



Verteilte Systeme - 4. Übung

Dr. Jens Brandt

Sommersemester 2011

Bully-Algorithmus

- a) Beschreiben Sie allgemein, wofür Auswahlalgorithmen benötigt werden und nennen Sie einige konkrete Einsatzzwecke.
- Übertragung von Algorithmen und Techniken eines Einzelsystems auf ein verteiltes System durch zentralen Koordinator
 - Beispiele: Zugriffs- und Ressourcenkontrolle, Deadlock-Erkennung
 - Algorithmen zur Rollenbestimmung für den Ausfall des Koordinators

Bully-Algorithmus

- b) In einem verteilten System, bestehend aus 10 Prozessen P_1, \dots, P_{10} , soll nach einem Ausfall des Koordinators mit Hilfe des Bully-Algorithmus dieser neu bestimmt werden. Legen Sie einen Prozess als ausgefallenen Koordinator sowie einen Prozess fest, der den Ausfall bemerkt und den Algorithmus initiiert, so dass ein möglichst hoher Kommunikationsaufwand vorliegt (worst case). Bestimmen Sie für diesen Fall, wie viele Nachrichten (ELECTION, OK, COORDINATOR) jeder der beteiligten Prozesse senden muss.
- Worst Case: P_1 bemerkt den Ausfall des Koordinators P_{10}

Bully-Algorithmus

Prozess	ELECTION	OK	COORDINATOR	Summe
P_1	9	0	0	9
P_2	8	1	0	9
P_3	7	2	0	9
P_4	6	3	0	9
\vdots	\vdots	\vdots	\vdots	\vdots
P_9	1	8	8	17
P_{10}	0	0	0	0
<hr/> <hr/>				
	45	36	8	89

Bully-Algorithmus (Kommunikationsaufwand)

$$K = \sum ELECTION + \sum OK + \sum COORDINATOR \quad (1)$$

$$= \sum_{i=1}^{n-1} i + \sum_{i=1}^{n-2} i + (n-2) \quad (2)$$

$$= 2 \sum_{i=1}^{n-2} i + (n-1) + (n-2) \quad (3)$$

$$= 2 \sum_{i=1}^{n-2} i + 2n - 3 \quad (4)$$

$$= 2 \frac{(n-2)(n-1)}{2} + 2n - 3 \quad (5)$$

$$= n^2 - n - 2n + 2 + 2n - 3 \quad (6)$$

$$= n^2 - n - 1 \quad (7)$$

$$\Rightarrow O(n^2) \quad (8)$$

(Gaußsche Summenformel: $\sum_{i=1}^n i = \frac{n(n+1)}{2}$)

Bully-Algorithmus

- c) Untersuchen Sie, ob der Bully-Algorithmus durch den Einsatz von Multicast-Kommunikation optimiert werden könnte.
- Durch den Einsatz von Multicastnachrichten können Nachrichten eingespart werden:
 1. Prozess P_k , der den Ausfall bemerkt sendet eine ELECTION-Nachricht per Multicast und wartet Zeit T
 2. Prozesse P_l mit $l > k$ schicken eine OK-Nachricht per Multicast und warten Zeit T
 3. Prozess P_i wird neuer Koordinator wenn innerhalb der Zeit T keine OK-Nachricht von P_j mit $j > i$ empfangen wurde und sendet eine COORDINATOR-Nachricht per Multicast

Bully-Algorithmus

Prozess	ELECTION	OK	COORDINATOR	Summe
P_1	1	0	0	1
P_2	0	1	0	1
P_3	0	1	0	1
\vdots	\vdots	\vdots	\vdots	\vdots
P_9	0	1	1	2
P_{10}	0	0	0	0
<hr/>				
	1	8	1	10

Kommunikationsaufwand: $O(n)$

Bully-Algorithmus

- d) Angenommen zwei Prozesse stellen gleichzeitig den Ausfall des Koordinators fest und initiieren beide eine Wahl mit Hilfe des Bully-Algorithmus. Was passiert?
- Jeder Prozess mit einer höheren Nummer als die initiierenden Prozesse empfängt zwei ELECTION-Nachrichten. Die zweite Nachricht wird ignoriert, so dass der Algorithmus normal weiter läuft.

Ring-Algorithmus zur Auswahl

- a) Erläutern Sie die Funktionsweise des in der Vorlesung vorgestellten Ring-Algorithmus zur Auswahl eines neuen Koordinators.
- Knoten sind in einem Ring angeordnet (ggf. logisch)
 - ELECTION-Nachricht wird im Ring verschickt
 - Jeder Prozess hängt die eigene Prozess-ID an
 - Initiierender Knoten empfängt die eigene ELECTION-Nachricht und wandelt diese in eine COORDINATOR-Nachricht um

Ring-Algorithmus zur Auswahl

- b) Bei diesem Algorithmus kann es vorkommen, dass zwei Wahlnachrichten gleichzeitig im Ring zirkulieren. Für die Auswahl des Koordinators ist dies nicht schädlich, allerdings wird zusätzlicher Traffic erzeugt. Wie kann dieser reduziert werden und welche weiteren Auswirkungen kann dies haben?
- Empfängt ein Prozess eine ELECTION-Nachricht und hat selbst bereits eine verschickt, vergleicht er die eigene ID mit der in der Nachricht.
 - Ist die ID in der ELECTION-Nachricht kleiner als die eigene, so wird die Nachricht verworfen.

Gegenseitiger Ausschluss

- a) Nennen Sie verschiedene generische Mechanismen für die Realisierung von gegenseitigem Ausschluss.
- Mutual Exclusion (Mutex)
 - (Read-Write-) Lock: Ressource wird (zum Lesen oder Schreiben) gesperrt
 - Semaphore: Es werden nur x gleichzeitige Zugriffe pro Ressource gestattet
 - Aktive Warten: Regelmäßiges Polling ob die Ressource verfügbar ist
 - Passives Warten: Prozess wird benachrichtigt, wenn Ressource verfügbar ist
- b) Beschreiben Sie den in der Vorlesung vorgestellten zentralisierten Algorithmus zu Realisierung von gegenseitigem Ausschluss.
- Koordinator für kritische Region
 - Prozess muss Erlaubnis des Koordinators erfragen
 - Token zum Betreten der Region
 - Rückgabe des Token nach Verlassen der Region

Gegenseitiger Ausschluss

- c) Wie wird in diesem Verfahren auf den Verlust des Koordinators reagiert und welche Nebeneffekte können hierbei auftreten?
- Koordinator ist Single Point of Failure
 - Ein Ausfall des Koordinators führt zu blockenden Prozessen
 - Der Ausfall eines Prozesses im kritischen Bereich führt ebenso zu einer Blockierung
- d) Skizzieren Sie wie dieser Algorithmus verbessert werden kann um die Robustheit gegen den Ausfall des Koordinators zu erhöhen.
- Koordinator muss jede Anfrage direkt zulassen oder ablehnen
 - Ein Ausfall kann durch Ausbleiben der Antwort festgestellt werden.

Gegenseitiger Ausschluss

- e) Beschreiben Sie den verteilten Algorithmus von Ricard und Agrawala für gegenseitigem Ausschluss.
- Kein Koordinator, logische Uhren
 - Zum Betreten eines kritischen Bereichs:
 - Request per Multicast an alle Prozesse
 - Nach Empfang aller OK-Antworten Betreten möglich
 - Senden einer OK-Nachricht nach Verlassen des Bereichs
 - Ist der kritische Bereich belegt, werden Requests durch den belegenden Prozess gespeichert und später bearbeitet.
- e) Erläutern Sie welche Vorteile dieser Algorithmus gegenüber der zentralisierten Variante bietet.
- Keine, der Algorithmus ist komplizierter, langsamer und weniger robust als der zentralisierte Algorithmus

Fragen?

`brandt@ibr.cs.tu-bs.de`

Nächste Übung: 28.06.2011 09:45 - 11:15 Uhr
Nächstes Übungsblatt ab 21.06.2011 auf den Webseiten