Optimierung des Intruder-Problems durch Potentialvergabe

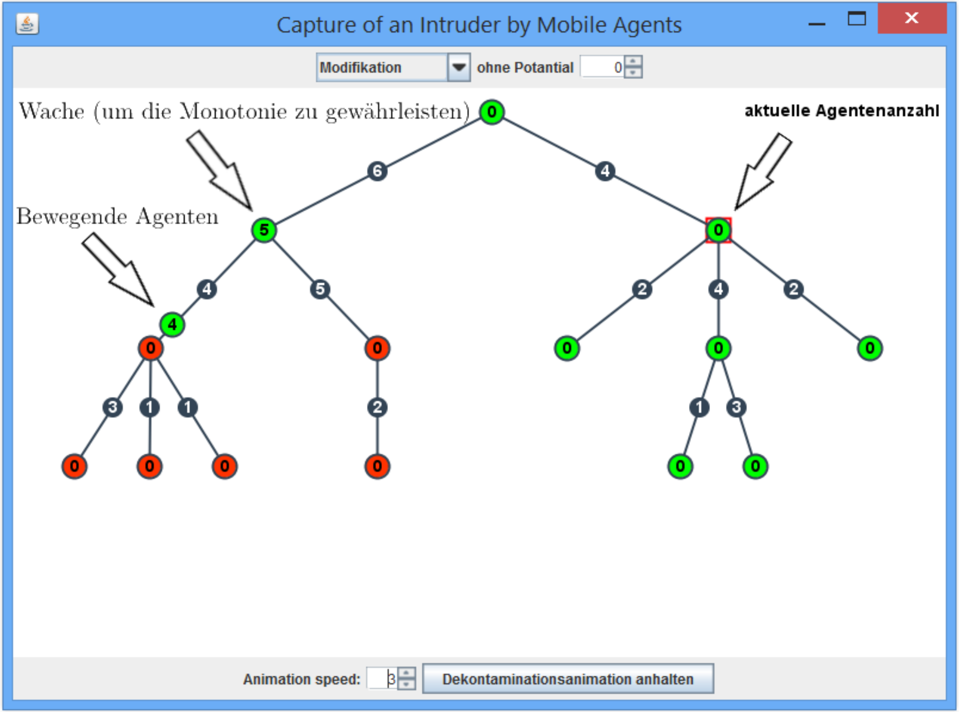
Jens Harder

9. November 2015

Inhaltsverzeichnis

Das Intruder-Problem auf einem Baum

- Gesucht wird ein Eindringling
- Minimale Anzahl an Agenten
- Systematisches Absuchen des Baumes
- wichtige Eigenschaft: Monotonie


Capture of an Intruder by Mobile Agents
Modifikation
ohne Potantial
Wache (um die Monotonie zu gewährleisten)
aktuelle Agentenanzahl
Bewegende Agenten
Animation speed:
Dekontaminationsanimation anhalten

Ziele der Bachelorarbeit

- Modifikation des ursprünglichen Algorithmus
- Algorithmus mit Potentialvergabe
  - $k = 1$ auf einer Kante
  - $k \geq 1$ auf einer Kante
  - $k > 1$ auf mehrere Kanten verteilt
- Implementierung der Algorithmen

Nachrichtenberechnung

- $\omega(e) \geq 1$ ist das Kantengewicht einer Kante e
- $\omega(x) = \max_e \omega(e)$ für jede zu x inzidente Kante e. Dann ist $\omega(x)$ das Knotengewicht des Knoten x
- $\mu(x)$ ist die minimale Anzahl an Agenten, die man für den Baum braucht, wenn man von diesem Knoten aus startet
- $\lambda_y(e)$ ist der Nachrichtenwert, der über Kante e zu Knoten y geschickt wird

Nachrichtenberechnung in 3 Fällen

1. Fall: Blattknoten ($\lambda_y = \omega(x)$)

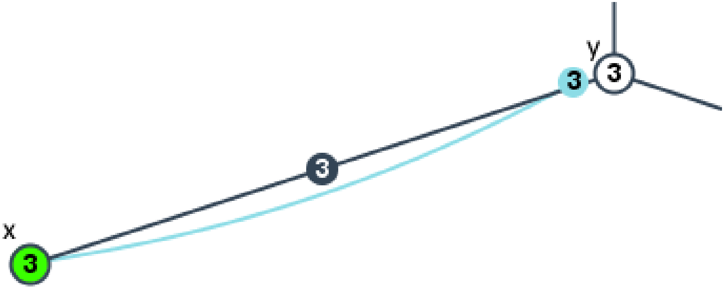


Abbildung: Das Knotengewicht des Blattknotens x (grün) bestimmt den Wert Nachricht $\lambda_y = 3$ (blau) zum Nachbarknoten y.

2. Fall: $n-1$ empfangene Nachrichten ($\lambda_y = max\{l_1, l_2 + \omega(x)\}$)

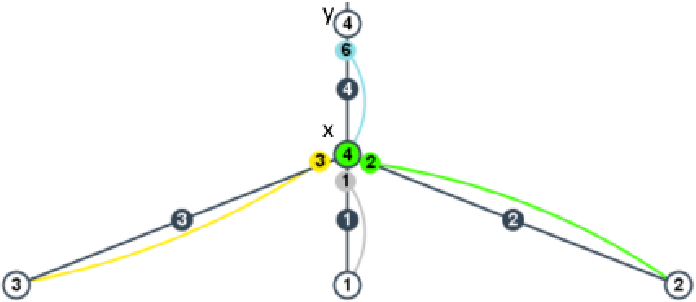


Abbildung: Der Wert der neuen Nachricht λ_y(blau) von Knoten x zu Knoten y beträgt 6, da $l_2 + \omega(x) = 6$ (grün) größer ist als $l_1 = 3$ (gelb).

Nachrichtenberechnung in 3 Fällen

3. Fall: n empfangene Nachrichten ($\lambda_y = max\{l_1, l_2 + \omega(x)\}$)

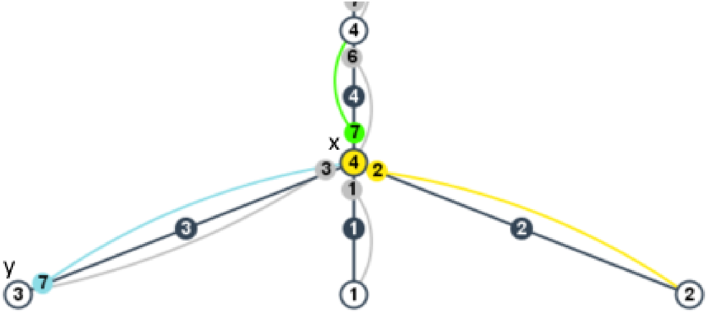


Abbildung: Die neue Nachricht λ_y (blau) von Knoten x zu Knoten y hat den Wert 7, da $l_1 = 7$ (grün) größer ist als $l_2 + \omega(x) = 6$ (gelb). Die Nachricht von y zu x (mit dem Wert $= 3$) wird bei dieser Berechnung ignoriert.

Berechnung der minimalen Agentenzahl

- Berechne $\mu(x)$ für jeden Knoten x: $\mu(x) = max\{l_1, l_2 + \omega(x)\}$
- Durchlauf durch alle Knoten, um das minimale μ zu finden
- Knoten x mit minimalem μ ist die neue Homebase

Wofür brauchen wir die Modifikation?

Ohne Modifikation

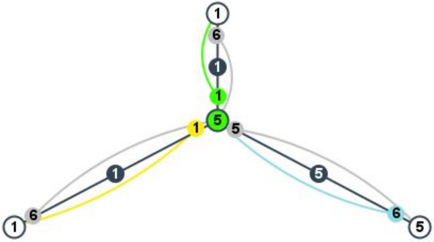


Abbildung: Berechnung der Nachrichten mit dem ursprünglichen Algorithmus.

Mit Modifikation

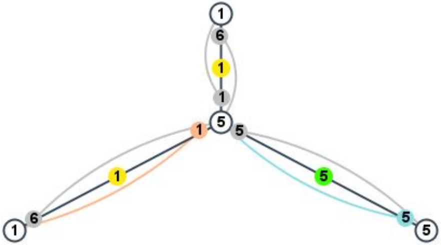


Abbildung: Berechnung der Nachrichten mit der modifizierten Variante.

Modifizierte Nachrichtenberechnung

$\lambda_y = edge_1 + edge_2$

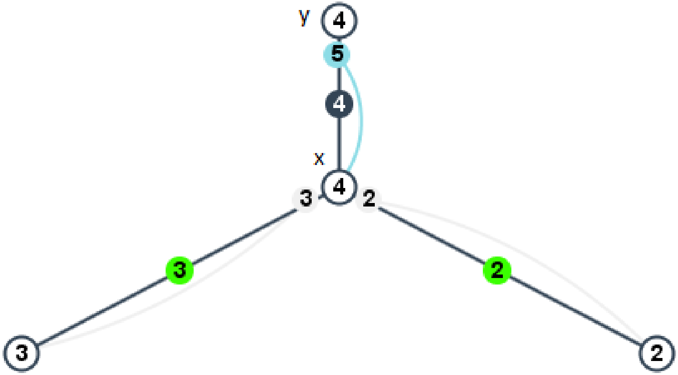


Abbildung: Die Nachricht λ_y (blau) von x zu y hat den Wert 5, da $edge_1$ und $edge_2$ (beide grün) die entscheidenden Kanten sind.

Modifizierte Nachrichtenberechnung

$\lambda_y = \max\{\lambda_y, \omega(x)\}$

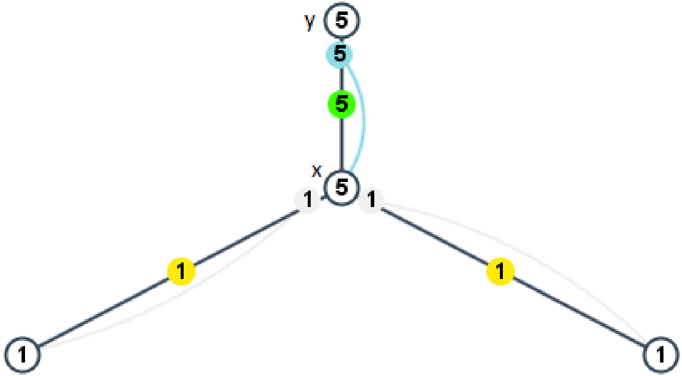


Abbildung: Da das Knotengewicht durch die Kante zwischen x und y bestimmt wird (grün), und dieses größer ist als die Summe der beiden normalen Kanten $edge_1$ und $edge_2$ (gelb), bestimmt diese Kante den Wert der Nachricht $\lambda_y = 5$ (blau).

Modifizierte Nachrichtenberechnung

$\lambda_y = \max\{\lambda_y, l_1\}$

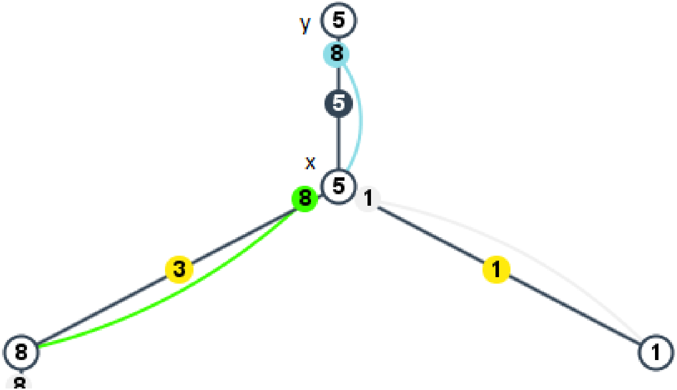


Abbildung: Die größte Nachricht (grün), die an x ankommt, ist größer als die bisher berechnete Nachricht und bestimmt daher den Nachrichtenwert $\lambda_y = 8$ (blau).

Berechnung der minimalen Agentenzahl

- Berechne $\mu(x)$ für jeden Knoten x:
  $\mu(x) = max\{edge_1 + edge_2, l_1\}$
- Durchlauf durch alle Knoten, um das minimale μ zu finden
- Knoten x mit minimalem μ ist die neue Homebase

Das Potential-Problem

- Reduziere Kante(n) mit einem Potential k
- Kann das Potential immer sinnvoll angewendet werden?
- Auf welche Kanten soll das Potential angewendet werden
- Algorithmus

Das Potential-Problem $k = 1$

- Idee: Protokolliere die Kanten, von denen der Nachrichtenwert abhängt
- 3 Fälle, von denen die Nachrichten abhängen können:
  a) $\lambda_y \leftarrow edge_1 + edge_2$ (aus den beiden größten Kanten)
  b) $\lambda_y \leftarrow \omega(x)$ (aus der Kante, über welche die Nachricht gerade verschickt wird (diese entspricht dem Knotengewicht))
  c) $\lambda_y \leftarrow l_1$ (aus der größten angekommenen Nachricht)

Das Potential-Problem $k = 1$

- Fall a: $\lambda_y \leftarrow edge_1 + edge_2$

if $edge_1 == edge_3$ **then**

    /* (mindestens) drei gleich große Kanten */

    protokolliere: „flag“

else if $edge_2 == edge_3$ **then**

    /* maximale Kante ist eindeutig */

    protokolliere: $edge_1$

else

    /* größte und zweitgrößte Kante ist eindeutig */

    protokolliere: $edge_1$ und $edge_2$

end if

Das Potential-Problem $k = 1$

- Fall b: $\lambda_y \leftarrow \omega(x)$

  Die Kante, über die die Nachricht geschickt wird, bestimmt den Nachrichtenwert:

  protokolliere: $e = \{x, y\}$

- Fall c: $\lambda_y \leftarrow l_1$

  Die größte angekommene Nachricht bestimmt den Nachrichtenwert:

  protokolliere: das gleiche wie l_1

Das Potential-Problem $k = 1$

- Berechnung aller $\mu(x)$ analog zur Nachrichtenberechnung
- in jedem Knoten wird protokolliert, von welchen Kanten sie abhängen ($edge_1$ udn $edge_2$ oder l_1)
- Potential kann auf eine beliebige Kante e angewendet werden, die in einem Knoten mit minimalem μ protokolliert wurde

Das Potential-Problem $k \geq 1$ auf einer Kante

- Idee: Protokolliere die Kanten, die am effektivsten reduziert werden können
- Es sind nur 4 Fälle pro Nachrichtenberechnung interessant:
  a) $edge_1$ (die größte Kante)
  b) $edge_2$ (die zweitgrößte Kante)
  c) $edge_{xy}$ (die Kante zwischen x und y)
  d) die in l_1 gespeicherten Kanten
- Wende das Potential k auf alle vier Möglichkeiten an, und speichere das beste Ergebnis β (Es muss immer gelten $\beta \geq 1$)

Das Potential-Problem $k \geq 1$ auf einer Kante

- speichere in jeder Nachricht drei Parameter:
  - α: die normal berechnete Nachricht (im modifizierten Algorithmus)
  - β: die modifizierte Nachricht, die durch das Potential am stärksten reduzierte Nachricht
  - e_1/e_2: die Kante(n), auf die das Potential angewendet worden ist
- analog bei Berechnung der minimalen Agentenzahl $\mu(x)$

$\rightarrow$ Jeder Knoten hat die Information

  - auf welchen Wert β er reduziert werden kann
  - welche Kante(n) e_1/e_2 dafür reduziert werden muss

Das Potential-Problem $k > 1$ auf mehreren Kanten

Es kann tatsächlich notwendig sein, das Potential k auf mehrere Kanten zu verteilen:

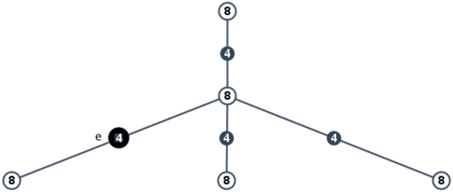


Abbildung: Alle Kanten haben Gewicht 4. Alle Knoten benötigen mindestens 8 Agenten.

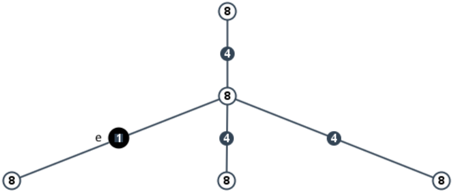


Abbildung: Eine Kante wurde auf Gewicht 1 reduziert, alle anderen Kanten haben weiterhin Gewicht 4. Alle Knoten benötigen trotzdem mindestens 8 Agenten.

Das Potential-Problem $k > 1$ auf mehreren Kanten

Es gibt Bäume, bei denen das Potential auf linear vielen Kanten verteilt werden muss (hier auf $n - 2$):

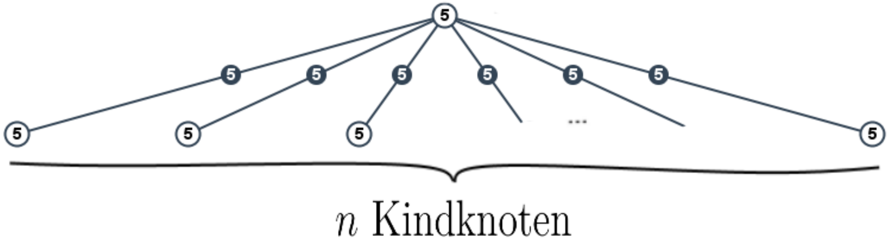


Abbildung: Die Mindestanzahl an benötigten Kanten kann bei der Potentialverteilung eine Komplexität linear zur Knotenanzahl haben.

Fazit und Aussicht

In dieser Bachelorarbeit war es möglich

- eine Modifikation des Algorithmus zu entwickeln
- einen Algorithmus für das Potential-Problem $k = 1$ zu entwickeln
- das Potential-Problem $k \geq 1$ auf einer Kante zu lösen
- zu Argumentieren, dass es nötig sein kann, das Potential auf mehrere Kanten aufzuteilen

Das Potential-Problem $k > 1$ auf verteilten Kanten konnte im Rahmen dieser Bachelorarbeit nicht gelöst werden, und könnte in Form einer zukünftigen Arbeit weiter untersucht werden.

Vielen Dank für Ihre Aufmerksamkeit!

Quellen

- Dieser Vortrag basiert auf der Bachelorarbeit „Optimierung des Intruder-Problems durch Potentialvergabe“
- Alle Abbildungen stammen aus dem im Rahmen dieser Bachelorarbeit entwickelten Applet
- Die Bachelorarbeit basiert auf dem Paper:

L. Barrière, P. Flocchini, P. Fraigniaud, N. Santoro (2002): Capture of an Intruder by Mobile Agents. (SPAA'02) Winnipeg, Manitoba, Canada.