

## **Zusammenfassung**

In dieser Arbeit zeigen wir, dass es sich bei dem "Coalition Skill Game" um ein superadditives Spiel handelt in welcher die große Koalition die notwendigerweise beste Strategie für alle Agenten darstellt um deren Gewinn zu maximieren. Hierfür führen wir zunächst formal die Begriffe ein und zeigen dass sich jedes CSG bezüglich der Strategie äquivalent zu einem ist, indem jeder Agent nur über eine Kapazität verfügt. Anschließend zeigen wir die Superadditivitätseigenschaft für das Spiel und geben ein Algorithmus an mit dem sich die optimale Ressourcenverteilung aus der Sicht der Agenten berechnen lässt.

# AOTidf Bagger Game

Denis Erfurt, Tobias Behrens

29. November 2016

## 1 Voraussetzungen

Um das gegebene Problem strukturell analysieren zu können werden wir hier der Sprache der HOL (Higher Order Logic) bedienen. Zunächst werden wir in (1.2) vier Signaturen einführen mit den dazugehörigen Modellklassen einführen auf denen das Problem analysiert wird, anschließend werden wir in (1.3) Beziehungen zwischen den Modellklassen betrachten und schließlich in (1.4) die Superadditivität des Spiels zu zeigen.

**TODO:** Vereinfachungen - fixes binäres budget

### 1.1 Grundlagen

**TODO:** HOL

Im weiteren werden Funktionen mit kleinem Anfangsbuchstaben, sowie Relationen mit einem großen Anfangsbuchstaben geschrieben. Ebenfalls sind alle Funktionen total und mit 0 initialisiert, falls für eine Struktur und Eingabeparameter nicht näher definiert sind. Ebenfalls werden wir die Prädikatenschreibweise und die Mengenschreibweise äquivalent verwenden:  $a \in A \equiv A(a)$

## 1.2 Terminologie

### 1.2.1 Signaturen und Modellklassen

Wir werden vier verschiedene Signaturen mit den dazugehörigen Modellklassen einführen:

1. **CSGS** - Coalition Skill Game Setting
2. **CSG** - Coalition Skill Game
3. **SCSG** - Simple Coalition Skill Game
4. **SCSGS** - Simple Coalition Skill Game Setting

**Definition 1** (CSGS):

Eine Coalitional Skill Game Setting (CSGS) Signatur:

$$\sigma_{CSGS} := \quad (1)$$

$$\{Agent_{/1}, Baustelle_{/1}, supply_{/2}, demand_{/2}, budget_{/1}, kosten_{/3}\} \quad (2)$$

#### Intuition

$Agent(x)$	$:\Leftrightarrow$	x ist ein Agent
$Baustelle(x)$	$:\Leftrightarrow$	x ist eine Baustelle
$supply(x, t) \mapsto n$	$:\Leftrightarrow$	Agent x besitzt n Einheiten vom typ t
$demand(x, t) \mapsto n$	$:\Leftrightarrow$	Baustelle x benötigt n Einheiten vom typ t
$budget(x) \mapsto n$	$:\Leftrightarrow$	Baustelle x ist maximal bereit einen Gewinn von n bei fertigstellung auszahlungen
$kosten(t, n, x, y) \mapsto n$	$:\Leftrightarrow$	Kosten für Agent x für den Transport von n Ressourcen t an die Baustelle y.

Im folgenden werden wir die  $\sigma_{CSGS}$ -Struktur  $\mathcal{S}$  als ein Setting bezeichnen sowie die  $M_{CSGS}$  als die Modellklasse aller valider Settings. Auf eine formale Definition der validität eines Settings wie z.b. die Forderung des die Menge der Baustellen und Agenten disjunkt ist, wird verzichtet.

Intuitiv definiert ein Setting ein gegebenes Szenario:

**Beispiel 1:**

Sei  $\mathcal{S}$  eine  $\sigma_{CSGS}$ -Struktur mit:

$$Agent^{\mathcal{S}} := \{a_1, a_2, a_3\}$$

$$Baustelle^{\mathcal{S}} := \{b_1, b_2\}$$

$$supply^{\mathcal{S}}(a_1, t_1) \mapsto 2$$

$$supply^{\mathcal{S}}(a_1, t_2) \mapsto 7$$

$$supply^{\mathcal{S}}(a_2, t_1) \mapsto 3$$

$$supply^{\mathcal{S}}(a_2, t_2) \mapsto 5$$

$$supply^{\mathcal{S}}(a_3, t_1) \mapsto 20$$

$$supply^{\mathcal{S}}(a_3, t_2) \mapsto 5$$

$$demand^{\mathcal{S}}(b_1, t_1) \mapsto 10$$

$$demand^{\mathcal{S}}(b_1, t_2) \mapsto 5$$

$$demand^{\mathcal{S}}(b_2, t_1) \mapsto 2$$

$$demand^{\mathcal{S}}(b_2, t_2) \mapsto 2$$

$$budget^{\mathcal{S}}(b_1) \mapsto 10$$

$$budget^{\mathcal{S}}(b_2) \mapsto 3$$

**Definition 2 (CSG):**

Eine Coalitioal Skill Game Signatur

$$\sigma_{CSG} := \sigma_{CSGS} \cup \{m_{/3}, v_{/2}\}.$$

**Intuition**

$m(x, t, y) \mapsto n :\Leftrightarrow$  Agent x sendet n Ressourcen des Types t an die Baustelle y

$v(x, y) \mapsto n :\Leftrightarrow$  Agent x bekommt von Baustelle y eine vergütung von n

Wir bezeichnen eine  $\sigma_{CSG}$ -Struktur als ein Game  $\mathcal{G}$ . Intuitiv ist ein Game ein valides Setting zusammen mit einer möglichen validen Lösung (e.g. valide verteilung der Ressourcen und Vergütungen). Dabei werden wir  $M_{CSG}$  als die Klasse aller valider Games bezeichnen. Hier wird ebenfalls auf eine

formale Definition valider kritären verzichtet. Informell sind die wichtigsten validitätskritären dass ein Agent nicht mehr ressourcen ausgibt als er besitzt, eine Baustelle nicht mehr Geld ausgibt als sie besitzt und dass eine Baustelle nur geld ausgibt, falls diese vollständig mit benötigten Ressourcen versorgt wird.

**Beispiel 2:**

Sei  $\mathcal{G}$  eine erweiterung der  $\mathcal{A}$  Struktur zu einer  $\sigma_{CSG}$ -Struktur mit:

$$m(a_1, t_1, b_1) \mapsto 2$$

$$m(a_1, t_2, b_1) \mapsto 5$$

$$m(a_2, t_1, b_1) \mapsto 3$$

$$m(a_3, t_1, b_1) \mapsto 5$$

$$m(a_3, t_1, b_2) \mapsto 2$$

$$m(a_3, t_2, b_2) \mapsto 2$$

Wie wir später noch zeigen werden lässt sich jedes Setting und Game zu einem Vereinfachen, bei der jeder Agent nur eine Ressource besitzt. Diese vereinfachung erleichtert uns den Beweis der Superadditivität und die Betrachtung möglicher algorithmen zur verteilungsberechnung. Hierfür benötigen wir jedoch eine neue Relation  $AgentOwner_{/2}$  mit der wir die zuteilung der ursprünglichen Agenten zu Ressourcen merken.

**Definition 3:**

SCSGS Eine Simple Coalition Skill Game Setting Signatur:

$$i\sigma_{SCSGS} := \{AgentOwner_2, Agent_{/1}, Baustelle_{/1}, type_{/1}, demand_{/2}, budget_{/1}, kosten_{/3}\}$$

**Intuition**

$AgentOwner(a, x)$	$:\Leftrightarrow$	a "besitzt" agent x
$Agent(x)$	$:\Leftrightarrow$	x ist ein Agent
$Baustelle(x)$	$:\Leftrightarrow$	x ist eine Baustelle
$type(x) \mapsto t$	$:\Leftrightarrow$	Agent x ist vom typ t
$demand(x, t) \mapsto n$	$:\Leftrightarrow$	Baustelle x benötigt n Einheiten vom typ t
$budget(x) \mapsto n$	$:\Leftrightarrow$	Baustelle x ist maximal bereit einen Gewinn von n bei fertigung ausbezahlen
$kosten(t, n, a, y) \mapsto n$	$:\Leftrightarrow$	Kosten für Agentbesitzer x für den Transport von n Agenten des Typ an die Baustelle y.

Wir bezeichnen hier eine  $\sigma_{SCSGS}$ -Struktur  $\mathcal{SS}$  als ein Simple Setting und die dazugehörige Modellklasse von validen Strukturen  $M_{SCSGS}$ . Auf die Definition der validitätskriterien wird auch hier verzichtet.

**Definition 4:**

SCSG Eine Simple Coalition Skill Game Signatur:

$$\sigma_{SCSG} := \sigma_{SCSGS} \cup \{M_{/3}, v_{/2}\}$$

**Intuition**

$M(x, y) :\Leftrightarrow$  Agent x sendet seine Ressource an die Baustelle y

$v(x, y) \mapsto n :\Leftrightarrow$  Agent x bekommt von Baustelle y eine vergütung von n

Wir bezeichnen eine  $\sigma_{SCSG}$ -Struktur  $\mathcal{SG}$  als ein Simple Game mit der dazugehörigen validen Modellklasse  $M_{SCSG}$ . Auf die Definition der validitätskriterien wird verzichtet.

### 1.3 bez

Bei der gegebenen Aufgabenstellung interessieren wir uns für ein Mechanismus der bei einem gegebenen Setting ein Game berechnet. Formal gesehen:

$$m : M_{SCSGS} \leftarrow M_{SCSG} \tag{3}$$

Nach welchen Kriterien dieser Mechanismus arbeitet und bewertet wird wird vorerst offen gelassen.

Die nachfolgenden Betrachtungen wollen wir jedoch auf einer vereinfachten Strukturen machen, bei denen jeder Agent nur eine Ressource besitzt. Um dennoch Aussagen über CSG machen zu können werden wir zeigen dass jedes Setting bzw. Game zu einem Simple Setting bzw. Simple Game überführen lässt. Dieses erlaubt uns auch die Betrachtung von Mechanismen die bei einem Simple Setting ein Simple Game berechnen. Formal:

$$\begin{array}{ccc}
 M_{CSGS} & \xrightarrow{\pi'^{-1} \circ m' \circ \pi} & M_{CSG} \\
 \downarrow \pi & & \uparrow \pi'^{-1} \\
 M_{SCSGS} & \xrightarrow{m'} & M_{SCSG}
 \end{array}$$

Dabei müssen folgende Eigenschaften gelten:

$$\pi \quad - \quad \text{total, injektiv} \quad (4)$$

$$\pi^{-1} \quad - \quad \text{surjektiv} \quad (5)$$

$$\pi' \quad - \quad \text{total, injektiv} \quad (6)$$

$$\pi'^{-1} \quad - \quad \text{surjektiv} \quad (7)$$

$$\pi^{-1} \circ \pi \quad = \quad id_{M_{CSGS}} \quad (8)$$

$$\pi'^{-1} \circ \pi' \quad = \quad id_{M_{CSG}} \quad (9)$$

$$(10)$$

Weiter werden wir nur Mechanismen betrachten die nichts am Setting ändern sondern nur matchings und vergütungsverteilung berechnen. Auf eine ausführliche Definition der gesuchten Funktionen wird hier ebenfalls verzichtet. Stattdessen wird eine intuition gegeben: Ein Setting bzw. Game lässt

sich in ein Simple Setting bzw. Simple Game überführen indem jede resource eines Agenten als eigenständiger Agent betrachtet wird wobei die zugehörigkeit von agenten und Ressourcen sich in der *AgentOwner* Relation gemerkt wird. Kostenfunktion bleibt dabei bestehen und wird lediglich auf die Agentbesitzer übertragen. Bei den Matches sowie vergütung wird ähnlich verfahren. Hierdurch gehen keine informationen verloren und alle Betrachtungen können auf den vereinfachten Modellen durchgeführt werden.

## 1.4 supadd

Um die Superadditivität des Spieles zu zeigen werden zunächst der Begriffe formal festgehalten:

### Definition 5:

Sei  $K \subseteq A$  eine beliebige Koalition. Wir definieren die Menge  $PM(K)$  der potentiellen Matches:

$$PM(K) := \{ \quad (11)$$

$$m : Agent \times Resource \times Baustelle \rightarrow \mathbb{N} \mid (12)$$

$$\phi_{Match}(m) \wedge (13)$$

$$\forall x. \forall t. \forall y. m(x, t, y) > 0 \rightarrow x \in K \} (14)$$

Dabei definiert  $\phi_{Match}(m)$  die validen Matches:

$$\phi_{Match}(m) := \forall x. \forall t. \left( \sum_{y \in Baustelle} m(x, t, y) \right) \leq supply(x, t) \quad (15)$$

Die Zeile (14) sagt das nur solche Matches betrachtet werden, deren "aktive" Agenten auch teil der Koalition sind.

### Definition 6:

Sei  $K$  eine beliebige Koalition, dann definiert  $B(K)$  die Menge der möglichen Baustellen, die von  $K$  gleichzeitig gebaut werden können.

$$B(K) := \{ B_k \subseteq Baustelle \mid (16)$$

$$\exists m \in PM(K). \forall b \in B_k. \forall t. \left( \sum_{x \in K} m(x, t, b) \right) \geq demand(b, t) \} (17)$$



Da es sich um ein Simple Game handelt, Könnenn wir o.B.d.A. annehmen, dass eine Agent nur einer Koalition zugeordnet ist:

$$K \neq S \Rightarrow K \cap S = \emptyset \quad (18)$$

Hieraus folgt notwendigerweise, dass eine Baustelle in einem Spiel nur von einer Koalition gebaut werden kann, da diese pro Spiel nur einmal gebaut werden kann. Um dieses formal festzuhalten definieren wir eine Outcome Relation:

**Definition 7:**

Seien  $K, S$  Koalitionen. Eine Outcome Menge weist beiden Koalitionen ein Tupel der Baustellen zu, die sie jeweils zur gleichen Zeit bauen können:

$$Outcome(K, S) := \{(B_K, B_S) \subseteq Baustelle \times Baustelle \mid \quad (19)$$

$$B_K \in B(K) \wedge \quad (20)$$

$$B_S \in B(S) \wedge \quad (21)$$

$$B_K \cap B_S = \emptyset\} \quad (22)$$

Zur weiteren vereinfachung definieren wir uns noch eine Bewertungsfunktion für eine Menge von Baustellen:

**Definition 8:**

Sei  $B \subseteq Baustelle$  beliebig. Dann definiert sich der wert von  $B$ :

$$v(B) := \sum_{b \in B} budget(b) \quad (23)$$

**Definition 9:**

Nun können wir den Wert betrachten, die zwei Koalitionen erzielen können. Seien  $K, S$  beliebige Koalition:

$$v(K) + v(S) := \max_{(B_K, B_S) \in Outcome(K, S)} (v(B_K) + v(B_S)) \quad (24)$$

Dadurch dass die Baustellen die von einer Koalition gebaut werden nicht mehr von einer anderen koalition gebaut werden können, müssen wir diese zeitgleich betrachten.

**Bemerkung:** Nach Definition ist das Ergebniss pareto effizient.

Exestiert eine Koalition isoliert, lässt sich diese auch isoliert betrachten:

**Definition 10:**

Sei  $K$  eine beliebige Koalition.

$$v(K) := \max_{B' \in B(K)} (v(B')) \quad (25)$$

Insbesondere gilt dann auch für zwei Koalitionen  $K, S$ :

$$v(K \cup S) = \max_{B' \in B(K \cup S)} (v(B')) \quad (26)$$

Beachte auch das für einen Outcome zweier Koalitionen eine mögliche Baustellenmenge der potentiellen Baustellen der vereinigung der Koalitionen exestiert die alle Baustellen des seperaten Outcomes beinhaltet:

$$\forall (B_K, B_S) \in Outcome(K, S). \exists B' \in B(K \cup S). B_K \cup B_S \subseteq B' \quad (27)$$

Hieraus können wir die Superadditivität des Spiels schließen:

$$v(K \cup S) \geq v(K) + v(S) \quad (28)$$

Insbesondere ist das Spiel Konvex. Nach einem Theorem (TODO) ist bei einem konvexen Spiel das Shapley Value im Core enthalten, sodass die beste Lösung dieses Spiels die große Koalition ist und eine stabile und faire Lösung immer exestiert.