

Redistribuční algoritmus pro úlohy ve stromě

Richard Smolka

December 19, 2024

Uvažujeme stromovou strukturu (N, \leq) , kde:

- N : Množina uzlů, z nichž každý reprezentuje zdroj.
- \leq : Částečné uspořádání definující vztah rodič-potomek.
- T : Množina úloh, přičemž každá má časový požadavek $\tau(t)$.

Každý uzel $n \in N$ má kapacitu $t(n)$, a úlohy jsou přiřazeny tak, aby platilo:

$$\sum_{t \in T_n} \tau(t) \leq t(n),$$

kde T_n je množina úloh přiřazená uzlu n .

Cílem je redistribuovat úlohy napříč stromovou strukturou tak, aby:

- Všechny úlohy T byly dokončeny.
- Žádný uzel n nepřekročil svou kapacitu $t(n)$.

Pokud celková pracovní zátěž překročí celkovou kapacitu, systém je neproveditelný.

Základní algoritmus a rámec problému

Základní algoritmus pro redistribuci úloh funguje následovně:

- 1 Spočítejte celkovou pracovní zátěž pro každý uzel.
- 2 Pokud pracovní zátěž uzlu překračuje jeho kapacitu, redistribuujte přebytečné úlohy k jeho potomkům.
- 3 Postup opakujte rekurzivně pro všechny potomky.
- 4 Pokud listové uzly nemohou úlohy zpracovat, systém je neproveditelný.

Pro zvýšení robustnosti algoritmu jsou zavedeny následující rozšíření:

- 1 Globální kontrola proveditelnosti: Před zahájením redistribuce ověřte, zda celková pracovní zátěž nepřekračuje celkovou kapacitu.
- 2 Redistribuce založená na prioritách: Přidělte úlohám priority a redistribuujte nejprve úlohy s vysokou prioritou.
- 3 Optimalizace redistribuce: Seřad'te potomky podle dostupné kapacity, aby se minimalizovala hloubka redistribuce.
- 4 Explicitní řešení neproveditelnosti: Jasně řešte případy, kdy redistribuce selže u listových uzlů.

Zvažujeme strom se 4 uzly a následujícími kapacitami:

- N_1 : Kořenový uzal, kapacita $t(N_1) = 10$
- N_2 : Potomek uzlu N_1 , kapacita $t(N_2) = 8$
- N_3 : Potomek uzlu N_1 , kapacita $t(N_3) = 6$
- N_4 : Potomek uzlu N_2 , kapacita $t(N_4) = 5$

Máme 4 úlohy s následujícími časovými požadavky:

- T_1 : Čas $\tau(T_1) = 7$
- T_2 : Čas $\tau(T_2) = 4$
- T_3 : Čas $\tau(T_3) = 3$
- T_4 : Čas $\tau(T_4) = 5$

Na začátku jsou všechny úlohy přiřazeny uzlu N_1 , čímž překračují jeho kapacitu:

$$\tau(T_1) + \tau(T_2) + \tau(T_3) + \tau(T_4) = 19 > t(N_1) = 10.$$

Úlohy jsou redistribuovány, aby splňovaly kapacitní omezení.

- 1 Přřadte T_1 ($\tau(T_1) = 7$) uzlu N_2 . Zbývající kapacita uzlu N_1 :
 $19 - 7 = 12$.
- 2 Přřadte T_4 ($\tau(T_4) = 5$) uzlu N_3 . Zbývající kapacita uzlu N_1 :
 $12 - 5 = 7$.
- 3 Přřadte T_3 ($\tau(T_3) = 3$) uzlu N_4 (potomek N_2). Zbývající kapacita uzlu N_1 : $7 - 3 = 4$.
- 4 Ponechejte T_2 ($\tau(T_2) = 4$) v uzlu N_1 .

Konečné přiřazení úloh je:

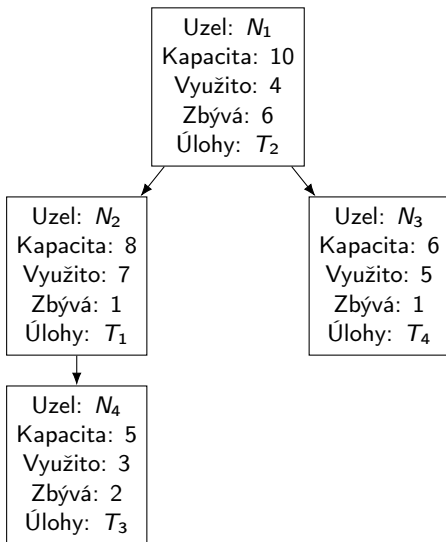
$N_1 : T_2$ (Kapacita: 10, Využito: 4, Zbývá: 6),

$N_2 : T_1$ (Kapacita: 8, Využito: 7, Zbývá: 1),

$N_3 : T_4$ (Kapacita: 6, Využito: 5, Zbývá: 1),

$N_4 : T_3$ (Kapacita: 5, Využito: 3, Zbývá: 2).

Reprezentace stromu



Redistribuce úloh (Alternativní řešení)

Zvažujeme stejnou stromovou strukturu a úlohy:

- N_1 : Kořenový uzel, kapacita $t(N_1) = 10$
- N_2 : Potomek uzlu N_1 , kapacita $t(N_2) = 8$
- N_3 : Potomek uzlu N_1 , kapacita $t(N_3) = 6$
- N_4 : Potomek uzlu N_2 , kapacita $t(N_4) = 5$
- T_1 : Čas $\tau(T_1) = 7$
- T_2 : Čas $\tau(T_2) = 4$
- T_3 : Čas $\tau(T_3) = 3$
- T_4 : Čas $\tau(T_4) = 5$

Na začátku jsou všechny úlohy přiřazeny uzlu N_1 , čímž překračují jeho kapacitu:

$$\tau(T_1) + \tau(T_2) + \tau(T_3) + \tau(T_4) = 19 > t(N_1) = 10.$$

Úlohy jsou redistribuovány, aby žádný uzel nepřekročil svou kapacitu.

- 1 Přřadte T_1 ($\tau(T_1) = 7$) uzlu N_2 . Zbývající kapacita uzlu N_1 :
 $19 - 7 = 12$.
- 2 Přřadte T_3 ($\tau(T_3) = 3$) uzlu N_3 . Zbývající kapacita uzlu N_1 :
 $12 - 3 = 9$.
- 3 Přřadte T_4 ($\tau(T_4) = 5$) uzlu N_4 (potomek N_2). Zbývající kapacita uzlu N_1 : $9 - 5 = 4$.
- 4 Ponechejte T_2 ($\tau(T_2) = 4$) v uzlu N_1 .

Úlohy a časové požadavky

- T_1 : Čas $\tau(T_1) = 7$
- T_2 : Čas $\tau(T_2) = 4$
- T_3 : Čas $\tau(T_3) = 3$
- T_4 : Čas $\tau(T_4) = 5$

Na začátku jsou všechny úlohy přiřazeny uzlu N_1 , čímž překračují jeho kapacitu:

$$\tau(T_1) + \tau(T_2) + \tau(T_3) + \tau(T_4) = 19 > t(N_1) = 10.$$

Úlohy jsou redistribuovány, aby žádný uzel nepřekročil svou kapacitu.

- 1 Přiřadte T_1 ($\tau(T_1) = 7$) uzlu N_2 . Zbývající kapacita uzlu N_1 :
 $19 - 7 = 12$.
- 2 Přiřadte T_3 ($\tau(T_3) = 3$) uzlu N_3 . Zbývající kapacita uzlu N_1 :
 $12 - 3 = 9$.
- 3 Přiřadte T_4 ($\tau(T_4) = 5$) uzlu N_4 (potomek N_2). Zbývající kapacita uzlu N_1 :
 $9 - 5 = 4$.
- 4 Ponechte T_2 ($\tau(T_2) = 4$) v uzlu N_1 .

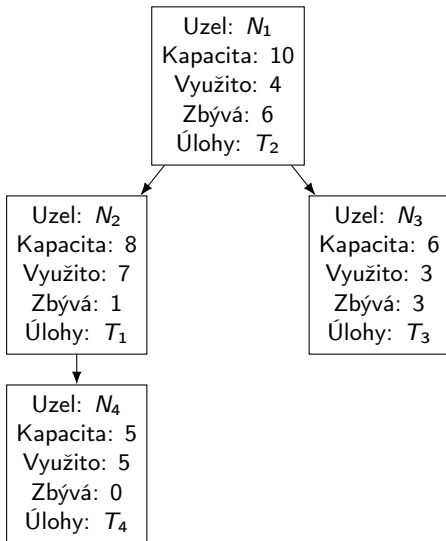
$N_1 : T_2$ (Kapacita: 10, Využito: 4, Zbývá: 6),

$N_2 : T_1$ (Kapacita: 8, Využito: 7, Zbývá: 1),

$N_3 : T_3$ (Kapacita: 6, Využito: 3, Zbývá: 3),

$N_4 : T_4$ (Kapacita: 5, Využito: 5, Zbývá: 0).

Reprezentace stromu



Ilustrační příklad: Proces přípravy pizzy

Pro demonstraci algoritmu redistribuce uvažujeme hierarchický strom reprezentující proces přípravy pizzy v pizzerii.

Hierarchie sestává ze čtyř uzlů:

- N_1 : Manažer, dohlížející na proces a zpočátku odpovědný za všechny úlohy.
- N_2 : Šéfkuchař, odpovědný za úkoly související s vařením.
- N_3 : Asistent, zajišťující úkoly přípravy.
- N_4 : Doručovatel, odpovědný za doručování pizzy zákazníkům.

Každý uzel má kapacitu, která udává počet úkolů, které může zvládnout:

$$t(N_1) = 5, \quad t(N_2) = 10, \quad t(N_3) = 8, \quad t(N_4) = 6.$$

Máme sadu úkolů potřebných k přípravě a doručení pizzy:

- T_1 : Hnětení těsta ($\tau(T_1) = 3$),
- T_2 : Příprava polevy ($\tau(T_2) = 4$),
- T_3 : Pečení pizzy ($\tau(T_3) = 5$),
- T_4 : Balení pizzy ($\tau(T_4) = 2$),
- T_5 : Doručení pizzy ($\tau(T_5) = 3$).

Ilustrační příklad: Proces přípravy pizzy

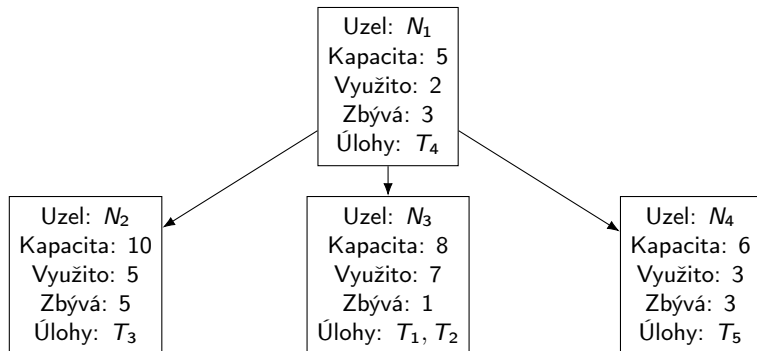
čátku jsou všechny úkoly přiřazeny uzlu N_1 , čímž překračují jeho kapacitu:

$$\tau(T_1) + \tau(T_2) + \tau(T_3) + \tau(T_4) + \tau(T_5) = 17 > t(N_1) = 5.$$

Úkoly jsou redistribuovány napříč hierarchií, aby byla pracovní zátěž vyvážená.

Algoritmus redistribuce přiřazuje úkoly následovně:

- 1 Přiřad'te T_3 ($\tau(T_3) = 5$) uzlu N_2 , protože pečení spadá do odpovědnosti šéfkuchaře. Zbývající kapacita uzlu N_1 : $17 - 5 = 12$.
- 2 Přiřad'te T_1 ($\tau(T_1) = 3$) a T_2 ($\tau(T_2) = 4$) uzlu N_3 , protože přípravné úkoly spadají do odpovědnosti asistenta. Zbývající kapacita uzlu N_1 : $12 - (3 + 4) = 5$.
- 3 Přiřad'te T_5 ($\tau(T_5) = 3$) uzlu N_4 , protože doručování je odpovědností doručovatele. Zbývající kapacita uzlu N_1 : $5 - 3 = 2$.
- 4 Ponechejte T_4 ($\tau(T_4) = 2$) v uzlu N_1 , protože balení vyžaduje dohled manažera.



Tento příklad ukazuje, jak jsou úkoly redistribuovány podle specializace a kapacity jednotlivých uzlů:

- Specializované uzly (např. šéfkuchař pro pečení, doručovatel pro doručování) zvládají své úkoly efektivně.
- Manažer si ponechává minimální množství úkolů (např. balení) a redistribuuje většinu zátěže.
- Hierarchická struktura zajišťuje, že žádný uzel nepřekročí svou kapacitu, což umožňuje vyvážený pracovní tok.

Děkuji moc za pozornost.