

**Informaticup 2023**

**Profit**

von

**Lisa Binkert, Leopold Gaupe, Yasin Koschinski**

RWU Hochschule Ravensburg-Weingarten University of Applied Sciences

Abgabedatum: 15.01.2023

---

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>2</b>
<b>2</b>	<b>Profit</b>	<b>3</b>
2.1	Spielregel . . . . .	3
2.2	Implementierung . . . . .	4
2.3	Tests . . . . .	4
<b>3</b>	<b>Task Generator</b>	<b>5</b>
<b>4</b>	<b>Reinforcement Learning</b>	<b>6</b>
	<b>Literaturverzeichnis</b>	<b>7</b>

# 1 Einleitung

---

## 2 Profit

Beim Spiel Profit werden Ressourcen von Depots abgebaut und in Fabriken zu Produkten verarbeitet. Für diese Produkte gibt es eine gewisse Anzahl an Punkten. Ziel des Spiels ist es so viele Punkte wie möglich zu erreichen, wobei die Rundenzahl sowie die vorhandenen Ressourcen begrenzt sind.

### 2.1 Spielregel

Das Spielfeld ist ein beliebig großes Rastergitter. Ein Raster ist entweder durch ein Objekt besetzt. Die Abbildung 1 zeigt ein mögliches Spielfeld. Das linke obere Raster

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
0																														
1		+	-	-	-	-						+	x										+	-	-	-	-	-	-	
2		-	0	0	0	-						x	x										-	2	2	2	2	2	2	-
3		-	0	0	0	-						x	x										-	2	2	2	2	2	2	-
4		-	0	0	0	-						x	x										-	2	2	2	2	2	2	-
5		-	-	-	-	↖						x	x										-	2	2	2	2	2	2	-
6												x	x										-	2	2	2	2	2	2	-
7												x	x										-	-	-	-	-	-	↖	
8												x	↖																	
9												+	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
10												x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	↖
11																														
12																														
13																														
14		+	-	-	-	-																								
15		-	1	1	1	-																								
16		-	1	1	1	-																								
17		-	1	1	1	-																								
18		-	-	-	-	↖																								
19																														

Abbildung 1: Beispiel eines Spielfeldes der Größe 30x30

befindet sich an der Stelle (0,0), die rechte untere an (Breite-1, Höhe-1). Die Größe des Spielfeldes ist pro Spiel vorgegeben.

Zu Beginn des Spiels sind bereits Hindernisse (Obstacle) und Depots (Deposits) mit Ressourcen vorhanden. Ebenso sind die Produkte die produziert werden können definiert.

Die Menge an Ressourcen eines Deposits werden durch die Größe des Deposits festgelegt, diese wird durch die Menge an Rastern  $\times 5$  festgelegt. Ist also ein Deposit  $3 \cdot 3$  dann enthält es  $3 \cdot 3 \cdot 5 = 45$  Ressourcen eines bestimmten Subtyps. Insgesamt gibt es 8 Subtypen (0-7).

Obstacle im Spielfeld stellen Felder dar, in denen kein anderes Gebäude gebaut werden kann. Diese sind beliebig groß.

## Produkte

Für jedes Spiel ist mindestens ein Produkt definiert, maximal acht. Ein Produkt benötigt eine festgelegte Anzahl an mindesten einer maximal acht Ressourcen. Wie viel von dieser Ressource gebraucht wird ist ebenfalls definiert. Beispielsweise kann das Produkt 0 dreimal die Ressource 0 und einmal die Ressource 1 brauchen. Jedes Produkt gibt eine bestimmte Anzahl an Punkte.

## Mine

Um die Ressourcen in den Deposits abzubauen gibt es Minen. Sie ist  $4 \cdot 2$  oder  $2 \cdot 4$  Felder Groß und hat vier Subtypen, die je unterschiedlich ausgerichtet sind. Die Abbildung 2 zeigt die 4 verschiedenen Subtypen von Minen. Jede Mine hat einen Eingang (+) und einen

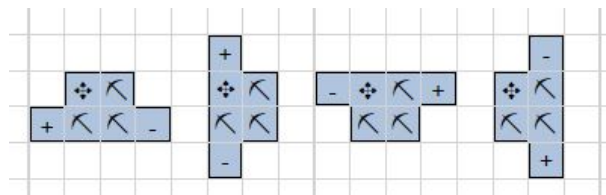


Abbildung 2: Minen in unterschiedlichen Ausrichtungen

Ausgang (-). Der Eingang muss an das Deposit angebracht werden um die Ressourcen abzubauen. An den Ausgang können Eingänge von Conveyor, Combiner oder Factories angebracht werden.

## Conveyor

## Combiner

## Factory

## 2.2 Implementierung

### Environment

### Buildings

### Game rules

## 2.3 Tests

### Environment

### Buildings

### Game rules

---

## 3 Task Generator

## 4 Reinforcement Learning

---

Dota

- OpenAI five

- Long-Short-Term-Memory - [Be19]

- Das Reinforcement Learning von Dota baut auf die Arbeit von Deep Networks in Atari Games auf: -LSTM -DQN, CNN basierter Agent  $\epsilon$ -greedy Policy [Oe15]

- Starcraft:

- Multi-Agent Reinforcement Learning MARL [Se19]

---

## Literaturverzeichnis

- [Be19] BERNER, Christopher ; ET. AL: *Dota 2 with Large Scale Deep Reinforcement Learning*. <http://dx.doi.org/10.48550/ARXIV.1912.06680>. Version: 2019
- [Oe15] OH, Junhyuk ; ET. AL: Action-Conditional Video Prediction using Deep Networks in Atari Games. In: CORTES, C. (Hrsg.) ; LAWRENCE, N. (Hrsg.) ; LEE, D. (Hrsg.) ; SUGIYAMA, M. (Hrsg.) ; GARNETT, R. (Hrsg.): *Advances in Neural Information Processing Systems* Bd. 28, Curran Associates, Inc., 2015
- [Se19] SAMVELYAN, Mikayel ; ET. AL: *The StarCraft Multi-Agent Challenge*. <http://dx.doi.org/10.48550/ARXIV.1902.04043>. Version: 2019