation.

3.3.1 Efficient Allocation

We begin by focusing on the efficiency of unconstrained allocations, i.e. when we do not demand the constraint of ex-ante envy-freeness. We show that any unconstrained efficient allocation is also ex-post efficient and vice versa. Hence, this gives us a reference point for the price of envy-freeness and also for any future approximation.

Lemma 3.1. *An allocation is ex-ante efficient if and only if the allocation is ex-post efficient for every $\omega \in \Omega$.*

Proof. By linearity of expectation, ex-post efficiency implies ex-ante efficiency. For the reverse, we assume for contradiction that an allocation A is ex-ante efficient but not ex-post efficient for a number of events $\Psi \subseteq \Omega$. Using A and an allocation $A^e = (a_1^e, \ldots, a_n^e)$ that is ex-post efficient for every $\omega \in \Psi$, i.e. $\sum_{i \in [n]} v_i(a_i^e(\omega)) > \sum_{i \in [n]} v_i(a_i(\omega))$, we create allocation A' with $a_i'(\omega) = a_i(\omega)$ if $\omega \notin \Psi$ and $a_i'(\omega) = a_i^e(\omega)$ if $\omega \in \Psi$. The efficiency of A^e for all $\omega \in \Psi$ and the linearity of expectation imply that A' has a higher social welfare than A. Hence, A cannot be ex-ante efficient. □

The construction of Lemma 3.1 shows that we can create an ex-ante efficient allocation by using the ex-post efficient allocations for every event. This calculation for one event can be represented as an optimisation problem with the ex-ante social welfare as the optimisation function and two linear constraints which require that the allocation does not exceed the available amount and that the allocation is not negative for any agent. Hence, we have that, for concave utility functions, the optimisation function is concave and the problem can be solved in polynomial [21]. Since there are m events we can find the entire allocation in polynomial time.

Simultaneously, this means that there is no difference between unconstrained ex-ante and ex-post efficiency. This appears negative since no ex-ante efficiency improvement is possible. However, since allocations under ex-post envy-freeness can be very inefficient this does not affect our work. Moreover, the result also means that when measuring the social welfare improvements under ex-ante envy-freeness in comparison to the unconstrained efficiency we have to consider only one reference value.

3.3.2 Envy-Free Allocation

In contrast to unconstrained efficiency, ex-ante envy-free allocations can vary significantly from ex-post envy-free allocations (see also the example in Section 3.2). Nevertheless, we first establish that an ex-ante envy-free allocation always exists. The ex-ante envy-free allocation that always exist is the *equal share* allocation, which allocates everything equally to all agents in all events. Formally:

Definition 3.2 (Equal Share Allocation)**.** The *equal share allocation* A_{ES} is the allocation with $a_i(\omega) = \frac{\omega}{n}$ for all $i \in [n]$ and $\omega \in \Omega$.

In order to establish that this allocation is envy-free we prove that, for envy-free allocations, the set of ex-post allocations is a subset of the set of ex-ante allocations.

Lemma 3.3. *If an allocation function is ex-post envy-free for every $\omega \in \Omega$, it is also ex-ante envy-free.*

Proof. Let $A \in \Lambda$ be a valid allocation satisfying the lemmas statement, i.e. $v_i(a_i(\omega)) \geq v_i(a_j(\omega))$ for all $i, j \in [n]$ and $\omega \in \Omega$. Then, $V_i(a_i) = \sum_{\omega \in \Omega} v_i(a_i(\omega)) f(\omega) \geq \sum_{i \in [n]} \sum_{\omega \in \Omega} v_i(a_j(\omega)) f(\omega) = V_i(a_j)$ for all $i, j \in [n]$. □

Note, the opposite is not necessarily true and this is exactly why it is possible to find allocations with increased ex-ante social welfare. Additionally, since the equal share allocation is trivially ex-post envy-free, it is also ex-ante envy-free by Lemma 3.3.

Corollary 3.4. *The equal share allocation is ex-ante envy-free.*

3.3.3 Price of Envy-freeness

We next consider the extent to which both efficiency and envy-freeness can be achieved. Specifically, we show in Theorem 3.8 that, without restricting the utility functions, the price of envy-freeness is at least in the order of the number of agents. This follows since, in the case of linear and non-equal utility functions, the most efficient solution is to give everything to one agent (see Lemma 3.6), and that, under ex-ante envy-freeness, it is not possible to achieve a higher efficiency than the equal share allocation (see Lemma 3.7). Finally, we show that the bound is asymptotically tight for concave utility functions (see Theorem 3.9).

For this section, we assume all utility functions are linear of the form $v_i(x) = c_i \cdot x$ with $c_i \in \mathbb{R}_+$ for $i \in [n]$. We only consider linear functions with y-intercepts equal to zero since otherwise the utility functions would violate the no valuation for zero assumption (see Section 3.1). In this case of linear utility functions an ex-ante efficient allocation, which we call *maximal slope allocation*, gives everything to an agent with the highest slope. The ex-ante social welfare of this allocation is the product of the maximal slope and the expected amount.

Definition 3.5 (Maximal Slope Allocation)**.** A *maximal slope allocation* is the allocation where an agent $j \in \arg\max_{i\in[n]}\{c_i\}$ gets all of the resource.

It is straightforward to see that this achieves the highest social welfare. For completeness, the proof is in the appendix (see Section A.1).

Lemma 3.6. *A maximal slope allocation is efficient and has an ex-ante social welfare of* $c_j \cdot \mathrm{E}[X]$ *for* $j \in \arg\max_{i\in[n]}\{c_i\}$.

In comparison, for linear functions, the maximal possible social welfare under envy-freeness is achieved by the equal share allocation and is a function of the average of the slopes of all agents. This demonstrates that there is an inherent trade-off between social welfare and envy-freeness.

Lemma 3.7. *No ex-ante envy-free allocation can have more ex-ante social welfare than* $\mathrm{mean}_{i\in[n]}$ $\{c_i\} \cdot \mathrm{E}[X]$ *which is matched by the equal share allocation (where* mean *is the arithmetic mean, see Nomenclature for definition).*

Proof. The key point is that, under linearity of the utilities, ex-ante envy-freeness implies that the expected allocations are the same, i.e. $\mathrm{E}[a_i] = \mathrm{E}[a_j]$ for all $i, j \in [n]$. Hence, the ex-ante social welfare is determined by the allocations and the average slope, i.e. $W(A) = n{\cdot}\mathrm{mean}_{i\in[n]}\{c_i\}{\cdot}\mathrm{E}[a_k]$ for any $k \in [n]$. Then, since every allocation is limited by the available amount, we have that $n \cdot \mathrm{E}[a_k] \leq E[X]$ which implies the claimed limit on the achievable ex-ante social welfare. It is straightforward to show that this is matched by the equal share allocation (see Section A.1 in the appendix for the details). □

Finally, we construct an instance for the price of envy-freeness.

Theorem 3.8. *The division of a homogeneous resource has a price of envy-freeness of* $\Omega(n)$.

Proof. Let X be arbitrary but have at least one event with positive amount and probability. Let the utility function of the first agent be $v_1(x) = 2x$ and the remaining utility functions

$v_i(x) = \frac{1}{n-1} \cdot x$ for $i \in \{2, \dots, n\}$. Then, the ratio of the unconstrained efficient allocation to the envy-free allocation is, by Lemma 3.6 and 3.7, $(c_1 \cdot \mathrm{E}[X])/(\mathrm{mean}_{i\in[n]}\{c_i\} \cdot \mathrm{E}[X]) = 2/3 \cdot n$ which implies the claim. □

For arbitrary utility functions the upper bound can be unbounded. For example, similar to the instance of Theorem 3.8 within a group of agents with low utilities could be one agent with an arbitrarily high utility for getting entire events. Additionally, this agent might have an arbitrarily low utility for any fraction of the events. Hence, in comparison to the theorem, the agent with the high utility does not contribute any utility to the equal share allocation and therefore the difference between the two allocations can be arbitrarily bad. However, for the realistic assumption of concave utility functions the bound is asymptotically tight.

Theorem 3.9. *For concave utility functions, the division of a homogeneous resource has a price of envy-freeness of at most n.*

This follows from the monotonicity of the utility functions and the fact that concavity limits the utility of each event by n times the equal share allocation (see Section A.1 in the appendix for the detailed proof).

Theorem 3.8 and Theorem 3.9 show that, in the worst case, the degradation of social welfare under ex-ante envy-freeness is unfortunately as bad as under ex-post envy-freeness. Nevertheless, these results are based on a setting where all agents have linear utilities. As the example in Section 3.2 shows even simple piecewise linear utility functions are not restricted to requiring that resources are equally shared under ex-ante envy-freeness and therefore circumvent this worst case. Interestingly, the results hold irrespective of the probability distribution.

3.4 Complexity

The results from the previous section show that, for linear utility functions, equal share already attains maximum possible efficiency. However, the example in Section 3.2 shows that, if the utility function is not linear for all agents, there are allocations with higher ex-ante social welfare. Hence, the question we answer in this section is: how difficult would it be to find more efficient allocations in general? To this end, we show that the problem of maximising ex-ante social welfare under ex-ante envy-freeness is strongly NP-hard. In order to prove this, we consider the decision version of our problem which we call *decision version of uncertain amount fair division (D-UAFD)*.

Definition 3.10 (Decision Version of Uncertain Amount Fair Division (D-UAFD))**.**

Instance: $\langle X, (v_i)_{i\in[n]}, B\rangle$ where $X : \Omega \to [0, 1]$ with finite $\Omega \subsetneq [0, 1]$ is a discrete random variable, $(v_i)_{i\in[n]}$ with $v_i \in \Theta$ are utility functions and $B \in \mathbb{R}$ is a bound.

Problem: Does there exist a valid allocation $A = (a_i)_{i\in[n]} \in \mathcal{F}^n$ such that $V_i(a_i) \geq V_i(a_j)$ for all $i, j \in [n]$ and $W(A) \geq B$.

Θ, $\mathcal{F}$, V_i and $W(A)$ are defined as in the preliminaries.

This problem is strongly NP-complete as stated by the following theorem. We prove the theorem formally at the end of this section after presenting the construction and a number of supporting definitions and statements.

Theorem 3.11. *D-UAFD is strongly NP-complete.*

Before we proceed to the formal proof, we introduce necessary definitions and lemmas. In brief, the proof is by reduction from the 3-partition problem, that is known to be NP-complete in the strong sense [40].

Definition 3.12 (3-Partition Problem)**.**

Instance: $\langle S, B\rangle$, where $S \subset \mathbb{N}$ is a finite multiset of $3m$ elements $s_1 \dots s_{3m}$ and $B \in \mathbb{Z}^+$ is a bound such that $B/4 < s_i < B/2\ \forall i \in [3m]$ and $\sum_{i\in[3m]} s_i = mB$.

Problem: Can S be partitioned into m disjoint subsets $S_1, \dots, S_m$ such that $\sum_{x\in S_i} x = B$ for all $i \in [m]$.

(Notice that the item size requirement implies that each subset must contain exactly 3 elements.)

We reduce a given instance of the 3-partition problem to an instance of our fair division problem which we call *envy partition instance.* In short, we scale the 3-partition instance down and transform it into an instance of D-UAFT where the sets are the events and the 3-partition elements are the agents. To achieve this, the agents' utility functions are chosen so that the opposing goals of envy-freeness and efficiency require that every agent gets allocated exactly the amount specified by the corresponding 3-partition element in a single event. Therefore, allocating this amount to an agent corresponds to choosing a set for the corresponding 3-partition element. Additionally, to address that every event has to have a different amount, we slightly increase the size of the events and add additional agents who desire exactly these amounts.

Definition 3.13 (Envy Partition Instance)**.** An *envy partition instance* $\langle X, (v_i)_{i\in[n]}, B'\rangle$ is an instance of D-UAFD constructed given a 3-partition instance $\langle S, B\rangle$ in the following way. We assume without loss of generality that the 3-partition elements are ordered in a non-decreasing way, i.e. $s_1 \leq s_2 \leq \ldots \leq s_{3m}$.

Firstly, there are $4m$ agents in total; i.e. $n = 4m$. The agents $m+1, \dots, 4m$ correspond to the elements of the 3-partition instance. The remaining m agents correspond to both the partitioning subsets of the 3-partition instance and the events as defined for this instance. In particular, let $s'_i := {}^{s_i}/_{(mB)}$ for all $i \in [3m]$, let ε be chosen such that $0 < \varepsilon < {}^2/_{(m^2B(m+1))}$, and let $Z := \sum_{i=1}^{m} i\cdot\varepsilon$. Then the utility functions are defined as follows.

$$v_i(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ i\cdot x & \text{if } 0 < x \leq \hat{s}_i \\ i\cdot \hat{s}_i & \text{if } x > \hat{s}_i \end{cases}$$

$$\text{with } \hat{s}_i = \begin{cases} i\cdot\varepsilon & \text{if } i \leq m \\ s'_{i-m} & \text{if } m+1 \leq i \leq 4m \end{cases}$$

for $i \in [4m]$. We refer to $\hat{s}_i$ as agent i's saturation amount since any bigger amount has no additional value to the agent. In other words Note that the $\hat{s}_i$ are weakly increasing in i.

The second step is the construction of the random variable X. The set of events consists of m elements which are of size $\frac{1}{m}$ plus an additional small and increasing offset. Formally, $\Omega = \{\omega_1, \dots, \omega_m\} = \left\{\frac{1}{m} + \varepsilon, \frac{1}{m} + 2 \cdot \varepsilon, \dots, \frac{1}{m} + m \cdot \varepsilon\right\}$. Additionally, our random variable is uniformly distributed, i.e. has the probability mass function $f(\omega) = \frac{1}{|\Omega|} = \frac{1}{m}$ for all $\omega \in \Omega$. Note that $\sum_{\omega \in \Omega} \omega = 1 + Z = \sum_{i \in [4m]} \hat{s}_i$.

Finally, let the bound B' be defined as $B' := \frac{1}{m} \cdot \sum_{i=1}^{3m} i \cdot s'_i + \frac{\varepsilon}{6}\left(2m^2 + 3m + 1\right) + 1$.

This construction can be done in polynomial time, the constructed instance is of polynomial size, and the constructed values are polynomial in the values of the 3-partition instance.

Given the constructed envy partition instance, we first show that an allocation A achieves the required social welfare B' and satisfies ex-ante envy-freeness only if it allocates to each agent, except maybe agent 1, their own saturation amount in exactly one event and zero in all other events; that is, $a_i(\omega') = \hat{s}_i$ for one $\omega' \in \Omega$ and $a_i(\omega) = 0$ for all $\omega \in \Omega \backslash \{\omega'\}$. We then show that such an allocation, satisfying envy-freeness and having social welfare of at least B', exists if and only if a 3-partition exists.

Intuitively, to maximise social welfare we need to give the agents with higher index larger allocations, as they have higher utility, but we cannot allocate too much to them as the agents with lower index would envy them.

For the purpose of our proof, we introduce a new representation of an allocation which we call *allocation by pieces*. In this representation, we allocate up to the entire amount of resource over all events $\sum_{\omega \in \Omega} \omega = 1 + Z$ to agents as *pieces*, independent of the events. Each agent receives m pieces. We then map events to pieces.

Definition 3.14 (Allocation by Pieces)**.** An allocation by pieces consists of functions $a_i^p : [m] \to [0,1]$ for each agent $i \in [4m]$ such that $0 \le a_i^p(j) \le \hat{s}_i$, $\forall j \in [m]$ and $\sum_{i \in [n]} \sum_{j \in [m]} a_i^p(j) \le 1 + Z$, and a function $\Phi : [n] \times \Omega \to [m]$ where $\Phi(i, \omega) \ne \Phi(i, \omega')$, $\forall i \in [n]$, $\forall \omega \ne \omega' \in \Omega$.

Social welfare and envy-freeness for a_i^p's are defined as in the preliminaries. Furthermore, an allocation by pieces is valid if $a_i^p(\Phi(i, \omega))$ is valid $\forall i \in [n], \forall \omega \in \Omega$, where validity is defined as in the preliminaries. Given a_i^p's and given a Φ, we say that Φ is valid if all $a_i^p(\Phi(i, \omega))$ are valid.

We will show that taking any allocation by pieces and modifying it so that the pieces for a given agent i are reduced to a number of pieces of size $\hat{s}_i$ and maximum one piece of size less than $\hat{s}_i$ (see Definition 3.16) does not change the social welfare (see Lemma 3.17). Furthermore, if the a_i^p are ex-ante envy-free and a valid Φ exists then the $a_i^p(\Phi(i, \omega))$ are ex-ante envy-free (see Lemma 3.18).

Note that an agent's utility in any given event does not increase after reaching their saturation amount. Therefore, reducing their own allocation to their saturation amount and changing nothing else maintains the social welfare and envy-freeness.

Observation 3.15. *It suffices to consider allocations A with $a_i(\omega) \le \hat{s}_i$ for all $i \in [n]$.*

From now on we only consider allocations in which all agents receive in each event at most their own saturation amount.

Any allocation $A = (a_1, \dots, a_n)$ is equivalent to the allocation by pieces A^p with $a_i^p(j) = a_i(\omega_j)$ and $\Phi(i, \omega_j) = j$. Considering the whole amount of the allocation, we can observe that the pieces can be reduced to a number of pieces allocating exactly the saturation amount and at most one piece with less than the saturation amount. This means that we can create a new allocation by pieces which allocates to each agent the same total amount as in A. We denote this allocation as *total amount allocation*.

Definition 3.16 (Total Amount Allocation)**.** An agent i's allocation can be represented as a total amount allocation $A_i = n_i \cdot \hat{s}_i + d_i$ with $0 \le d_i < \hat{s}_i$. This corresponds to the allocation in pieces $a_i^p(j) = \hat{s}_i$ for $j \in [n_i]$ with $n_i \in ([m] \cup \{0\})$ and $a_i^p(n_i + 1) = d_i$.

Given an allocation by pieces, a total amount allocation can be created in a stepwise manner. For example, assume that agent i has two pieces $(1, 2)$ below their saturation amount and that their allocation under 1 is weakly less than their allocation under 2; i.e. $a_i^p(1) \le a_i^p(2) < \hat{s}_i$. Then it is possible to reduce the allocation by $d := \min\{a_i^p(2), \hat{s}_i - a_i^p(1)\}$ in piece 2 ($a_i^{p'}(2) := a_i^{p'}(2) + d$) and increase it in piece 1 ($a_i^{p'}(1) := a_i^{p'}(1) + d$).

We remark again that we ignore Φ here. Generally, a valid Φ does not have to exist for all total amount allocations. However, the aim is to show that for any allocation to achieve a social welfare of B', any agent i's allocation has to be exactly $\hat{s}_i$ in one event and zero in all other events. This is independent of the choice of the event in which the agent gets their saturation amount. In this respect, the total amount allocation is suitable since its social welfare is the same as the social welfare of the allocation it is based on, and if the original allocation is ex-ante envy-free and a valid Φ exists, then the allocation $(a_i^p(\Phi(i, \omega)))_{i \in [n], \omega \in \Omega}$ is also ex-ante envy-free. The social welfare is unaffected since the utility functions are linear.

Lemma 3.17. *The ex-ante social welfare of an allocation and its representing total amount allocation are equivalent.*

It is straightforward to see that shifting allocation from smaller events to bigger events can transform an allocation to a total amount representation. The details of the technical proof are in the appendix (see Section A.2.2)

The envy-freeness of $(a_i^p(\Phi(i, \omega)))_{i \in [n], \omega \in \Omega}$ can be argued with the shifts to create a total amount allocation. An agent i is indifferent between the allocation before and after a shift since they receive the same amount in total and all events are equally likely. Every other agent j's expected utility of agent i's allocation after the shift is weakly less than before the shift, since it is the same amount but all expected utilities are subject to agent j's saturation amount. Hence, the difference in the ex-ante envy-freeness inequalities can only increase (for details see Section A.2.2 in the appendix).

Lemma 3.18. *If the a_i^p are ex-ante envy-free then the total amount allocation is ex-ante envy-free. Moreover, if the a_i^p are ex-ante envy-free and a valid Φ exists then $(a_i(\Phi(i, \omega)))_{i \in [n], \omega \in \Omega}$ is ex-ante envy-free.*

The following lemmata are used in the shifting procedure in the proof of Theorem 3.11. If any shift based on the following lemmata ends in an allocation for which a valid Φ exists then by Lemma 3.18 we have reached an envy-free allocation.

To begin with, shifting allocation to agents with a higher index increases social welfare based on the order of the agents (see Section A.2.3 in the appendix for details).

Lemma 3.19. *Fixing an allocation A and an agent $1 \leq k < 4m$, if all agents i where $i < k$ have a total amount allocation of $A_i = \hat{s}_i$ and agent k's total amount allocation is $A_k > \hat{s}_k$, then shifting allocation to create allocation A' such that any excess is shifted to the next agent, i.e. $A'_k = \hat{s}_k$ and $A'_{k+1} = A_{k+1} + A_k - \hat{s}_k$, increases social welfare.*

Moreover, envy-freeness stipulates that, if an agent receives less than their saturation amount, then all agents of higher index also receive less than their saturation amounts (see Section A.2.3 in the appendix for details).

Lemma 3.20. *For two agents $i, k \in [4m]$ with $i > k$, if agent k has a total amount allocation of less than, or equal to, their saturation amount, i.e. $A_k \leq \hat{s}_k$, then the same must be true for agent i, i.e. $A_i \leq \hat{s}_i$. The same holds for the strict case, i.e. if $A_k < \hat{s}_k$ then $A_i < \hat{s}_i$.*

Further, envy-freeness imposes the following conditions on Φ.

Lemma 3.21. *If for an envy-free allocation every agent has an allocation of once their saturation amount then the agents $2, \dots, 4m$ have to have their saturation amount in exactly one event or piece.*

Proof. If the allocation of $\hat{s}_i$ for $i \in [n] \backslash \{1\}$ is in one event, the expected utility of agent 1 for the allocation of agent i will be $\hat{s}_1$ (the remainder $\hat{s}_i - \hat{s}_1$ is of no value to agent 1). By contradiction, if $\hat{s}_i$ is split between two or more events, then agent 1 will receive additional expected utility from $\hat{s}_i - \hat{s}_1$ and hence their expected utility for agent i's allocation will be more than $\hat{s}_1$, implying that agent 1 envies agent i's allocation. Hence, the allocation of agent i has to be in one event. □

Finally, we are able to prove the main theorem.

Proof of Theorem 3.11. It is easy to see that given an allocation, we can calculate the social welfare and verify the envy-freeness in polynomial time. Therefore D-UAFD is in NP. To prove NP-completeness, we show that the constructed envy partition instance (see Definition 3.13) is a yes-instance if and only if the given 3-partition instance is a yes-instance.

First, consider the case where there is a valid partition $S_1, \dots, S_m$ for the 3-partition instance. Consider the allocation A where for $s_i \in S_j$ agent $i+m$ is allocated $\hat{s}_{i+m}$ in ω_j for all $i \in [3m]$ and for $i \in [m]$ agent i gets assigned $\hat{s}_i$ in ω_i. Recall that $\hat{s}_{i+m} = s'_i = \frac{s_i}{mB}$ for all $i \in [3m]$, and $\hat{s}_i = i \cdot \varepsilon$ for all $i \in [m]$. It is thus easy to verify that allocation A is valid as $\sum_{s_i \in S_j} s'_i = \sum_{s_i \in S_j} \frac{s_i}{mB} = \frac{1}{m}$. The expected utility for an agent $i \in [n]$ with respect to their own allocation is $V_i(a_i) = \frac{i}{m} \cdot \hat{s}_i$, and their expected utility with respect to the allocation of agent $j \in [n]$ is $V_i(a_j) = \frac{i}{m} \cdot \hat{s}_i$ if $j > i$ and $V_i(a_j) = \frac{i}{m} \cdot \hat{s}_j \leq \frac{i}{m} \cdot \hat{s}_i$ if $i > j$. Hence, A is ex-ante envy-free. Furthermore, the sum of the expected utilities is $\sum_{i \in [n]} V_i(a_i) = \frac{1}{m} \left(\sum_{i=1}^{3m} (i+m) \cdot s'_i + \sum_{i=1}^{m} i^2 \cdot \varepsilon \right) = B'$, which concludes this case.

Now, consider the case where there is an envy-free allocation A with social welfare at least B' for the constructed envy partition instance. By Observation 3.15, we can assume that every agent receives in every event at most their saturation amount. Therefore, A can be represented as an allocation by pieces (Definition 3.14), and thus has a total amount allocation representation $(A_1, \ldots, A_{4m})$ (Definition 3.16) that has the same ex-ante social welfare as A (by Lemma 3.17) and is ex-ante envy-free (by Lemma 3.18). This gives us a framework in which we show that, independent of Φ, to achieve B' every agent, except maybe agent 1, has to get their saturation amount exactly once; i.e. $a_i(\omega') = \hat{s}_i$ for one $\omega' \in \Omega$ and $a_i(\omega) = 0$ for all $\omega \in \Omega \setminus \{\omega'\}$, for all agents i, $2 \leq i \leq 4m$.

Starting from the total amount allocation $(A_1, \ldots, A_{4m})$ and considering agents one by one in increasing order of indices, we construct another allocation A' using the following procedure. If the current agent i is agent $4m$ or has a total amount allocation $A_i < \hat{s}_i$, we stop. If the current agent i has a total amount allocation $A_i > \hat{s}_i$ the additional amount $(A_i - \hat{s}_i)$ is shifted to the next agent.

If the procedure reaches agent $4m$ and no amount has been shifted, then for each agent $i < 4m$ we have that $A_i = \hat{s}_i$ and, by Lemma 3.20, we have that $A_{4m} \leq \hat{s}_{4m}$. An allocation where every agent is allocated exactly their saturation amount has social welfare of B'. Thus, if $A_{4m} < \hat{s}_{4m}$ then $W(A) < B'$ which contradicts our assumption. If $A_{4m} = \hat{s}_{4m}$ then, by Lemma 3.21, every agent in A, except maybe agent 1, is allocated theit saturation amount exactly once in one event.

If the procedure reaches agent $4m$ and any amount has been shifted during the procedure, then by Lemma 3.19, $W(A') > W(A)$. Furthermore, each agent $i < 4m$ is allocated exactly their saturation amount in A' and the total amount of resource available stipulates that agent $4m$ is allocated at most their saturation amount. An allocation where every agent is allocated exactly their saturation amount has social welfare of B'. Therefore $W(A') \leq B'$, and hence $W(A) < B'$ which contradicts our assumption.

If the procedure stops at an agent $i < 4m$, then we have that (1) every agent $j < i$ is allocated exactly their saturation amount in A', and (2) since $A_i < \hat{s}_i$ then, by Lemma 3.20, for each agent $j \geq i$ we have that $A_j < \hat{s}_j$ and hence each such agent is allocated less than their saturation amount in A'. An allocation where every agent is allocated exactly their saturation amount has social welfare of B'. Therefore $W(A') < B'$. By Lemma 3.19, $W(A') \geq W(A)$, and hence $W(A) < B'$ which contradicts our assumption. Hence we can conclude that either each agent, except maybe agent 1, is allocated their saturation amount exactly once, or $W(A) < B'$ which is a contradiction.

So far we have established that allocation A is equivalent to the allocation by pieces A^p where $a_i^p(1) = \hat{s}_i$ and $a_i^p(j) = 0$ for $j \in ([m] \backslash \{1\})$, for all $i \in [4m] \setminus \{1\}$. In the remainder of the proof we show that a valid Φ exists only if there exists a 3-partition and, moreover, that agent 1 receives their saturation amount exactly once in A as well. Since by assumption A is valid, a valid Φ must exist and thus a 3-partition exists and, moreover, every agent receives their saturation amount exactly once in A.

For easier presentation, we multiply saturation amounts and the amount of events by mB which results in saturation amounts of agents $m+1 \ldots 4m$ to be equal to 3-partition elements, which are all integers, and the amount of resource in each event $j \in [m]$ to be equal to $B + B \cdot m \cdot j \cdot \varepsilon$.

Recall that integer B denotes the size of each set. Furthermore, by the choice of ε, $B \cdot m \cdot j \cdot \varepsilon < 1$ for all $j \in [m]$.

We now investigate what must hold for a valid Φ to exist. We first claim that each agent i with $2 \leq i \leq m$ must be assigned to event i. If that were not the case, then there must be an event $j \in [m]$ with a remaining unallocated amount of less than B. Hence, since no event has an amount of $B+1$ or more, this implies that we have a total amount of less than mB that we can assign to agents $m+1 \ldots 4m$. But then these agents' saturation amounts add up to mB so it impossible to have these agents assigned. A similar argument holds if agent 1 is not assigned to event 1, and hence we conclude that each agent $i \in [m]$ is assigned to event i. Therefore, the remaining amount in each event is exactly B. We have that the saturation amounts of agents $m+1 \ldots 4m$ are equal to their corresponding 3-partition elements, hence the existence of a valid Φ implies that there exists a 3-partition, which can be directly derived from Φ.

Therefore, the envy partition instance is a yes-instance if and only if the 3-partition instance is a yes-instance. Finally, since the envy partition instance can be constructed in polynomial time and is of polynomial size this concludes the strong completeness (for details see Lemma A.3 and Lemma A.4 in Section A.2.1). □

3.5 Two Agents Linear Satiable Utility Functions

Following the two negative results we use the insights of those to decide the limit of the number of agents in order to investigate the effects of the probability distribution over the events and provide algorithms. We further use the instances of the negative results to decide what utility functions to consider. Firstly, we used an instance with linear utility functions to prove the price of envy-freeness (Theorem 3.8) and, secondly, we used an instance with utility functions that are linear up to a point and then constant to prove the NP-hardness (Theorem 3.11). Moreover, the hardness instance uses only uniform probabilities. Hence, the distribution plays less of a role and the distribution and the utility functions cannot be further relaxed. Additionally, the complexity proof highlights that the complexity is based on the number of agents. Therefore, taking all these points together, we relax the number of agents and consider two agents with utilities that are linear up to a maximal value. Accordingly, unless explicitly stated otherwise, $n = 2$ for the rest of this section. This allows us to explore the algorithmic possibilities for two agents and the influence of the probability distribution.

We begin examining the setting by defining the utility functions which are linear up to a maximal value (see Section 3.5.1). This includes how previous definitions of expected utility and ex-ante envy-freeness can be simplified as a consequence. Following on, we establish that for this 2 agent setting, similar to the case of linear utilities, specific settings do not allow allocations with more social welfare than that provided by equal share (See Section 3.5.2). Hence, we simultaneously declare to focus on the instances where we can gain ex-ante social welfare under ex-ante envy-freeness (see Assumption 3.29 and Assumption 3.31). We proceed by presenting our positive and negative results separately, considering those instances with uniform and arbitrary probability distributions over the events. In Section 3.5.3, we discuss the algorithm which provides the optimal and approximate results. This is followed by the result that for uniform probabilities over the events the algorithm indeed produces an optimal allocation (see Section 3.5.4). The contrasting result that for arbitrary probability distributions the problem is NP-hard is presented in Section 3.5.5.1. Following the hardness result, we discuss the approximation guarantees (see Section 3.5.5.2). The counter-intuitive result that, when events are ordered by the expected amount, the algorithm cannot approximate the optimal solution is demonstrated in Section 3.5.5.2. For the other ordering by amount only, we show in Section 3.5.5.2 that the algorithm's allocation structurally cannot be too different to an optimal allocation. We conclude by comparing the two orderings empirically in Section 3.5.7, demonstrating that both perform well on 'average' instances. This comparison relies on the difference to the maximal achievable ex-ante envy-free allocation, determined via our integer programming problem (see Section 3.5.6).

3.5.1 Preliminaries

We begin with the definitions of the utility functions which are linear up to a maximal value. These functions, referred to as *linear satiable utilities*, are determined by a *saturation amount* q and a *maximal value* u.

Definition 3.22 (Linear Satiable Utilities)**.** A *linear satiable utility* $v : [0,1] \to \mathbb{R}^+$ is determined by the tuple (q,u) and is defined as

$$v(x) = \begin{cases} \frac{u}{q} \cdot x & \text{if } x \leq q \\ u & \text{else} \end{cases}.$$

The set of all linear satiable utility functions is denoted by $\mathcal{L}_{\mathcal{M}} = \{(q,u) | q \in [0,1], u \in \mathbb{R}^+\}$.

Besides limiting the amount that is valuable to an agent, the agent's saturation amount also limits the amount up to which an agent might desire another agent's allocation. Hence, for brevity, we denote an agent i's allocation limited by any other agent j's saturation amount by a_i^j.

Definition 3.23 (Saturation Amount Limited Utility Function)**.** For any allocation A and any event ω let $a_i^j(\omega)$ denote the allocation function a_i of agent i limited by the saturation amount q_j of agent j. More precisely, let $a_i^j(\omega) := \min\{a_i(\omega), q_j\}$.

The linearity and the saturation amount of the linear satiable utilities allow us to rewrite the expected utility as well as the equations representing envy-freeness. For these utility functions the expected utility simplifies mostly to the sum of allocated amount subject to the saturation amount.

Observation 3.24. *The utility for agent* $i \in [n]$ *with* $(u_i, q_i) \in ([0,1], \mathbb{R}^+)$ *is*

$$V_i(a) = \frac{u_i}{q_i} \cdot \sum_{\omega \in \Omega} \min\{a(\omega), q_i\} f(\omega).$$

This observation shows that the expected utility separates into the allocated amount and a factor of the fraction of saturation amount q_i and maximal value u_i. This means that the envy-freeness depends solely on the difference, subject to the saturation amount, of the allocated amounts of any agent and any another agent. Particularly, that means that envy-freeness is independent of the maximal values.

Observation 3.25. *The envy-freeness of agent $i \in [n]$ with respect to agent $k \in [n]$ is represented by the equation*

$$EF(i,k) := \sum_{\omega \in \Omega} (\min\{a_i(\omega), q_i\} - \min\{a_k(\omega), q_i\})\, f(\omega) \geq 0.$$

3.5.2 Assumptions on Agent's Utility and Saturation Amounts

The setting in the example in Section 3.2 allows for a more efficient allocation than equal share whilst maintaining ex-ante envy-freeness. However, certain combinations of the saturation amount and maximal value still do not allow more social welfare than what equal share admits. As the following lemma shows, this is the case for four combinations of maximal values, saturation amounts and sizes of the events.

Lemma 3.26. *If one of the following four cases is satisfied then the ex-ante social welfare of equal share is maximal considering ex-ante envy-free allocations:*

1. $q_1 = q_2$ *or* *2.* $u_2 \geq u_1$ *and* $q_1 > q_2$ *or* *3.* $\frac{u_1}{q_1} = \frac{u_2}{q_2}$

or *4.* $\omega \geq 2 \cdot \min\{q_1, q_2\}$ *for all* $\omega \in \Omega$

Case 2 obviously also applies when changing the order of the agents.

Before we prove this lemma, we adjust equal share to cover the edge cases where events are significantly bigger than the smaller saturation amount of the two agents.

Definition 3.27 ((Adjusted) Equal Share Allocation)**.** The *(adjusted) equal share allocation* is the allocation where for all $\omega \in \Omega$

$$a_i(\omega) = \begin{cases} \frac{\omega}{2} & \text{if } \omega < 2 \cdot q_i \\ q_i & \text{otherwise} \end{cases} \quad \text{and} \quad a_j(\omega) = \begin{cases} \frac{\omega}{2} & \text{if } \omega < 2 \cdot q_i \\ \min\{\omega - a_i(\omega), q_j\} & \text{otherwise} \end{cases}$$

where $i = \arg\min_{k \in [2]}\{q_k\}$ and $j \in [2] \backslash \{i\}$ holds.

For the rest of this section this definition supersedes equal share (see Definition 3.2) and any reference of equal share refers to the adjusted equal share allocation.

Additionally, we want to observe that the conclusion of Observation 3.15 also applies here. We capture this in the following observation.

Observation 3.28. *It suffices to consider allocations A with $a_i(\omega) \leq q_i$ for all $i \in [n]$.*

Using preceding definition of equal share and the observation, we prove the lemma on the cases that can be excluded.

Proof of Lemma 3.26. The statement can be verified by considering the envy-freeness constraint for linear satiable utilities (see Observation 3.25). We start with the case where $q_1 = q_2$. Since we assume no more resource is allocated than an agent would value (see Observation 3.28), we can remove the references to the saturation amount from the envy-freeness: $EF(i,k) = \sum_{\omega\in\Omega} (a_i(\omega) - a_k(\omega))\, f(\omega) \geq 0$. Since this equations holds in both directions, $\sum_{\omega\in\Omega} a_i(\omega)\cdot f(\omega) \geq \sum_{\omega\in\Omega} a_k(\omega)\cdot f(\omega)$ and $\sum_{\omega\in\Omega} a_k(\omega)\cdot f(\omega) \geq \sum_{\omega\in\Omega} a_i(\omega)\cdot f(\omega)$, we can see that the allocated amount has to be the same. Hence, any allocation is just a version of equal share.

For the second case, i.e. if $u_2 > u_1$ and $q_1 > q_2$ holds, we can observe that envy-freeness implies that any expected amount given to agent 2 has to be also given to agent 1 since agent 1's saturation amount is higher. Moreover, since an allocation to agent 2 has more value we should give agent 2 as much allocation as possible. Both of these observations together mean again that any optimal allocation has the same value as equal share.

For the third case, i.e. $\frac{u_1}{q_1} = \frac{u_2}{q_2}$, it is easy to see that the social welfare is just the total amount multiplied by one of the two marginal utility fractions:

$$V_1(a_1) + V_2(a_2) = \frac{u_1}{q_1} \cdot \left(\sum_{\omega\in\Omega} (a_1(\omega) + a_2(\omega)) f(\omega) \right).$$

Hence, any two allocations which allocate all available resource that can be allocated (i.e. where for every event $\omega \in \Omega$ we have $a_1(\omega) + a_1(\omega) = \min\{\omega, q_1 + q_2\}$) have the same value. This especially includes all ex-ante envy-free allocations including equal share.

Finally, in the last case, to allocate all available resource we have to allocate at least q_2 to the agent with the higher saturation amount. However, that means that the other agent has to have an allocation of q_2 in every event as well. This, again, is the equal share allocation. □

As a consequence of this lemma, we consider only instances for which allocations with a higher ex-ante social welfare than the respective equal share allocation exist.

Assumption 3.29. *Based on Lemma 3.26, if we can gain more ex-ante social welfare, one agent has a higher saturation amount and this agent's maximal value has to be higher than that of the other agent. Without loss of generality, we assume that it is agent 1 who satisfies this. Hence, we assume $q_1 > q_2$ and $u_1 > u_2$. Moreover, we assume that there is at least one event $\omega \in \Omega$ which is less than $2 \cdot q_2$.*

For the rest of this work we consider only settings as specified by Assumption 3.29. Nevertheless, Case 3. of Lemma 3.26 still allows two different settings. The settings are that either the first agent's marginal utility is greater than the second agent's marginal utility, i.e. $\frac{u_1}{q_1} > \frac{u_2}{q_2}$, or the second agent's marginal utility is greater than the first agent's marginal utility, i.e. $\frac{u_2}{q_2} > \frac{u_1}{q_1}$. Of these two cases the former case is more interesting and is the focus of the rest of this section. In the latter case, where the second agent contributes marginally more to the social welfare, an optimal allocation can be computed straightforwardly. We prove this fact in the following lemma that gives a sample of the general approaches used to solve the case where the first agent's marginal utility is bigger.

Lemma 3.30. *In the context of Assumption 3.29 if $\frac{u_2}{q_2} > \frac{u_1}{q_1}$ then the optimal allocation can be found in polynomial time.*

Proof. We start with a procedure to compute an allocation which we show is optimal. The procedure starts with setting the allocation of agent 2 to be as much as possible in all events, i.e. $a_2(\omega) = \min\{\omega, q_2\}$. Agent 1 gets as much as available or desired of the remaining resource $a_1(\omega) = \min\{\omega - a_2(\omega), q_1\}$. Following this, we modify the allocation to be envy-free. If agent 1 is not envy-free we shift resource from agent 2 to agent 1 beginning with the largest event until agent 1 is not envious any more, i.e. $EF(1,2) = 0$. Agent 2 has to be envy-free since their saturation amount is smaller than agent 1's saturation amount, and we give agent 1 only so much to not be envious any more. The claim is that this allocation is optimal.

It is clear that any allocation where agent 2 has an allocation of less than the amount of this allocation must have less social welfare. The reason is that any shift of ε from agent 2 to agent 1 decreases the social welfare of an allocation, i.e. $-\varepsilon\frac{u_2}{q_2} + \varepsilon\frac{u_1}{q_1} < 0$. Hence, the social welfare has to be smaller.

Moreover, there cannot be an allocation with more social welfare. The created allocation gives allocation preferentially to agent 2. At the end of the procedure agent 2 has as much as their saturation amount in every event or agent 1 has just enough to be not envious any more. It is clear that the if agent 2 has as much as their saturation amount in every, such an allocation cannot be improved. Similarly, if agent 1 has just enough to be not envious any more, any allocation given to agent 2 would make agent 1 not envy-free since their saturation amount is bigger than that of agent 2. Hence, such an allocation could not be improved. Consequentially, the allocation is optimal.

Finally, the procedure to get this allocation can be computed in polynomial time. Firstly, the procedure considers every event at most once. At some point the procedure might considers an event where setting $a_1 = \min\{\omega, q_1\}$ and $a_2 = \min\{\omega - a_1(\omega), q_2\}$ would result in agent 1 being strictly not envy-free, i.e. $EF(1,2) > 0$. In this case we have to solve a set of linear equations to get the allocation of this event which results in the entire allocation being exactly envy-free, i.e. $EF(1,2) = 0$. This can also be done in polynomial time which means the whole procedure can be done in polynomial time. □

Having resolved this setting we focus on the setting where the first agent's marginal utility is bigger.

Assumption 3.31. *Based on Lemma 3.30 a setting where agent 2's utility is marginally bigger can be solved optimally in polynomial time. Hence, we assume that agent 1's utility is marginally bigger, i.e.* $\frac{u_1}{q_1} > \frac{u_2}{q_2}$.

In this case, allocating as much allocation as possible to the first agent maximises social welfare. However, agent 2 has to be still ex-ante envy-free. To achieve these two requirements (maximise agent 1's allocated amount and have agent 2 envy-free), we seek allocations which give agent 2 as little allocation as possible while still being envy-free. We refer to the fact that agent 2 has only as much allocation as necessary as *just* envy-free.

Definition 3.32 (Just Envy-Free)**.** An agent i is *just ex-ante envy-free* if the expected utility of their own allocation is equal to the expected utility of the other agents j's allocation, i.e. $V_i(a_i) = V_i(a_j)$ or $EF(i,j) = 0$.

In the following two lemmata we show that having agent 2 just envy-free is sufficient to show that an allocation is valid and envy-free and that, in an optimal allocation, agent 2 has to be just envy-free.

Lemma 3.33. *If agent 2 is just ex-ante envy-free then agent 1 is ex-ante envy-free.*

Proof. The claim of the statement (i.e. $EF(1,2) \geq 0$) is implied in the following way:

$$\sum_{\omega \in \Omega} \min\{a_1(\omega), q_1\} f(\omega) \overset{(*)}{\geq} \sum_{\omega \in \Omega} \min\{a_1(\omega), q_2\} f(\omega)$$
$$\overset{(**)}{=} \sum_{\omega \in \Omega} \min\{a_2(\omega), q_2\} f(\omega) \overset{(***)}{=} \sum_{\omega \in \Omega} \min\{a_2(\omega), q_1\} f(\omega).$$

In this chain of inequalities, the first sum is the allocated amount to agent 1. Since agent 2's saturation amount is smaller than agent 1's saturation amount ($q_1 \geq q_2$) the amount allocated to agent 1 cannot be smaller than agent 2's perception of this allocated amount (Inequality $(*)$). Moreover, by just envy-freeness ($EF(2,1) = 0$) agent 2's perception of agent 1's allocation has to be equal to the allocated amount of agent 2 (Equality $(**)$). Finally, since $q_1 \geq q_2$ and we do not allocate more than necessary (Observation 3.28) agent 1's perception of agent 2's allocated amount has to be the same (Equality $(***)$). Altogether, this implies $EF(1,2) \geq 0$. □

Lemma 3.34. *In an optimal allocation agent 2 is just ex-ante envy-free.*

Proof. We assume for contradiction that in an optimal allocation agent 2 is more than just envy-free, i.e. $V_2(a_2) > V_2(a_1)$. In this case we choose an event $\omega \in \Omega$ where agent 2 has a positive allocation and agent 1 has an allocation of less than q_1. Such an event has to exist since if no such event exists the two agents have to have the same amount allocated and $V_2(a_2)$ cannot be greater than $V_2(a_1)$. Taking any of the allocation in ω from agent 2 to give to agent 1 increases the allocation of agent 1 and since agent 2 was more than envy-free we can do this without agent 2 becoming envious. Consequently, since agent 1 has a higher marginal utility of the transferred allocation this also increases the social welfare. Hence, the allocation cannot be optimal which is a contradiction. □

3.5.3 Algorithm

In order to calculate the optimal allocation for uniform distributions and to approximate the optimal allocation for arbitrary distributions, we devise one general algorithm. The procedure of the algorithm depends on the order of the events and we define two orderings that give us two variants of the algorithm (for ordering by amount see Definition 3.37 and Algorithm 2 in Section 3.5.4, for ordering by expected amount see Definition 3.57 and Algorithm 3 in Section 3.5.5). Based on this order, the algorithm allocates 'bigger' events preferentially to agent 1 and allocates 'smaller' events preferentially to agent 2 so that agent 2 is just envy-free. To cover both variants and all cases, we describe the algorithm in general for any ordering and arbitrary probability distributions.

The algorithm called GreedyTake is formally presented in Algorithm 1 and works as follows. Initially, agent 1 receives as much allocation as possible in every event and agent 2 receives the remaining amount (line 1 and 2). Clearly, in this allocation, agent 2 would be envious. In order to rectify this, the second step (line 3 - 8) is to find the event such that giving agent 2 as much as possible in all events up to this event would make agent 2 not envious. Hence, the index of the current event j is moved, starting with the smallest (depending on the ordering) event (line 3, note that the first event is event 1), in increasing order (line 5) until agent 2 is not envious any more (line 4). In every step of this loop agent 2 gets allocated as much resource as possible in the current event (line 6), and agent 1 gets the remaining amount of the available amount (line 7). After the event j that makes agent 2 no longer envious is identified, that event has to be split to make agent 2 just envy-free (line 9 - 17). Due to the importance of this event for later arguments, we name this event the *split event*

Definition 3.35 (Split Event)**.** For any run of Algorithm 1 the event j which is fixed by line 9 is called *split event* and denoted as j^*.

The split could be achieved by increasing agent 2's allocation in event j until agent 2 is just envy-free. However, the algorithm determines this point (either line 13 or 15) using three values depending on the allocation in all other events, the amount of ω_{j*} and the saturation amount of agent 2. The first value represents agent 2's envy without event j^* (A, line 9). The second value reflects the available amount that affects agent 2's utility based on agent 2's saturation amount (B, line 10). Lastly, the third value indicates if agent 2 is satisfied before agent 1 gets more than agent 2's saturation amount (C, line 11).

The two possible cases covered by the *if* in line 12 depend on the allocation to agent 1 with respect to agent 2's saturation amount. If agent 2 is satisfied by a split in which agent 1 gets less than agent 2's saturation amount (line 12), then agent 2 can get essentially half of the amount that is available (subject to agent 2's saturation amount, value B) and the amount required to rectify the difference in allocation (value A). However, if the split means that agent 1's allocation increases beyond agent 2's saturation amount, then agent 2's allocation is agent 2's adjusted saturation amount (line 15). Finally, agent 1 gets the remaining resource in event j (line 17).

Informally, it is clear that a split to achieve just envy-freeness for agent 2 is possible. Agent 2 is envious when agent 1 gets as much resource as possible in the split event. On the other hand, agent 2 is envy-free when agent 2 gets as much resource as possible in the split event. Hence, there has to be a split to achieve just envy-freeness. Proving this statement requires us just to consider all possible cases and to show that line 13 and 15 are determining an allocation that is ex-ante envy-free. The proof is omitted here and can be found in the appendix (see Section A.3).

Lemma 3.36. *An allocation determined by* GreedyTake *is valid and ex-ante envy-free.*

3.5.4 Uniform Probability Distributions

The first case we are considering is the case of uniform probability distributions. As stated in the introduction of this section (see Section 3.5), an allocation determined by GreedyTake when events are ordered by their amount is optimal. In what follows, we formally define this ordering, referring to it as *amount ordering*, and the respective variant of the algorithm, and then prove the optimality claim.

Algorithm 1 GreedyTake
Algorithm that prefers agent 1 and tries to make agent 2 non envious.

Input: $(u_i, q_i)_{i \in [2]}$, Ω
$a_1(\omega) = \min\{\omega, q_1\}\ \forall \omega \in \Omega$
$a_2(\omega) = \min\{\omega - a_1(\omega) q_2\}\ \forall \omega \in \Omega$
$j = 0$
while $\sum_{\omega \in \Omega} v_2(a_2(\omega)) \cdot f(\omega) < \sum_{\omega \in \Omega} v_2(a_1(\omega)) \cdot f(\omega)$ **do**
    $j = j + 1$
    $a_2(\omega_j) = \min\{\omega_j, q_2\}$
    $a_1(\omega_j) = \min\{\omega_j - a_2(\omega_j), q_1\}$
end while
$A = \sum_{j' \in [m] \setminus \{j\}} a_1^2(\omega_{j'}) \cdot f(\omega_{j'}) - \sum_{j' \in [m] \setminus \{j\}} a_2(\omega_{j'}) \cdot f(\omega_{j'})$
$B = (\min\{\omega_j, q_2\} + \min\{\omega_j - \min\{\omega_j, q_2\}, q_2\}) \cdot f(\omega_j)$
$C = (q_2 - \min\{\omega_j - \min\{\omega_j, q_2\}, q_2\}) \cdot f(\omega_j)$
if $\sum_{\omega \in \Omega \setminus \{\omega_j\}} a_2(\omega) \cdot f(\omega) + \min\{\omega_j, q_2\} \cdot f(\omega_j) - C \leq \sum_{\omega \in \Omega \setminus \{\omega_j\}} a_1^2(\omega) \cdot f(\omega) + q_2 \cdot f(\omega_j)$ **then**
    $a_2(\omega_j) = \frac{A+B}{2 \cdot f(\omega_j)}$
else
    $a_2(\omega_j) = \frac{A}{f(\omega_j)} + q_2$
end if
$a_1(\omega_j) = \min\{\omega_j - a_2(\omega_j), q_1\}$

Definition 3.37 (Amount Ordering)**.** The *amount ordering* sorts the events increasing with respect to ω for $\omega \in \Omega$.

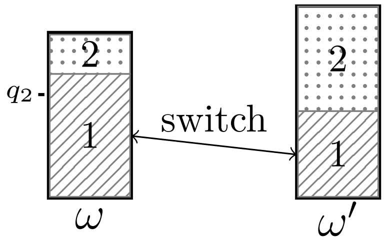
Accordingly, we define Algorithm 2 to refer to the variant of GreedyTake that processes the events ordered by the amount ordering as GreedyTake-Amt.

Algorithm 2 GreedyTake-Amt
Variant of GreedyTake (Algorithm 1) that uses the amount ordering (see Definition 3.37).

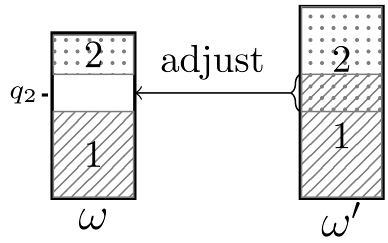
Input: $(u_i, q_i)_{i \in [2]}$, Ω
Sort Ω increasing by ω for $\omega \in \Omega$
GreedyTake($(u_i, q_i)_{i \in [2]}$, Ω)

To prove the optimality of the allocation determined by the algorithm denoted ALG, our main argument, which we prove in Theorem 3.39, is that any optimal allocation OPT can be transformed into an allocation like ALG. The algorithm's allocation is characterised by the split event and the events that are smaller (preferentially given to agent 2) and the events that are bigger (preferentially given to agent 1). The proof argues how the structure of an optimal allocation can be transformed into the structure of the algorithm's allocation. To demonstrate this, the proof relies on the fact that, once we know that all events bigger than the split event are allocated in the same way, we can immediately see that the ex-ante social welfare is the same. We prove this in the following supporting lemma.

Lemma 3.38. *If an optimal allocation OPT has the same allocation as the allocation of Algorithm 2 in all events j whose index is greater than the split event $j > j^*$ then the two allocations have the same ex-ante social-welfare.*



(A) In the switch step the allocation of agent 1 is switched between the two events.



(B) In the adjust step the allocation of agent 2 is adjusted such that the overall allocation is valid again.

FIGURE 3.2: Example of the switch/adjust procedure used in Theorem 3.39 with the left event $\omega \in \Omega_l$ and the right event $\omega' \in \Omega_r$. 3.2A and 3.2B indicate the switch and the adjust step, respectively.

Proof. We prove this claim by contradiction, i.e. we assume that there is an optimal allocation OPT whose allocation in the events j with $j > j^*$ is the same as the algorithm's allocation but the optimal allocation has more ex-ante social welfare.

We know from the procedure of the algorithm that, if agent 2 gets everything possible from event 1 to $j^* - 1$, then this is still not enough for agent 2 to be envy-free. Moreover, by the order of the events, this gives the least amount of events with the smallest opportunity for agent 1 to have allocation greater than agent 2's saturation amount to agent 2. Hence, we can conclude that agent 2 needs at least this amount to be envy-free.

If agent 2 does not have this amount allocated in the events 1 to $j^* - 1$ then some of it has to be allocated in the split event j^* (otherwise agent 2 would not be envy-free). In this case, we shift the allocation of OPT so that agent 2 gets allocated everything possible in the events 1 to $j^* - 1$. Following this shift, the only difference of the two allocations is the split event j^*. However, the algorithm gives as much as possible to agent 1. Hence, if the allocation of OPT in j^* after the shift would be different to the algorithm's allocation, then OPT has to have less social welfare than the allocation of the algorithm. Therefore, the allocations have to have the same ex-ante social welfare. □

Using this utility lemma, we are able to prove that the allocation ALG produced by Algorithm 1 is an optimal allocation.

Theorem 3.39. *For uniform probability distributions over the events the allocation ALG of* GREEDYTAKE-AMT *is optimal.*

Proof. Similar to the previous lemma, we prove this statement by assuming the opposite and using an exchange argument where we shift the optimal allocation to become the allocation of the algorithm. Formally, we assume that there is an optimal allocation OPT which has a higher ex-ante social welfare than the allocation of the algorithm ALG.

Before we begin, we separate the set of events in two sets based on the order and the split event. Let the set of all events smaller than the split event, called left events, be denoted as $\Omega_l = \{\omega_1, \dots, \omega_{j^*-1}\}$, and the set of all events greater than the split event, called right events, be denoted as $\Omega_r = \{\omega_{j^*+1}, \dots, \omega_m\}$. We change the optimal allocation by exchanging allocation from left to right using the following two-stage *switch/adjust* procedure (see also Figure 3.2).

For the *switch/adjust* procedure, we assume there is a left event ω where agent 1's allocation is greater than agent 2's saturation amount, i.e. $\omega \in \Omega_l \cup \{j^*\}$ with $a_1(\omega) > q_2$, and a right event ω' where agent 1's allocation is less than agent 2's saturation amount, i.e. $\omega' \in \Omega_r$ with $a_1(\omega') < q_2$. In the first step, the *switch* step, we switch agent 1' allocation of the two events, i.e. we set $a_1'(\omega) = a_1(\omega')$ and $a_1'(\omega') = a_1(\omega)$. In the second step, the *adjust* step, we shift the allocation for agent 2 from ω' to ω so that the allocation is valid again.

We can prove that this adjust step has to be possible by assuming the opposite, that the allocation of agent 2 in event ω' is big and cannot be shifted to event ω, i.e. $a_1(\omega) - a_1(\omega') > q_2 - a_2(\omega)$. However, this, together with the order of the events, implies that the allocation has allocated resource with no value. In detail, the assumption is equivalent to $q_2 + a_1(\omega') < a_1(\omega) + a_2(\omega)$ and the order of events means that $a_1(\omega) + a_2(\omega) \leq a_1(\omega') + a_2(\omega')$. Together, we have that the allocation of agent 2 in the right event before the procedure is $a_2(\omega') > q_2$. However, this contradicts our observation that there is no reason to allocate resource which give no further value (see Observation 3.15).

Now, we use this *switch/adjust* procedure to change the optimal allocation towards the allocation of the algorithm. If Ω_r of the optimal allocation is different to ALG and it is possible to shift allocation for agent 1 from $\Omega_l \cup \{j^*\}$ to Ω_r we perform the procedure. It is clear that this does not affect the social welfare.

After a number of applications of the *switch/adjust* procedure there will be no remaining left event and/or no remaining right event that satisfies our conditions for the procedure. At this point we try to shift allocation for agent 1 to the right events and for agent 2 to the left events. We do this if there is a right event where agent 1 has an allocation of at least agent 2's saturation amount but less than agent 1's saturation amount, and a left event where agent 2 has less than their saturation amount. If agent 1's allocation in the left event is bigger than agent 2's saturation amount, this is a shift that does not change social welfare or envy-freeness. If agent 1's allocation in the left event is at most agent 2's saturation amount we can decrease agent 2's allocation since shifting allocation for agent 1 decreases agent 2's expected utility of agent 1's allocation. Hence, we can maintain envy-freeness while increasing social welfare.

After the shifting of allocation, if the allocation in Ω_r of the changed OPT is the same as ALG, then the ex-ante social welfare has to be the same by Lemma 3.38. In contrast, we assume the changed OPT allocation is still different to ALG in Ω_r but we cannot switch and adjust or shift any more resource. Since we have no events left to perform a switch/adjust or a shift and the events are in increasing order, agent 1 has to have less allocation beyond agent 2's saturation amount. Consequentially, agent 2 has to have more allocation in OPT than in ALG. This implies that the ex-ante social welfare of OPT is smaller than the one of ALG. This is clearly contradicting the optimality of OPT. Hence, ALG has to be optimal. □

3.5.5 Non-Uniform Probability Distributions

In comparison to the case of uniform probability distributions, for arbitrary probability distributions the problem becomes NP-hard even for the linear satiable utility functions. We prove this claim in the following section by a reduction from a version of the partition problem and subsequently present an approximation algorithm.

3.5.5.1 Complexity

Similar to the previous complexity proof, we consider the respective decision version of the problem to prove the complexity.

Definition 3.40 (Decision Version of Two Player Uncertain Amount Fair Division (D-2-UAFD))**.**

Instance: $\langle X, v_1, v_2, B\rangle$ where $X : \Omega \to [0,1]$ with finite $\Omega \subsetneq [0,1]$ is a discrete random variable, $v_1 \in \Theta$ and $v_2 \in \Theta$ are utility functions and $B \in \mathbb{R}$ is a bound.

Problem: Does there exist a valid allocation $A = (a_1, a_2) \in \mathcal{F}^2$ such that $V_i(a_i) \geq V_i(a_j)$ for all $i, j \in [2]$ and $W(A) \geq B$.

Θ, $\mathcal{F}$, V_i and $W(A)$ are defined as in the preliminaries.

This version of the problem is NP-complete. Similar to before, we formally prove the theorem after the construction and supporting statements at the end of this section.

Theorem 3.41. *D-2-UAFD is NP-complete.*

We prove the complexity by reduction from a version of the partition problem. The only difference to the partition problem is that we assume that no two elements have the same amount.

Definition 3.42 (Distinct Partition Problem)**.**

Instance: $\langle S\rangle$, where $S \subset \mathbb{N}$ is a finite set without duplicates.

Problem: Is there a subset $S' \subset S$ such that $\sum_{s \in S'} s = \sum_{s \in S \setminus S'} s$.

This minor difference does not change the complexity but makes the reduction and complexity proof less complicated.

Corollary 3.43. *The distinct partition problem is NP-complete.*

Proof. For the proof we refer to the proof of the complexity of the partition problem in 'Computers and Intractability' [40, Theorem 3.5]. Their proof is via a reduction from 3-dimensional matching (3DM) which uses binary representations of the 3DM instance for the elements of S. Hence, all elements are distinct as we have in the distinct partition problem. □

Based on a distinct partition problem instance we create an instance of D-2-UAFD called *two player envy partition instance*. Briefly, for each element of the partition problem, we create one event whose amount as well as probability is determined by the element. Moreover, we have two agents representing the subsets that the elements have to be divided into. One agent values the resource higher and requires more resource and should therefore get as much resource as possible for efficiency. The second agent requires strictly less than any of the events but because of envy-freeness has to get half of the elements allocated.

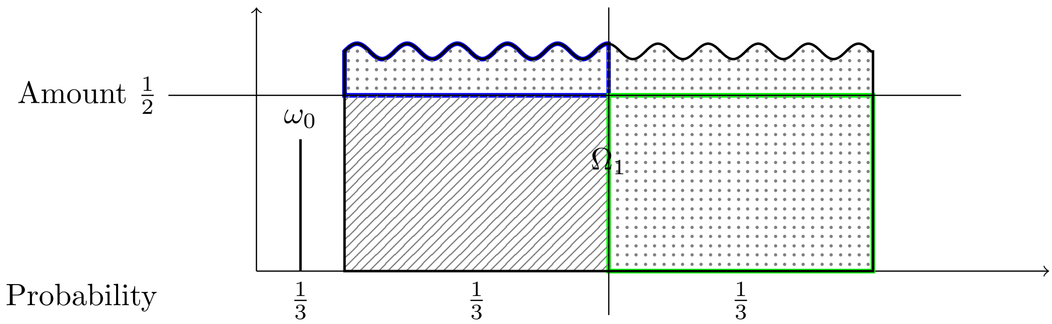


FIGURE 3.3: Illustration of the two player envy partition instance (see Definition 3.44 and the split of the partition allocation (see Definition 3.45). The elements of Ω_1 are bigger than 0.5. The elements of Ω_1 are not ordered by size in this illustration but rather represent the two halves of a possible partition. The wavy pattern indicates the different sizes due to the offset on top of 0.5 given by the respective s. The probability of the events in Ω_1 are chosen to add up to $\frac{2}{3}$ which means one half (i.e. a partition) is $\frac{1}{3}$. ω_0 is smaller than 0.5 but of the same value as the blue outlined area. The vertical lines indicate an allocation to agent 2 and the dots indicate an allocation to agent 1. Agent 2 perceives the green and the blue area of agent 1.

Definition 3.44 (Two Player Envy Partition Instance)**.** A *two player envy partition instance* $\langle X, v_1, v_2, B\rangle$ is an instance of D-2-UAFD constructed given a distinct partition problem instance $\langle S\rangle$ in the following way (see also Figure 3.3).

Firstly, we number the partition elements $S = s_1, \ldots, s_m$ from 1 to $m := |S|$ where the order can be arbitrary. We use those elements and the constant

$$b > \max\left\{2, \sum_{s\in S} s^2 \cdot s_{min}^{-1} \cdot \left(\sum_{s\in S} s\right)^{-1}\right\}$$

to define the events. Every event's amount is made up of a *base amount* of 0.5 and a *partition amount* that is relative to the normalised size of the partition element, depending on b. The probability of the events is chosen such that the probability of all these m events sum up to $\frac{2}{3}$ and the individual event's probability is relative to the respective normalised element.

$$\omega_i = 0.5 + \frac{s_i}{b \cdot \sum_{s\in S} s}, \quad Pr(\omega_i) = \frac{2}{3} \cdot \frac{s_i}{\sum_{s\in S} s}, \qquad i \in [m]$$

In addition to these events, we define one more event which occurs with probability $\frac{1}{3}$ and whose expected amount is half of the sum of the expected partition amounts (blue area in Figure 3.3).

$$\omega_0 = \frac{\sum_{s\in S} s^2}{b \cdot \left(\sum_{S\in S} s\right)^2}, \quad Pr(\omega_0) = \frac{1}{3}$$

Concluding the events' definition, let $\Omega_1 := \{\omega_1, \ldots, \omega_m\}$ and $\Omega := \Omega_1 \cup \{\omega_0\}$.

For the agents, we have one agent who wants as much resource as possible. Their utility v_1 is defined by the saturation amount $q_1 = 1$ and maximal value $u_1 = 2$. The other agent is happy with the base amount. Their utility v_2 is defined by saturation amount $q_2 = 0.5$ and maximal value $u_2 = 0.5$.

Finally, the bound B is set to $B := \frac{1}{2} + \frac{5}{3b} \cdot \sum_{S} s^2 \cdot \left(\sum_{s \in S} s \right)^{-2}$.

It is straightforward to see that the instance is valid and can be computed in polynomial time.

We prove the fact that an allocation for this instance can achieve B if and only if a partition exists in several steps. In particular, if there is a partition we can use it to separate the events of the two player partition instance into two halves for the two agents as illustrated in Figure 3.3. An allocation based on this split is ex-ante envy-free and achieves B. For the case that no partition exists we show in four steps that no allocation can achieve an ex-ante social welfare of B. Firstly, agent 1 needs a minimal amount for the allocation to achieve B. Secondly, agent 2 needs a minimal amount to be ex-ante envy-free. Thirdly, for agent 2 to have the minimal amount, agent 1's allocation has to have specific structural properties. That is, the allocation to agent 1 should have as many events where agent 1 gets all of the resource as possible. Finally, the previous three statements together show that if there is no partition, any allocation must have ex-ante social welfare of less than B.

A Partition Instance We begin with considering the case that there is a partition. In this case, we can split the events in Ω_1 in two halves based on the expected amounts. The following definition describes this more formally:

Definition 3.45 (Partition Allocation)**.** Let $\langle S \rangle$ have a partition and let S' be the set of indices of this partition. Then, the *partition allocation* A is defined as follows:

$$
\begin{aligned}
&a_1(\omega_i) = \omega_i, && \forall i \in S'; && a_1(\omega_i) = \omega_i - 0.5, && \forall i \in S \setminus S'; && a_1(\omega_0) = 0 \\
&a_2(\omega_i) = 0, && \forall i \in S'; && a_2(\omega_i) = 0.5, && \forall i \in S \setminus S'; && a_2(\omega_0) = \omega_0
\end{aligned}
$$

This partition is ex-ante envy-free since the second agent has, based on the expected amount, half of the events in Ω_1 to compensate for the other half plus the event ω_0 to compensate for the amount above the base amount in the events allocated to agent 2.

Lemma 3.46. *The partition allocation is ex-ante envy-free.*

Proof. It is obvious from the definition of the partition allocation (see Definition 3.45) that agent 1 has more allocation than agent 2 and therefore cannot be envious.

Agent 2's allocation is 0.5 in half of the events in Ω_1 and their allocation in ω_0 is the entire amount. Moreover, since agent 2 has a saturation amount of 0.5 they value the events in Ω_1 fully allocated to agent 1 only up to the amount of 0.5 (see green box in Figure 3.3). Furthermore, by construction (see Definition 3.44), ω_0 has exactly the same expected amount as the expected amount of agent 1's allocation in the events in Ω_1 where agent 2 has an allocation of 0.5 (see blue box in Figure 3.3). Hence, agent 2's utility of agent 1's allocation is the same as agent 2's own utility, which means agent 2 is not envious. □

By design the allocation also achieves the welfare bound.

Lemma 3.47. *The partition allocation has an ex-ante social welfare of B.*

Proof. To show this we can simply calculate the agent's expected utilities for the partition allocation.

The first agent's expected utility, denoted as U_1, is half of the base amount of Ω_1 (see green box in Figure 3.3) plus the entire partition amount ($\omega - 0.5$ in all events $\omega \in \Omega_1$), i.e.

$$U_1 := \frac{1}{2} \cdot \sum_{i \in [m]} \frac{1}{2} \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s \in S} s} + \frac{2}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2}.$$

The second agent's expected utility, denoted as U_2, is the other half of the base amount of Ω_1 (see vertical lined area in Figure 3.3) and event ω_0, i.e.

$$U_2 := \frac{1}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2} + \frac{1}{2} \cdot \sum_{i \in [m]} \frac{1}{2} \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s \in S} s}$$

Summing-up both of these expected utilities yields exactly the social welfare bound (remember $m = |S|$), i.e.

$$\begin{aligned}
&V_1(a_1) + V_2(a_2) = 2 \cdot U_1 + U_2 \\
&= \sum_{i \in [m]} \frac{1}{2} \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s \in S} s} + \frac{4}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2} + \frac{1}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2} + \frac{1}{2} \cdot \sum_{i \in [m]} \frac{1}{2} \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s \in S} s} \\
&= \frac{1}{3} + \frac{5}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2} + \frac{1}{6} = \frac{3}{6} + \frac{5}{3b} \cdot \sum_S s^2 \cdot \left(\sum_{s \in S} s\right)^{-2} = B
\end{aligned}$$

□

Minimal Allocation Agent 1 Having established the case where a partition exists, we consider the case where no partition exists. As introduced above, we perform this in four steps. The first step establishes a minimal amount of resource agent 1 requires for any allocation to achieve bound B. Since agent 1 has a higher marginal utility this agents contributes more to the social welfare. Consequentially, to achieve the social welfare bound agent 1's allocation has to be at least half of the expected partition amount (the amount above 0.5, see Definition 3.44). We state this more precisely in the following lemma:

Lemma 3.48. *Agent 1's allocation of the partition amount has to satisfy*

$$\sum_{\omega \in \Omega} |a_1(\omega) - 0.5| \cdot f(\omega) \geq \frac{1}{3} \cdot \sum_{s \in S} s^2 \cdot b^{-1} \cdot \left(\sum_{S \in S} s\right)^{-2}$$

for an allocation's ex-ante social welfare to achieve at least B.

Proof. For contradiction, we consider an allocation A for which the opposite holds and show that A's social welfare is less than B.

We start with considering the amount both agents have to have so that agent 2 is still envy-free. Since the partition amount of every event is beyond the saturation amount of agent 2 only, the base amount (part of the events that is less than 0.5) of the allocation has to be divided in half so that agent 2 is not envious. Hence, both agents' allocation has to contain half of the available amount minus the partition amount allocated to agent 1. We denote this amount with C.

$$C := \underbrace{\sum_{i\in[m]} \left(0.5 + \frac{s_i}{b \cdot \sum_{s\in S} s}\right) \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s\in S} s}}_{\text{Base amount in } \Omega_1} + \underbrace{\frac{\sum_{s\in S} s^2}{3 \cdot b \cdot \left(\sum_{S\in S} s\right)^2}}_{\omega_0} - \underbrace{\sum_{\omega\in\Omega} |a_1(\omega) - 0.5| \cdot f(\omega)}_{\text{Partition amount in } \Omega_1}$$

Using this the ex-ante social welfare can be expressed as follows.

$$\begin{aligned}
W(A) &= 2 \cdot \left[\frac{1}{2} \cdot C + \sum_{\omega\in\Omega} |a_1(\omega) - 0.5| \cdot f(\omega)\right] + \frac{1}{2} \cdot C \\
&= \frac{3}{2} \cdot \sum_{i\in[m]} \left(0.5 + \frac{s_i}{b \cdot \sum_{s\in S} s}\right) \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s\in S} s} + \frac{1}{2} \cdot \sum_{\omega\in\Omega} |a_1(\omega) - 0.5| \cdot f(\omega) + \frac{3}{2} \cdot \frac{\sum_{s\in S} s^2}{3 \cdot b \cdot \left(\sum_{S\in S} s\right)^2} \\
&= \frac{1}{2} + \frac{3}{2} \cdot \sum_{i\in[m]} s_i^2 \cdot b^{-1} \cdot \left(\sum_{s\in S} s\right)^2 + \frac{1}{2} \cdot \sum_{\omega\in\Omega} |a_1(\omega) - 0.5| \cdot f(\omega)
\end{aligned}$$

If we then apply the upper bound assumption on the partition amount of agent 1's allocation (opposite of the lemma's statement) we can bound the allocation A's maximal social welfare as follows.

$$\begin{aligned}
W(A) &< \frac{1}{2} + \frac{3}{2} \cdot \sum_{i\in[m]} s_i^2 \cdot b^{-1} \cdot \left(\sum_{s\in S} s\right)^2 + \frac{1}{2} \cdot \frac{1}{3} \cdot \sum_{s\in S} s^2 \cdot b^{-1} \cdot \left(\sum_{S\in S} s\right)^{-2} \\
&= \frac{1}{2} + \frac{5}{3} \cdot \sum_{i\in[m]} s_i^2 \cdot b^{-1} \cdot \left(\sum_{s\in S} s\right)^2 = B
\end{aligned}$$

However, this is a contradiction the assumption that A has at least a social welfare of B. □

Minimal Allocation Agent 2 Having established the minimal amount agent 1 requires for an allocation to achieve B, the immediate follow-up question is how much allocation has to be given to the second agent so that the allocation is envy-free. This required amount is half of the expected base amount.

Before we prove this, we make two observations about any allocation in this setting which simplifies the proof of the claimed minimal amount. Both observations are based on the fact that agent 1 can get more allocation than agent 2 by maximising the partition amount allocated to agent 1 since this amount is more than agent 2's saturation amount. The first observation is that it cannot be beneficial for agent 1 to have a non zero allocation in event ω_0 since it is smaller than agent 2's saturation amount.

Observation 3.49. *Agent 1 cannot gain any utility in ω_0 which is beyond agent 2's utility since ω_0 is smaller than the base amount $\omega_0 < \frac{1}{2}$.*

The second observation considers how the aim to maximise the partition amount in agent 1's allocation affects the structure of the allocation. More precisely, the partition amount in agent 1's allocation can be increased by having agent 2's allocation distributed over less events.

Observation 3.50. *Condensing Agent 2's allocation allows for an increased allocation for agent 1. If we have $a_2(\omega') < 0.5$ for at most one event $\omega' \in \Omega$ and for all other events $\omega \in \Omega_1 \backslash \{\omega'\}$ we have $a_2(\omega) \in \{0, 0.5\}$ then agent 1 can have more full events, i.e. $a_1(\omega) = \omega$, which allows agent 1 to have more allocation than agent 2.*

Finally, we can use these two observations in the proof for the claimed minimal allocation to agent 2, which is formally stated in the following.

Lemma 3.51. *The allocation to agent 2 has a lower bound of $\sum_{\omega \in \Omega_1} a_2(\omega) f(\omega) \geq \frac{1}{6}$ for agent 2 to be ex-ante envy-free.*

Proof. We assume for the sake of contradiction that $\sum_{\omega} a_2(\omega) f(\omega) < \frac{1}{6}$ and claim that even if we assume the statements of above observations, agents 2 is still envious.

1. ω_0 is allocated to agent 2, i.e. $a_2(\omega_0) = \omega_0$ (see Observation 3.49)
2. Agent 2's allocation is condensed, i.e. $a_2(\omega') < 0.5$ for at most one $\omega' \in \Omega$ and for all other events $\omega \in \Omega_1 \backslash \{\omega'\}$ it holds that $a_2(\omega) \in \{0, 0.5\}$ (see Observation 3.50)

We start with observing that since the allocation to agent 2 is less than $\frac{1}{6}$ and the allocation is condensed, we know that the allocation to agent 1 must contains at least half of the expected amount in Ω_1 as well as the partition amount in the other events. More precisely,

$$V_2(a_1) = \sum_{\omega \in \Omega_1} a_2(\omega) \cdot f(\omega) \geq \frac{1}{2} \cdot \sum_{i \in [m]} \frac{1}{2} \cdot \frac{2}{3} \cdot \frac{s_i}{\sum_{s \in S} s} + \frac{1}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2} = \frac{1}{6} + \frac{1}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2}.$$

In comparison, agent 2's allocation contains ω_0 and, as assumed, is strictly less than $\frac{1}{6}$ in Ω_1. Hence, agent 2's expected utility is

$$V_2(a_2) = v_2(\omega_0) \cdot f(\omega_0) + \sum_{\omega \in \Omega_1} v_2(a_2(\Omega_1)) \cdot f(\omega) < \frac{1}{3b} \cdot \frac{\sum_S s^2}{\left(\sum_{s \in S} s\right)^2} + \frac{1}{6}.$$

However, both of the equations together clearly imply $V_2(a_2) < V_2(a_1)$, which contradicts ex-ante envy-freeness and therefore our assumption. □

Structure Allocation Agent 1 The two lemmata of the previous two sections give us a minimal amount of allocation for both agents. The last step before we are able to prove the complexity claim of the theorem is that we can combine the two lemmata to prove that the allocation of agent 1 has to be somewhat condensed. In other words, agent 1 cannot have allocation above agent 2's saturation amount in too many events. This is expressed more precisely in the following lemma. In words, the claim is that the sum of the expected amount of events where agent 1 has an allocation beyond the base amount (0.5), can only be half the expected amount of Ω_1.

Lemma 3.52. *The allocation to agent 1 has to satisfy* $\sum_{\substack{\omega\in\Omega,\\ a_1(\omega)>0.5}} a_1(\omega)\cdot f(\omega) \leq \frac{1}{6}$.

Proof. Again, for contradiction we assume the opposite, i.e. agent 1 has more than $\frac{1}{6}$ in events where agent 1's allocation is greater than 0.5.

Let $\widehat{\Omega}_1$ be the set of events where the first agent has more than the base amount, i.e. $\widehat{\Omega}_1 := \{\omega | a_1(\omega) > 0.5, \omega \in \Omega\}\setminus\{\omega'\}$. Since half of the expected base amount of Ω_1 is $\frac{1}{6}$ ($\sum_{\omega\in\Omega_1} 0.5\cdot f(\omega) = \frac{1}{6}$), there has to be at least one more event $\omega' \in \widehat{\Omega}_1$ that extends agent 1's allocation to have more than halve the expected base amount of Ω_1.

Simultaneously, we assume that agent 2 has 0.5 in all events not part of agent 1's allocation, i.e. $a_1(\omega) = 0.5$ for all $\omega \in \Omega_1\setminus\widehat{\Omega}_1$. Since agent 2 has exactly the expected utility of 0.5 in those events and since there is at least one ω', agent 2's utility of those events has to be less than $\frac{1}{6}$. However, by Lemma 3.51 agent 2 requires at lest $\frac{1}{6}$. Hence, we have to assume that the events in $\widehat{\Omega}_1$ are not all full events, i.e. there exist $\omega \in \widehat{\Omega}_1$ with $a_1(\omega) < \omega$. On top, we assume ω' is the smallest event. This can only decreases the amount agent 2 requires to be envy-free and is therefore a valid assumption.

However, even making these assumption does not allow us to compensate for the missing allocation to agent 2 so that agent 2 is ex-ante envy-free, even if all of the partition amount in $\widehat{\Omega}_1$ is allocated to agent 2. The reason is that by the choice of b the expected partition amount is less than the expected base amount. Starting with expected base amount we have

$$\frac{1}{2}\cdot\frac{2}{3}\cdot\frac{s_{min}}{\sum\limits_{s\in S} s} \overset{\cdot\frac{b}{b}}{>} \frac{1}{3}\cdot\frac{s_{min}}{\sum\limits_{s\in S} s}\cdot\frac{\sum\limits_{s\in S} s^2}{b\cdot s_{min}\cdot\sum\limits_{s\in S} s} = \frac{1}{3}\cdot\frac{\sum\limits_{s\in S} s^2}{b\cdot\left(\sum\limits_{s\in S} s\right)^2}.$$

Hence, agent 2 cannot have the required allocation of $\frac{1}{6}$ (see Lemma 3.51), which is a contradiction to our assumption. □

Complexity Proof Finally, we can use above statements to show that there has to be a partition for any allocation to achieve the bound B which lets us proof the overall theorem.

Proof of Theorem 3.41. If there is a partition (the partition instance is a yes-instance) then by Lemma 3.46 and 3.47 the partition allocation (Definition 3.45) shows that the two-player envy partition (see Definition 3.44) instance is a yes-instance.

Contrarily, we consider any allocation without a partition (the partition instance is a no-instance). Then, we require by Lemma 3.51 that agent 2 has to have half of the expected base amount (0.5 of the events in Ω_1) and by Lemma 3.52 we require that agent 1 has to have half of the expected base amount through events where agent 1's allocation is more than the base amount. However, if there is no partition there is no way to split the events in two halves in accordance with these two requirements. Hence, it is impossible to satisfy both at the same time.

Consequentially, agent 1 has to have less allocation so that agent 2 is ex-ante envy-free. Therefore, by Lemma 3.48 the allocation has to have ex-ante social welfare of less than B. Overall, this means the two-player envy partition instance a no-instance. □

3.5.5.2 Approximation

Clearly the hardness proven in Theorem 3.41 has the common algorithmic consequences. In comparison to the optimisation formulation of the n agent problem we present below in Section 3.5.6 (which gives us an algorithm that is helpful for small instances), we are interested in what the possibility under polynomial runtime is. Investigating this, as stated in Section 3.5.3, we look at two different variants of GreedyTake determined by the ordering of events. The first variant is the same one we used in the uniform case (see Section 3.5.4) and the second variant is one where we sort the events by the expected amount. Initially it appears that taking the expectation of the events into consideration should lead to better results. Unfortunately, it turns out that theoretically we cannot guarantee an approximation when we sort by expected amounts. For the case where we sort by just the amount, we show that the algorithm's allocation is structurally close to an optimal allocation we specify. We consider the results for these two algorithm variants in this order: first for the sorting by the expected amount and second for the sorting by just the amount.

Before we present these results, we present a framework in which an approximation for this problem can hold and which is in line with maximising social welfare. This is necessary since we allow arbitrary utilities which means approximation guarantees expressed in social welfare can be arbitrarily bad as we show in the following Lemma 3.55. Instead of considering approximation ratios in terms of social welfare, we consider the approximation in terms of the amount of allocation to the first agent which exceeds the second agent's saturation amount. Since this allocation does not affect the envy-freeness of the second agent we refer to this amount as *(agent 2) indifferent* amount.

Definition 3.53 ((Agent 2) Indifferent Allocation)**.** For an allocation A, we denote as *agent 2 indifferent* or simply *indifferent* the resource amount $k_\omega^A := max(a_1(\omega) - q_2, 0)$ for all $\omega \in \Omega$. If the allocation A is clear from context we write k_ω instead of k_ω^A.

We prove that maximising the indifferent allocation is in line with maximising social welfare in Lemma 3.56.

Definition 3.54 (Indifferent Amount Framework)**.** The indifferent amount framework considers maximising the amount of the *indifferent* allocation as a proxy for maximising social welfare.

Prior to proving that the indifferent amount framework is aligned with maximising social welfare, we prove that an approximation ratio expressed in social welfare can be arbitrarily bad for any approximation algorithm. Informally, let us assume that in an optimal allocation one of the two agents has more resource than the other. Since an approximation algorithm is not optimal the difference in the allocation between the two agents has to be smaller in comparison to the optimal algorithm. The utility of this difference is crucial for the approximation ratio. However small the difference is, increasing the difference between the utility of the agents could increase the social welfare arbitrarily. We show this formally in the following lemma:

Lemma 3.55. *The approximation ratio in terms of social welfare can be arbitrarily bad.*

Proof. For $i \in \{1,2\}$ let a_i^{opt} be an optimal allocation function and a_i be the allocation function of an approximation algorithm. Due to our assumption that the optimal allocation OPT has more social welfare, agent 1 has to have more allocation (see also Assumption 3.29 and Assumption 3.31). Moreover, this allocation can only be indifferent allocation and it has to be smaller for the algorithm's allocation ALG. We denote this difference $\sum_{\omega\in\Omega} k_\omega^{opt} - \sum_{\omega\in\Omega} k_\omega$ with k. Additionally, we assume that agent 2 is also just envy-free in the approximation algorithm (see also Lemma 3.34) which can only be beneficial for the approximation algorithm. Consequentially, if the entire amount that can be allocated is allocated, the amount k has to be equally distributed to both agents with none of it being part of the indifferent allocation. Hence, the difference between the social welfare of ALG and the social welfare of OPT is as follows:

$$\begin{aligned} ALG &= OPT - \frac{u_1}{q_1}\cdot k + \frac{u_1}{q_1}\cdot\frac{1}{2}\cdot k + \frac{u_2}{q_2}\cdot\frac{1}{2}\cdot k \\ &= OPT + k\cdot\left(-\frac{u_1}{q_1} + \frac{1}{2}\cdot\frac{u_1}{q_1} + \frac{1}{2}\cdot\frac{u_2}{q_2}\right) \\ &= OPT + \frac{1}{2}\cdot k\cdot\left(-\frac{u_1}{q_1} + \frac{u_2}{q_2}\right). \end{aligned}$$

Considering the last expression, since agent 1's marginal utility is greater than agent 2's marginal utility we have that the $-\frac{u_1}{q_1} + \frac{u_2}{q_2}$ has to be negative. Moreover, this difference increases as the difference between the two agent's utilities also increases. Consequentially, the ratio of the algorithm and the optimal allocation increases. Hence, the approximation ratio continues to decrease with an increase of agent 1's utility. □

In contrast, the next lemma shows that expressing the approximation ratio in terms of the indifferent amount is sensible because it is in line with maximising social welfare.

Lemma 3.56. *Maximising social welfare is equivalent to maximising the indifferent amount.*

Proof. We assume for contradiction that the opposite is true. For that, let A be an allocation that maximises social welfare and A' be an allocation that maximises the indifferent amount. Hence, we have $W(A') < W(A)$.

We show that the difference between the allocations depends only on the indifferent amount. For that, we firstly express the total allocation R in terms of the allocation to the first agent. By Lemma 3.34 agent 2's allocation is the same as agent 1's allocation without the indifferent amount. By the same arguments, agent 2 has to be just envy-free in the allocation that maximises the indifferent amount. Hence, we have that

$$R \coloneqq \sum_{\omega\in\Omega} a_2(\omega)f(\omega) + \sum_{\omega\in\Omega} a_1(\omega)f(\omega) = 2\cdot\sum_{\omega\in\Omega} a_1(\omega)f(\omega) - \sum_{\omega\in\Omega} k_\omega f(\omega) \tag{3.1}$$

holds. The same holds for A' with k'_ω.

Using these equations we can express the social welfare of A' in terms of the total allocation and the indifferent amount. This gives us the following:

$$
\begin{aligned}
W(A') &= \frac{u_1}{q_1}\sum_{\omega\in\Omega}(a_1'(\omega)-k_\omega')f(\omega)+\frac{u_2}{q_2}\sum_{\omega\in\Omega}(a_2'(\omega))f(\omega)+\frac{u_1}{q_1}\sum_{\omega\in\Omega}k_\omega' f(\omega)\\
&\overset{(*)}{=}\left(\frac{u_1}{q_1}+\frac{u_2}{q_2}\right)\sum_{\omega\in\Omega}(a_1'(\omega)-k_\omega')f(\omega)+\frac{u_1}{q_1}\sum_{\omega\in\Omega}k_\omega' f(\omega)\\
&\overset{(**)}{=}\left(\frac{u_1}{q_1}+\frac{u_2}{q_2}\right)\frac{R}{2}-\left(\frac{u_1}{q_1}+\frac{u_2}{q_2}\right)\frac{\sum_{\omega\in\Omega}k_\omega' f(\omega)}{2}+\frac{u_1}{q_1}\sum_{\omega\in\Omega}k_\omega' f(\omega)\\
&=\left(\frac{u_1}{q_1}+\frac{u_2}{q_2}\right)\frac{R}{2}+\left(\frac{u_1}{q_1}-\frac{u_2}{q_2}\right)\frac{\sum_{\omega\in\Omega}k_\omega' f(\omega)}{2}
\end{aligned}\tag{3.2}
$$

In this we used the fact that the allocation to agent 2 is the allocation to agent 1 without the indifferent amount in Equality $(*)$ and we used Equations (3.1) in Equality $(**)$ by substituting $\sum_{\omega\in\Omega}a_1'(\omega)f(\omega)-\sum_{\omega\in\Omega}k_\omega' f(\omega)$ with $\frac{R}{2}-\frac{\sum_{\omega\in\Omega}k_\omega' f(\omega)}{2}$.

We rewrite the social welfare of A in the same way:

$$
W(A)=\left(\frac{u_1}{q_1}+\frac{u_2}{q_2}\right)\frac{R}{2}+\left(\frac{u_1}{q_1}-\frac{u_2}{q_2}\right)\frac{\sum_{\omega\in\Omega}k_\omega f(\omega)}{2}. \tag{3.3}
$$

From Equation (3.2) and (3.3) we can see that the difference in social welfare between A and A' depends on the indifferent amount as present in the second term.

Considering the difference between Equation (3.2) and (3.3), we chose A' to maximise the indifferent amount which means $\sum_{\omega\in\Omega}k_\omega' f(\omega)\geq\sum_{\omega\in\Omega}k_\omega f(\omega)$. Hence, it is easy to see that the social welfare of A' is at least that of A.

$$
W(A')-W(A)=\left(\frac{u_1}{q_1}-\frac{u_2}{q_2}\right)\frac{\sum_{\omega\in\Omega}k_\omega' f(\omega)}{2}-\left(\frac{u_1}{q_1}-\frac{u_2}{q_2}\right)\frac{\sum_{\omega\in\Omega}k_\omega f(\omega)}{2}\geq 0
$$

However, this contradicts our assumption that A' does not maximise social welfare. □

Expected Amount Using the indifferent amount framework (see Definition 3.54), we start with the case of sorting events by $\omega\cdot f(\omega)$ which we call the *expected amount ordering.*

Definition 3.57 (Expected Amount Ordering)**.** The *expected amount ordering* sorts the event increasing with respect to $\omega\cdot f(\omega)$ for $\omega\in\Omega$.

Correspondingly, we define Algorithm 3 to refer to the variant of GreedyTake that processes the events ordered by the expected amount ordering as GreedyTake-Exp.

As we introduced in the Section 3.5.5.2, it seems reasonable to assume that taking the probabilities into consideration is better than just considering the amount of the events. However, using this ordering can actually produce worse results. We show this by considering the instance depicted in Figure 3.4. In the following theorem, we show that GreedyTake-Exp does not allocate the only event in this instance with the possibility of an indifferent amount to agent 1. Consequentially, the approximation is arbitrarily bad.

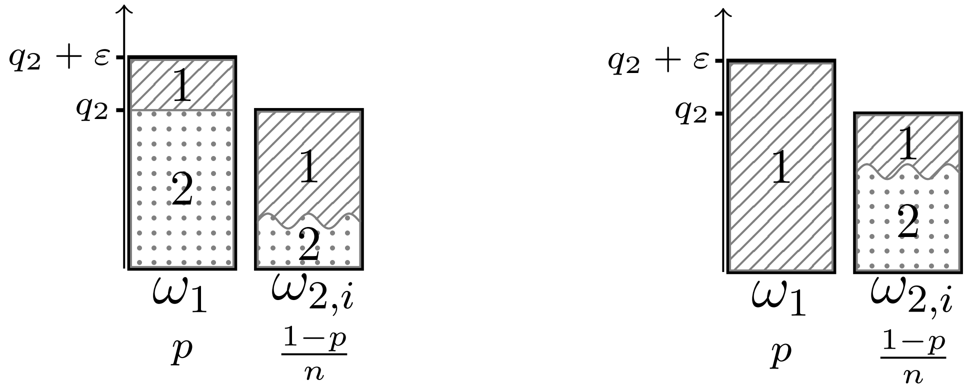
Algorithm 3 GreedyTake-Exp

Variant of GreedyTake (Algorithm 1) that uses the expected amount ordering (see Definition 3.57).

Input: $(u_i, q_i)_{i\in[2]}$, Ω

Sort Ω increasing by $\omega \cdot f(\omega)$ for $\omega \in \Omega$

GreedyTake($(u_i, q_i)_{i\in[2]}$, Ω)



(A) The allocation produced by GreedyTake-Exp (Algorithm 3). Agent 2 gets allocated the full saturation amount in event ω_1.

(B) An optimal allocation for this instance. In comparison to the algorithm, event ω_1 is fully allocated to the first agent.

Figure 3.4: The instance that shows that the expected amount ordering (GreedyTake-Exp) cannot approximate the indifferent amount (see Theorem 3.58). The instance has $n+1$ events ω_1 and $(\omega_{2,i})_{i\in[n]}$. Event ω_1 has a probability of p and an amount of $q_2+\varepsilon$. All events $\omega_{2,i}$ for $i \in [n]$ have a probability of $\frac{1-p}{n}$ and an amount of q_2. 3.4A and 3.4B show the allocations for the algorithm and the optimal solution, respectively. The wavy line indicates that the allocation of the events $(\omega_{2,i})_{i\in[n]}$ depends on the choice of q_1, q_2, p and ε.

Theorem 3.58. GreedyTake-Exp *(see Algorithm 3) may not approximate the indifferent amount.*

Proof. In order to prove the statement, we construct an instance for any number of events and any saturation amounts (subject to Assumption 3.29 and Assumption 3.31) and show that Algorithm 3 cannot approximate the indifferent amount for this instance. In this proof we assume for brevity that we can have several events of the same amount. The results are not affected by this since the events can easily be replaced by a number of events with slightly different amounts.

Our instance has one event ω_1 of size $q_2+\varepsilon$ with $0<\varepsilon<q_2$ and n events $(\omega_{2,i})_{i\in[n]}$ of size q_2. We fix the events' probabilities so that the expected amount of event ω_1 is less than that of any of the other events. In that case the algorithm will allocate q_2 in ω_1 to agent 2 which means that agent 1's indifferent allocation will be zero (see Figure 3.4A). In comparison, the optimal allocation can allocate the whole of event ω_1 to agent 1 which means agent 1 has a non zero indifferent amount (see Figure 3.4B). For the majority of the proof we show that the allocations exists as depicted and the claim follows straightforward from that.

In more detail, let event ω_1 have a probability of p and all other events $\omega_{2,i}$ with $i \in [n]$ have a probability of $\frac{1-p}{n}$. In order to have event ω_1 the first event in the expected amount ordering it must hold that $p \cdot \omega_1 < \frac{1-p}{n} \cdot \omega_{2,i}$ for all $i \in [n]$. With $\omega_1 = q_2+\varepsilon$ and $\omega_{2,i} = q_2$ for all $i \in [n]$. This implies that the value of p should satisfy

$$p = p \cdot \frac{q_2 + \varepsilon + \frac{q_2}{n}}{q_2 + \varepsilon + \frac{q_2}{n}} < \frac{1-p}{n} \cdot q_2 \cdot \frac{1}{q_2 + \varepsilon + \frac{q_2}{n}} + p \cdot \frac{\frac{q_2}{n}}{q_2 + \varepsilon + \frac{q_2}{n}} = \frac{q_2}{(n+1) \cdot q_2 + n \cdot \varepsilon} \tag{3.4}$$

The next step is to consider the allocations in both cases. While ω_1 is allocated as described and depicted in Figure 3.4, for the remaining events we have to allocate their total amount of $\sum_{i \in [n]} \frac{1-p}{n} \cdot q_2 = (1-p) \cdot q_2$ so that agent 2 is not envious.

In the case of the algorithm this means we have to find $x \in [0,1]$ such that the allocation to agent 2 equals to the allocation of agent 1 (restricted to q_2), i.e.

$$p \cdot q_2 + (1-p) \cdot x \cdot q_2 = p \cdot \varepsilon + (1-p) \cdot (1-x) \cdot q_2. \tag{3.5}$$

Using this equation (in the second step) we get that the value of x can be determined as follows.

$$\begin{aligned} x &= x \cdot \frac{2 \cdot (1-p) \cdot q_2 + p \cdot q_2 - p \cdot q_2}{2 \cdot (1-p) \cdot q_2} \\ &\overset{(3.5)}{=} \frac{p \cdot \varepsilon + (1-p) \cdot (1-x) \cdot q_2 + x \cdot (1-p) \cdot q_2 - p \cdot q_2}{2 \cdot (1-p) \cdot q_2} \\ &= \frac{p \cdot \varepsilon + (1-p) \cdot q_2 - x \cdot (1-p) \cdot q_2 + x \cdot (1-p) \cdot q_2 - p \cdot q_2}{2 \cdot (1-p) \cdot q_2} \\ &= \frac{p \cdot (\varepsilon - q_2) + (1-p) \cdot q_2}{2 \cdot (1-p) \cdot q_2} \end{aligned} \tag{3.6}$$

We are not interested in the precise value of x but that there exists a possible allocation. Since x has to be at least 0 is has to hold that

$$p = p \cdot \frac{2 \cdot q_2 - \varepsilon - \frac{q_2}{p} + \frac{q_2}{p}}{2 \cdot q_2 - \varepsilon} = \frac{p \cdot (q_2 - \varepsilon) - q_2 \cdot (1-p) + q_2}{2 \cdot q_2 - \varepsilon} \leq \frac{q_2}{2 \cdot q_2 - \varepsilon} \tag{3.7}$$

where the inequality is implied by $-x \leq 0$ (see Equation (3.6)).

Simultaneously, since x can be at most 1, Equation (3.6) implies that $p \cdot (\varepsilon - q_2) + (1-p) \cdot q_2 \leq 2 \cdot (1-p) \cdot q_2$ which means p has to satisfy

$$\begin{aligned} p &= p \cdot \frac{\varepsilon - 2 \cdot q_2 + 2 \cdot q_2 + 2 \cdot \frac{q_2}{p} - 2 \cdot \frac{q_2}{p}}{\varepsilon} \\ &= \frac{p \cdot (\varepsilon - q_2) + (1-p) \cdot q_2 - 2 \cdot (1-p) \cdot q_2 + q_2}{\varepsilon} \\ &\leq \frac{2 \cdot (1-p) \cdot q_2 - 2 \cdot (1-p) \cdot q_2 + q_2}{\varepsilon} \\ &= \frac{q_2}{\varepsilon}. \end{aligned} \tag{3.8}$$

However, neither equation restricts our choice of p further. Considering that q_2 is greater than ε, Equation (3.7) has to be smaller than Equation (3.8) and Equation (3.7) is clearly bigger than Equation (3.4) since $n \geq 1$.

Considering the optimal allocation we similarly have to determine $y \in [0,1]$ such that agent 2 is envy-free, i.e. $(1-p) \cdot y \cdot q_2 = p \cdot q_2 + (1-p) \cdot (1-y) \cdot q_2$. This implies

$$\begin{aligned} y &= y \cdot \frac{2 \cdot (1-p) \cdot q_2}{2 \cdot (1-p) \cdot q_2} \\ &\overset{(*)}{=} \frac{p \cdot q_2 + (1-p) \cdot (1-y) \cdot q_2 + (1-p) \cdot y \cdot q_2}{2 \cdot (1-p) \cdot q_2} \\ &= \frac{p \cdot q_2 + (1-p) \cdot q_2 - (1-p) \cdot y \cdot q_2 + (1-p) \cdot y \cdot q_2}{2 \cdot (1-p) \cdot q_2} \\ &= \frac{1}{2 \cdot (1-p)}, \end{aligned} \tag{3.9}$$

where we use the envy-freeness equality in Equality $(*)$. Again, we are only interested in the fact that there is a possible allocation.

It is easy to see that the expression in Equation (3.9) is positive for $p < 1$. For the case of $y \le 1$ using $-1 \ge -2 \cdot (1-p)$ we have

$$p = 1 - \frac{2 \cdot (1-p)}{2} \le 1 - \frac{1}{2} = \frac{1}{2}.$$

However, as in the previous case, this does not restrict our choice of p further since the denominator of Equation (3.4) is greater than $2 \cdot q_2$ for $n \ge 1$ which clearly means that the equation is smaller than $\frac{1}{2}$.

Altogether, if we choose p to satisfy Equation (3.4) we have that the optimal allocation has an indifferent amount of ε and the allocation of GreedyTake-Exp has no indifferent allocation. This means that this algorithm cannot approximate the indifferent amount. □

Actual Amount Unfortunately, as demonstrated in Theorem 3.58, taking the event's probabilities into consideration does not allow us to provide an approximation ratio in general. The proof relies on the fact that the sorting by expected amount might place events with indifferent amounts before events without indifferent amounts. This characteristic of sorting by the expected amount cannot occur if we simply order the events by their amount. Therefore, we conjecture that GreedyTake-Amt which sort the events by their amount ω (see Algorithm 2 and Definition 3.37) can approximate the optimal solution. While we do not provide an approximation ratio, we prove that the structural difference between the algorithm allocation and the optimal allocation is limited.

In particular, we show that, in the same way that the split event of the algorithm partitions the events into two sets, there is an optimal allocation that has two sets of events and a split event. We then consider those events that are classified differently by the optimal solution and the algorithm. For these event sets we show that switching events simply between these two sets cannot change the indifferent amount. Which indicates that the approximation ratio is dependent on the split event and how far it allows to shift allocation.

In more detail, we begin with the partitioning of the events for optimal allocations and the algorithm. Starting with GreedyTake-Amt, it separates the events into events smaller and bigger than the split event j^* (see Definition 3.35) For the smaller events the resource is preferentially given to agent 2. For the bigger events the resource is preferentially given to agent 1. Formally, we call this an *algorithm induced partition of* Ω.

Definition 3.59 ((Algorithm) Induced Ω Partition)**.** Let A^{amt} be an allocation obtained by running GreedyTake-Amt and j^* the respective split event. Then $\Omega_1 := \{\omega | \omega < \omega_{j^*}, \omega \in \Omega\}$ and $\Omega_2 := \{\omega | \omega > \omega_{j^*}, \omega \in \Omega\}$.

Similarly, there is an optimal allocation with two sets of events differentiated by if the events are preferentially given to the first or the second agent. We show this in the following lemma.

Lemma 3.60. *There is an optimal allocation* A^{opt} *with* $\Omega_1^{opt} := \{\omega | a_2^{opt}(\omega) = q_2 \vee a_2^{opt}(\omega) = \omega\}$, $\Omega_2^{opt} := \{\omega | a_1^{opt}(\omega) = q_1 \vee a_1^{opt}(\omega) = \omega\}$ *and at most one event* $\omega_{j^*_{opt}}$ *with* $a_1^{opt}(\omega_{j^*_{opt}}) < \min\{q_2, \omega_{j^*_{opt}}\}$ *and* $a_2^{opt}(\omega_{j^*_{opt}}) < \min\{q_2, \omega_{j^*_{opt}}\}$.

Proof. The main argument is that if there is more than one event $\omega' \in \Omega$ (with $a_1^{opt}(\omega') < \min\{q_2, \omega'\}$ and $a_2^{opt}(\omega') < \min\{q_2, \omega'\}$), then we can shift allocation to get an allocation of the same social welfare with one less of such events.

Let us assume that there are at least two such events $\omega', \omega'' \in \Omega$. We have two options: either increase or decrease agent 1's allocation in the first event. The allocation of agent 2 is shifted accordingly taking the probabilities into consideration. Considering both options and continuing to shift means that at some point either ω' or ω'' has to satisfy the conditions of Ω_1^{opt} or Ω_2^{opt}. Hence, we have removed at least one of the events not satisfying Ω_1^{opt} or Ω_2^{opt}. Continuing this procedure means that at most one event, denoted by $\omega_{j^*_{opt}}$, can remain. □

As a reminder, we are interested in the events that differ between the allocation of Definition 3.59 and Lemma 3.60. We denote those events that differ between Ω_1 and Ω_1^{opt} as well as Ω_2 and Ω_2^{opt} as *flip events*. The split event of the algorithm ω_{j*} may be a flip event in either Ω_1^{opt} or Ω_2^{opt}.

Definition 3.61 (Flip Sets)**.** For the events that are in Ω_1 in the algorithm's allocation A^{amt} and in Ω_2^{opt} in the optimal allocation A^{opt} we define the flip set Ω_1^F:

$$\Omega_1^F := \{\omega | \omega \in \Omega_1 \cup \{\omega_{j^*}\}, \omega \in \Omega_2^{opt}\}.$$

Similarly, for the events that are in Ω_2 in the algorithm's allocation A^{amt} and in Ω_1^{opt} in the optimal allocation A^{opt} we define the flip set Ω_2^F:

$$\Omega_2^F := \{\omega | \omega \in \Omega_2 \cup \{\omega_{j^*}\}, \omega \in \Omega_1^{opt}\}.$$

Informally, these two sets are the events where, in comparison to the algorithm's allocation, the optimal allocation gains indifferent amount (events in Ω_1^F) and the events where the optimal allocation loses indifferent amount (events in Ω_2^F). We show in Lemma 3.65 that these flips are limited in the sense that without considering flip events, the events in Ω_2^F are not sufficient as compensation to maintain envy-freeness for the events in Ω_1^F. Before we prove these results, we provide a technical lemma and two observations on the events in the flip sets to aid the proof.

We start with considering the size of the events involved in the flip sets. Since we are maximising the indifferent amount there is no benefit for allocating an event that is smaller than the second agent's saturation amount q_2 to agent 1 when it was given to agent 2 in the algorithm's allocation. Hence, we can observe that events which can increase the optimal allocations value over the algorithm's allocation have to be bigger than the second agent's saturation amount q_2.

Observation 3.62. *For all events $\omega \in \Omega_1^F$ it holds that they are bigger than the second agent's saturation amount, i.e. $\omega > q_2$.*

Additionally, events cannot be too big if they have an impact on the flips. If events are bigger than both saturation amounts, their allocation cannot be changed without reducing the overall allocated amount. Hence, the allocation in these events is always the same.

Observation 3.63. *All events in the flip sets $\omega \in \Omega_1^F \cup \Omega_2^F$ have to be smaller than the sum of saturation amounts, i.e. $\omega < q_1 + q_2$.*

Lastly, before the proof of Lemma 3.65, we provide a technical lemma that gives us the option to combine a set of events into a pseudo event which represents the sum of the events in the set. This allows us to be more succinct in Lemma 3.65.

Lemma 3.64. *For a subset of events $\widehat{\Omega} \subset \Omega$ which satisfy $a_i(\omega) = \min\{\omega, q_i\}$ and $a_j(\omega) = \min\{\max\{\omega - q_i, 0\}, q_j\}$ with $i \in [2]$ and $j \in [2]\backslash\{i\}$ for all $\omega \in \widehat{\Omega}$, there is a pseudo event $\widehat{\omega}$ with amount and probability*

$$\widehat{\omega} = \begin{cases} \frac{\sum_{\omega \in \widehat{\Omega}} \omega f(\omega)}{\sum_{\omega \in \widehat{\Omega}} f(\omega)} & \text{if } \omega \leq q_i \ \forall \omega \in \widehat{\Omega} \\ q_i \left(\frac{\sum_{\omega \in \widehat{\Omega}} \min\{\max\{\omega - q_i, 0\}, q_j\} f(\omega)}{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)} + 1 \right) & \text{else} \end{cases}$$

and

$$f(\widehat{\omega}) = \begin{cases} \sum_{\omega \in \widehat{\Omega}} f(\omega) & \text{if } \omega \leq q_i \ \forall \omega \in \widehat{\Omega} \\ \frac{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)}{q_i} & \text{else} \end{cases},$$

respectively, that can represent the events of the subset. The allocation to agent i and agent j in the subset of events is equivalent to the allocation

$$a_i(\widehat{\omega}) = \begin{cases} \widehat{\omega} & \text{if } \omega \leq q_i \ \forall \omega \in \widehat{\Omega} \\ q_i & \text{else} \end{cases} \quad \text{and} \quad a_j(\widehat{\omega}) = \begin{cases} 0 & \text{if } \omega \leq q_i \ \forall \omega \in \widehat{\Omega} \\ \widehat{\omega} - q_i & \text{else} \end{cases},$$

in the pseudo event. respectively.

Proof. We consider the two cases of $\omega \leq q_i$ for all $\omega \in \widehat{\Omega}$ and $\omega > q_i$ for at least one $\omega \in \widehat{\Omega}$ individually. The first case is that all event are smaller than the saturation amount of agent i, i.e. $\omega \leq q_i$ for all $\omega \in \widehat{\Omega}$.

It is straightforward to see that agent i has the same expected allocation in the events in the subset and in the pseudo event:

$$a_i(\widehat{\omega}) = \widehat{\omega} f(\widehat{\omega}) = \frac{\sum_{\omega \in \widehat{\Omega}} \omega f(\omega)}{\sum_{\omega \in \widehat{\Omega}} f(\omega)} \cdot \sum_{\omega \in \widehat{\Omega}} f(\omega) = \sum_{\omega \in \widehat{\Omega}} \omega f(\omega).$$

The other agent has no allocation which means there is nothing to show.

The other case is that at least one $\omega \in \widehat{\Omega}$ is bigger than agent i's saturation amount, i.e. $\omega > q_i$. In this case it is easy to see that $\widehat{\omega}$ is greater than q_i. Hence, we show that the expected amount and the expected allocated amount of the pseudo events is the same as the expected amount and expected allocated amount of the sum of events in the subset of events, respectively. Multiplying the pseudo event's amount with the pseudo event's probability yields the equivalence to the expected amount of all events.

$$\begin{aligned}
\widehat{\omega} f(\widehat{\omega}) &= \frac{q_i \sum_{\omega \in \widehat{\Omega}} \min\{\max\{\omega - q_i, 0\}, q_j\} f(\omega) + q_i \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)}{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)} \frac{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)}{q_i} \\
&= \sum_{\omega \in \widehat{\Omega}} \min\{\max\{\omega - q_i, 0\}, q_j\} f(\omega) + \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega) \\
&= \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i + q_j\} f(\omega).
\end{aligned}$$

Showing the equivalence in allocation we start with agent j.

$$\begin{aligned}
& a_j(\widehat{\omega}) f(\widehat{\omega}) \\
&= (\widehat{\omega} - q_i) f(\widehat{\omega}) \\
&= \left(\frac{q_i \sum_{\omega \in \widehat{\Omega}} \min\{\max\{\omega - q_i, 0\}, q_j\} f(\omega) + q_i \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)}{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)} - q_i \right) \frac{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)}{q_i} \\
&= \sum_{\omega \in \widehat{\Omega}} \min\{\max\{\omega - q_i, 0\}, q_j\} f(\omega) + \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega) - \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega) \\
&= \sum_{\omega \in \widehat{\Omega}} \min\{\max\{\omega - q_i, 0\}, q_j\} f(\omega)
\end{aligned}$$

Similarly, the equivalence in allocation holds for agent i.

$$a_i(\widehat{\omega}) f(\widehat{\omega}) = q_i f(\widehat{\omega}) = q_i \frac{\sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)}{q_i} = \sum_{\omega \in \widehat{\Omega}} \min\{\omega, q_i\} f(\omega)$$

Hence, for both cases the expected amount and allocation is the same. □

Finally, we prove the limitation on the extent of exchanges via the flip sets which lessens the quantity of possible changes between the algorithm's allocation and the optimal allocation.

Lemma 3.65. *In order to increase ex-ante social welfare while maintaining ex-ante envy-freeness the flip sets cannot simply be subsets of* Ω_1 *and* Ω_2*.*

Proof. We prove this by contradiction and therefore assume the opposite which means that there are flip sets $\Omega_1^F \subset \Omega_1$ and $\Omega_2^F \subset \Omega_2$ which can be flipped while maintaining envy-freeness and increasing social welfare. For succinctness, we use Lemma 3.64 to transform this setting into flipping two pseudo events. We refer to the pseudo events and their probabilities as ω_1 with $f_1 = f(\omega_1)$ for the events in Ω_1^F and ω_2 with $f_2 = f(\omega_2)$ for the events in Ω_2^F.

We show the result by considering the just envy-freeness of agent 2 after the flip and the change in the indifferent amount and show that the combination of the two cannot be greater than zero. We start with considering just envy-freeness of agent 2 before and after a flip. The allocation of the algorithm is just envy-free (see Algorithm 1 and Section 3.5.3) and by Lemma 3.34 the optimal allocation is just envy-free. Moreover, by Observation 3.62 all events are bigger than the second agent's saturation amount q_2 and by Observation 3.63 smaller than the sum of saturation amounts $q_1 + q_2$. Hence, the changes in the flip events have to maintain the just envy-freeness which means the following equation has to hold.

$$\begin{aligned}
& -q_2 f_1 + \max\{\omega_1 - q_1, 0\} f_1 + (q_2 - \max\{\omega_2 - q_1, 0\}) f_2 \\
& \quad = -q_2 f_2 + \min\{\omega_2 - q_2, q_2\} f_2 + (q_2 - \min\{\omega_1 - q_2, q_2\}) f_1
\end{aligned}$$

In detail, the changes are as follows. Flipping ω_1 means agent 2 might retain some allocation if ω_1 is big enough in comparison to agent 1's saturation amount q_1 but otherwise loses its allocation, i.e. agent 2 looses $\max\{\omega_1 - q_1, 0\} - q_2$. In comparison, agent 1 from the perspective of agent 2 gains the difference of the maximal perceivable amount (saturation amount q_2) and what agent 1 had allocated before, i.e. $q_2 - \min\{\omega_1 - q_2, q_2\}$. On the other hand, flipping ω_2 means agent 2 gains allocation minus their allocation before if the event was big in comparison to agent 2's allocation, i.e. $q_2 - \max\{\omega_2 - q_1, 0\}$. In comparison, agent 1 loses the amount allocated to agent 2 but retains as much as possible from the rest, i.e. $\min\{\omega_2 - q_2, q_2\} - q_2$.

We can rearrange this equation to get a dependency between the probabilities of the two pseudo events f_1 and f_2.

$$\begin{aligned}
& -q_2 f_1 + \max\{\omega_1 - q_1, 0\} f_1 + (q_2 - \max\{\omega_2 - q_1, 0\}) f_2 \\
& \quad = -q_2 f_2 + \min\{\omega_2 - q_2, q_2\} f_2 + (q_2 - \min\{\omega_1 - q_2, q_2\}) f_1 \\
\Leftrightarrow\ & (2q_2 - \min\{\omega_2 - q_2, q_2\} - \max\{\omega_2 - q_1, 0\}) f_2 \\
& \quad = (2q_2 - \min\{\omega_1 - q_2, q_2\} - \max\{\omega_1 - q_1, 0\}) f_1 \\
\Leftrightarrow\ & f_1 = \frac{2q_2 - \min\{\omega_2 - q_2, q_2\} - \max\{\omega_2 - q_1, 0\}}{2q_2 - \min\{\omega_1 - q_2, q_2\} - \max\{\omega_1 - q_1, 0\}} f_2
\end{aligned} \tag{3.10}$$

The denominator has to be bigger than zero since the other two terms cannot add up to $2q_2$. The term $\min\{\omega_2 - q_2, q_2\}$ is at most q_2. Moreover, if $\max\{\omega_1 - q_1, 0\}$ would be bigger than q_2 that would mean the first event ω_1 has an amount of at least the sum of saturation amounts. Hence, the second event ω_2 would have to be bigger than the sum of saturation amount. However, according to Observation 3.63 that would not work since such events cannot be flipped.

Following our observation on the probabilities from the envy-freeness, we consider how much the indifferent amount would change from the flips. Flipping an event in Ω_2^F means that agent 1 loses $(\min\{\omega_2, q_1\} - q_2) f_2$ indifferent amount and retains $\max\{\omega_2 - 2q_2, 0\} f_2$ in the case that event ω_2 is big, i.e. $\omega_2 > 2q_2$. In comparison, flipping an event in Ω_1^F means that agent 1 gains $(\min\{\omega_1, q_1\} - q_2) f_1$ indifferent amount but might have had already $\max\{\omega_1 - 2q_2, 0\} f_1$ in the case that event ω_1 is big, i.e. $\omega_1 > 2q_2$. Hence, the change in indifferent amount is $-(\min\{\omega_2, q_1\} - q_2) f_2 + \max\{\omega_2 - 2q_2, 0\} f_2 + \min\{\omega_1 - q_2, q_2\} f_1 - \max\{\omega_1 - 2q_2, 0\} f_1$.

Applying Equation (3.10) and rearranging the equation yields the following equation for the change in indifferent amount.

$$\begin{aligned}
&-(\min\{\omega_2,q_1\}-q_2)f_2+\max\{\omega_2-2q_2,0\}f_2\\
&\qquad+(\min\{\omega_1,q_1\}-q_2)f_1-\max\{\omega_1-2q_2,0\}f_1\\
&=f_2(-\min\{\omega_2,q_1\}+q_2+\max\{\omega_2-2q_2,0\})\\
&\quad+(\min\{\omega_1,q_1\}-q_2-\max\{\omega_1-2q_2,0\})\frac{2q_2-\min\{\omega_2-q_2,q_2\}-\max\{\omega_2-q_1,0\}}{2q_2-\min\{\omega_1-q_2,q_2\}-\max\{\omega_1-q_1,0\}}f_2\\
&=f_2(((-\min\{\omega_2,q_1\}+q_2+\max\{\omega_2-2q_2,0\})(2q_2-\min\{\omega_1-q_2,q_2\}-\max\{\omega_1-q_1,0\})\\
&\quad+(\min\{\omega_1,q_1\}-q_2-\max\{\omega_1-2q_2,0\})(2q_2-\min\{\omega_2-q_2,q_2\}-\max\{\omega_2-q_1,0\}))\\
&\quad\cdot(2q_2-\min\{\omega_1-q_2,q_2\}-\max\{\omega_1-q_1,0\})^{-1})
\end{aligned}\tag{3.11}$$

We use Equation (3.11) to show that no improvement is possible by showing that the equation is at most zero. The value depends on the sizes of the events and saturation amounts expressed in the minimum and maximum functions. Which means the value of the equation is determined by the following four inequalities:

(a) $\omega_1 - q_2 < q_2$

(b) $\omega_2 - q_2 < q_2$

(c) $\omega_2 - q_1 < 0$

(d) $\omega_1 - q_1 < 0$

Every inequality can be either true or false resulting in one or the other value of a minimum or maximum function. In total there are 16 combinations based on if (a) - (d) are true or false. However, not all combinations are possible. We observe 4 implications among those inequalities from the fact that the second event is bigger than the first ($\omega_2 > \omega_1$) which allow us to determine which are the possible combinations.

- Not (a) implies the strict version of not (b): $\omega_2 - q_2 > \omega_1 - q_2 \geq q_2$
- (b) implies (a): $\omega_1 - q_2 < \omega_2 - q_2 < q_2$
- (c) implies (d): $\omega_1 - q_1 < \omega_2 - q_1 < 0$
- Not (d) implies the strict version of not (c): $\omega_2 - q_1 > \omega_1 - q_1 \geq 0$

Considering all 16 combinations with these four implications gives us the following 9 combinations where $(\cdot)$ means the inequality is true and $\neg(\cdot)$ means the inequality is false.

1. $(a)(b)(c)(d)$
2. $(a)(b)\neg(c)(d)$
3. $(a)(b)\neg(c)\neg(d)$
4. $(a)\neg(b)(c)(d)$
5. $(a)\neg(b)\neg(c)\neg(d)$
6. $(a)\neg(b)\neg(c)(d)$
7. $\neg(a)\neg(b)(c)(d)$
8. $\neg(a)\neg(b)\neg(c)\neg(d)$

9. $\neg(a)\neg(b)\neg(c)(d)$

We consider the nine cases one after the other. Most of the cases only require to determine the value of the minimum and maximum functions and rearrange Equation (3.11). We focus on the following term of the equation which determines if the equation is negative, positive or zero.

$$\begin{aligned}G &:= (-\min\{\omega_2, q_1\} + q_2 + \max\{\omega_2 - 2q_2, 0\})(2q_2 - \min\{\omega_1 - q_2, q_2\} - \max\{\omega_1 - q_1, 0\})\\ &\quad + (\min\{\omega_1, q_1\} - q_2 - \max\{\omega_1 - 2q_2, 0\})(2q_2 - \min\{\omega_2 - q_2, q_2\} - \max\{\omega_2 - q_1, 0\})\end{aligned}$$

Case 1.

The first case is $(a)(b)(c)(d)$ which means the inequalities are as follows: $\omega_1 - q_2 < q_2$, $\omega_2 - q_2 < q_2$, $\omega_2 - q_1 < 0$ and $\omega_1 - q_1 < 0$. Applying this to G yields:

$$\begin{aligned}G &= (-\omega_2 + q_2 + 0)(2q_2 - \omega_1 + q_2 - 0) + (\omega_1 - q_2 - 0)(2q_2 - \omega_2 + q_2 - 0)\\ &= (-\omega_2 + q_2)(3q_2 - \omega_1) + (\omega_1 - q_2)(3q_2 - \omega_2)\\ &= -3\omega_2 q_2 + \omega_1\omega_2 + 3q_2^2 - \omega_1 q_2 + 3\omega_1 q_2 - \omega_1\omega_2 - 3q_2^2 + \omega_2 q_2\\ &= -3\omega_2 q_2 - \omega_1 q_2 + 3\omega_1 q_2 + \omega_2 q_2\\ &= 2q_2(-\omega_2 + \omega_1).\end{aligned}$$

The last expression clearly indicates that G has to be negative since ω_2 is bigger than ω_1.

Case 2.

The second case is $(a)(b)\neg(c)(d)$ which means the inequalities are as follows: $\omega_1 - q_2 < q_2$, $\omega_2 - q_2 < q_2$, $\omega_2 - q_1 \geq 0$ and $\omega_1 - q_1 < 0$. Applying this to G yields:

$$\begin{aligned}G &= (-_1 + q_2 + 0)(2q_2 - \omega_1 + q_2 - 0) + (\omega_1 - q_2 + 0)(2q_2 - \omega_2 + q_2 - \omega_2 + q_1)\\ &= (-q_1 + q_2)(3q_2 - \omega_1) + (\omega_1 - q_2)(3q_2 + q_1 - 2\omega_2)\\ &= -3q_1 q_2 + \omega_1 q_1 + 3q_2^2 - \omega_1 q_2\\ &\qquad + 3\omega_1 q_2 + \omega_1 q_1 - 2\omega_1\omega_2 - 3q_2^2 - q_1 q_2 + 2\omega_2 q_2\\ &= -4q_1 q_2 + 2\omega_1 q_1 + 2\omega_1 q_2 - 2\omega_1\omega_2 + 2\omega_2 q_2\\ &= 2(q_2(\omega_1 + \omega_2 - 2q_1) + \omega_1(q_1 - \omega_2))\end{aligned}$$

Since (d) implies that ω_1 is smaller than q_1 we can upper bound this further.

$$\begin{aligned}G &< 2(q_2(q_1 + \omega_2 - 2q_1) + \omega_1(q_1 - \omega_2))\\ &= 2((\omega_2 - q_1)(q_2 - \omega_1))\end{aligned}$$

This has to be negative since (c) is false which means ω_2 has to be at least of size q_1 and ω_1 is bigger than q_2 (see Observation 3.62).

Case 3.

The third case is $(a)(b)\neg(c)\neg(d)$ which means the inequalities are as follows: $\omega_1 - q_2 < q_2$, $\omega_2 - q_2 < q_2$, $\omega_2 - q_1 > 0$ and $\omega_1 - q_1 \geq 0$. Applying this to G yields:

$$
\begin{aligned}
G &= (-q_1 + q_2 + 0)(2q_2 - \omega_1 + q_2 - \omega_1 + q_1) + (q_1 - q_2 - 0)(2q_2 - \omega_2 + q_2 - \omega_2 + q_1) \\
&= (-q_1 + q_2)(3q_2 + q_1 - 2\omega_1) + (q_1 - q_2)(3q_2 + q_1 - 2\omega_2) \\
&= -3q_1q_2 - q_1^2 + 2\omega_1q_1 + 3q_2^2 + q_1q_2 - 2\omega_1q_2 \\
&\quad + 3q_1q_2 + q_1^2 - 2\omega_2q_1 - 3q_2^2 - q_1q_2 + 2\omega_2q_2 \\
&= 2(q_1(\omega_1 - \omega_2) + q_2(\omega_2 - \omega_1)) \\
&= 2((\omega_1 - \omega_2)(q_1 - q_2))
\end{aligned}
$$

The expression has to be negative since one of the factors is negative and the other is positive. Since ω_1 is less than ω_2 the left factor is negative, i.e. $\omega_1 - \omega_2 < 0$, and since q_1 is bigger than q_2 the second factor has to be positive, i.e. $q_1 - q_2 > 0$.

Case 4.

The fourth case is $(a)\neg(b)(c)(d)$ which means the inequalities are as follows: $\omega_1 - q_2 < q_2$, $\omega_2 - q_2 \geq q_2$, $\omega_2 - q_1 < 0$ and $\omega_1 - q_1 < 0$. Applying this to G yields:

$$
\begin{aligned}
G &= (-\omega_2 + q_2 + \omega_2 - 2q_2)(2q_2 - \omega_1 + q_2 - 0) + (\omega_1 - q_2 + 0)(2q_2 - q_2) \\
&= -q_2(3q_2 - \omega_1) + (\omega_1 - q_2)q_2 \\
&= q_2(2\omega_1 - 4q_2) \\
&= 2q_2(\omega_1 - 2q_2)
\end{aligned}
$$

The expression has to be negative since by inequality (a) we have that ω_1 is less than $2q_2$.

Case 5.

The fifth case is $(a)\neg(b)\neg(c)\neg(d)$ which means the inequalities are as follows: $\omega_1 - q_2 < q_2$, $\omega_2 - q_2 > q_2$, $\omega_2 - q_1 > 0$ and $\omega_1 - q_1 \geq 0$. Applying this to G yields:

$$
\begin{aligned}
G &= (-q_1 + q_2 + \omega_2 - 2q_2)(2q_2 - \omega_1 + q_2 - \omega_1 + q_1) + (q_1 - q_2 - 0)(2q_2 - q_2 - \omega_2 + q_1) \\
&= (\omega_2 - q_1 - q_2)(3q_2 - 2\omega_1 + q_1) + (q_1 - q_2)(q_1 + q_2 - \omega_2) \\
&= (q_1 + q_2 - \omega_2)(q_1 - q_2 - 3q_2 + 2\omega_1 - q_1) \\
&= 2(q_1 + q_2 - \omega_2)(\omega_1 - 2q_2)
\end{aligned}
$$

The expression has to be negative since one of the factors is negative and the other is positive. The left factor is positive since ω_2 is less than the sum of saturation amounts, i.e. $q_1 + q_2 - \omega_2 > 0$. In comparison, the second factor has to be negative since by inequality (a) we have that ω_1 is less than twice the second agent's saturation amount, i.e. $\omega_1 - 2q_2 < 0$.

Case 6.

The sixth case is $(a)\neg(b)\neg(c)(d)$ which means the inequalities are as follows: $\omega_1 - q_2 < q_2$, $\omega_2 - q_2 \geq q_2$, $\omega_2 - q_1 \geq 0$ and $\omega_1 - q_1 < 0$. Applying this to G yields:

$$
\begin{aligned}
G &= (-q_1 + q_2 + \omega_2 - 2q_2)(2q_2 - \omega_1 + q_2) + (\omega_1 - q_2 - 0)(2q_2 - q_2 - \omega_2 + q_1) \\
&= (\omega_2 - q_1 - q_2)(3q_2 - \omega_1) + (\omega_1 - q_2)(q_1 + q_2 - \omega_2) \\
&= (q_1 + q_2 - \omega_2)(-3q_2 + \omega_1 + \omega_1 - q_2) \\
&= 2(q_1 + q_2 - \omega_2)(\omega_1 - 2q_2)
\end{aligned}
$$

This expression is the same as the one of case 5 and the same relevant inequalities hold which means this expression is negative as well.

Case 7.

The seventh case is $\neg(a)\neg(b)(c)(d)$ which means the inequalities are as follows: $\omega_1 - q_2 \geq q_2$, $\omega_2 - q_2 \geq q_2$, $\omega_2 - q_1 < 0$ and $\omega_1 - q_1 < 0$. Applying this to G yields:

$$
\begin{aligned}
G &= (-\omega_1 + q_2 + \omega_2 - 2q_2)(2q_2 - q_2 - 0) + (\omega_1 - q_2 - \omega_1 + 2q_2)(2q_2 - q_2 - 0) \\
&= -q_2^2 + q_2^2
\end{aligned}
$$

Clearly, this case allows no change in the indifferent amount.

Case 8.

The eighth case is $\neg(a)\neg(b)\neg(c)\neg(d)$ which means the inequalities are as follows: $\omega_1 - q_2 \geq q_2$, $\omega_2 - q_2 \geq q_2$, $\omega_2 - q_1 > 0$ and $\omega_1 - q_1 \geq 0$. Applying this to G yields:

$$
\begin{aligned}
G &= (-q_1 + q_2 + \omega_2 - 2q_2)(2q_2 - q_2 - \omega_1 + q_1) + (q_1 - q_2 - \omega_1 + 2q_2)(2q_2 - q_2 - \omega_2 + q_1) \\
&= (\omega_2 - q_1 - q_2)(q_1 + q_2 - \omega_1) + (q_1 + q_2 - \omega_1)(q_1 + q_2 - \omega_2) \\
&= (\omega_2 - q_1 - q_2)(q_1 + q_2 - \omega_1 - q_1 - q_2 + \omega_2) \\
&= (\omega_2 - q_1 - q_2)(\omega_2 - \omega_1)
\end{aligned}
$$

The expression has to be negative since one of the factors is negative and the other is positive. The first term has to be negative since by Observation 3.63 the amount of ω_2 has to be less than the sum of saturation amounts. The second term has to be positive since ω_2 is bigger than ω_1.

Case 9.

The ninth case is $\neg(a)\neg(b)\neg(c)(d)$ which means the inequalities are as follows: $\omega_1 - q_2 > q_2$, $\omega_2 - q_2 \geq q_2$, $\omega_2 - q_1 \geq 0$ and $\omega_1 - q_1 < 0$. Applying this to G yields:

$$\begin{aligned} G &= (-q_1 + q_2 + \omega_2 - 2q_2)(2q_2 - q_2 - 0) + (\omega_1 - q_2 - \omega_1 + 2q_2)(2q_2 - q_2 - \omega_2 + q_1) \\ &= (\omega_2 - q_1 - q_2)q_2 + q_2(q_1 + q_2 - \omega_2) \\ &= q_2(\omega_2 - \omega_2 - q_1 + q_1 - q_2 + q_2) \end{aligned}$$

Like case 7, this case allows no change in the indifferent amount.

Altogether, in seven cases G is negative and in two cases G is zero. This clearly shows that flipping just events in Ω_1 and Ω_2 at best does not affect the indifferent amount and at worst decreases it, which contradicts our assumption. □

Lemma 3.65 indicates that the difference in allocation hinges on the allocation in the split event which is determined by the structure of the instance. Based on this we conjecture that the approximation ratio of GreedyTake-AMT is generally better than that of GreedyTake-EXP.

3.5.6 An Optimal Algorithm

Following on from the theoretical considerations of the algorithm variants in the previous section, we first devise the integer programming problem that we use to calculate the optimal allocation in order to compare the two algorithm variants empirically in Section 3.5.7. In general, it is straightforward to represent the problem in this work as an optimisation problem. However, depending on the utility functions, the problem may be non-linear and non-concave. Nevertheless, in the setting of linear satiable utility functions we can rewrite the expected utility functions and envy-freeness constraint with minima functions. In turn, these can be transformed into a series of constraints to reformulate the optimisation problem into an integer programming problem to calculate the optimal envy-free solution.

In more detail, the initial optimisation problem with decision variables x_{ij}, indicating the allocation of agent $i \in [n]$ in event $j \in [m]$, can be formulated as shown in Optimisation Problem 2.

Optimisation Problem 2

$$\max \quad \sum_{i\in[n]} \frac{u_i}{q_i} \cdot \sum_{j\in[m]} Pr(\omega_j) \cdot \min\{x_{ij}, q_i\} \tag{3.12}$$

$$\text{s.t.} \quad \sum_{i\in[n]} x_{ij} \leq \omega_j \qquad \forall j \in [m] \tag{3.13}$$

$$\sum_{j\in[m]} Pr(\omega_j)\,(\min\{x_{ij}, q_i\} - \min\{x_{kj}, q_i\})\,f(\omega) \geq 0 \qquad \forall i \in [n], k \in [n] \tag{3.14}$$

$$x_{ij} \geq 0 \qquad \forall i \in [n], j \in [m] \tag{3.15}$$

Here, the optimisation function (Equation (3.12)) is the social welfare rewritten using Observation 3.24. Constraint (3.13) limits the allocation x_{ij} for agent $i \in [n]$ in event $j \in [m]$ to the available amount. Constraint (3.14) is the envy-freeness constraint rewritten using Observation 3.25. Finally, Constraint (3.15) ensures that only positive allocations are attained.

In this formulation, neither the optimisation function nor the envy-freeness constraint appear linear. However, in the following series of replacements we replace those with linear constraints and integer variables.

Firstly, by Observation 3.28, allocating more to an agent than their saturation amount does not increase the value. Furthermore, it can only negatively affect envy-freeness since the agents' utilities do not increase but another agent might be envious of the increased amount. Consequently, we can replace one of the minimum functions in the optimisation problem.

Lemma 3.66. *The expression* $\min\{x_{ij}, q_i\}$ *can be replaced with* x_{ij} *and constraint* $x_{ij} \leq q_i$ *for all* $i \in [n]$ *and* $j \in [m]$.

The other minimum in the envy-freeness equation, i.e. $\min\{x_{kj}, q_i\}$, cannot be replaced that easily. Hence, we apply linearisation techniques and replace the minimum function with three more types of variables and a number of constraints. An overview of linearisation techniques can be found in the work by Liberti [53].

In more detail, firstly, we substitute the minimum function $\min\{x_{kj}, q_i\}$ with a variable x^i_{kj}. Secondly, we use a second integer variable y^i_{kj} to ensure that the substitution is valid.

Lemma 3.67. *The equation* $x^i_{kj} = \min\{x_{kj}, q_i\}$ *for all* $i, k \in [n], j \in [m]$ *holds for constraints* $q_i \cdot y^i_{kj} \leq x^i_{kj} \leq q_i$ *and* $x_{kj} \cdot (1 - y^i_{kj}) \leq x^i_{kj} \leq x_{kj}$ *with* $y^i_{kj} \in \{0, 1\}$.

The general idea behind the technique is as follows: for the equation to hold, the variable x^i_{kj} has to be smaller than both values in the minimum; yet at the same time it also has to be greater than one of the two, i.e. be exactly of that value. By setting y^i_{kj} to one or zero we can tighten one or the other constraint to be of the minimal value. Considering the four cases of the two possible values of the minimum and the two possible values of y^i_{kj}, one can see that y^i_{kj} can only be chosen so that the equation $x^i_{kj} = \min\{x_{kj}, q_i\}$ holds.

However, the constraint $x_{kj} \cdot (1 - y^i_{kj}) \leq x^i_{kj}$ still contains a non-linear expression in the form of a product. We replace that product $x_{kj} \cdot y^i_{kj}$ with a new variable z^i_{kj} which yields constraint $x^i_{kj} \geq x_{kj} - z^i_{kj}$ and further constraints.

Lemma 3.68. *The equation* $z^i_{kj} = x_{kj} \cdot y^i_{kj}$ *for all* $i, k \in [n], j \in [m]$ *holds for constraints* $0 \leq z^i_{kj} \leq x_{kj}$ *and* $x_{kj} + y^i_{kj} - 1 \leq z^i_{kj} \leq y^i_{kj}$.

The main observations for this linearisation are that $x_{kj} \in [0, 1]$ and that y^i_{kj} is binary. This implies that z^i_{kj} has to be smaller than both factors. Furthermore, the constraint $x_{kj} + y^i_{kj} - 1$ is either x_{kj} or non-positive and therefore assures that the result holds.

Finally, replacing the minimum functions with the variables and constraints from the Lemmata 3.66, 3.67 and 3.68 yields an integer linear programming problem.

We can verify this by observing that all of the resulting constraints are linear and we have continuous variables x_{kj}, x^i_{kj} and z^i_{kj} as well as the binary variables y^i_{kj}. The full linear integer problem can easily be obtained by applying the linearisations to Optimisation Problem 2. For completeness, the linear integer problem version can be found in the appendix (see Section A.4)

3.5.7 Experimental Comparison of GreedyTake Variants

Theorem 3.58 showed that, counter-intuitively, we have no social welfare guarantee in general when sorting by the expected amount, whereas Lemmata 3.60 and 3.65 show that sorting by the amount produces allocations that are structurally similar to the optimal allocation. That raises the questions whether the instance of Theorem 3.58 is a degenerate special case and if the amount ordering is actually better than the expected amount ordering in general. We addressed these questions empirically with a number of different experiments with different amounts and probabilities of the events. These experiments show that, in most cases, GreedyTake-Amt achieves better results. However, GreedyTake-Exp performs almost as good and, as we show below, in some cases even better.

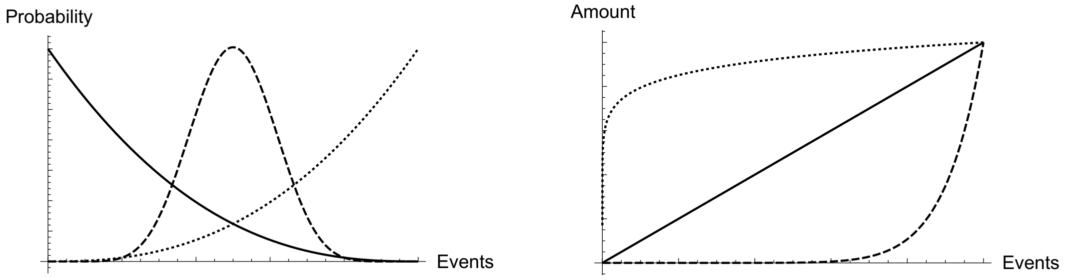
3.5.7.1 Setting

For every experiment we generate the events, the saturation amounts and the utilities. In brief, we set the events' amounts and probabilities as well as the saturation amounts systematically and generate the utilities randomly. We then repeat each setting 50 times to achieve statistical significance. Considering these parts in turn we start with the events generation. Instead of generating events randomly, we generate amounts and probabilities individually in order to systematically cover different cases. For every instance we choose a combination of an amount generation function and a probability mass function (PMF) to generate the amount and probability of all events. More precisely, for an event $i \in [1, m]$, where m is the number of events, we use the value of the amount and probability function at i for the amount and probability, respectively. For generating the amounts we use three different types of functions: linear, root, and one-term polynomial. For the probability, we use the PMF of the Beta-binomial distribution. An overview of the used function can be found in Table 3.1.

In more detail, for the amount generation we considered two different approaches: one where all events are almost equal in size and one where the amount increases from very small to maximal amount. The former case is represented by an almost horizontal function where $\omega_i = 0.001 \cdot i + b$ with some $b \in [0, 1)$. We refer to this as near horizontal. For the latter case we consider a linear mapping as well as the root and the exponentiation of the events' index (see Figure 3.5B). The *linear* function is a simple base case (see solid line Figure 3.5B). The *root* (see dotted line Figure 3.5B) represents the situation where we have a relatively small number of smaller events and more larger events. Finally, the exponentiation (see dashed line Figure 3.5B), denoted as *power*, represents the situation where we have a larger number of smaller events and a few bigger events. The power and the root function represent situations where we have a lot or only a few of events with shortfall, respectively.

For the probability generation, we use the PMF of the Beta-binomial distribution. This distribution gives us the option to adjust the centre of the probability to our needs. We use this to cover three important situations (see Figure 3.5A): a high probability on the events with smaller amounts (solid line), a high probability on the events with bigger amounts (dotted line) and a higher probability on the events in the middle (dashed line).

With regards to the saturation amounts and utility of the agents, for every combination of amount and probability, we set the saturation amount of the first agent to 0.9 and for the second agent we choose a saturation amount of 0.1, 0.3, 0.5 and 0.7. The maximal values of the agents are randomly chosen according to Assumption 3.29. Finally, we run experiments for 2 to 15 events. Overall this results in 1176 different instances which, as mentioned, are run 50 times giving us a total of 58800 experiments. We use 50 runs since statistical measures are already stable, as we have determined by performing larger runs of up to 200.



(A) Plot of the functions to generate the probabilities of the events: beta decrease (solid line), beta Gauß (dashed line) and beta increase (dotted line).

(B) Plot of the functions to generate the amount of the events: linear (solid line), power (dashed line) and root (dotted line).

FIGURE 3.5: Plots of the functions to generate the events shown in Table 3.1, except the near horizontal linear amount function.

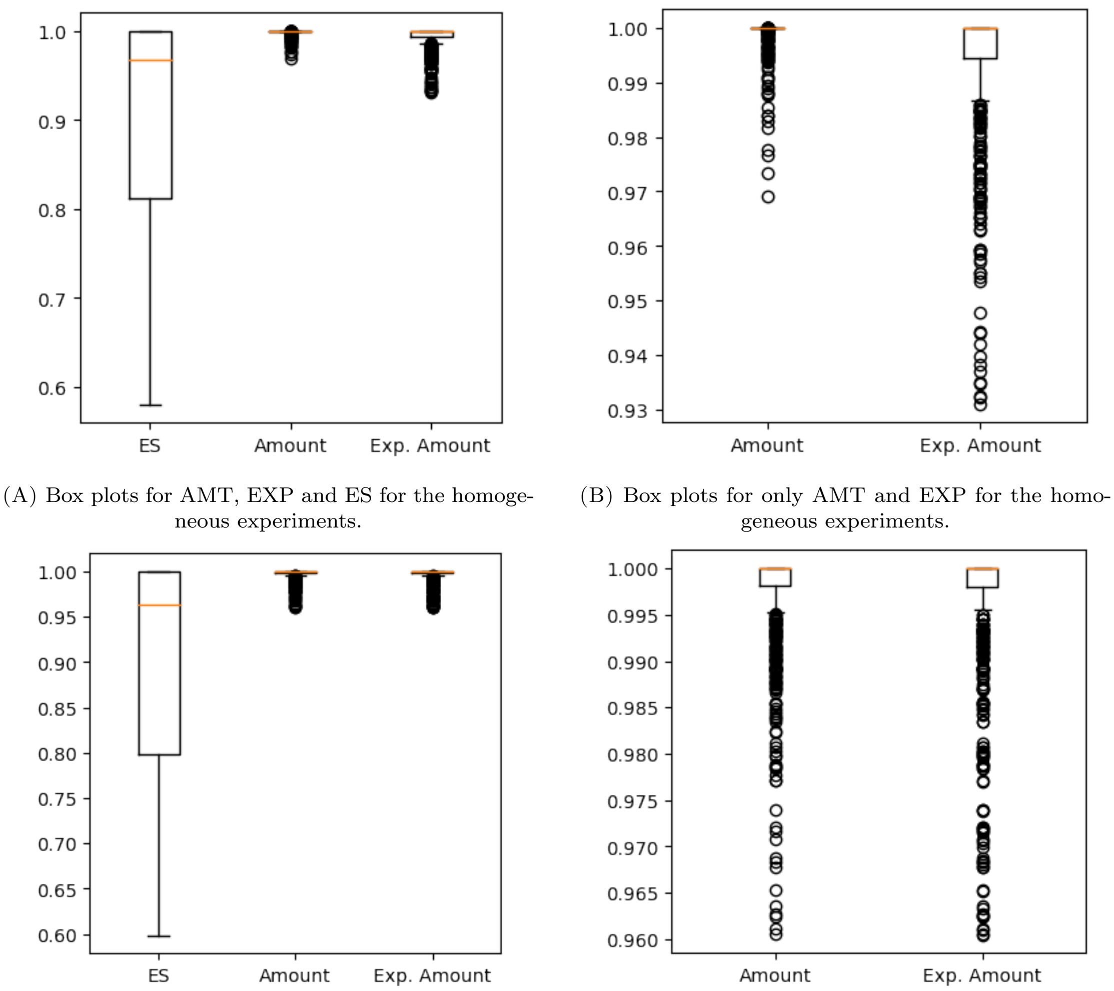
Events Part	Name	Function	Parameter values
Amount	Linear	$\omega_i = \frac{1}{m} \cdot i$	-
	Power	$\omega_i = (\frac{i}{m})^{10}$	-
	Root	$\omega_i = (\frac{i}{m})^{\frac{1}{10}}$	-
	Near Horizontal b	$\omega_i = 0.001 \cdot i + b$	-
Probability	Beta decrease	$Pr(\omega_i) = \binom{m}{i}\frac{B(i+\alpha, m-i+\beta)}{B(\alpha,\beta)}$	$\alpha = 1, \beta = 3.5$
	Beta Gauß		$\alpha = 10, \beta = 10$
	Beta increase		$\alpha = 3.5, \beta = 1$

TABLE 3.1: Overview of the functions used to generate the events' amount and probability.

3.5.7.2 Experimental Results

We measure the performance of the two variants and, for comparison, of equal share (ES) in relation to the maximal achievable expected social welfare, as determined by our linear integer problem (see Section 3.5.6). Hence, the social welfare is presented in relation to the optimal value which is set to 1. For brevity, we refer to GreedyTake-Amt with *AMT* (see Algorithm 2 in Section 3.5.4), to GreedyTake-Exp with *EXP* (see Algorithm 3 in Section 3.5.5.2) and to equal share with *ES* in this Section.

Overall, both variants of the algorithm perform well (see Table 3.2). The two variants, AMT and EXP, achieve an average social welfare of 0.998 (standard deviation 0.005) and 0.995 (standard deviation 0.011), respectively. This is in comparison to ES with a lower average of 0.889 and a much higher standard deviation of 0.147. Looking at individual experiments, the two algorithms are never worse than ES. Moreover, at least one of the two variants is better than ES in 85% of experiments. Additionally, the two variants achieve maximal social welfare frequently: EXP achieves maximal social welfare in 48.3% of experiments whereas AMT performs better and achieves this in 57% of experiments. In comparison, ES achieves maximal social welfare in only 8.9% of experiments.



(A) Box plots for AMT, EXP and ES for the homogeneous experiments.

(B) Box plots for only AMT and EXP for the homogeneous experiments.

(C) Box plots for AMT, EXP and ES for the diverse experiments.

(D) Box plots for only AMT and EXP for the diverse experiments.

FIGURE 3.6: Box plots for the performance of equal share (ES), GREEDYTAKE-AMT (AMT) and GREEDYTAKE-AMT (EXP) in the homogeneous and diverse experiments.

In order to highlight the performance difference between AMT and EXP, we have divided our experiments into two sets. The first set, labelled *diverse*, contains all instances where we use the linear, the root and the power function for the amount. The second set, labelled *homogeneous*, contains all instances where we use the near horizontal linear function. Considering the performance of the algorithms in these sets, more diversity means that the two variants and ES achieve maximal social welfare less often with AMT in 53.37%, EXP in 37.7% and ES in 4.17% of experiments. In contrast to the homogeneous case where they achieve maximal social welfare in 59.67% for AMT, 56.4% for EXP and 12.5% for ES of experiments. Otherwise, on average, the two variants are still close to maximal social welfare (see Table 3.2). Moreover, the variants perform similarly in comparison to ES (see Table 3.3). This includes having a much smaller dispersion than ES (Figures 3.6A and 3.6C).

Comparing the two algorithms to each other (see also Table 3.3), EXP performs better in the homogeneous set than in the diverse set. In the diverse set, AMT performs better than EXP in a third of the experiments and the opposite only holds for a small number of experiments (3.97%). Additionally, the dispersion is visibly bigger (see Figure 3.6B) However, the gap is smaller in the

	ES			AMT			EXP		
Set	min	Mean	SD	min	Mean	SD	min	Mean	SD
Overall	0.581	0.889	0.147	0.961	0.998	0.005	0.931	0.995	0.011
Diverse	0.581	0.897	0.138	0.969	0.999	0.003	0.969	0.994	0.013
Homogeneous	0.599	0.882	0.152	0.961	0.997	0.006	0.96	0.996	0.008

TABLE 3.2: Overview of the minimum (min), mean and standard deviation (SD) of percentage of social welfare of equal share (ES), GREEDYTAKE-AMT (AMT) and GREEDYTAKE-AMT (EXP). The social welfare is as a fraction of the maximal achievable ex-ante social welfare, i.e. $W(OPT) \equiv 1$.

homogeneous experiments where AMT is less often better than EXP (18.15%) and EXP is more often better (10.42%). Moreover, the dispersion is similar (see Figure 3.6D). Nevertheless, considering the maximal difference in performance over all experiments the social welfare difference is less than 0.05 (see maximal difference column in Table 3.3). This is substantially better than ES which is outperformed by AMT and EXP by up to 0.401 and 0.419 in the homogeneous and diverse set, respectively.

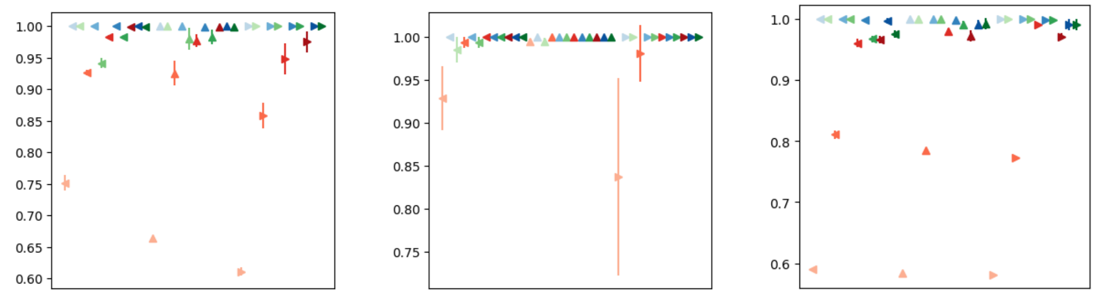
Experiment	A	B	$A > B$	$max A - B$
Diverse	AMT & EXP	ES	86%	0.419
	AMT	EXP	33.33%	0.069
	EXP	AMT	3.97%	0.026
Homogeneous	AMT & EXP	ES	84%	0.401
	AMT	EXP	18.15%	0.037
	EXP	AMT	10.42%	0.021

TABLE 3.3: Comparison of the performance of the two algorithm variants with equal share and with each other. The percentage indicates in what percentage of cases one variant (or the respective ordering) performs better than equal share or the other variant. $max A - B$ indicates the maximal difference in social welfare over all experiments of the category (experiment) and the respective comparison (A and B). (The social welfare is as a fraction of the maximal achievable ex-ante social welfare, i.e. $W(OPT) \equiv 1$.)

Finally, besides the separation into homogeneous and diverse experiments, there is no further clear indicator whereby EXP outperforms AMT. Figures 3.7 and 3.8 separate the experiments further by the function which was used for the amount and probability generation. These provide interesting insights about ES. Specifically, Figures 3.7A, 3.7C and 3.8C indicate that the bigger the difference between the first and the second agents saturation amount, the lower the social welfare of ES is. However, there is no significant difference between AMT and EXP considering these characteristics. These figures highlight that AMT and EXP perform very similarly.

3.6 Conclusion

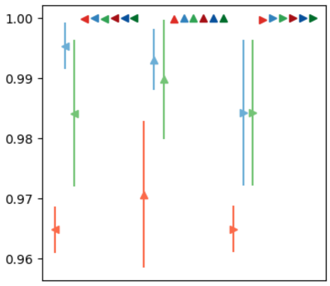
The presented results provide our contribution to the research area of fair division of homogeneous resources (Problem A). We consider a fair division variant where the amount of a homogeneous resource is uncertain which is reflected by a random variable over a finite set of discrete events. We establish this problem as a promising research area by presenting the results summarised in the research contributions (see Section1.3) which cover our research challenges (see Section 1.2). We highlight differences and similarities of ex-ante envy-freeness to ex-post



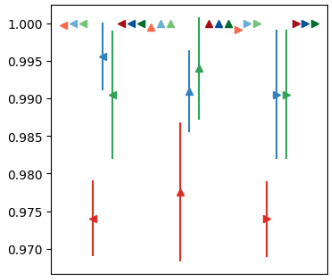
(A) Linear amount function. (B) Power amount function. (C) Root amount function.

FIGURE 3.7: Performance of ES (orange to red colours), AMT (blue colours) and EXP (green colours) for the set of diverse experiments. The hue of the colours indicate the second agents saturation amount from the lightest indicating 0.1 to the darkest indicting 0.7 (this especially means that there is no continuous and consecutive data on the x-axis). The arrows' heads direction indicates the probability distribution. Pointing to the left indicates beta decrease, pointing to the top indicates beta Gauß and pointing to the right indicates beta increase (see Table 3.1). Finally, the experiments are separated into three sets based on the amount functions: (A) shows the experiments for the linear function, (B) shows the experiments for the power function and (C) shows the experiments for the root function.

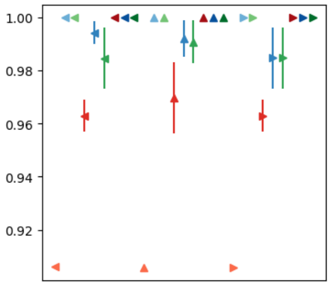
envy-freeness, especially that in the worst case the price of envy-freeness is tightly linear in the number of agents. While ex-ante envy-freeness offers opportunities for improvement, optimising efficiency under ex-ante envy-freeness (i.e. determining optimal allocations) is NP-hard for any number of agents. Considering this result and in pursuit of Research Challenge A.5, as well as to investigate the effects of uncertainty, we consider the settings of two agents. For uniform probabilities over the events the problem of finding optimal allocations is solvable in polynomial time. In contrast, for arbitrary probabilities over the events, the problem is also NP-hard. Nevertheless, we provide two variants of a polynomial greedy algorithm whose allocations outperform equal share allocations. The first version does not have a guaranteed theoretical approximation but is empirically close to the second version and outperforms it on a smaller number of instances. The second version is structurally close to a specific optimal allocation and performs empirically well.



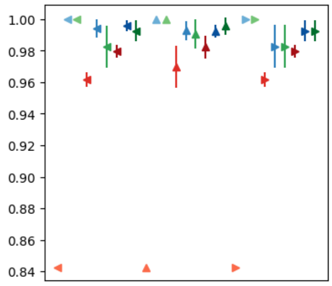
(A) Near horizontal amount function y-intercept $b = 0.5$.



(B) Near horizontal amount function y-intercept $b = 0.6$.



(C) Near horizontal amount function y-intercept $b = 0.7$.



(D) Near horizontal amount function y-intercept $b = 0.8$.

FIGURE 3.8: Performance of ES (orange to red colours), AMT (blue colours) and EXP (green colours) for the set of diverse experiments. The hue of the colours indicate the second agents saturation amount from the lightest indicating 0.1 to the darkest indicting 0.7 (this especially means that there is no continuous and consecutive data on the x-axis). The arrows' heads direction indicates the probability distribution. Pointing to the left indicates beta decrease, pointing to the top indicates beta Gauß and pointing to the right indicates beta increase (see Table 3.1). Finally, the experiments are separated into four sets based on the near horizontal amount function's y-intercept b: (A), (B), (C) and (D) show experiments for value 0.5, 0.6, 0.7 and 0.8, respectively.

Chapter 4

Single Robot Adversarial Patrolling Against a Full-Knowledge Adversary on a Perimeter

In this chapter we address Problem B (see Section 1.2), i.e. improving the runtime for full-knowledge multi-robot adversarial patrolling on polylines. We start by introducing the formal model in Section 4.1. This includes the graphs, the movement types and the random process of the robots' strategies. In total we have four settings from the combination of the two variants of polylines (open and closed) and two types of the robot's movements (determined by if the robot has to turn around to change direction or not). We use one of the settings to explain our approach (see Section 4.2). This includes the characteristics that influence the different paths the robots can walk and, the important novelty, how this can be translated into lattice path characteristics (see Section 4.2.5). Using the lattice path representation, we count the number of paths. This immediately allows us to explicitly state the probability of detecting the adversary in different vertices of the polyline. Following this, we consider the other three settings (see Section 4.3). We focus for each of the three settings on the difference to the first settings. Hence, we limit the presentation to statements that are additionally required for the respective setting or the differences in modelling of the respective case. Section 4.2.5 as well as Section 4.3 focus on the first step of determining the optimal probabilities and therefore conclude with the respective probability functions. In contrast, in the last section (see Section 4.4 we focus on the second step and the (calculation of the) optimal strategies. The presented results in this section are theoretical and empirical insights into the first setting. In the first part (see Section 4.4.1) we show how the system of equations can be reduced based on the reachability of different parts of the graph. In the second part (see Section 4.4.2) we present experimental results on the probabilities and the algorithm's runtime. Additionally, we present empirically patterns of the optimal strategies which also indicates further runtime reductions.

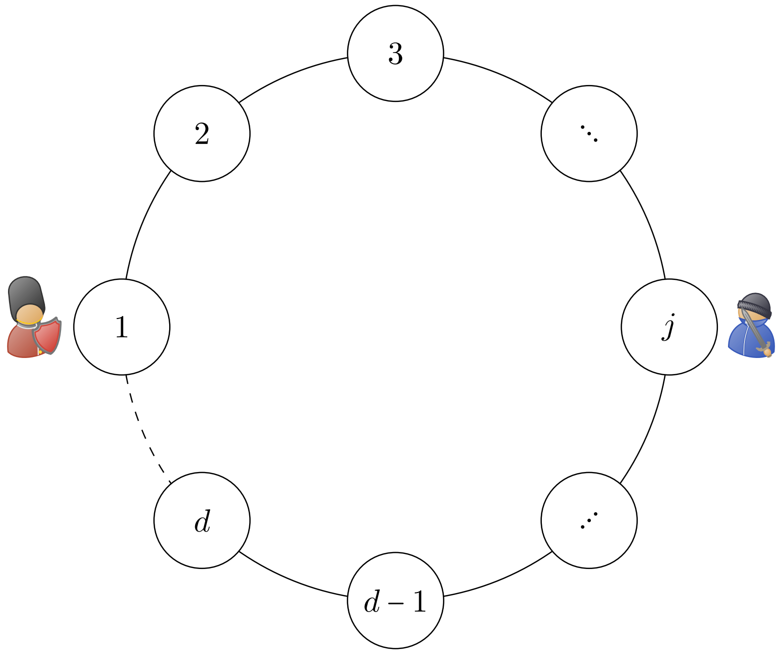


FIGURE 4.1: The polyline graph with d segments. The dashed line indicates the edge that is present in the circle (perimeter) but not in the line (fence), i.e. the perimeter is a cyclic setting whereas the fence is a setting with defined ends. The guard with the shield represents the robot positioned in segment 1 and the person in the jacket and the knit cap with the sword represents the adversary penetrating in segment j.

4.1 Preliminaries

We are considering multi-robot patrolling in a discrete environment assuming discrete time. In this settings k homogeneous robots shall be used to detect penetration attempts on a polyline with $d \in \mathbb{N}$ vertices, called segments, by a full-knowledge adversary. In this a penetration attempt is the presence of an adversary at a target segment $1 \leq j \leq d$ with the intent of passing the perimeter or fence. The adversary is detected if a robot reaches the targeted segment within the t time steps. The polyline can be open or closed to represent a perimeter or a fence, respectively (see Figure 4.1) [1].

1. **Perimeter / Circle**: A closed polyline.
2. **Fence / Line**: An open polyline.

The polyline restricts the robot's movement to two directions: clockwise and anticlockwise, or right and left. We differentiate two movement types based on if the robot can freely move into either adjacent segment at any moment, or if it needs to turn around first [1].

1. **Omnidirectional**: The omnidirectional movement allows a robot to freely walk into either adjacent segment in one time step. On the circle this is the next segment in either the clockwise or anticlockwise direction and on the line the segment to the left or the right of the current segment (see Figure 4.2).
2. **Directional**: The robot has a direction pointing along the axis of the polyline in on or the other direction. The robot can walk into the segment ahead of the current segment in

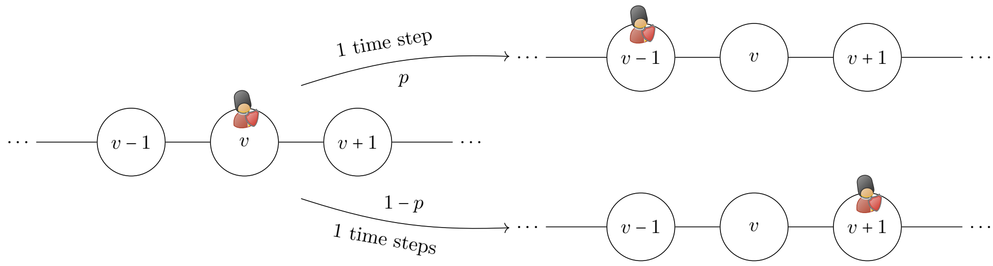


FIGURE 4.2: The movement of the robot (represented by the guard with shield) in the omni-directional case. If the robot is in a segment v then with probability p it moves into segment $v-1$ to the left in the next time step and with probability $1-p$ it moves into segment $v+1$ to the right in the next time step.

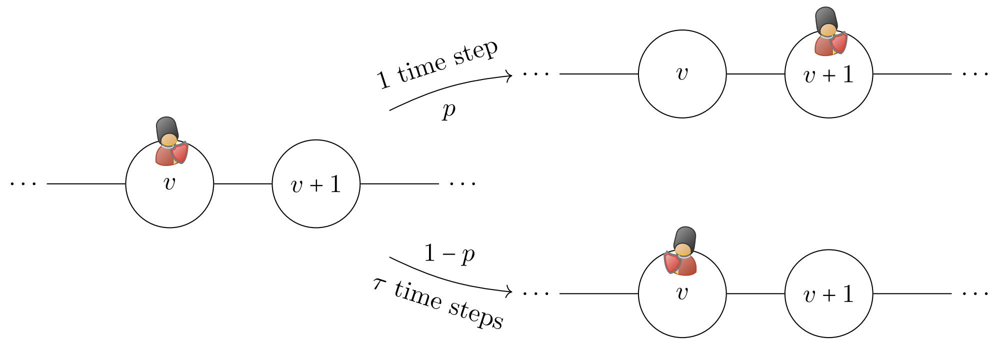


FIGURE 4.3: The movement of the robot (represented by the guard with shield) in the directional case. If the robot is in a segment v then with probability p it moves into segment $v+1$ to the right in the next time step and with probability $1-p$ it turns around and stays in segment v. Turning around takes τ time steps. The case where the robot faces left and would walk ahead into segment $v-1$ is analogues.

one time step but needs to turn around to change direction which takes $\tau \geq 1$ time steps (see Figure 4.3).

As stated in the introduction, since the adversary has full knowledge of the robots' strategy, the robot has to act randomly. Hence, in line with the fact that there are two movement possibilities, e.g. clockwise and anticlockwise, and since we are considering memoryless strategies, a random strategy's walk of a robot is a Bernoulli process. In every round the robot does, for both movement types, one of the two possibilities with probability p and the other action with probability $1-p$. We aim to determine the optimal value of p that maximises the probability of detecting the adversary. However, since the adversary has full knowledge they will choose the segment j with the lowest probability of the robot reaching it in time. Thus, to optimise p we need to maximise the probability of the segment with minimal probability of a robot detecting the adversary, i.e. we seek $p \in \arg\max_{p' \in [0,1]} \min_{j \in [d]} Pr(j', j, t, d)$ where $Pr(j'j, t, d)$ is the probability that a robot in segment j' detects the adversary in segment j within the t time steps on a graph of d segments, i.e. the robot reaches segment j from j' within the t time steps. This is also referred to as *minmax* approach [1].

The probability depends on the paths the robots can walk. In general, a path is a sequence of consecutive segments representing the robots clockwise, anticlockwise, left, right, forward or turn movements.

Definition 4.1 (Path, Right/Clockwise Step, Left/Anticlockwise Step)**.** A *path* of length l is a sequence of consecutive segments, i.e. a vector $(s_1, s_2, \dots, s_l) \in [d]^l$ where every pair of segments (s_i, s_{i+1}) for $1 \leq i < l$ are neighbours in the graph. For a path a *step to the right/clockwise step* or a *step to the left/anticlockwise step* is a tuple (s_{i-1}, s_i) where the second segment is right of/next to in clockwise direction or left of/next to in anticlockwise direction of the first segment, respectively.

Since the random walk is a Bernoulli process, describing the probability of one path is straightforward. The outcome set Ω consists of all paths from a start segment (the current location of the robot) with a length at most t and the probability measure P_W is the obvious Bernoulli process measure.

Definition 4.2 (Probability of a Path)**.** The *probability of a path* $\omega \in \Omega$ is $P_W(\omega) = p^Y \cdot (1-p)^X$ where X and Y are the number of actions depending on the movement type.

Detecting the adversary means that within the penetration time t there is a time step in which the robot is in the segment where the adversary penetrates. This means, if the adversary penetrates segment $j \in [d]$ and the robot walks a path that contains j, the robot will detect the adversary. Hence, we can define the probability measure of detecting the adversary $Pr(j', j, t, d)$ as the probability of reaching the adversary:

$$Pr(j'j, t, d) := P(\{\omega | \omega \in \Omega \wedge j \in \omega\}) = \sum_{\omega \in \Omega : j \in \omega} P_W(\omega).$$

4.2 Omnidirectional Movement in the Circle

We use the case of omnidirectional movement in the circle to introduce the general modelling. This encompasses the different types of paths, how the lattice path problem, based on the different types of paths, looks like and how this is combined to determine the probability of reaching the adversary. For the rest of this section we consider one robot and a segment j as the specified target. Without loss of generality the robot is placed in segment 1. This can be done since the segment numbering is arbitrary and we can always shift the segments for several robots according to their position.

The probability of detecting the adversary depends on the possible paths and the probability of one path $P_W = p^Y \cdot (1-p)^X$ where X is the number of clockwise steps and Y is the number of anticlockwise steps. We determine the valid paths and their number in five steps. Firstly, we limit the range of the penetration time t. This is necessary since for short penetration times the adversary can always penetrate successfully while for large t the robots can always detect the adversary. Secondly, the symmetry of the circle means that we only have to consider reaching a segment in one direction and the other directions follow similarly. Thirdly, by the definition of detection we only need to consider paths that end in j and are of maximal length t. Fourthly, we determine the freedom that leads to the different possible paths. Finally, the major part is modelling the different paths as lattice paths and counting them. Altogether, this gives us the probabilities of the robot reaching the different segments.

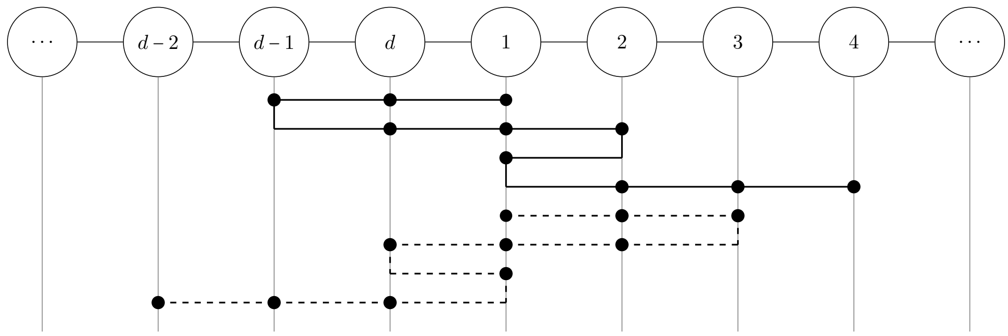


FIGURE 4.4: Example of a path and its mirrored path. The solid path starts in segment 1 and the target segment is segment 4. Transforming every step from this path into a step in the opposite direction results in the dashed path. This path starts in 1 and the target segment is segment $d - (4 - 2) = d - 2$ as stated by Lemma 4.6.

4.2.1 Penetration Time Range

Firstly, the problem makes sense only for a certain range of t. If the adversary requires a long penetration time, the robot can always detect the penetration; if the adversary requires only a short penetration time, the adversary can always avoid the robot using its full knowledge. We only sketch the proof since the result is analogous to Agmon et al. [1].

Lemma 4.3. *The penetration time for the omnidirectional robot defending a circle is* $\lfloor d/2 \rfloor \le t \le d - 2$.

Proof Sketch. For small t, i.e. $t < \lfloor d/2 \rfloor$ the robot cannot reach the adversary in segment $\lfloor d/2 \rfloor + 1$ in either direction.

Contrarily, for large t, i.e. $t \ge d - 1$, the robot can always deterministically walk around the circle and reach every segment in $d - 2$ steps and thus detect the adversary. □

4.2.2 Path Symmetry

Secondly, by virtue of a circle, for every path reaching some segment in clockwise direction there is a path that mirrors it, i.e. switches clockwise and anticlockwise steps, and therefore reaches the segment with the same distance to the robot's location in anticlockwise direction. More formally presented in the following definition, we call the latter path the mirrored path of the former path.

Definition 4.4 (Mirrored Path)**.** For a path u of length l let $(s_1, \dots, s_{l-1})$ be a sequence of clockwise and anticlockwise steps, i.e. $s_i \in \{clockwise, anticlockwise\}$ for $i \in [l - 1]$. The path u's *mirrored path* is the path whose clockwise and anticlockwise step sequence is $(s'_1, \dots, s'_{l-1})$ with $s'_i \in \{clockwise, anticlockwise\}$ and $s'_i = clockwise$ if and only if $s_i = anticlockwise$ for all $i \in [l - 1]$.

From this definition it is obvious that the number of clockwise and anticlockwise steps is the other way around in the mirrored path.

Observation 4.5. *For a path with X clockwise steps and Y anticlockwise steps, the mirrored path has Y clockwise steps and X anticlockwise steps.*

This symmetry means that, a set of paths and the set of their mirrored paths has to have the same cardinality.

Lemma 4.6. *Moving from right into segment $j \in \{2, \dots, \lceil d/2 \rceil\}$ is mirrored by moving from the left into $d - (j-2) \in \{\lceil d/2 \rceil + 1, \dots, d\}$ and vice versa.*

Proof. Let P be the set of all paths that start in segment 1 and end in segment $j \in \{2, \dots, \lceil d/2 \rceil\}$ which was entered in clockwise direction. Overall, these paths all walk $j - 1 = X - Y$ steps in clockwise direction. However, every path in P has a mirrored path which walks $X - Y$ steps in anticlockwise into segment $d - (X - Y - 1) = d - (j - 2)$. The opposite follows similarly. □

Consequently, we can focus on one direction and the other direction follows. We focus on in the next sections until (inclusive) Section 4.2.6 on the clockwise direction.

4.2.3 Restriction of Necessary Paths

Thirdly, we restrict the number of paths we have to consider since any path passing j is covered by a path that ends in j. More specifically, any path of length t passing through j is a path that detects the adversary (see Section 4.1). Additionally, we can divide all paths into sets of paths which share a common prefix, i.e. the path from 1 to the first occurrence of j. Let ω' be such a prefix that ends in j and has length $\ell \leq t$. Then the probability that the robot walks any path of length t with this prefix is $P_W(\omega') \cdot \sum_{i=0}^{t-\ell} p^i (1-p)^{t-\ell}$. The sum of the suffixes is one which also makes sense from the probability argument that they are the space of all possible paths of length $t - \ell$. By that the probability of the set of paths with the common prefix ω' is equivalent to the probabilities of the prefix. Hence, we can conclude that we only need to consider the prefixes (paths ending in j).

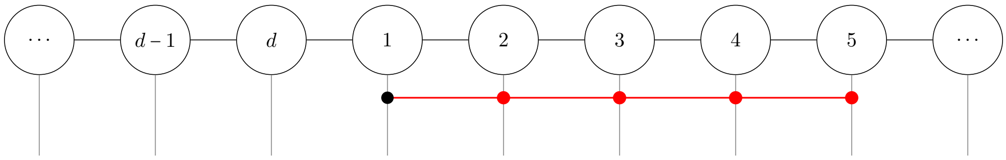
Observation 4.7. *It suffices to consider the subset of paths which end in segment j, without having passed it before, and which are at most of length t.*

Therefore, we have to count all valid paths of length up to t that reach the specified segment j.

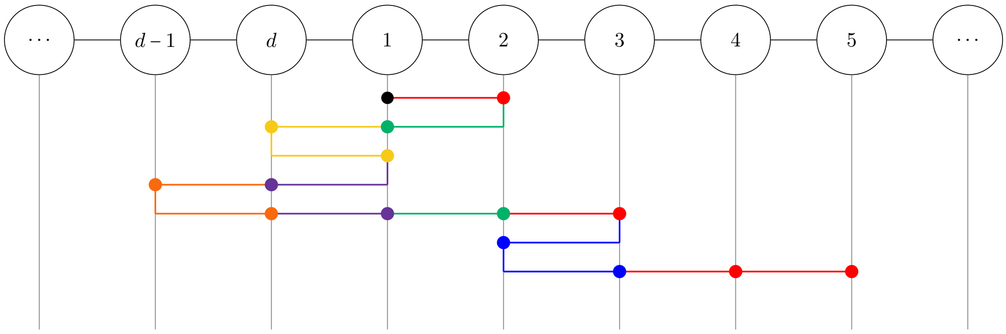
4.2.4 Freedom of Movement

Fourthly, the constraints of considering paths of maximal length t that end in j mean that the robot has the freedom to walk a number of steps equivalent to the difference of t and the steps necessary to reaching segment j, i.e. the distance (see Figure 4.5). As a preparation for modelling the robot's paths as lattice paths we partition this set of all paths which end in segment j into several sets of lattice paths based on the number of *additional steps*.

In more detail, the robot can walk up to t steps. Within these t steps the robot needs to reach j at a distance of $dist := dist(1, j)$. This distance is $j - 1$ for the clockwise direction and $d - j + 1$



(A) In the direct path each step of the robot is towards the target segment.



(B) Besides the direct path, the robot can perform further steps which results in alternate paths. The depicted path is one possibility with 10 extra steps. It can be observed that the direct steps are still present (steps in red), and that all other steps come in pairs (other colours than red).

FIGURE 4.5: Example of the freedom of movement for a path where a robot is in segment 1, the target segment is segment 5 and we assume a clockwise direction. The two figures show the direct path (Figure A) and a path with extra steps (Figure B).

for the anticlockwise direction. By Lemma 4.3 the robot can reach every segment within t so we assume that $t \geq dist$. Hence, since the robot needs $dist$ moves to reach j we have $t - dist$ additional moves. Considering the direct path as the starting point, we observe that any step away from j has to be countered by a step towards j at some point.

Observation 4.8. *The robot may walk up to $\lfloor {}^{(t-dist)}/2 \rfloor$ away from j which has to be countered by the same number of moves towards j.*

We note here that towards and away depend on the direction. For the clockwise direction steps towards segment j are clockwise steps and steps away from j are anticlockwise steps. For the anticlockwise direction it is the other way around.

Based on Observation 4.8 and 4.7 for every length $t - dist + 2i$ with $i \in \{0, 1, \dots \lfloor {}^{dist}/2 \rfloor\}$ we have a different amount of steps and their placement determines the robot's path.

4.2.5 Modelling as Lattice Paths

Finally, after our initial observations on which paths have to be considered, we use lattice paths to model and determine the number of paths within one partition as presented in Section 4.2.4. The overall result: the number of paths for fixed d, j and i is as stated in the following theorem. Since the number matches an entry of Catalan's triangle [79] we use this as a notation for the number.

Theorem 4.9. *The number of paths of exactly length $dist + 2i$ is $C(dist - 1 + i, i)$ with $C(n,k) := \binom{n+k}{k} - \binom{n+k}{k-1}$.*

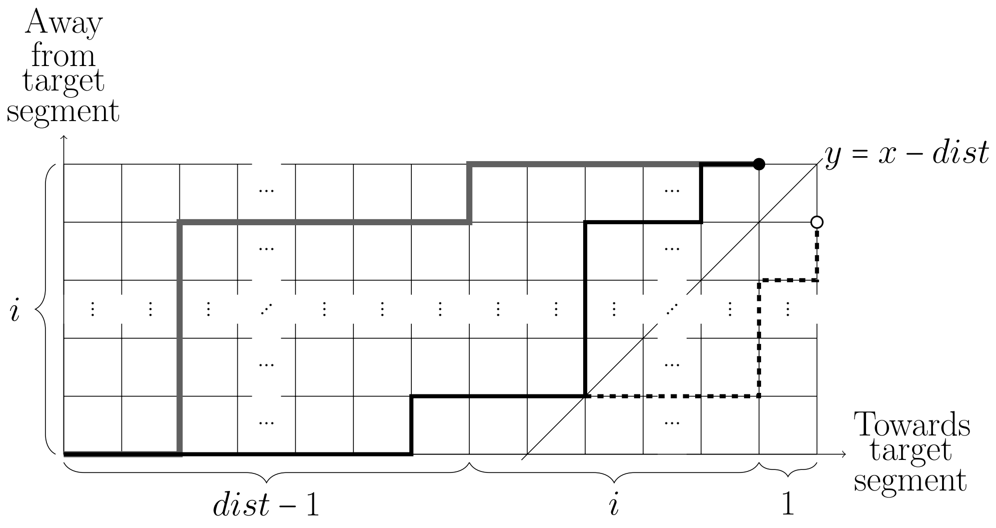


FIGURE 4.6: The lattice that represents the movement pattern for the case of reaching the target segment in clockwise direction. The x-axis are steps in clockwise direction and the y-axis are steps in anticlockwise direction. The start is the bottom left corner and the finish is the top right corner. The disk (filled circle) and the circle are the end points of the valid and the invalid paths, respectively. The thicker grey line is a valid path, whereas the solid black path is an illegal path that touches the line $y = x - dist$. The dashed path shows how reflecting every move after the first touch of the path and the line leads to the circle.

In order to establish this number we define the lattice path setting (see Figure 4.6) as follows. The robot's initial position corresponds to $(0, 0)$, the bottom left corner. Starting there, we allow the following lattice paths.

Definition 4.10 (Lattice Path)**.** A *lattice path* is a sequence of horizontal and vertical unit steps beginning at $(0, 0)$ and ending in a specified point.

The horizontal direction represents movement towards the target (clockwise for reaching the target segment in clockwise direction) and the vertical direction represents movement away from the target (anticlockwise for reaching the target segment in clockwise direction). Hence, the point $(x, y) \in \mathbb{Z}^2$ corresponds to the robot being in segment $y - x + d + 1$ if $x < y$ or else segment $x - y + 1$ (for the anticlockwise direction it is the other way round). The fact that we consider only paths that end in j is incorporated by considering only paths that do not cross the line $y = x - dist$.

Finally, while paths that end in j after $dist + 2i$ steps correspond to paths that end in $(dist + i, i)$, for technical reasons, we will consider paths to coordinate $(dist - 1 + i, i)$. The number of paths is not affected by this. Logically, every path that ends in j after $dist + 2i$ steps must end in $j - 1$ after $dist - 1 + i$ steps. Graphically, considering Figure 4.6, the only way to reach $(dist + i, i)$ is by going through $(dist - 1 + i, i)$.

4.2.6 Counting the Lattice Paths

The described lattice path setting allows us to show that the number of lattice paths from $(0, 0)$ to $(dist - 1 + i, i)$ amounts to the number of paths stated in Theorem 4.9. We do this by counting all lattice paths from the origin to the end point and subtract all paths that touch or cross the line $y = x - dist$.

Lemma 4.11. *The number of lattice paths from* $(0,0)$ *to* $(dist - 1 + i, i)$ *not crossing line* $y = x - dist$ *is* $C(dist - 1 + i, i)$.

Proof. The number of unrestricted lattice paths, not restricted by a line, from $(0,0)$ to (a,b) is $\binom{a+b}{b}$ (see Theorem 10.3.1 in *Lattice Path Enumeration* [49]).

For the restricted paths, we can use the general ballot theorem [42] or André's reflection principle [49] to count these. According to the reflection principle, we can change any path that touches or crosses the line so that horizontal steps become vertical steps and vice versa which reflects the endpoint of these paths to $(dist + i, i - 1)$ (see Figure 4.6). Hence, the total number of paths, using above binomial coefficient for the legal and illegal paths, is

$$\binom{dist - 1 + 2i}{i} - \binom{dist - 1 + 2i}{i - 1} = \frac{dist}{i}\binom{dist - 1 + 2i}{i - 1}$$

which is $C(dist - 1 + i, i)$. □

The number of the robot's paths follows immediately.

Proof of Theorem 4.9. By the correspondence between the robot's paths and the lattice paths (see Section 4.2.5), Lemma 4.11 directly implies the claim. □

4.2.7 Probability of Detecting the Adversary

Finally, having the number of paths, we can immediately state the probability that a robot in segment 1 reaches segment j in at most t steps given the Bernoulli parameter p.

Theorem 4.12. *The probability of an omnidirectional robot detecting an adversary in segment* j *of a perimeter of size* d *is*

$$\begin{aligned} Pr(1,j,t,d) = &\sum_{i=0}^{\lfloor \frac{t-d+j-1}{2} \rfloor} C(d-j+i,i)\cdot p^{d+i-j+1}\cdot(1-p)^i \\ &+ \sum_{i=0}^{\lfloor \frac{t-j+1}{2} \rfloor} C(j-2+i,i)\cdot p^i\cdot(1-p)^{i+j-1} \end{aligned}$$

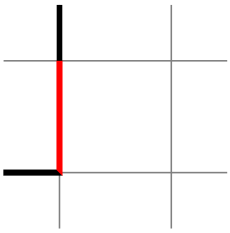
Proof. The probability function follows by combining the statements in this section. Firstly, Observation 4.5 and Lemma 4.6 imply that the results hold for both directions. As stated in Section 4.2.4 by the Observations 4.7 and 4.8 we can sum over all prefixes to get all paths. Finally, Theorem 4.9 gives the number of paths and Definition 4.2 gives the probability of a single path. □

4.3 Remaining Settings

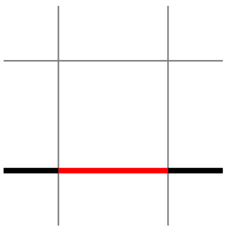
In the previous section we presented the general modelling technique and how to count the number of paths. The probability in the other three settings can be found similarly by counting lattice paths with different features. Due to the similarities we focus on the important differences. We present the modelling and the probabilities for the cases of the directional movement in the circle and the omnidirectional movement on the line. For the directional movement on the line we simply present the probability since no further ideas or approaches besides those from the other three cases are required.

4.3.1 Directional Movement in the Circle

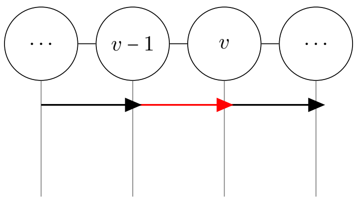
For the directional movement we have a path probability of $P_W = p^X \cdot (1-p)^Y$ where, in this case, X is the number of steps and Y is the number of turns. Like before, the robot needs *dist* moves to reach j. However, in comparison to the previous case, X and Y do not directly correspond to the clockwise and anticlockwise steps which means we need a different approach for counting the paths. Moreover, we separate the additional moves into three different types. The first type are additional steps which are, similar to the previous case, steps besides those required to reach



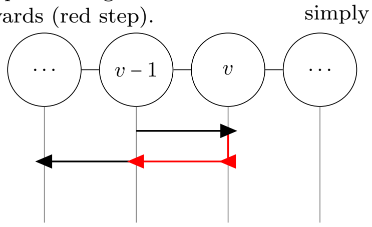
(A) Examples of a change in direction. The path changes direction upwards (red step).



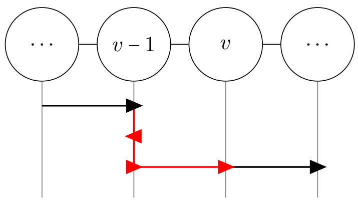
(B) Examples of a path without a change in direction. The path simply continues.



(C) Examples of a step forwards without change in direction.



(D) Example of a turn. After reaching segment v the robot changes direction and continues in the new direction.



(E) Examples of a spin. In segment $v-1$ the robot changes direction twice before continuing in the original direction.

Figure 4.7: Illustrations of robot paths and lattice paths with and without directional changes. Figures (A) and (B) are parts of lattice paths. Figures (C), (D) and (E) show robot paths on the graph. At the top are the segments of the polyline. The robot's movement is indicated by the arrows on the grey lines under the segments. The arrowheads on the grey lines represent the positioning of the robot as well as the direction the robot is facing, and the lines are the connecting steps of the robot. These images show how turns and steps can be modelled using lattice paths but spins have to be considered separately. A simple walk in one direction like in Figure (C) would be simply a lattice path that does not change direction as in Figure (B). Similarly, the turn shown in Figure (D) can be represented as a turn in a lattice path as shown in Figure (A). However, a spin in the robot's path as in Figure (E) would appear in the lattice path in the same way as the path without any directional changes like in Figure (B).

the target segment. The other two movement types are two kinds of changes in direction called turn and spin, separated for technical reasons.

Definition 4.13 (Turn / Spin)**.** A *turn* is a change of direction followed by a step into the new direction (see Figure 4.7D) and a *spin* are two consecutive changes of direction (see Figure 4.7E).

As before the majority of the results are concerned with the number of paths for one combination of direction, target segment and amount of additional moves. Overall, we will prove that one combination represented in the following lemma has the number of paths stated in the preceding definition. We prove the lemma towards the end of the segment just before the statement of the probability functions for this setting.

Definition 4.14 (C_{DMC})**.** Let the number C_{DMC} be defined as follows:

$$\begin{aligned} &C_{DMC}(dist, k, i, m) \\ &= \binom{dist + 2k + i - m - 1}{i - m} \cdot \left(\binom{dist - 1 + k}{m} \binom{k}{m} - \binom{dist + k}{m} \binom{k - 1}{m} \right) \end{aligned}$$

Lemma 4.15. *The number of paths from* $(0, 0)$ *to* $(dist - 1 + k, k)$ *with* $2i$ *turns and spins, from which* $2m$ *are turns,* $2k$ *additional steps, and not crossing line* $y = x - dist$ *is* $C_{DMC}(dist, k, i, m)$*.*

In more detail, similar to the omnidirectional case we have $t - dist$ time for the robot to perform movements other than reaching the target. Firstly, like in the previous setting, all of them come in pairs again as moving or turning away from the target has to be reversed at some point (see also Section 4.2.4 and Observation 4.8.

Observation 4.16. *Every additional movement of the robot (extra steps, turns and spins) has to be countered by the same but opposite movement type at some point.*

We determine the result of Lemma 4.15 in three steps. Firstly, we establish the number of additional moves, turns and spins. Secondly, we count the number of paths given the number of turns and additional steps. Thirdly, we factor in the number of ways for the robot to perform spins on the paths. The separation of turns and spins is necessary since turns can be represented in the lattice as presented in Section 4.2.5. Similar to additional steps, turns can be represented in the lattice since they are followed by a step (see Figure 4.7). This is not the case for spins which can happen at any point in the path. Nevertheless, since a spin can be performed at any point of a path they can be factored in after considering the additional steps and turns. Moreover, we assume the initial direction is clockwise which means that reaching a segment in anticlockwise direction needs an additional change in direction.

Beginning with the number of additional moves, since a turn takes τ steps the robot can perform at most $\left\lfloor \frac{(t-dist)}{(2\cdot\tau)} \right\rfloor$ many changes in direction. Similar to the previous case (see Section 4.2.4), the robot may perform any fixed number of these $i \in \left\{0, 1, \dots, \left\lfloor \frac{(t-dist)}{(2\cdot\tau)} \right\rfloor\right\}$, considering the different types of paths.

Of these i turns and spins, we denote the number of pairs of turns with m. This number must be within the range of all changes of direction, i.e. in the range from 0 to $\min\left\{i, \left\lfloor \frac{(t-dist)}{(2\cdot\tau+2)} \right\rfloor\right\}$, where the additional 2 comes from the accompanying steps. Consequently, a set number of turns

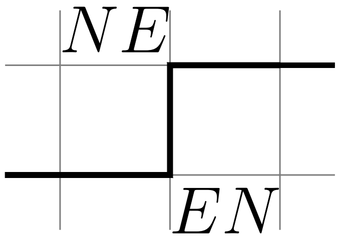


FIGURE 4.8: A EN- and NE-turn on a lattice path. The shown part of a lattice path contains both types of turns. As indicated, the first turn is an EN-turn where a horizontal step is followed by a vertical step. This is followed by a NE-turn where a vertical step is followed by a horizontal step.

implies a fixed number of $i - m$ spins. Finally, the remaining moves can be used for additional steps $k \in \left\{m, \cdots, \left\lfloor \frac{(t-dist-2i)}{2} \right\rfloor\right\}$ where we require the $2m$ steps after a turn.

Altogether, this gives us a setting where we can count the paths for fixed i, m and k. In the second step, we use the lattice as above (see Section 4.2.5) to count the number of paths given additional steps k and turns m. Turns can be represented as a feature of the lattice paths since they appear as either a *north east turn* or a *east north turn* (see Figure 4.8).

Definition 4.17 (NE-turn / EN-turn)**.** A point on a lattice path is a *north east turn (NE-turn)* if it is the end point of a vertical step and the starting point of a horizontal step. Similarly, a point on a lattice path is a *east north turn (EN-turn)* if it is the end point of a horizontal step and the starting point of a vertical step.

We count the number of turn pairs m by the number of NE-turns since those represent the total number of turns.

Lemma 4.18. *A robot's path with* $2m$ *turns is a lattice path with* m *NE-turns.*

Proof. With two exceptions every turn of the robot shows up as a NE-turn or an EN-turn (see the thick grey path in Figure 4.6 with two turns of either type). The exceptions are walking vertically from $(0,0)$ which is a turn of direction (equivalent to EN-turn) and walking vertically into the target segment. The latter of walking into the target segment is not possible in our setting (compare Observation 4.7). Consequently, the paths with m NE-turns represent all path with $2m$ turns. □

Counting paths with a specified number of NE-turns can be performed by observing the fact that the positions of the NE-turns determine a lattice path [49]. Placing them onto the possible (x, y) coordinates yields the following lemma. This is a special case of Equation 10.120 in Section 10.14 of *Lattice Path Enumeration* [49]).

Lemma 4.19. *The number of paths from* $(0,0)$ *to* (a,b) *with* ℓ *NE-turns is*

$$\binom{a}{\ell}\binom{b}{\ell}.$$

In the third step we factor in the spins which can be performed in every segment along a path. Hence, we can include them by distributing them along the path. This amounts to distributing the $i-m$ spins over the $dist-1+2k+1$ positions. This can be seen as counting the combinations of putting indistinguishable balls into distinguishable boxes. This number is an elementary result in enumerative combinatorics (see e.g. *Enumerative Combinatorics* [76]).

Lemma 4.20. *The number of ways n indistinguishable balls can be put into m distinguishable boxes, allowing multiple balls per box, is $\binom{m+n-1}{n}$.*

Finally, we prove Lemma 4.15 by combining these results.

Proof of Lemma 4.15. We count the number of paths for $2k$ additional steps, $2i$ turns and spins and $2m$ turns. The term is comprised of two factors: 1) the number of paths given turns and additional steps, 2) the ways to include the spins.

Firstly, by Lemma 4.18 we can count the number of path given the number of steps and turns by counting all lattice paths with m NE-turns. The number of all of these paths is given by Lemma 4.19 since we have paths from $(0,0)$ to $(dist-1+k,k)$. As in Lemma 4.11 we get the number of all legal path by subtracting the number of lattice paths that cross the line from the number of all paths using the reflection principle.

Secondly, we have to include the spins to get the overall result. We achieve this by multiplying the number of paths, given turns and spins, by the number of possibilities of distributing the $i-m$ spins over the paths of length $dist-1+2k+1$. This number is given by Lemma 4.20. The product of the two factors gives us the overall result. □

Finally, the probability functions are simply a combination of i, m and k similar to Theorem 4.12.

Theorem 4.21. *The probability of a directional robot detecting an adversary in segment j on a perimeter of size d is*

$$\begin{aligned}
Pr(1,j,t,d) = &\sum_{i=0}^{\lfloor \frac{t-d+j-2}{2\cdot\tau} \rfloor} \sum_{m=0}^{\min\{i,\lfloor \frac{t-d+j-2}{2\cdot\tau} \rfloor\}} \sum_{k=m}^{\lfloor \frac{t-d+j-2-2i}{2} \rfloor} C_{DMC}(d-j+1,k,i,m) \\
&\cdot p^{d-j+2k+2m+1} \cdot (1-p)^{2m+2(i-m)+1} \\
&+ \sum_{i=0}^{\lfloor \frac{t-j}{2\cdot\tau} \rfloor} \sum_{m=0}^{\min\{i,\lfloor \frac{t-j}{2\cdot\tau} \rfloor\}} \sum_{k=m}^{\lfloor \frac{t-j-2i}{2} \rfloor} C_{DMC}(j-1,k,i,m) \\
&\cdot p^{j-1+2k+2m} \cdot (1-p)^{2m+2(i-m)}
\end{aligned}$$

Proof. For both directions, i, m and k determine the number of changes in direction, the number of pairs of turns, and the number of pairs of additional steps, respectively. This depends on the distance which is, as before, different for the two cases of reaching the target segment in clockwise and anticlockwise direction. Also as before, all possible combinations result in different numbers of paths. Hence, considering all combinations for both directions yields the sums.

For both directions we have to determine for a given i, m and k combination what is the probability of the path. As a reminder, p is determined by the number of steps and $1-p$ the number of changes in direction. Hence, we have to add up the respective numbers of those per case. For both cases, the steps are determined by the distance, the pairs of additional steps and

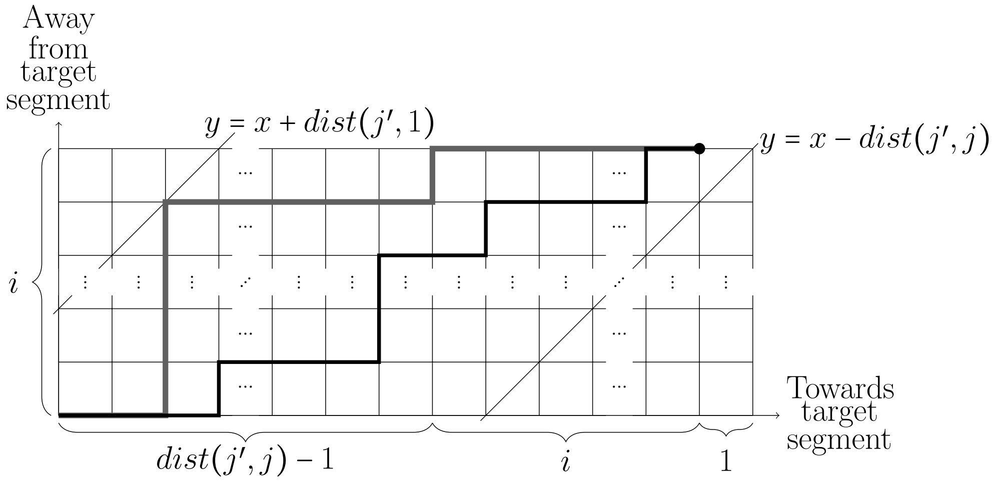


FIGURE 4.9: The lattice for the omnidirectional case on the line. In general the grid is like the one in Figure 4.6. As in the previous case, the line $y = x - dist(j', j)$ is to exclude paths that enter the target segment j prior to having exhausted the additional steps. Additionally, the setting on the line graph has a second line $y = x - dist(j', 1)$ which represents the end of the line/fence. In comparison to the $y = x - dist(j', j)$ restricting line, a path may touch the $y = x - dist(j', 1)$ line, like in the case of the thicker grey line, but cannot cross it. The black path is a valid path that touches neither line.

the pairs of steps after a turn. Therefore, we have $j - 1 + 2k + 2m = j + 2k + 2m$ for the clockwise case and $d - j + 1 + 2k + 2m = d - j + 2k + 2m + 1$ for the anticlockwise case. For the changes in direction these are determined by the pairs of turns and the pairs of spins. Hence, we have $d-j+1+2k+2m = d-j+2k+2m$ for the clockwise case and $d-j+1+2k+2m = d-j+2k+2m+1$ for the anticlockwise case, which requires an additional change in direction under the assumption that the robot starts clockwise.

Finally, Lemma 4.15 gives the number of lattice path for a given case representing the number of path the robot can walk. □

4.3.2 Omnidirectional Movement on the Line

In comparison to the circle, the previous symmetry does not hold any more because of the line's defined ends. This also affects the probabilities of paths since the robot has to turn around at the end of the line with probability 1. In general, the movement is now split into X right steps with probability $(1 - p)$, Y left steps with probability p and Z steps turning around in the end segments, implying the probability measure $P_W(\omega) = p^Y \cdot (1 - p)^X \cdot 1^Z$. This is not reflected in the lattice described in Section 4.2.5 since in the circle the robot had no restrictions, except the available steps, for walking away. Hence, the crucial part of this setting is to reflect the restrictions of the robot's path by restrictions of the lattice paths. This can be done by introducing a second line in the lattice path representation which represents the end of the line graph (see Figure 4.9).

Before we address this, we also have to consider that we do not have the same symmetry as before. Furthermore, we cannot assume a fixed initial position of the robot as before and have to consider all pairs of segments. Fortunately, we have a different symmetry in this setting and we still only need to consider paths that end in the target segment (see also Section 4.7 and Observation 4.7). This means that the robot can only reach one end of the line, and every setting where the target segment is left of the initial segment is similar to a setting where the target is left of the initial segment and the distance to end and the target segment are the same.

Lemma 4.22. *The number of paths from segment j' to segment j with $j' < j$ is the same as the number of paths from a segment $d - dist(1, j')$ to segment $d - dist(1, j)$ with $dist(j', j) := |j - j'|$.*

This allows us to concentrate on the case where the initial segment j' is left of the target segment j, i.e. $j' < j$. In order to count the number of paths we have to adjust the lattice positions and add, as we mentioned in the beginning of this section, another line.

In more detail, in principal the lattice is the same as in the setting on the circle (see Section 4.2.5, Figure 4.6). However, point $(0, 0)$ corresponds to j'. As before, the line $y = x - dist(j', j)$ represents segment j that cannot be entered prematurely. Additionally, the line $y = x + dist(j', 1)$ represents the end of the line.

Again, as before any path can have $i \in \left\{0, 1, \ldots, \left\lfloor \frac{(t-dist(j',j))}{2} \right\rfloor\right\}$ additional moves. The difference is that since reaching the end of the line changes the probability, we have to consider not only paths of a certain freedom of movement but how often a path reaches the end of the line. More precisely, we have to determine the number of paths given that the end is reached k times, i.e. the number of lattice path that touch (not intersect) the introduced line $y = x + dist(j', 1)$ exactly k times. The range of k is limited by the distances and the penetration time $0 \le k \le \left\lfloor \frac{(t-dist(1,j')-dist(1,j))}{2} \right\rfloor$.

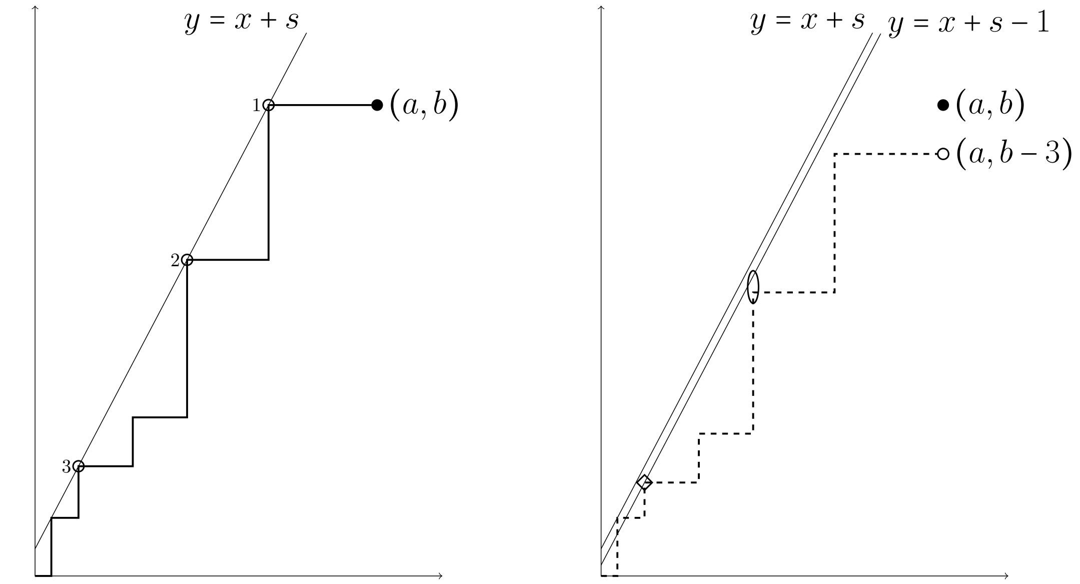
We can count the number of paths reaching the line k times by adapting a bijection used by Spivey [75]. We describe a bijection between the lattice paths that touch the line k times and the lattice paths that touch the line once.

Lemma 4.23. *There is an explicit bijection between paths from $(0, 0)$ to (a, b) touching the line $y = x + s$ exactly k times and paths from $(0, 0)$ to $(a, b - k)$ touching the line exactly once.*

Proof. We can transform a path from the first group to a path in the second group by taking any path from the first group and removing, beginning with the last touch, any vertical step that ends on the line until only one touch is left (see Figure 4.10).

For the reverse we take any path from the second group. We start from the last point where the path touches or intersects $y = x + s - 1$ and add into the path one vertical step. We repeat this until the path ends in (a, b). This procedure will always produce a path in the first group since at least one touch is guaranteed, by the one touch of line $y = x + s$, and shifting the path up by one vertical step results in an intersection of the path and $y = x + s - 1$. By always choosing the last touch or intersection this procedure reverses the mapping. □

Hence, we can split all paths into a path from the origin to a point on the line and a path from that point to the changed target point. Moreover, summing up over all possible midpoints gives us the total number of paths. We state this formally in the following lemma.



(A) A path in the first group. The path starts at $(0,0)$, ends in (a,b) and touches the line 4 times. The numbered circles indicate the order with which the touches are processed in the bijection.

(B) A path in the second group. The path starts at $(0,0)$, ends in $(a,b-3)$ and touches the line once. The diamond indicates the touch of the line $y = x + s - 1$

FIGURE 4.10: Illustration of the bijection of Lemma 4.23. The bijection takes a path from the first group and removes, beginning with the last touch (touch 1 in Figure A), one vertical step at the intersection of the path and the line. This results in the path illustrated in Figure A. For the reverse, the vertical steps are added back in. The diamond in Figure A indicates how the path touching before at 3 (Figure A) is now touching the line $y = x + s - 1$. Adding in a vertical step restores touch 3 and results in a new touch of the path with the line $y = x+s-1$ (at the oval). Repeating this process until the path ends in (a,b) again completes the bijection.

Lemma 4.24. *The number of paths from $(0,0)$ to (a,b) touching the line $y = x + s$ exactly k times is*

$$\sum_{m=0}^{b-k-s} |(0,0) \to (m, m+s)| \cdot |(m, m+s) \to (a, b-k)|,$$

where $|(u,v) \to (x,y)|$ denotes the number of paths from (u,v) to (x,y).

Proof. By Lemma 4.23 for any path touching the line k times there is a path touching it once in the first touch point. Hence, if we sum up the paths over every possible point on the line we get the total number of paths. □

The entire number of lattice paths can be obtained by counting the number of paths between the two lines. For ease of notation we denote this number with D_{OML}.

Definition 4.25 (D_{OML})**.** Let the number D_{OML} be defined as follows:
$D_{OML}(a,b,c,d,e,g)$

$$= \sum_{\ell=1}^{\lfloor (e-g+1)/2 \rfloor} \frac{4}{e-g+2} \left(2 \cdot \cos\left(\frac{\pi \ell}{e-g+2}\right)\right)^{c+d-a-b} \cdot \sin\left(\frac{\pi \ell (a-b+e+1)}{e-g+2}\right) \cdot \sin\left(\frac{\pi \ell (c-d+e+1)}{e-g+2}\right)$$

That this number is the number of paths between two lines can be obtained using standard lattice path results. The following Lemma is Theorem 10.3.4 in *Lattice Path Enumeration* [49]).

Lemma 4.26. *The number of paths from* (a,b) *to* (c,d) *staying below the line* $y = x + e$ *and above the line* $y = x + g$ *is* $D_{OML}(a,b,c,d,e,g)$.

Consequentially, we can use D_{OML} to refine the expression in Lemma 4.24 stating the number of paths touching the line k times. We denote this with C_{OML}.

Definition 4.27 (C_{OML})**.** Let the number C_{OML} be defined as follows:

$$\begin{aligned} &C_{OML}(j,j',\hat{j},i,k) \\ =& \sum_{m=0}^{i-k-|\hat{j}-j'|} D_{OML}(0,0,m,m-|\hat{j}-j'|,|\hat{j}-j'|,-|j-j'|) \\ &\qquad \cdot D_{OML}(m,m-|\hat{j}-j'|,|j-j'|+i-1,i-k,|\hat{j}-j'|,-|j-j'|) \end{aligned}$$

Since the lattice paths represent the robot's paths we can use this to formally state the number of paths for the robot.

Lemma 4.28. *There are* $C_{OML}(j,j',\hat{j},i,k)$ *paths for a robot in segment* j' *walking to segment* j *which reaches segment* $\hat{j}$ *exactly* k *times using* i *additional steps.*

Proof. The results follows from observing how the robot's path fit into Lemma 4.26. Firstly, the robot has to reach j and use all additional steps. Hence, the lattice has to reflect that and the end of the paths (a,b) has to be $(|j-j'|+i-1,i)$. Additionally, any path of the robot is limited by the distance to the end of the line, i.e. $e = |\hat{j}-j'|$, and the distance to the target segment, i.e. $g = -|j-j'|$. □

Finally, again, combining these results yields the probability functions.

Theorem 4.29. *The probability of an omnidirectional robot in segment* j' *detecting an adversary in segment* j *on a line of size* d *is*

$$Pr(j',j,t,d) = \sum_{i=0}^{\left\lfloor \frac{t-|j-j'|}{2} \right\rfloor} \sum_{k=0}^{\left\lfloor \frac{t-|1-j'|-|1-j|}{2} \right\rfloor} C_{OML}(j,j',1,i,k) \cdot p^{i} \cdot (1-p)^{|j-j'|+i+k}$$

if segment j' *is left of segment* j *and*

$$Pr(j',j,t,d) = \sum_{i=0}^{\left\lfloor \frac{t-|j-j'|}{2} \right\rfloor} \sum_{k=0}^{\left\lfloor \frac{t-|1-j'|-|1-j|}{2} \right\rfloor} C_{OML}(j,j',d,i,k) \cdot p^{|j-j'|+i+k} \cdot (1-p)^{i}$$

if segment j' *is right of segment* j.

Proof. It is clear that the two cases of the robot being left or right of the target are symmetric (see also Lemma 4.22). Hence, we focus on the case where the robot is left of the target segment.

The sets of possible paths are all combinations of additional steps and how often the robot reaches the end of the line. All of these cases are covered by the two sums. For the probabilities, the number of left steps (exponent of p) is one step of every additional step pair, i.e. i. The number of right steps (exponent of $1-p$) is given by the distance between the start and the target segment, one step of every additional step pair and one step for every instance of reaching the end of the fence, i.e. $|j - j'| + i + k$.

Finally, $C_{OML}(\cdot)$ as specified in Definition 4.27 is the number of paths for the cases as established by Lemma 4.28. □

4.3.3 Directional Movement on the Line

The probability functions in the case of the robot with directional sensors and the associated movement defending the fence can be determined using the ideas and approaches of the other three cases. Since there are no new results required but merely the application of above we simply state the results here.

Similar to the previous case, we denote the number of lattice paths between two lines with D_{DML}.

Definition 4.30 (D_{DML})**.** Let the number D_{DML} be defined as follows:
$D_{DML}(a, b, c, d, e, g, \ell)$

$$= \sum_{k \in \mathbb{Z}} \left(\binom{c-a-k(e-g)}{\ell+k} \binom{d-b+k(e-g)}{\ell-k} - \binom{c-b-k(e-g)+g-1}{\ell+k} \binom{d-a+k(e-g)-g+1}{\ell-k} \right)$$

That this number describes the number of lattice paths between two lines is again an established result in lattice path literature. The following Lemma is Theorem 10.14.3 in *Lattice Path Enumeration* [49]).

Lemma 4.31. *Let $a + t \geq b \geq a + s$ and $c + t \geq d \geq c + s$. The number of all paths from (a, b) to (c, d) staying weakly below the line $y = x + t$ (,i.e. may touch the line but not cross it) and above the line $y = x + s$ with exactly ℓ NE-turns is given by $D_{DML}(a, b, c, d, e, g, \ell)$.*

Also similar to the other settings we denote the resulting formula for the number of robot paths with C_{DML}.

Definition 4.32 (C_{DML})**.** Let the number C_{DML} be defined as follows:

$$\begin{aligned} &C_{DML}(j, j', \hat{j}, i, k, \ell) \\ &= \binom{|j - j'| + 2k + i - m - 1}{i - m} \\ &\quad \cdot \sum_{m=0}^{i-k-|\hat{j}-j'|} \sum_{r=0}^{\ell} D_{DML}(0, 0, m, m + s, e, g, r) \\ &\qquad\qquad \cdot D_{DML}(m, m + s, a, b - k, e, g, \ell - r) \end{aligned}$$

This expression is similar to the one in the other settings above and the proof is omitted since it is similar to the one of Lemma 4.28.

Lemma 4.33. *The number of paths from initial segment j' to segment j with the respective end segment $\hat{j} \in \{1, d\}$, i and k is $C_{DML}(j, j', \hat{j}, i, k, \ell)$*

Finally, this lets us express the probability functions for this case.

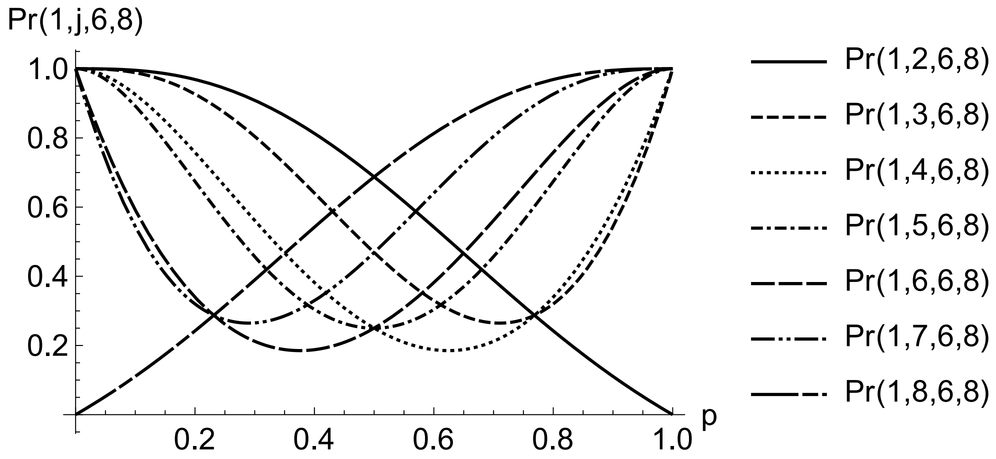


FIGURE 4.11: The probabilities of the robot reaching the segments for p ranging from 0 to 1 in a circle with 8 segments ($d = 8$) and a penetration time of 6 ($t = 6$).

Theorem 4.34. *The probability of a directional robot in segment j' detecting an adversary in segment j on a line of size d is*

$$Pr(j', j, t, d) = \sum_{i=0}^{\left\lfloor \frac{t-\delta-|j-j'|}{2\tau} \right\rfloor} \sum_{k=0}^{\left\lfloor \frac{t-\delta-|1-j'|-|1-j|-2}{2} \right\rfloor} \sum_{m=0}^{\min\left\{i, \left\lfloor \frac{t-\delta-|j-j'|}{2\cdot\tau} \right\rfloor\right\}} \sum_{\ell=m}^{\left\lfloor \frac{t-\delta-|j-j'|-2i}{2} \right\rfloor} C_{DML}(j, j', \hat{j}, i, k, \ell) \cdot p^{|j-j'|+2k+2m} \cdot (1-p)^{2m+2(i-m)+\delta}$$

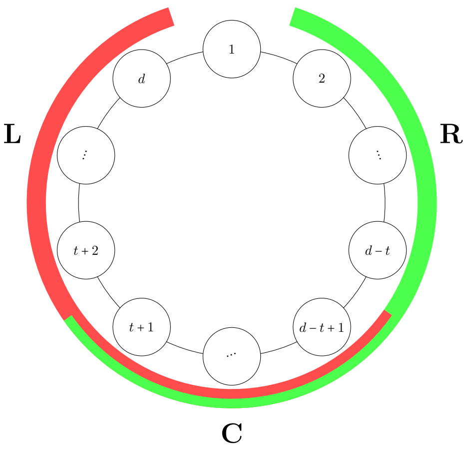
where δ is 1 if the robot is facing towards j and 0 if the robot is facing towards the end of the line segment, $\hat{j} \in \{1, d\}$ is then end segment, i.e. $\hat{j}$ is 1 if j' is left of segment j and d if segment j' is right of segment j.

The theorem follows from reasoning similar to Theorem 4.21 and Theorem 4.29.

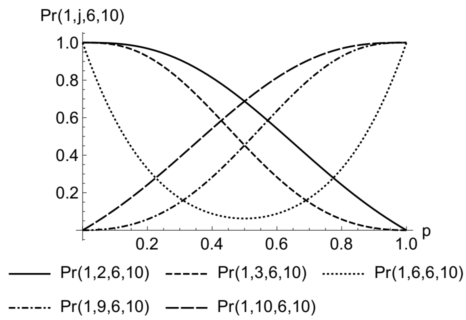
4.4 Examples

In this section we illustrate on the example of the omnidirectional robot on the circle (see Section 4.2) that our approach allows to gain further insight in the relationship between the optimal p and different values of d, t. Moreover, we illustrate the runtime of both steps to highlight the runtime previously required and the remaining necessary runtime.

Before we consider the calculation and give notes on the implementation, the analytic probability functions allow us to show an example of the resulting curves. Figure 4.11 illustrates, depending on p, the 7 probabilities of an omnidirectional robot reaching the segments 2 – 8 from segment 1 in a circle with 8 segments and a penetration time of 6. Two different groups of segments are apparent from the figure. The first group are the segments 2 and 8 where the penetration time does not allow the robot to reach the segment from both directions (clockwise and anticlockwise). Hence, over the range of p, the curves decrease or increase for reaching segment 2 clockwise or segment 8 anticlockwise, respectively. All the other segments make the second group, consisting of segments that are reachable from both directions. The graph also shows three local maxima



(A) The segment types indicated on the circle. Segments that satisfy the condition for R segments are green and segments that satisfy the condition for L segments are red. C segments are those segments where both conditions are true at the same time, which is indicated by both colours.



(B) An example of the plots of the probability functions of the segments of the different segment types for $d = 10$ and $t = 6$. $Pr(1, 2, 6, 10)$ and $P(1, 3, 6)$ are an examples of R segments, $P(1, 9, 6)$ and $P(1, 10, 6)$ are an examples of L segments, and $P(1, 6, 6)$ is an example of a C segment.

FIGURE 4.12: An illustration of the location and the probability functions of the segment types of Definition 4.35.

on the lower envelope, one in the centre at $p = 0.5$ and two equidistant to 0.5 near 0.225 and 0.775. The general observations of this small example appear to hold for all values of d and t. We show that this grouping is the same for any circle and penetration time in the next section.

4.4.1 Segment Types/Reducing the Search Space

As stated in the introduction, finding the optimal probability is a two step process where in the first step we determine the probability functions and in the second step we solve the system of equations for an optimal parameter. In this sense Theorem 4.12 solves the problem of the first step. Hence, these can be used for the second step to determine an optimal parameter. As stated by Agmon et al. [1] this means determining the maximum of the lower envelope of the $Pr(1, j, t, d)$ for $1 \leq j \leq d$. Moreover, they state that the maximum has to be a local maximum or an intersection of two functions. However, in comparison to Agmon et al. [1] the closed representations of the probabilities provided by Theorem 4.12 allows us to examine the functions and make observations regarding the lower envelope.

In this section we use this to show how the search space can be reduced by excluding a number of functions which are above the lower envelope and can be excluded in calculating the intersections and roots. These are the probability functions of reaching the segments immediately left and right of segment one. The amount of these segments depends on the value of t.

The intuitive reason why we can omit these functions from the analysis is that the robot can only reach them in one direction and the two segments furthest away in both directions have the smallest chance of being reached. Hence, only the probability of these two segments can affect the lower envelope, and all other segments closer to segment 1 can be ignored. This can be seen in Figure 4.12B where the probability of the segments further away from segment 1 (segment 3 and 9) is smaller than the probability of the segments closer to segment 1 (segment 2 and 10, respectively).

More precisely, these are the segments where only one of the two sums of $Pr(1, j, t, d)$ is non-zero. This results in three types of functions (see Figure 4.12B) where, depending on p, the probability function increases, decreases or first decreases and then increases again. The first two types are the segments which are only reachable from one directions and the third type are the segments reachable from both directions. We denote these three types of segments with R, L and C.

Definition 4.35 (R/L/C Segment)**.** The segments are divided into three types called R, L and C segments. The type of a segment j is determined by if it satisfies just inequality $j \leq d - t$, just inequality $j \geq t + 2$ or both inequalities (see Figure 4.12A). More formally, a segment j belongs to the segment type as follows:

R Segments satisfy $j \leq d - t$,

L Segments satisfy $j \geq t + 2$,

C Segments satisfy $d - t + 1 \leq j \leq t + 1$.

Using this definition we can formally state our claim that we can exclude all but one segment, and therefore their probability functions, from each of the L and R segments.

Theorem 4.36. *In each of the two types of L and R segments the segment adjacent to a C segment has a lower probability than any other segment of the same type. More precisely,* $Pr(1, j, t, d) < Pr(1, j', t, d)$ *for segments*

- $j = d - t$ *and* $j' < d - t$ *for R segments or*
- $j = t + 2$ *and* $j' > t + 2$ *for L segments.*

Before we prove this fact we prove, for completion, that the definition describes three distinct groups of segments.

Lemma 4.37. *The segment types are mutually exclusive.*

Proof. It is easy to see that type C segments are separate from type R and L.

For R and L segments, if they were overlapping that would mean that $t < \frac{d}{2}$ which is not possible due to the observation of Lemma 4.3. Hence, R and L segments are distinct. □

Following this, we prove the theorem that one segment from each of the types L and R has lower probability then the remaining segments of the respective type

Proof of Theorem 4.36. Like in previous statements, by symmetry (see Section 4.2.2) we can focus on one case and the other one follows similarly. Hence, we focus on the case of L segments and, for sake of contradiction, assume the opposite of the claim is true, i.e.

$$Pr(1, t + 2, t, d) \geq Pr(1, j, t, d) \tag{4.1}$$

with $j \in \{t+3,\dots,d\}$.

To begin, we can rewrite this equation. The left hand side is equivalent to

$$p^{d-j+1} \cdot p^{j-t-2} \cdot \left(\sum_{i=1}^{\lfloor \frac{2t-d+1}{2} \rfloor} C(d-t+i-2,i) \cdot p^i \cdot (1-p)^i + C(d-t-2,0) \right)$$

and the right hand side is equivalent to

$$p^{d-j+1} \cdot \sum_{i=1}^{\lfloor \frac{t-d+j-1}{2} \rfloor} C(d-j+i,i) \cdot p^i \cdot (1-p)^i + C(d-j,0).$$

This means we can express the inequality (Equation (4.1)) as a bound on the probability of the robot walking the distance between the two segments.

$$p^{j-t-2} \geq \frac{\sum_{i=1}^{\lfloor \frac{t-d+j-1}{2} \rfloor} C(d-j+i,i) \cdot p^i \cdot (1-p)^i + 1}{\sum_{i=1}^{\lfloor \frac{2t-d+1}{2} \rfloor} C(d-t+i-2,i) \cdot p^i \cdot (1-p)^i + 1} \tag{4.2}$$

In order to come to a contradiction, we show that this fraction has to be larger than 1. We can prove this by considering the difference in magnitude of the numerator and the denominator using two observations. Firstly, $j \geq t+2$ implies that the numerator has more terms since $t-d+j-1$ is strictly greater than $2t-d+1$

Secondly, the binomial coefficient factors are also bigger. To show this, we firstly introduce some notation for brevity. We denote the distance between the two segments with $k := j-t-2$. We can use this to define three products which express the difference of the binomials in the sums in the numerator and the denominator : $\alpha := \prod_{\ell=1}^{k} (d-j+2i-k+\ell)$, $\beta := \prod_{\ell=1}^{k} (d-j+i-k+\ell)$ and $\gamma := \prod_{\ell=1}^{k} (d-j+i-k+\ell+1)$.

If we consider the factor $C(d-j+i,i)$ in the numerator, we can express this as

$$\alpha \left(\frac{(d-j+2i-k)!}{i!(d-j+i-k)!} \cdot \frac{1}{\beta} - \frac{(d-j+2i-k)!}{(i-1)!(d-j+i-k+1)!} \cdot \frac{1}{\gamma} \right).$$

Considering the terms of β and γ it is clear that β is strictly smaller than γ. This means that we can bound $\frac{1}{\gamma}$ with $\frac{1}{\beta}$ and therefore the whole factor is greater than $\frac{\alpha}{\beta} \cdot C(d-t+i-2,i)$. Similarly, the value of α is at least as much as the value of β which means the whole term is at least as big as the factor in the denominator $C(d-t+i-2,i)$.

Therefore, since there are more terms in the numerator and the factors in the numerator are bigger, the numerator of the fraction in Equation (4.2) has to be bigger than the denominator.

This especially means that the right hand side of Equation (4.2) is bigger than 1 which contradicts the fact that the left hand side can be at most 1. Therefore, the claim must be true which concludes the proof. $\square$

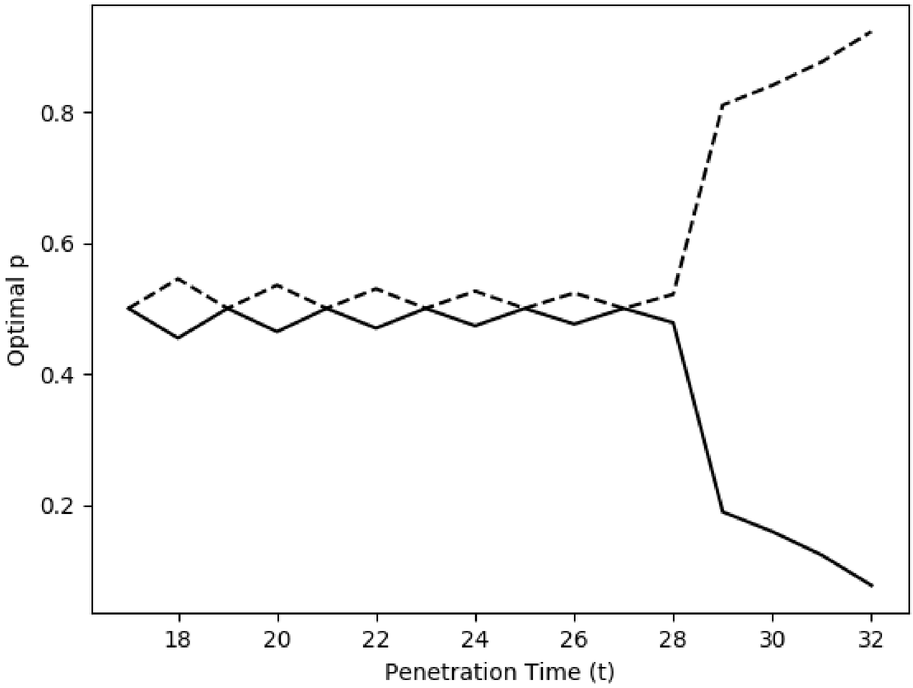


FIGURE 4.13: The optimal p for a circle of size $d = 34$ for all possible penetration times t ($17 \leq t \leq 32$). The solid and the dashed line show the one or two optimal p values.

Consequentially, as it suffices to consider the segments of type C as well as segment $d-t$ and $t+2$. Depending on t this might mean that significantly less than all d^2 pairs have to be considered.

Proposition 4.38. *The probability functions for* $2t - d + 3$ *segments have to be considered for finding an optimal* p.

Proof. By Theorem 4.36 of all L and R segments, only segment $d-t$ and $t+2$ can affect the lower envelope. Hence, we can exclude all the other segments of these two types. From the L segments we can exclude all $d - t - 2$ segments from segment $t + 3$ to segment d. Similarly, from the R segments we can exclude all $d - t - 1$ segments from segment 2 to segment $d - t - 1$. Subtracting these two from the total number of d segments yields the claim. □

4.4.2 Experimental Results

Lastly, we empirically show that the runtime of finding optimal strategies based on our probability functions is a fraction of the runtime which would be additionally needed for the Markov chain approach as well as giving a few insights into the optimal strategies and detection probabilities. In order to calculate the optimal strategies and detection probabilities we implemented an algorithm, similar to FINDP of Agmon et al. [1], in Python using Mathematica for efficient and accurate root-finding. The algorithm takes into consideration their results that the optimal value is a local maximum or an intersection of two functions on the lower envelope of the probability functions and that robots need to be synchronised and placed equidistant for optimality.

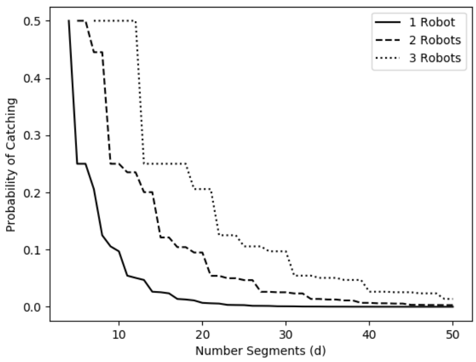


Figure 4.14: The probabilities of the robot detecting the adversary depending on the number of segments (d). The three curves show the probability for one, two or three robots patrolling a circle of up to 50 segments and a penetration time of $\frac{2}{3} \cdot d$.

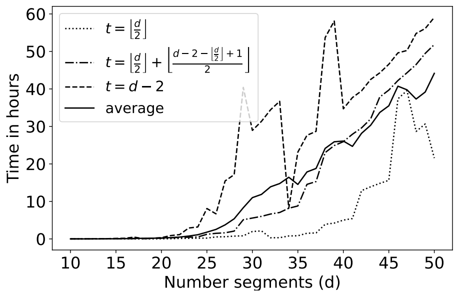


Figure 4.15: The runtime in hours for determining the probability functions for a robot with omnidirectional movement using the Markov chain approach.

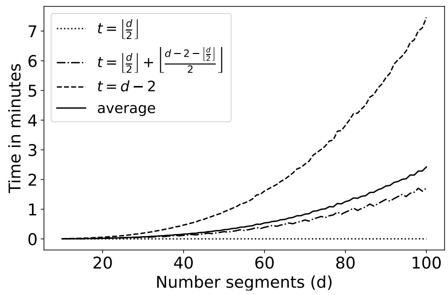


Figure 4.16: The runtime to calculate an optimal p for a robot with omnidirectional movement.

Additionally we note here two further specifications on the probability functions presented above which we require for the calculation. Firstly, a careful consideration of the functions shows that there are two edge cases at $p = 1$ and $p = 0$ which are, in general, mathematically not defined. However, for all edge cases it makes sense to define them to be the limit of the function which is always well defined in our case. Secondly, the functions are for one robot in segment 1 so for the case of multiple robots the respective functions have to be shifted for the different robots accordingly.

To compare the runtime of FindP using our explicit probability function we implemented the Markov chain approach algorithm FindFunc as described by Agmon et al. [1]. The execution of FindFunc provides the same probability functions as our explicit probability functions. We have used the execution time of FindFunc to set the remaining runtime of FindP into context. Previously the runtime for finding an optimal strategy would be the runtime of both algorithms. The runtimes were calculated by running the algorithms on the Iridis 5 Compute Cluster at the University of Southampton. Each execution of the respective algorithms ran on one core of an Intel Xeon E5-2670 processor situated in a node with 192 GB of DDR4 memory.

To give an overview of the runtime we have calculated the runtime for four different values of the penetration time t for increasing values of d on the circle with an omnidirectional robot. For each d we ran the algorithms for the smallest possible value of $\lfloor \frac{d}{2} \rfloor$, the biggest possible value of $d-2$, the middle value of $\lfloor \frac{d}{2} \rfloor + \left\lfloor \frac{d-2-\lfloor \frac{d}{2} \rfloor+1}{2} \right\rfloor$ as well as the average over all t values.

While the runtime of FindFunc is not as smooth as that of FindP it is clear that the runtime of FindP is several orders of magnitude smaller. The runtime of FindFunc as depicted in Figure 4.15, despite a number of outliers, increases polynomially where the rate of increase appears to diverge slowly. The average runtime reaches 44:09 hours and the minimum runtime is 21:33 hours at $d = 50$. In comparison, the runtime of FindP, as shown in Figure 4.16, is in the range of minutes rather than hours. For $d = 100$ the smallest t requires a time of 24 seconds and the middle t a time of 1:24. The maximal value of t clearly rises orders of magnitudes faster but is still only at 7:27 minutes at $d = 100$. Moreover, the runtime for the average value of t is closer to the runtime of the middle value of t and has a runtime of only 2.25 at $d = 100$.

Clearly, the runtime for smaller values of t benefits from the results of Section 4.4.1 resulting in runtimes that increase slower in comparison to the runtime at the maximal value of t. Moreover, another improvement factor for all values of t is that the runtime benefits from the fact that we can calculate the first and second derivative of the probability functions beforehand and thus those do not have to be calculated during the run. Altogether, by not having to execute FindFunc the runtime saving is in the range of several hours and the calculation of an optimal p can be done significantly faster.

Regarding optimal strategies and the probability of detection, we observe in Figure 4.14 that the highest possible probability of detecting the adversary is 0.5 and the probability decreases exponentially for increasing d. This is in line with previous research. For further insights, Figure 4.13 visualises a general pattern that the optimal p exhibits for all d and increasing t. An analysis of a larger range of d values up to 150 shows that for every d for increasing t there are either one or two optimal p values which tends towards 0.5 until $d - 2 - \lfloor \frac{d}{11} \rfloor$. At this point the two values diverge and tend towards 0 and 1. For smaller t values, every other optimal p, beginning with $\lfloor \frac{d}{2} \rfloor$, is 0.5. This suggests that for smaller penetration times the randomness is important whereas for relatively large t the best value tends towards the robot mostly walk in one direction. Moreover, in conjunction with the observations on the lower envelope in the beginning of Section 4.4 this indicates that only two probability functions have to be considered.

4.5 Conclusions

The presented results provide our contribution (see Section1.3) to the research area of determining optimal random strategies for multi-robot adversarial patrolling (Problem B). We provide results that address the research challenges (see Section 1.2) for closed and open polylines (perimeter and fence, respectively) with two different movement patterns. We show how lattice path techniques can be used to model and determine the number of possible paths and the probability of detecting the adversary for optimal random multi-robot adversarial patrolling strategies on a perimeter or a fence with two different movement patterns. The probabilities were previously determined using Markov chain based black box algorithms [1]. Our approach does not only reduce the runtime of the first step (determine the probability functions) but the analytic system of probability functions allows us to make observations allowing us to further improve the runtime of the second step (determining optimal strategies based on the probability functions). Using the example of an omnidirectional robot on the circle, we use the system of equations to show that two out of three sets of equations can be disregarded in the second step. Moreover, we empirically show the runtime reduction in hours and illustrate empirically an underlying structure of the probability functions and the change of the optimal parameter depending on the penetration time.

Chapter 5

Conclusions and Future Work

We explore two agent problems both with their own challenges within the common constraint of uncertainty. Specifically, we introduce the problem of distributing an uncertain homogeneous divisible resource among agents and we improve the runtime for multi-robot adversarial patrolling on polylines. We conclude this work with our contribution to the respective fields by linking our results back to the challenges stated in the introduction (see Section 1.2 in Chapter 1). We finish by highlighting interesting research that logically follows from our work or that has been opened up by our work.

5.1 Conclusions

The first problem we consider is a fair division variant where the amount of a homogeneous resource is uncertain. This uncertainty is reflected by a random variable over a finite set of discrete events. We show that, while the problem of optimising unconstrained efficiency can be solved in polynomial time for concave and other reasonable utility functions (Contribution A.I.), this is no longer the case if ex-ante envy-freeness is required (Contribution A.II.). In this case, an ex-ante envy-free allocation always exists but may have a significantly worse social welfare than the ex-ante efficient allocation (Research Challenge A.3). More specifically, the price of envy-freeness is tightly bounded by n for concave utility functions, where n is the number of agents (Contribution A.III.). Principally, we show that the problem of finding an ex-ante efficient allocation under ex-ante envy-freeness is strongly NP-hard, even under simple continuous utility functions and with uniform probability over the events (Contribution A.IV.). We devise an integer program for the optimal ex-ante envy-free solution for linear satiable utilities. Since the complexity depends on the number of agents, we focus on the case of two agents with linear satiable utility functions, in line with Research Challenge A.5 and Research Requirement 3. For the case of uniform probability distributions over the events, we devise a greedy algorithm that calculates an optimal solution in polynomial time (Contribution A.V.). This cannot be achieved for arbitrary probability distributions over the events since we show that this case is NP-hard as well (Contribution A.VI.). However, we show that the greedy algorithm sorting the events by their amount can be used to attain solutions that are structurally close to an optimal solution. Contrary to intuition, taking the probability distribution into consideration in the algorithm does not allow any general approximation guarantee. Nevertheless, we show empirically that both version of the algorithm perform well on average problem instances (Contribution A.VII. and Research Challenge A.5).

The second problem is multi-robot adversarial patrolling on closed and open polylines (perimeter and fence, respectively). Considering two different movement types on both graph types, we determine analytic forms for the probability of detecting the adversary (Contribution B.II.) We achieve this by using lattice path techniques to model and determine the number of possible paths a robot can walk (Contribution B.I.) Overall, this removes any runtime and space requirement that was previously required to determine the probabilities (Contribution B.III.) [1]. Hence, to determine optimal strategies we are only required to solve the system of equations given by the probability functions. Moreover, we illustrate an underlying structure of the probability functions and the change of the optimal parameter depending on the penetration time. For the closed polyline and robots that can freely walk left and right, we use the insights available from the analytic probability functions to improve the runtime of solving the system of equations. Firstly, we show that a significant number of probability functions can be ignored (Contribution B.IV.a)). Secondly, we use the analytic functions to show empirically that only two functions have to be considered, reducing the runtime to a constant runtime (Contribution B.IV.b)).

Together, the two problems highlight the difficulties and benefits of uncertainty. The allocation problem shows how taking uncertainty into consideration yields an interesting problem in an area of limited research consideration. Nevertheless, it clearly also highlights the difficulty of finding efficient solutions in settings of uncertainty. The patrolling problem shows how uncertainty can be used to find efficient strategies against an adversary. Our contributions highlights how some solutions are better at considering this uncertainty which contributes to finding improved solutions.

5.2 Future Work

Both problems highlight the importance of considering the uncertainty of problems as well as significant points when researching problems with uncertainty. Integrating uncertainty enriches the research and models problems closer to the real world. While difficult and with its own challenges it also opens possibilities to improve results. More problems should be considered under uncertainty. In addition to this general outlook, we conclude with more specific future research that follows on from this work.

The setting of allocating uncertain resources presented in this thesis invites various directions for future work. Firstly, we have provided algorithms for the case of two agents. These are not directly transferable to the general case of any number of agents. However, the increased heterogeneity of more agents has, as already seen in the case of linear satiable utilities (in comparison to the linear utilities case), the potential for further increases in social welfare [19]. Hence, our NP-hardness result for this case calls for a polynomial-time algorithm that approximates efficiency under ex-ante envy-freeness better than equal share. Secondly, we focused on linear and linear satiable utility. It would be interesting to investigate the possibilities and restrictions in other restricted classes of utility functions like piecewise linear utility functions. Thirdly, we have assumed that the utilities are known. While this makes sense in certain settings, it would be of interest to examine the strategic case where agents can misrepresent their utilities.

Likewise, the adversarial patrolling problem provides further directions. Specific to the investigated setting, we have observed empirically for instances up to 150 segments that the runtime can be reduced to a constant one. We conjecture that this is the case for all instances. While high degree polynomials allow no general algebraic solution [59], it would be interesting to analyse further patterns like monotonicity of local maxima to prove this conjecture. Moreover, it would be interesting to apply the technique of lattice path modelling to other structures and longer histories. More generally, these techniques have the potential to simplify the calculation, and gain further insight, of all similar scenarios [73, 74].

Bibliography

[1] Noa Agmon, Gal A. Kaminka and Sarit Kraus. 'Multi-robot adversarial patrolling: Facing a full-knowledge opponent'. In: *Journal of Artificial Intelligence Research* 42 (2011), pp. 887–916. ISSN: 10769757. DOI: 10.1613/jair.3365.

[2] Noa Agmon, Vladimir Sadov, Gal A. Kaminka and Sarit Kraus. 'The Impact of Adversarial Knowledge on Adversarial Planning in Perimeter Patrol'. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems*. Vol. 1. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2008, pp. 55–62. ISBN: 9780981738109. DOI: 10.5555/1402383.1402396.

[3] Steve Alpern, Thomas Lidbetter and Katerina Papadaki. 'Optimizing periodic patrols against short attacks on the line and other networks'. In: *European Journal of Operational Research* 273.3 (2019), pp. 1065–1073. ISSN: 03772217. DOI: 10.1016/j.ejor.2018.08.050.

[4] Steve Alpern, Alec Morton and Katerina Papadaki. 'Patrolling Games'. In: *Operations Research* 59.5 (2011), pp. 1246–1257. DOI: 10.1287/opre.1110.0983.

[5] Marcelo Wilson Furlan Matos Alves and Enzo Barberio Mariano. 'Climate justice and human development: A systematic literature review'. In: *Journal of Cleaner Production* 202 (2018), pp. 360–375. ISSN: 0959-6526. DOI: 10.1016/J.JCLEPRO.2018.08.091.

[6] Bo An, Milind Tambe and Arunesh Sinha. 'Stackelberg Security Games (SSG) Basics and Application Overview'. In: *Improving Homeland Security Decisions*. Ed. by Ali E. Abbas, Milind Tambe and Detlof von Winterfeldt. Cambridge: Cambridge University Press, 2017, pp. 485–507. DOI: 10.1017/9781316676714.021.

[7] Anthony B. Atkinson. 'On the Measurement of Inequality'. In: *Journal of Economic Theory* 2.3 (1970), pp. 244–263. ISSN: 10957235. DOI: 10.1016/0022-0531(70)90039-6.

[8] Yonatan Aumann and Yair Dombb. 'The Efficiency of Fair Division with Connected Pieces'. In: *ACM Trans. Econ. Comput.* 3.4 (2015), 23:1–23:16. ISSN: 2167-8375. DOI: 10.1145/2781776.

[9] Yonatan Aumann, Yair Dombb and Avinatan Hassidim. 'Auctioning Time: Truthful Auctions of Heterogeneous Divisible Goods'. In: *ACM Trans. Econ. Comput.* 4.1 (2016), 3:1–3:16. ISSN: 2167-8375. DOI: 10.1145/2833086.

[10] Yossi Azar, Joan Boyar, Lene M. Favrholdt, Kim S. Larsen and Morten N. Nielsen. 'Fair versus Unrestricted Bin Packing'. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 1851 (2000), pp. 200–213. ISSN: 16113349. DOI: 10.1007/3-540-44985-X.

[11] Haris Aziz and Simon Mackenzie. 'A Discrete and Bounded Envy-Free Cake Cutting Protocol for Any Number of Agents'. In: *CoRR* abs/1604.0 (2016), pp. 1–33.

[12] Haris Aziz and Simon Rey. 'Almost Group Envy-free Allocation of Indivisible Goods and Chores'. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. California: International Joint Conferences on Artificial Intelligence Organization, 2020, pp. 39–45. ISBN: 978-0-9992411-6-5. DOI: 10.24963/ijcai.2020/6.

[13] Siddharth Barman, Ganesh Ghalme, Shweta Jain, Pooja Kulkarni and Shivika Narang. 'Fair Division of Indivisible Goods Among Strategic Agents'. In: *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2019, pp. 1811–1813. ISBN: 9781450363099.

[14] Nicola Basilico, Giuseppe De Nittis and Nicola Gatti. 'Adversarial patrolling with spatially uncertain alarm signals'. In: *Artificial Intelligence* 246 (2017), pp. 220–257. ISSN: 00043702. DOI: 10.1016/j.artint.2017.02.007.

[15] Nicola Basilico, Nicola Gatti and Francesco Amigoni. 'Patrolling security games: Definition and algorithms for solving large instances with single patroller and single intruder'. In: *Artificial Intelligence* 184-185 (2012), pp. 78–123. ISSN: 00043702. DOI: 10.1016/j.artint.2012.03.003.

[16] Nicola Basilico, Nicola Gatti, Thomas Rossi, Sofia Ceppi and Francesco Amigoni. 'Extending Algorithms for Mobile Robot Patrolling in the Presence of Adversaries to More Realistic Settings'. In: *Proceedings - 2009 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2009*. Vol. 2. 2009, pp. 557–564. DOI: 10.1109/WI-IAT.2009.211.

[17] Xiaohui Bei, Ning Chen, Xia Hua, Biaoshuai Tao and Endong Yang. 'Optimal Proportional Cake Cutting with Connected Pieces'. In: *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*. Vol. 2. 2012, pp. 1263–1269. ISBN: 9781577355687.

[18] Dimitris Bertsimas, Vivek F. Farias and Nikolaos Trichakis. 'The Price of Fairness'. In: *Operations Research* 59.1 (2011), pp. 17–31. ISSN: 0030-364X. DOI: 10.1287/opre.1100.0865.

[19] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang and Ariel D. Procaccia. *Handbook of Computational Social Choice*. New York, NY, USA: Cambridge University Press, 2016, pp. 1–535. ISBN: 9781107446984. DOI: 10.1017/CBO9781107446984.

[20] Peter Brucker. *Scheduling Algorithms*. 5th ed. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2007. ISBN: 3642089070. DOI: 10.1007/978-3-540-69516-5.

[21] Sébastien Bubeck. 'Convex Optimization: Algorithms and Complexity'. In: *Foundations and Trends® in Machine Learning* 8.3-4 (2015), pp. 231–357. ISSN: 1935-8237. DOI: 10.1561/2200000050.

[22] Eric Budish. 'The Combinatorial Assignment Problem: Approximate Competitive Equilibrium from Equal Incomes'. In: *Journal of Political Economy* 119.6 (2011), pp. 1061–1103. DOI: 10.1086/664613.

[23] Jan Buermann, Enrico H. Gerding and Baharak Rastegar. 'Fair Allocation of Resources with Uncertain Availability'. In: *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*. Auckland, New Zealand: IFAAMAS, May 2020, 9 pages.

[24] Jan Buermann and Jie Zhang. 'Multi-Robot Adversarial Patrolling Strategies via Lattice Paths'. In: *Proc. of the 29th International Joint Conference on Artificial Intelligence (IJCAI 2020)*. Japan: IJCAI, July 2020, 7 pages.

[25] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos and Maria Kyropoulou. 'The Efficiency of Fair Division'. In: *Theory of Computing Systems* 50.4 (2012). Ed. by Stefano Leonardi, pp. 589–610. ISSN: 1433-0490. DOI: `10.1007/s00224-011-9359-y`.

[26] Yiling Chen, John K. Lai, David C. Parkes and Ariel D. Procaccia. 'Truth, Justice, and Cake Cutting'. In: *Games and Economic Behavior* 77.1 (2013), pp. 284–297. ISSN: 0899-8256. DOI: `10.1016/J.GEB.2012.10.009`.

[27] Yann Chevaleyre. 'Theoretical Analysis of the Multi-agent Patrolling Problem'. In: *Proceedingsof the IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2004*. 2004, pp. 302–308. ISBN: 0-7695-2101-0. DOI: `10.1109/IAT.2004.1342959`.

[28] George Christodoulou, Elias Koutsoupias and Annamária Kovács. 'Mechanism Design for Fractional Scheduling on Unrelated Machines'. In: *ACM Transactions on Algorithms* 6.2 (2010), 38:1–38:18. ISSN: 15496325. DOI: `10.1145/1721837.1721854`.

[29] R. Clark Robinson. *Introduction to Mathematical Optimization*. Evanston, Illinois: Department of Mathematics, Northwestern University, 2013, [Unpublished undergraduate etextbook].

[30] Julio B. Clempner. 'A continuous-time Markov Stackelberg security game approach for reasoning about real patrol strategies'. In: *International Journal of Control* 91.11 (2018), pp. 2494–2510. DOI: `10.1080/00207179.2017.1371853`.

[31] Vincent P. Crawford. 'Fair Division'. In: *The New Palgrave Dictionary of Economics*. 3rd. London: Palgrave Macmillan UK, 2018, pp. 4408–4410. ISBN: 978-1-349-95189-5. DOI: `10.1057/978-1-349-95189-5\_252`.

[32] Dariusz Dereniowski and Wiesław Kubiak. 'Shared processor scheduling of multiprocessor jobs'. In: *European Journal of Operational Research* 282.2 (2020), pp. 464–477. ISSN: 03772217. DOI: `10.1016/j.ejor.2019.09.033`.

[33] Jelena Diakonikolas, Maryam Fazel and Lorenzo Orecchia. 'Fair Packing and Covering on a Relative Scale'. In: *SIAM Journal on Optimization* 30.4 (2021), pp. 3284–3314. ISSN: 10526234. DOI: `10.1137/19M1288516`.

[34] Yehuda Elmaliach, Noa Agmon and Gal A. Kaminka. 'Multi-robot area patrol under frequency constraints'. In: *Annals of Mathematics and Artificial Intelligence* 57.3 (2009), pp. 293–320. ISSN: 1573-7470. DOI: `10.1007/s10472-010-9193-y`.

[35] Yehuda Elmaliach, Asaf Shiloni and Gal A. Kaminka. 'A Realistic Model of Frequency-Based Multi-Robot Polyline Patrolling'. In: *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS* 1 (2008), pp. 63–70. DOI: `10.1145/1402383.1402397`.

[36] Uriel Feige and Moshe Tennenholtz. 'On fair division of a homogeneous good'. In: *Games and Economic Behavior* 87 (2014), pp. 305–321. ISSN: 0899-8256. DOI: 10.1016/j.geb.2014.02.009.

[37] Amos Fiat and Ariel Levavi. 'Tight Lower Bounds on Envy-Free Makespan Approximation'. In: *Internet and Network Economics*. Ed. by Paul W. Goldberg. Vol. 7695 LNCS. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 553–558. ISBN: 978-3-642-35311-6. DOI: 10.1007/978-3-642-35311-6_46.

[38] Duncan K. Foley. 'Resource Allocation and the Public Sector'. In: *Yale Economic Essays* 7.1 (1967), pp. 45–98.

[39] Thibault Gajdos and Jean-Marc Tallon. 'Fairness under Uncertainty'. In: *Economics Bulletin* 4.18 (2002), pp. 1–7.

[40] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1979. ISBN: 0716710447.

[41] Yiannis Giannakopoulos, Elias Koutsoupias and Maria Kyropoulou. 'The Anarchy of Scheduling Without Money'. In: *Algorithmic Game Theory*. Ed. by Martin Gairing and Rahul Savani. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 302–314. ISBN: 978-3-662-53354-3. DOI: 10.1007/978-3-662-53354-3_24.

[42] I. P. Goulden and Luis G. Serrano. 'Maintaining the spirit of the reflection principle when the boundary has arbitrary integer slope'. In: *Journal of Combinatorial Theory, Series A* 104.2 (2003), pp. 317–326. ISSN: 0097-3165. DOI: 10.1016/j.jcta.2003.09.004.

[43] Li Huang, MengChu Zhou, Kuangrong Hao and Edwin Hou. 'A Survey of Multi-robot Regular and Adversarial Patrolling'. In: *IEEE/CAA Journal of Automatica Sinica* 6.4 (2019), pp. 894–903. ISSN: 2329-9266. DOI: 10.1109/JAS.2019.1911537.

[44] Manish Jain, Bo An and Milind Tambe. 'Security Games Applied to Real-World: Research Contributions and Challenges'. In: *Moving Target Defense II*. Ed. by Sushil Jajodia, Anup K. Ghosh, V.S. Subrahmanian, Vipin Swarup, Cliff Wang and X. Sean Wang. New York, NY, USA: Springer, New York, NY, 2013, pp. 15–39. ISBN: 978-1-4614-5416-8. DOI: 10.1007/978-1-4614-5416-8_2.

[45] Ramesh Johari and John N. Tsitsiklis. 'Efficiency Loss in a Network Resource Allocation Game'. In: *Mathematics of Operations Research* 29.3 (2004), pp. 407–435. DOI: 10.1287/moor.1040.0091.

[46] Ramesh Johari and John N. Tsitsiklis. 'Efficiency of Scalar-Parameterized Mechanisms'. In: *Operations Research* 57.4 (2009), pp. 823–839. DOI: 10.1287/opre.1080.0638.

[47] Frank Kelly. 'Charging and Rate Control for Elastic Traffic'. In: *European Transactions on Telecommunications* 8.1 (1997), pp. 33–37. DOI: 10.1002/ett.4460080106.

[48] Byung-Gook Kim, Shaolei Ren, Mihaela van der Schaar and Jang-Won Lee. 'Bidirectional Energy Trading and Residential Load Scheduling with Electric Vehicles in the Smart Grid'. In: *IEEE Journal on Selected Areas in Communications* 31.7 (2013), pp. 1219–1234. ISSN: 0733-8716. DOI: 10.1109/JSAC.2013.130706.

[49] Christian Krattenthaler. 'Lattice Path Enumeration'. In: *Handbook of Enumerative Combinatorics*. Ed. by Miklós Bóna. 1st ed. Discrete Mathematics and Its Applications. Boca Raton-London-New York: CRC Press, 2015. Chap. 10, pp. 589–678. ISBN: 9781482220858.

[50] Ilan Kremer and Kjell G. Nyborg. 'Divisible-good auctions: the role of allocation rules'. In: *RAND Journal of Economics* 35.1 (2004), pp. 147–159. ISSN: 07416261.

[51] Bertrand LeCun, Thierry Mautor, Franck Quessette and Marc Antoine Weisser. 'Bin packing with fragmentable items: Presentation and approximations'. In: *Theoretical Computer Science* 602 (2015), pp. 50–59. ISSN: 03043975. DOI: 10.1016/j.tcs.2015.08.005.

[52] Chaojie Li, Xinghuo Yu, Peter Sokolowski, Nian Liu and Guo Chen. 'A Stochastic Game for Energy Resource Trading in the Context of Energy Internet'. In: *2016 IEEE Power and Energy Society General Meeting (PESGM)*. IEEE, 2016, pp. 1–5. ISBN: 978-1-5090-4168-8. DOI: 10.1109/PESGM.2016.7741891.

[53] Leo Liberti. 'Reformulation Techniques in Mathematical Programming'. Thèse d'Habilitation à diriger des Recherches. Université de Paris IX - Dauphine, 2007.

[54] Christian List. 'Social Choice Theory'. In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Metaphysics Research Lab, Stanford University, 2013.

[55] LO3 Energy. *Brooklyn Microgrid*. 2018.

[56] Esther Mengelkamp, Benedikt Notheisen, Carolin Beer, David Dauer and Christof Weinhardt. 'A Blockchain-Based Smart Grid: Towards Sustainable Local Energy Markets'. In: *Computer Science - Research and Development* 33.1-2 (2018), pp. 207–214. ISSN: 1865-2034. DOI: 10.1007/s00450-017-0360-9.

[57] Met Office. *Clarity - Helping you understand the facts about weatherforecasting*. Tech. rep. 2009, p. 13.

[58] Ahuva Mu'Alem. 'On Multi-dimensional Envy-Free Mechanisms'. In: *Proceedings of the 1st International Conference on Algorithmic Decision Theory*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 120–131. ISBN: 978-3-642-04427-4. DOI: 10.1007/978-3-642-04428-1_11.

[59] Ferrante Neri. *Linear Algebra for Computational Sciences and Engineering*. Cham: Springer International Publishing, 2016. ISBN: 978-3-319-40341-0. DOI: 10.1007/978-3-319-40341-0_5.

[60] Noam Nisan, Tim Roughgarden, Éva Tardos and Vijay V. Vazirani. *Algorithmic Game Theory*. Ed. by Noam Nisan, Tim Roughgarden, Eva Tardos and Vijay V. Vazirani. Cambridge: Cambridge University Press, 2007. ISBN: 9780511800481. DOI: 10.1017/CBO9780511800481.

[61] Gabriele Oliva, Roberto Setola and Marco Tesei. 'A Stackelberg Game-Theoretical Approach to Maritime Counter-Piracy'. In: *IEEE Systems Journal* 13.1 (2019), pp. 982–993. ISSN: 1932-8184. DOI: 10.1109/JSYST.2018.2795892.

[62] Olabambo Ifeoluwa Oluwasuji, Obaid Malik, Jie Zhang and Sarvapali Ramchurn. 'Algorithms for Fair Load Shedding in Developing Countries'. In: *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2018, pp. 1590–1596. DOI: 10.24963/ijcai.2018/220.

[63] Dénes Pálvölgyi, Hans Peters and Dries Vermeulen. 'A strategic approach to multiple estate division problems'. In: *Games and Economic Behavior* 88 (2014), pp. 135–152. ISSN: 0899-8256. DOI: 10.1016/j.geb.2014.09.005.

[64] Georgios Piliouras, Evdokia Nikolova and Jeff S. Shamma. 'Risk Sensitivity of Price of Anarchy Under Uncertainty'. In: *ACM Trans. Econ. Comput.* 5.1 (2016), pp. 1–27. DOI: 10.1145/2930956.

[65] Matthias Pilz and Luluwah Al-Fagih. 'Game-Theoretic Approaches to Energy Trading: A Survey'. In: *CoRR* abs/1702.0 (2017), pp. 1–10.

[66] J. H. Reijnierse and J. A. M. Potters. 'On finding an envy-free Pareto-optimal division'. In: *Mathematical Programming* 83.1 (1998), pp. 291–311. ISSN: 1436-4646. DOI: 10.1007/BF02680564.

[67] Ariel Rosenfeld, Oleg Maksimov and Sarit Kraus. 'Optimal Cruiser-Drone Traffic Enforcement Under Energy Limitation'. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. Vol. 2018-July. International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 3848–3855. DOI: 10.24963/ijcai.2018/535.

[68] Tiago Sak, Jacques Wainer and Siome Klein Goldenstein. 'Probabilistic Multiagent Patrolling'. In: *Advances in Artificial Intelligence - SBIA 2008*. Ed. by Gerson Zaverucha and Augusto Loureiro da Costa. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 124–133. ISBN: 978-3-540-88190-2. DOI: 10.1007/978-3-540-88190-2_18.

[69] Vourchteang Sea, Ayumi Sugiyama and Toshiharu Sugawara. 'Frequency-Based Multiagent Patrolling Model and Its Area Partitioning Solution Method for Balanced Workload'. In: *Integration of Constraint Programming, Artificial Intelligence, and Operations Research*. Cham: Springer International Publishing, 2018, pp. 530–545. ISBN: 978-3-319-93031-2. DOI: 10.1007/978-3-319-93031-2_38.

[70] E. Segal-Halevi, A. Hassidim and Y. Aumann. 'Envy-free Cake-cutting in Two Dimensions'. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. Vol. 2. Austin, Texas: AAAI Press, 2015, pp. 1021–1028. ISBN: 0-262-51129-0.

[71] Erel Segal-Halevi. 'Fair multi-cake cutting'. In: *Discrete Applied Mathematics* 291 (2021), pp. 15–35. ISSN: 0166218X. DOI: 10.1016/j.dam.2020.10.011.

[72] Hadas Shachnai and Tami Tamir. 'On Two Class-Constrained Versions of the Multiple Knapsack Problem'. In: *Algorithmica* 29.3 (2001), pp. 442–467. ISSN: 1432-0541. DOI: 10.1007/s004530010057.

[73] Efrat Sless, Noa Agmon and Sarit Kraus. 'Multi-Robot Adversarial Patrolling: Facing Coordinated Attacks'. In: *13th International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2014*. Vol. 2. 2014, pp. 1093–1100. ISBN: 9781634391313.

[74] Efrat Sless Lin, Noa Agmon and Sarit Kraus. 'Multi-Robot Adversarial Patrolling: Handling Sequential Attacks'. In: *Artificial Intelligence* 274 (2019), pp. 1–25. ISSN: 0004-3702. DOI: 10.1016/j.artint.2019.02.004.

[75] Michael Z. Spivey. 'Enumerating Lattice Paths Touching or Crossing the Diagonal at a Given Number of Lattice Points'. In: *Electronic Journal of Combinatorics* 19.3 (2012), pp. 1–6. ISSN: 10778926.

[76] Richard P. Stanley. *Enumerative Combinatorics, Volume 1*. 2nd ed. New York, NY, USA: Cambridge University Press, 2011. ISBN: 1107602629.

[77] Koichi Tadenuma and William Thomson. 'The fair allocation of an indivisible good when monetary compensations are possible'. In: *Mathematical Social Sciences* 25.2 (1993), pp. 117–132. ISSN: 0165-4896. DOI: 10.1016/0165-4896(93)90047-M.

[78] Noga Talmor and Noa Agmon. 'On the Power and Limitations of Deception in Multi-Robot Adversarial Patrolling'. In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. 2017, pp. 430–436. DOI: 10.24963/ijcai.2017/61.

[79] The OEIS Foundation Inc. *On-Line Encyclopedia of Integer Sequences - Catalan's triangle*. 2020.

[80] A.G.R. Vaz, B. Elsinga, W.G.J.H.M. van Sark and M.C. Brito. 'An Artificial Neural Network to Assess the Impact of Neighbouring Photovoltaic Systems in Power Forecasting in Utrecht, the Netherlands'. In: *Renewable Energy* 85 (2016), pp. 631–641. ISSN: 0960-1481. DOI: 10.1016/J.RENENE.2015.06.061.

[81] David P. Williamson and David B. Shmoys. *The Design of Approximation Algorithms*. Cambridge: Cambridge University Press, 2011, p. 518. ISBN: 9780511921735. DOI: 10.1017/CBO9780511921735.

[82] Sichao Yang and Bruce Hajek. 'VCG-Kelly Mechanisms for Allocation of Divisible Goods: Adapting VCG Mechanisms to One-Dimensional Signals'. In: *IEEE Journal on Selected Areas in Communications* 25.6 (2007), pp. 1237–1243. ISSN: 0733-8716. DOI: 10.1109/JSAC.2007.070817.

Chapter A

Fair Allocation of Resources with Uncertain Availability

A.1 Price of Envy-freeness

We prove Lemma 3.6 by considering the optimisation problem representing the problem of maximising ex-ante social welfare (see Optimisation Problem 3) and showing that the maximal slope allocation satisfies the KKT conditions.

Optimisation Problem 3 Find the optimal resource allocation for utility functions $(v_{i,\omega})_{i\in[n]} \in \mathcal{F}^n$ with $n \in \mathbb{N}$ for an amount of resource of $\omega \in \mathbb{R}$.

$$\max \quad \sum_{i\in[n]} v_i(x_i) \tag{1}$$

$$\text{s.t.} \quad \sum_{i\in[n]} x_i \leq \omega \tag{2}$$

$$x_i \geq 0 \qquad \forall i \in [n] \tag{3}$$

Lemma A.1. *A maximal slope allocation is efficient and has an ex-ante social welfare of* $c_j \cdot \mathrm{E}[X]$ *for* $j \in \arg\max_{i\in[n]}\{c_i\}$.

Proof. We will prove that a maximal slope allocation satisfies the KKT conditions for Problem 3. For that, let $\omega \in \Omega$ be arbitrary but fixed, $f(x) \coloneqq \sum_{i\in[n]} v_i(x_i)$, $g(x) \coloneqq \sum_{i\in[n]} x_i$ and $h_i(x) \coloneqq -x_i$ for all $i \in [n]$. Further, let $(\lambda, \mu_i, \dots, \mu_n)$ be the KKT multiplier for g and h_i, respectively.

We assume without loss of generality that the slope of agent 1's utility is maximal among the slopes of all agent's utilities, i.e. $1 \in \arg\max_{i\in[n]}\{c_i\}$, and thus $x_1 = \omega$. Then it is clear that the other agents cannot get any resource or in other words from Condition 2 it must hold that $x_i = 0$ for all $i \in [n]\setminus\{1\}$. Based on that, the complementary slackness for h_1 gives us $\mu_1 = 0$ since $h_1(x_1) = -\omega$.

If we look at the stationarity condition we must have $0_n = Df(x) - \lambda \cdot Dg(x) - \sum_i \mu_i \cdot Dh_i(x) = (c_1, \ldots, c_n) - \lambda \cdot (1, \ldots, 1) - (-\mu_1, \ldots, -\mu_n)$. This implies $\lambda = c_1$ and thus $\mu_i = c_1 - c_i$ for all $i \in [n] \backslash \{1\}$. Since $x_i = 0$ and $\mu_i = c_1 - c_i$ this satisfies complementary slackness for h_i for all $i \in [n] \backslash \{1\}$. Further, we have for g that $\lambda(\omega - g(x)) = c_1(\omega - \omega) = 0$ which therefore also satisfies complementary slackness. Finally, if we look at all KKT multiplier $(\lambda, \mu_i, \ldots, \mu_n) = (c_1, 0, c_1 - c_2, \ldots, c_1 - c_n)$ they must be at least zero since c_1 is of maximal slope and hence the solution is a maximiser.

We can conclude the overall efficiency from Lemma 3.1.

For the social welfare let A be the allocation with $a_j(\omega) = \omega$ and $a_i(\omega) = 0$ for all $i \in [n] \backslash \{1\}$ which is valid by definition. Then, a simple manipulation gives us the following. $W(A) = \sum_{i \in [n]} \sum_{\omega \in \Omega} v_i(a_i(\omega)) f(\omega) = \sum_{\omega \in \Omega} c_j \cdot \omega \cdot f(\omega) = c_j \cdot \mathrm{E}[X]$ □

On the other hand we have that equal share provides maximal social welfare under ex-ante envy-freeness. We show in this addition to the proof of Lemma 3.7 that the maximal achievable social welfare is indeed reached by the equal share allocation.

Lemma A.2. *No ex-ante envy-free allocation can have more ex-ante social welfare than* $\mathrm{mean}_{i \in [n]} \{c_i\} \cdot \mathrm{E}[X]$ *which is matched by the equal share allocation (where* mean *is the arithmetic mean, see Nomenclature for definition).*

Addition to Proof of Lemma 3.7. Let A_{ES} be the equal share allocation, i.e. $a_i(\omega) = \frac{\omega}{n}$ for all $i \in \mathbb{N}$. Similar to before, a rearrangement of the social welfare formula gives us the claim. $W(A) = \sum_{i \in [n]} \sum_{\omega \in \Omega} c_i \cdot \frac{\omega}{n} \cdot f(\omega) = \sum_{i \in [n]} \frac{c_i}{n} \cdot \sum_{\omega \in \Omega} \omega \cdot f(\omega) = \mathrm{mean}_{i \in [n]} \{c_i\} \cdot \mathrm{E}[X]$ □

Theorem 3.9. *For concave utility functions, the division of a homogeneous resource has a price of envy-freeness of at most* n.

Proof. Let $i \in [n]$ be arbitrary but fixed. We use concavity to bound the difference between the ex-ante efficient allocation without the restriction of ex-ante envy-freeness $A_E = (a_i)_{i \in [n]}$ and the equal share solution A_{ES} as a lower bound on the ex-ante efficient solution under ex-ante envy-freeness. Starting with the unrestricted allocation A_E, like any allocation this allocation is bound by the utility of the event amounts by the monotonicity of the utility functions: $W(A_E) = \sum_{i \in [n]} \sum_{\omega \in \Omega} v_i(a_i(\omega)) f(\omega) \leq \sum_{i \in [n]} \sum_{\omega \in \Omega} v_i(\omega) f(\omega)$.

However, by concavity any utility of an event's amount is bound by n times the amount of equally sharing this event.

$$\begin{aligned} v_i(\omega) &= n \cdot \left(\left(1 - \frac{1}{n} \right) \cdot v_i(0) + \frac{1}{n} \cdot v_i(\omega) \right) \\ &\leq n \cdot \left(v_i \left(\left(1 - \frac{1}{n} \right) \cdot 0 + \frac{1}{n} \cdot \omega \right) \right) \\ &= n \cdot v_i \left(\frac{\omega}{n} \right) \end{aligned}$$

Hence, using this we can further bound A_E as $W(A_E) \leq \sum_{i \in [n]} \sum_{\omega \in \Omega} n \cdot v_i\left(\frac{\omega}{n}\right) f(\omega)$. Finally, this is the same as n times the equal share allocations' social welfare, i.e. $n \cdot W(A_{ES})$, which is what we claimed. □

A.2 Complexity

A.2.1 Time and Size of Construction

The following two lemmata prove that the complexity proof of the problem of maximising ex-ante social welfare under ex-ante envy-freeness for any number of agents (see Section 3.4) preserves the strong NP-completeness of 3-Partition. Firstly, the envy partition instance (see Definition 3.13) can be constructed in polynomial time from a 3-partition problem instance (see Definition 3.12).

Lemma A.3. *An envy partition instance can be constructed in polynomial time.*

Proof. The agent's utility functions can be represented by two values. For that, we have to calculate the values of the $4m$ saturation amounts which depend on the summation of the $3m$ 3-partition elements y. Hence, for every utility function a constant number of multiplication or division with the 3-partition elements are necessary. Furthermore, a value for ε has to be chosen where the bound depends on m and y and there are m events. Therefore, the values of both of them can be calculated in a constant number of operations. Additionally, the random variable is uniform distributed and has therefore only one division. Altogether, the calculation is polynomial in the 3-partition instance. □

Secondly, the size of the instance is not polynomially of comparable size and the maximal value is polynomially bound.

Lemma A.4. *The length of an envy partition instance is polynomially not smaller than the respective 3-partition instance and the maximal value is polynomially bound by the maximal value and the length of the 3-partition instance.*

Proof. The construction has m events and $4m$ utility functions which are polynomial in the $3m$ values of 3-partition. Therefore, the instance is not smaller than the 3-partition instance. Furthermore, the biggest value is $4m \cdot \hat{s}_{4m}$ which can be expressed as a polynomial of the $3m$ values and the biggest integer of 3-partition s_{4m}. Hence, the value is polynomial bounded by the 3-partition instance and maximal value. □

A.2.2 Equivalence and Representation

The following four lemmata are the technical formulations and proofs of the statements or ideas mentioned in Section 3.4 which showing the equivalence of representations.

The proof that allocations can be represented as total amount representations (Lemma 3.17) uses the following shift of allocation.

Definition A.5 (Total Amount Representation Shift)**.** A total amount representation shift from allocation A to A' takes an agent $i \in [4m]$ who has two events where the allocation is below the saturation amount, i.e. $0 < a_i(\bar{\omega}) < \hat{s}_i$ and $0 < a_i(\tilde{\omega}) < \hat{s}_i$, and shifts the allocation from one event to the other. Without loss of generality $a_i(\bar{\omega}) \geq a_i(\tilde{\omega})$. For the construction of A' there are two mutually exclusive cases:

1. In the first case, the sum of both allocated amounts greater than the saturation amount, i.e. $a_i(\tilde{\omega}) + a_i(\bar{\omega}) > \hat{s}_i$. In this case we construct the new allocation by setting $a'_i(\bar{\omega}) = \hat{s}_i$ and $a'_i(\tilde{\omega}) = a_i(\tilde{\omega}) - (\hat{s}_i - a_i(\bar{\omega}))$.

2. The second case is the negate of the first and we construct the allocation by setting $a'_i(\bar{\omega}) = a_i(\bar{\omega}) + a_i(\tilde{\omega})$ and $a'_i(\tilde{\omega}) = 0$.

Using this we can prove Lemma 3.17.

Lemma A.6. *The ex-ante social welfare of an allocation and its representing total amount allocation are equivalent.*

Proof. We prove the claim by showing that a single total amount representation shift (see Definition A.5) preserves utilities and therefore expected utility and ex-ante social welfare.

The shift does not change the utility in the first case since,

$$\begin{aligned}\frac{1}{m} \cdot \sum_{\omega \in \Omega} v_i(a'_i(\omega)) &= \frac{1}{m} \cdot \sum_{\omega \in \Omega \setminus \{\bar{\omega}, \tilde{\omega}\}} v_i(a_i(\omega)) + v_i(a'_i(\bar{\omega})) + v_i(a'_i(\tilde{\omega})) \\ &= \frac{1}{m} \cdot \sum_{\omega \in \Omega \setminus \{\bar{\omega}, \tilde{\omega}\}} v_i(a_i(\omega)) + v_i(a_i(\bar{\omega})) + v_i(a_i(\tilde{\omega})) \\ &= \frac{1}{m} \cdot \sum_{\omega \in \Omega} v_i(a_i(\omega)).\end{aligned}$$

Neither does it change the utility in the second case since,

$$\begin{aligned}\frac{1}{m} \cdot \sum_{\omega \in \Omega} v_i(a'_i(\omega)) &= \frac{1}{m} \cdot \sum_{\omega \in \Omega \setminus \{\bar{\omega}, \tilde{\omega}\}} v_i(a_i(\omega)) + v_i(a'_i(\bar{\omega})) + v_i(a'_i(\tilde{\omega})) \\ &= \frac{1}{m} \cdot \sum_{\omega \in \Omega \setminus \{\bar{\omega}, \tilde{\omega}\}} v_i(a_i(\omega)) + v_i(a_i(\bar{\omega}) + a_i(\tilde{\omega})) + v_i(0) \\ &= \frac{1}{m} \cdot \sum_{\omega \in \Omega \setminus \{\bar{\omega}, \tilde{\omega}\}} v_i(a_i(\omega)) + i \cdot a_i(\bar{\omega}) + i \cdot a_i(\tilde{\omega}) + 0 \\ &= \frac{1}{m} \cdot \sum_{\omega \in \Omega} v_i(a_i(\omega)).\end{aligned}$$

Hence, one shift preserves ex-ante social welfare and if no further shift is possible the allocation must be in total amount representation. □

Besides preserving social welfare a total amount representation shift also has to preserve envy-freeness as stated by Lemma 3.18.

Lemma A.7. *If the a_i^p are ex-ante envy-free then the total amount allocation is ex-ante envy-free. Moreover, if the a_i^p are ex-ante envy-free and a valid Φ exists then $(a_i(\Phi(i,\omega)))_{i \in [n], \omega \in \Omega}$ is ex-ante envy-free.*

Proof. In order to prove the necessity of the envy-freeness we assume for sake of contradiction that the total amount representation is not ex-ante envy-free. However, before we use that, we first show that the utility of agent i's allocation measured by another agent j decreases by a total amount representation shift (see Definition A.5). This means that the difference in the ex-ante envy-free allocations increases by considering the total amount representation. We will show that this is true for both cases.

1. For the first case we consider two sub-cases

   a) $\hat{s}_j \geq a_i(\tilde{\omega}) - (\hat{s}_i - a_i(\bar{\omega}))$,

   b) $\hat{s}_j < a_i(\tilde{\omega}) - (\hat{s}_i - a_i(\bar{\omega}))$,

   based on the value of the saturation amount of agent j with respect to the value of the shift.

1.a) If the saturation amount of agent j is at least of that value, i.e. $\hat{s}_j \geq a_i(\tilde{\omega}) - (\hat{s}_i - a_i(\bar{\omega}))$, then the following holds.

$$\begin{aligned}
&\sum_{\omega\in\Omega} v_j(a'_i(\omega)) \\
&= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + v_j(a'_i(\bar{\omega})) + v_j(a'_i(\tilde{\omega})) \\
&= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + j\cdot\hat{s}_i^j + j\cdot a_i(\tilde{\omega}) - j(\hat{s}_j - a_i(\bar{\omega}))
\end{aligned}$$

Then, by definition we know that $\hat{s}_i^j \leq \hat{s}_j$ and adding a zero gives us $j\cdot\hat{s}_i^j \leq j\cdot a_i(\bar{\omega}) + j(\hat{s}_j - a_i(\bar{\omega}))$. Using this gives us:

$$\begin{aligned}
\sum_{\omega\in\Omega} v_j(a'_i(\omega)) &\leq \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + j\cdot a_i(\bar{\omega}) + j\cdot a_i(\tilde{\omega}) \\
&\quad + j(\hat{s}_j - a_i(\bar{\omega})) - j(\hat{s}_j - a_i(\bar{\omega})) \\
&= \sum_{\omega\in\Omega} v_j(a_i(\omega)).
\end{aligned}$$

1.b) Contrarily, if the opposite would hold, i.e. $\hat{s}_j < a_i(\tilde{\omega}) - (\hat{s}_i - a_i(\bar{\omega}))$, then we know that the saturation amount of agent j is less than the one of agent i:

$$\hat{s}_j < a_i(\tilde{\omega}) - (\hat{s}_i - a_i(\bar{\omega})) \leq 2\cdot\hat{s}_i - \hat{s}_i = \hat{s}_i.$$

Hence, $\hat{s}_j$ is smaller than the allocation of agent i in both events and for both allocations A and A' which means agent j values them the same:

$$\begin{aligned}
\sum_{\omega\in\Omega} v_j(a'_i(\omega)) &= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + v_j(a'_i(\bar{\omega})) + v_j(a'_i(\tilde{\omega})) \\
&= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + j\cdot\hat{s}_j + j\cdot\hat{s}_j \\
&= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + j\cdot a_i(\bar{\omega}) + j\cdot a_i(\tilde{\omega}) \\
&= \sum_{\omega\in\Omega} v_j(a_i(\omega)).
\end{aligned}$$

2. For the second case agent j has at most the same value from the combined allocation in comparison to the initial one. Again, this means that the utility can only decrease from

the perspective of agent j. In the following more formally.

$$\begin{aligned}\sum_{\omega\in\Omega} v_j(a_i'(\omega)) &= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + v_j(a_i'(\bar{\omega})) + v_j(a_i'(\tilde{\omega}))\\ &= \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + v_j(a_i(\bar{\omega}) + a_i(\tilde{\omega})) + v_j(0)\\ &\le \sum_{\omega\in\Omega\setminus\{\bar{\omega},\tilde{\omega}\}} v_j(a_i(\omega)) + j\cdot a_i(\bar{\omega}) + j\cdot a_i(\tilde{\omega}) + 0\\ &= \sum_{\omega\in\Omega} v_j(a_i(\omega))\end{aligned}$$

Now, we can use this and the assumption $V_i'(A_i) < V_i'(A_j)$ to show that ex-ante envy-freeness must be violated. For the ex-ante envy-freeness of agent i we have

$$\begin{aligned}&\frac{1}{m}\cdot\sum_{\omega\in\Omega} v_i(a_i(\omega)) = \frac{1}{m}\cdot\sum_{\omega\in\Omega} i\cdot a_i(\omega) = \frac{1}{m}\cdot(n_i\cdot i\cdot\hat{s}_i + i\cdot d_i) = V_i'(A_i)\\ &< V_i'(A_j) = \frac{1}{m}\cdot(n_j\cdot i\cdot\hat{s}_j^i + i\cdot d_j^i) \le \frac{1}{m}\cdot\sum_{\omega\in\Omega} i\cdot a_j^i(\omega) = \frac{1}{m}\cdot\sum_{\omega\in\Omega} v_i(a_j(\omega)),\end{aligned}$$

where the second inequality comes from above arguments that the shifting can only reduce the utility which means that under the total amount representation the utility cannot be higher. This clearly contradicts the assumption and the claim follows. ☐

A.2.3 Transformation of Allocations

The following four lemmata are the technical formulations and proofs of the statements or ideas mentioned in Section 3.4 which are required for the shifting procedure.

Lemma 3.19 states that shifting allocation to agents with a higher index can only increase social welfare.

Lemma A.8. *Fixing an allocation A and an agent $1 \le k < 4m$, if all agents i where $i < k$ have a total amount allocation of $A_i = \hat{s}_i$ and agent k's total amount allocation is $A_k > \hat{s}_k$, then shifting allocation to create allocation A' such that any excess is shifted to the next agent, i.e. $A_k' = \hat{s}_k$ and $A_{k+1}' = A_{k+1} + A_k - \hat{s}_k$, increases social welfare.*

Proof. We consider an arbitrary but fixed agent $k \in [4m-1]$. The allocation of the statement means that we have $A_k = n_k\cdot\hat{s}_k + d_k$ and $A_{k+1} = n_{k+1}\cdot\hat{s}_{k+1} + d_{k+1}$. Then shifting allocation from agent k to agent $k+1$ accordingly means that agent $k+1$ has an allocation of $A_{k+1}' = n_{k+1}\cdot\hat{s}_{k+1} + d_{k+1} + (n_k-1)\cdot\hat{s}_k + d_k$. Since agent $k+1$ has at least the same saturation amount as agent k but a higher utility means that this shift increases social welfare. Mostly rearranging the ex-ante social welfare equation shows exactly that in the following.

$$
\begin{aligned}
& W(A') \\
&= W(A) - k \cdot (n_k \cdot \hat{s}_k + d_k) + k \cdot \hat{s}_k + (k+1) \cdot (n_k - 1) \cdot \hat{s}_k + (k+1) \cdot d_k \\
&= W(A) - k \cdot (n_k \cdot \hat{s}_k + d_k) + k \cdot \hat{s}_k \\
&\quad + (k+1) n_k \cdot \hat{s}_k + (k+1) \cdot d_k - (k+1) \cdot \hat{s}_k \\
&= W(A) + n_k \cdot \hat{s}_k + d_k - \hat{s}_k \\
&> W(A) + \hat{s}_k - \hat{s}_k \\
&= W(A)
\end{aligned}
$$

□

The following lemma is supplementary for Lemma 3.20 and states that the amount of times agents can have his or her saturation amount allocated, decreases with increasing saturation amount. This comes from envy-freeness since every time an agent of higher saturation amount has an allocation in some event, so must all the agents of lower saturation amounts.

Lemma A.9. *The number of saturation amounts in the context of the total amount representation decreases with increasing agent index, i.e.* $n_1 \geq n_2 \geq \ldots \geq n_{4m}$

Proof. We prove this by contradiction by assuming that there are two agents $i, j \in [4m]$ with $j < i$ for which this does not apply, i.e. $n_i \geq n_j + 1$. Then, using envy-freeness and the assumption, $n_i \geq n_j + 1 \Leftrightarrow n_i - n_j \geq 1$, the value of d_j can be lower bound by the saturation amount:

$$
\begin{aligned}
d_j = n_j \cdot \hat{s}_j + d_j - n_j \cdot \hat{s}_j &= \sum_{\omega \in \Omega} v_j(a_j(\omega)) - n_j \cdot \hat{s}_j \geq \sum_{\omega \in \Omega} v_j(a_i(\omega)) - n_j \cdot \hat{s}_j \\
&= n_i \cdot \hat{s}_j + d_i^j - n_j \cdot \hat{s}_j = \hat{s}_j \cdot (n_i - n_j) + d_i^j \geq \hat{s}_j + d_i^j \geq \hat{s}_j .
\end{aligned}
$$

This would mean that the allocation has one saturation amount more which contradicts the amount of allocation of agent j. Hence, the claim is true for agents i and j and therefore for all agents. □

More general, the same is true for the entire allocation in the total amount allocation.

Lemma A.10. *For two agents $i, k \in [4m]$ with $i > k$, if agent k has a total amount allocation of less than, or equal to, their saturation amount, i.e. $A_k \leq \hat{s}_k$, then the same must be true for agent i, i.e. $A_i \leq \hat{s}_i$. The same holds for the strict case, i.e. if $A_k < \hat{s}_k$ then $A_i < \hat{s}_i$.*

Proof. We consider this in two cases based on if the allocation for agent k is exactly the allocation amount or less than that.

1. $A_k = \hat{s}_k$ or $n_k = 1$
2. $A_k = d_k$ or $n_k = 0$

In the first case, we know by Lemma A.9 that the allocation of agent i has to satisfy $n_i \leq 1$. If it is smaller the claim follows immediately since $0 \leq d_i < \hat{s_i}$. If $n_i = 1$ would hold and we assume for contradiction that agent i has a higher allocation, i.e. $d_i > 0$ then we would have

$$\hat{s}_k = \frac{m}{k} \cdot \left(\frac{1}{m} \cdot v_k(\hat{s}_k) \right) \geq \frac{m}{k} \cdot \left(\frac{1}{m} \cdot (v_k(\hat{s}_i) + v_k(d_i)) \right) = \hat{s}_k + d_i^k > \hat{s}_k.$$

This is clearly a contradiction to what Lemma A.9 implied.

In the second case, Lemma A.9 implies $n_j = 0$ which again immediately gives the claim since $0 \leq d_j < \hat{s_j}$.

Altogether, the second claim with the strict inequality follows from the second case and the first claim follows from both cases. □

A.3 Correctness of Algorithm for Two Agents Linear Satiable Utility Functions

We show in this section that the algorithm GreedyTake (Algorithm 1) is correct for arbitrary probability distributions, i.e. we show that the allocation produced by the algorithm is valid and ex-ante envy-free (Lemma 3.36).

Lemma A.11. *An allocation determined by* GreedyTake *is valid and ex-ante envy-free.*

Proof. We prove this statement essentially by showing that, for all possible cases defined by the algorithm (especially line 4 and line 12), agent 2 is just envy-free which implies by Lemma 3.33 that both agents are ex-ante envy-free. Before we start we observe that the allocation in line 2 and 3 as well as 6 and 7 respects the available amount. Hence, we can focus on the validity of the allocation of event ω_{j^*} and the envy-freeness.

The algorithm has two different possible allocations (line 13 and 15) which are determined by the if statement in line 12. For both cases we show the validity and the just envy-freeness of agent 2. The envy-freeness of agent 1 follows (Lemma 3.33). Moreover, since agent 1 gets the remaining amount of event ω_{j^*} we can focus on the validity of the allocation to agent 2.

Before we consider the first case we observe that since the loop ended at j^* we have that

$$\min\{\omega_{j^*}, q_2\} \cdot f(\omega_{j^*}) \geq A + \min\{\omega_{j^*} - \min\{\omega_{j^*}, q_2\}, q_2\} \cdot f(\omega_{j^*}), \tag{4}$$

which is simply the fact that agent 2 is envy-free (the opposite of the loops statement) rewritten using A (agent 2's envy without the split event).

We start with the case that the if statement in line 12 is true, i.e. agent 2' allocation is set according to line 13. Using Equation (4) the allocation from line 13 can be bound as

$$\begin{aligned}\frac{A + B}{2 \cdot f(\omega_{j^*})} &= \frac{A + \min\{\omega_{j^*}, q_2\} \cdot f(\omega_{j^*}) + \min\{\omega_{j^*} - \min\{\omega_{j^*}, q_2\}, q_2\} \cdot f(\omega_{j^*})}{2 \cdot f(\omega_{j^*})} \\ &\overset{Eq. (4)}{\leq} \frac{2 \cdot \min\{\omega_{j^*}, q_2\} \cdot f(\omega_{j^*})}{2 \cdot f(\omega_{j^*})},\end{aligned}$$

which means the allocation is valid, i.e.

$$\frac{A+B}{2\cdot f(\omega_{j^*})} \leq \min\{\omega_{j^*}, q_2\}. \tag{5}$$

For the envy-freeness, we start with the easier case that the while loop ended with an equality of the equation in line 4. In this case Equation (4) becomes an equality which in turn means that Equation (5) is an equality, i.e. $\frac{A+B}{2\cdot f(\omega_{j^*})} = \min\{\omega_{j^*}, q_2\}$. Hence, the allocation of line 13 does not change the allocation (compare to line 6) which implies that agent 2 is just envy-free.

Contrarily, if the equation in line 4 is an inequality, i.e. Equation (4) and (5) are strict, we consider agent 2's utility of the allocations of the two agents, subject to agent 2's saturation amount. Since, agent 2's marginal utility ($\frac{u_2}{q_2}$) is applied to both sides we can restrict ourselves to consider the allocated ex-ante amount. The ex-ante amount of agent 2's own allocation is

$$\begin{aligned}
&\sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_2(\omega)\cdot f(\omega) + \frac{A+B}{2\cdot f(\omega_{j^*})}\cdot f(\omega_{j^*})\\
=&\frac{\sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_2(\omega)\cdot f(\omega) + \sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_1^2(\omega)\cdot f(\omega) + B}{2}\\
&+\frac{\sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_1^2(\omega)\cdot f(\omega) - \sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_1^2(\omega)\cdot f(\omega)}{2}\\
=&\sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_1^2(\omega)\cdot f(\omega) + \frac{-A+B}{2\cdot f(\omega_{j^*})}\cdot f(\omega_{j^*}),
\end{aligned}$$

and agent 2's ex-ante amount of agent 1's allocation is

$$\sum_{\omega\in\Omega\setminus\{\omega_{j^*}\}} a_1^2(\omega)\cdot f(\omega) + \min\left\{\omega_{j^*} - \frac{A+B}{2\cdot f(\omega_{j^*})}, q_2\right\}\cdot f(\omega_{j^*}).$$

Hence, we have to show that

$$\min\left\{\omega_{j^*} - \frac{A+B}{2\cdot f(\omega_{j^*})}, q_2\right\} = \frac{-A+B}{2\cdot f(\omega_{j^*})}. \tag{6}$$

This follows from the fact that the amount of ω_{j^*} and by that the amount of B is limited. We prove this by considering the value of B and a bound on B implied by this case. In more detail, the amount of B is bounded by the inequality of the if statement (line 12). Rewriting the expression in line 12 using A and B yields

$$B - q_2\cdot f(\omega_{j^*}) \leq A + q_2\cdot f(\omega_{j^*}) \tag{7}$$

which implies that $\frac{-A+B}{2\cdot f(\omega_{j^*})} \leq q_2$. Hence, B can be bounded as

$$\frac{B}{f(\omega_{j^*})} = \frac{-A+B}{2\cdot f(\omega_{j^*})} + \frac{A+B}{2\cdot f(\omega_{j^*})} < 2q_2 \tag{8}$$

where the second part comes from the strict version of Equation (5). Additionally, we can observe the general value of B (see line 10).

$$\frac{B}{f(\omega_{j^*})} = \min\{\omega_{j^*}, q_2\} + \min\{\omega_{j^*} - \min\{\omega_{j^*}, q_2\}\}, q_2\} = \begin{cases} 2q_2 & \text{if } \omega_{j^*} \geq 2q_2 \\ \omega_{j^*} & \text{otherwise} \end{cases} \quad (9)$$

Hence, considering Equation (8) it must hold that $B = \omega_{j^*} \cdot f(\omega_{j^*})$ which means that for the first expression in the minimum in Equation (6) the quality of Equation (6) holds, i.e. $\omega_{j^*} - \frac{A+B}{2 \cdot f(\omega_{j^*})} = \frac{B}{f(\omega_{j^*})} - \frac{A+B}{2 \cdot f(\omega_{j^*})} = \frac{-A+B}{2 \cdot f(\omega_{j^*})}$. Moreover, since we observed that $\frac{-A+B}{2 \cdot f(\omega_{j^*})}$ is smaller than or equal to q_2 the whole claim of Equation (6) and therefore the just envy-freeness claim of agent 2 follows.

Following the first case we assume for the second case that the if statement (line 12) is false or

$$B - q_2 \cdot f(\omega_{j^*}) > A + q_2 \cdot f(\omega_{j^*}) \quad (10)$$

(opposite of Equations 7). In this case the allocation to agent 2 is $a_2(\omega_{j^*}) = \frac{A}{f(\omega_{j^*})} + q_2$ (line 15).

The validity follows immediately from Equation (10) and the possible values of B (Equation (9)). Equation (10) is equivalent to $A < B - 2q_2 \cdot f(\omega_{j^*})$ and by Equation (9) we know that $B \leq 2q_2 \cdot f(\omega_{j^*})$. Hence, the two equations imply $A < 0$ which means $\frac{A}{f(\omega_{j^*})} + q_2 \leq q_2$.

Regarding envy-freeness, the ex-ante amount of the second agent equals to

$$\begin{aligned}
&\sum_{\omega \in \Omega \setminus \{\omega_{j^*}\}} a_2(\omega) \cdot f(\omega) + \left(\frac{A}{f(\omega_{j^*})} + q_2 \right) \cdot f(\omega_{j^*}) \\
&= \sum_{\omega \in \Omega \setminus \{\omega_{j^*}\}} a_2(\omega) \cdot f(\omega) + \sum_{\omega \in \Omega \setminus \{\omega_{j^*}\}} a_1^2(\omega) \cdot f(\omega) - \sum_{\omega \in \Omega \setminus \{\omega_{j^*}\}} a_2(\omega) \cdot f(\omega) + q_2 \cdot f(\omega_{j^*}) \\
&= \sum_{\omega \in \Omega \setminus \{\omega_{j^*}\}} a_1^2(\omega) \cdot f(\omega) + q_2 \cdot f(\omega_{j^*}).
\end{aligned}$$

In comparison, the ex-ante amount of agent 1 subject to agent 2's saturation amount is

$$\sum_{\omega \in \Omega \setminus \{\omega_{j^*}\}} a_1^2(\omega) \cdot f(\omega) + \min\left\{ \omega_j - \frac{A}{f(\omega_{j^*})} - q_2, q_2 \right\} \cdot f(\omega_{j^*}).$$

Hence, we have to prove that $\min\left\{\omega_j - \frac{A}{f(\omega_{j^*})} - q_2, q_2\right\} = q_2$.

This equality follows from the possible values of B and the if statement equation (Equation (10)). In more detail, Equation (9) implies $\omega_{j^*} \geq \frac{B}{f(\omega_{j^*})}$ and Equation (10) implies $-\frac{A}{f(\omega_{j^*})} - q_2 > -\frac{B}{f(\omega_{j^*})} + q_2$ which together imply $\omega_{j^*} - \frac{A}{f(\omega_{j^*})} - q_2 > q_2$. This means the minimum is q_2 and the equality holds. Therefore, agent 2 is just envy-free which concludes this case as well.

Finally, for both cases, agent 1 is ex-ante envy-free by Lemma 3.33 which informally fits since the algorithm favours this agent. □

A.4 Optimal Algorithm's Linear Optimisation Problem

Optimisation Problem 4 is the normal from linearised version of the optimal algorithm presented in Section 3.5.6. As a reminder x_{ij} is the amount allocated to agent i in event j and we maximise ex-ante social welfare (Objective Function (11)). The original constraints are Constraint (12), ensuring that any allocation is restricted to the available amount, the envy-freeness (Constraint (13)) and the positivity of any allocation (Constraint (23)). The linearisation is achieved using Lemmata 3.66, 3.67 and 3.68. Considering them in turn, Lemma 3.66 introduces Constraint (21) to remove one minimum function. Lemma 3.67 introduces two new variables (x^i_{kj} and y^i_{kj}) and 5 constraints to linearise the other minimum function. Firstly, the variable y^i_{kj} is set to be binary in Constraint (22). Secondly, the first set of inequalities in the lemma are Constraint (14) and Constraint (16). Lastly, the second set of inequalities are Constraint (15) and Constraint (17). Finally, Lemma 3.68 introduces one variable (z^i_{kj}) and 4 constraints to linearise the product introduced in one of the constraints of Lemma 3.67. The first inequalities in the lemma are Constraint (19) and Constraint (24). The second inequalities in the lemma are Constraint (20) and Constraint (18).

Optimisation Problem 4

$$\max \quad \sum_{i\in[n]}\sum_{j\in[m]} Pr(\omega_j)\cdot\frac{u_i}{q_i}\cdot x_{ij} \tag{11}$$

$$\text{s.t.} \quad \sum_{i\in[n]} x_{ij} \le \omega_j \qquad \forall j\in[m] \tag{12}$$

$$\sum_{j\in[m]} Pr(\omega_j)\cdot(x^i_{kj} - x_{ij}) \le 0 \qquad \forall i\in[n], k\in[n] \tag{13}$$

$$x^i_{kj} - q_i \le 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{14}$$

$$x^i_{kj} - x_{kj} \le 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{15}$$

$$-x^i_{kj} + q_i\cdot y^i_{kj} \le 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{16}$$

$$x_{kj} - z^i_{kj} - x^i_{kj} \le 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{17}$$

$$z^i_{kj} - y^i_{kj} \le 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{18}$$

$$z^i_{kj} - x_{kj} \le 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{19}$$

$$x_{kj} + y^i_{kj} - z^i_{kj} \le 1 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{20}$$

$$x_{ij} - q_i \le 0 \qquad \forall i\in[n], j\in[m] \tag{21}$$

$$y^i_{kj} \in \{0,1\} \qquad \forall i\in[n], k\in[n], j\in[m] \tag{22}$$

$$x_{ij} \ge 0 \qquad \forall i\in[n], j\in[m] \tag{23}$$

$$z^i_{kj} \ge 0 \qquad \forall i\in[n], k\in[n], j\in[m] \tag{24}$$