

# Распределённые алгоритмы

mk.cs.msu.ru → Лекционные курсы → Распределённые алгоритмы

## Семинар 2

Свойства безопасности и живости

Лектор:

**Подымов Владислав Васильевич**

E-mail:

**valdus@yandex.ru**

ВМК МГУ, 2023/2024, весенний семестр

## Задача 1.1 (живость)

Доказать, что предложенное утверждение  $\varphi$  верно для предложенной распределённой системы в условиях слабой справедливости для всех указанных действий

---

Система состоит из одного узла, устроенного так

Узел содержит одну переменную:  $x$  типа  $\mathbb{Z}$ , начальное значение — неизвестное число, большее нуля ( $n$ )

Код узла:

```
do {  
   $x := x - 1$ ;  
} while( $x > 0$ )
```

$\varphi$ : Вычисление системы рано или поздно достигает тупиковой конфигурации

## Задача 1.2 (живость)

Доказать, что предложенное утверждение  $\varphi$  **в общем случае неверно** для предложенной распределённой системы в условиях слабой справедливости для всех указанных действий

---

Система состоит из двух узлов,  $A_0$  и  $A_1$ , устроенных так

Каждый узел  $A_i$  содержит одну переменную:  $x_i$  типа  $\mathbb{Z}$ , начальное значение неизвестно ( $n_i$ )

Узел  $A_i$  содержит два действия:

$\mathcal{R}_i$ :

1.  $\text{receive}_{A_{1-i}}(v)$
2. Если  $x_i > v$ , то  $x_i := x_i - 1$ ;
3. Иначе: если  $x_i < v$ , то  $x_i := x_i + 1$ ;
4. Иначе действие навсегда становится недоступным

$\mathcal{S}_i$ :

1.  $\text{send}_{A_{1-i}}(x_i)$

$\varphi$ : В любом вычислении системы рано или поздно оба действия  $\mathcal{R}_i$  становятся недоступными

## Задача 1.3 (живость)

Доказать, что предложенное утверждение  $\varphi$  верно для предложенной распределённой системы в условиях слабой справедливости для всех указанных действий

---

Система состоит из двух узлов,  $A_0$  и  $A_1$ , устроенных так

Каждый узел  $A_i$  содержит одну переменную:  $x_i$  типа  $\mathbb{Z}$ , начальное значение неизвестно ( $n_i$ )

Узел  $A_i$  содержит два действия:

$\mathcal{R}_i$ :

1.  $\text{receive}_{A_{1-i}}(v)$
2. Если  $x_i > v + 1$ , то  $x_i := x_i - 1$ ;
3. Иначе: если  $x_i < v - 1$ , то  $x_i := x_i + 1$ ;
4. Иначе действие навсегда становится недоступным

$\mathcal{S}_i$ :

1.  $\text{send}_{A_{1-i}}(x_i)$

$\varphi$ : В любом вычислении системы рано или поздно оба действия  $\mathcal{R}_i$  становятся недоступными

## Задача 2.1 (безопасность)

Доказать, что утверждение  $\varphi$  является инвариантом предложенной распределённой системы, и на основании этого обосновать утверждение  $\psi$

---

Система состоит из одного узла, устроенного так

Переменные:

- ▶  $x$  типа  $\mathbb{Z}$ , начальное значение 2
- ▶  $y$  типа  $\mathbb{Z}$ , начальное значение 3

Единственное действие:

1.  $x := x - 1$ ;
2.  $y := y + 1$ ;

$\varphi: x + y = 5$

$\psi$ : В любой достижимой конфигурации верно неравенство  $x + y < 10$

## Задача 2.2 (безопасность BSWP)

Устройство узла  $p$  (узел  $q$  устроен «симметрично»)

- ▶  $\ell_p : \mathbb{N}_0 = 0$
- ▶  $r_p : \mathbb{N}_0 = 0$
- ▶  $in_p : ARR[\mathcal{T}]$
- ▶  $out_p : ARR[\mathcal{T}] = (\perp, \perp, \perp, \dots)$ ;

**Действие  $S_p$ :**  $\{\ell_p < r_p + c_p\}$

1. Выбрать  $i \in \mathbb{N}_0$ :  $\ell_p \leq i < r_p + c_p$
2.  $send(\mathbf{pack}, in_p[i], i)$

**Действие  $R_p$ :**  $\{\text{очередь } Q_p \text{ не пуста}\}$

1.  $receive(\mathbf{pack}, w, i)$
2. Если  $out_p[i] = \perp$ :
  - 2.1  $out_p[i] := w$ ;
  - 2.2  $\ell_p := \max(\ell_p, i - c_q + 1)$ ;
  - 2.3  $r_p := \min(j \mid out_p[j] = \perp)$ ;

**Действие  $L_p$**   $\{\text{очередь } Q_p \text{ не пуста}\}$

1.  $receive(\mathbf{pack}, w, i)$

## Задача 2.2 (безопасность BSWP)

$$p^0: \forall i \in \{0, 1, \dots, r_p - 1\} : out_p[i] \neq \perp$$

$$p^1: \forall i \in \mathbb{N}_0 : (\mathbf{pack}, w, i) \in Q_p \Rightarrow w = in_q[i] \ \& \ (i < r_q + c_q)$$

$$p^2: \forall i \in \mathbb{N}_0 : out_p[i] \neq \perp \Rightarrow out_p[i] = in_q[i] \ \& \ (\ell_p > i - c_q)$$

$$p^3: \ell_p \leq r_q$$

$$q^0: \forall i \in \{0, 1, \dots, r_q - 1\} : out_q[i] \neq \perp$$

$$q^1: \forall i \in \mathbb{N}_0 : (\mathbf{pack}, w, i) \in Q_q \Rightarrow w = in_p[i] \ \& \ (i < \ell_p + c_p)$$

$$q^2: \forall i \in \mathbb{N}_0 : out_q[i] \neq \perp \Rightarrow out_q[i] = in_p[i] \ \& \ (\ell_q > i - c_p)$$

$$q^3: \ell_q \leq r_p$$

$$P^{BSWP} = p^0 \ \& \ p^1 \ \& \ p^2 \ \& \ p^3 \ \& \ q^0 \ \& \ q^1 \ \& \ q^2 \ \& \ q^3$$

Доказать, что  $P^{BSWP}$  — инвариант BSWP

## Задача 2.2: начало решения

По определению инварианта, следует доказать два утверждения:

1. Для любой начальной конфигурации  $\gamma$  верно  $P_{BSWP}(\gamma)$
2. Если  $P_{BSWP}(\gamma)$  и  $\gamma \rightarrow \delta$ , то  $P_{BSWP}(\delta)$

1.

$$p^0: \forall i \in \{0, 1, \dots, r_p - 1\} : out_p[i] \neq \perp$$

В начальной конфигурации верно  $r_p = 0$

$$p^1: \forall i \in \mathbb{N}_0 : (\mathbf{pack}, w, i) \in Q_p \Rightarrow w = in_q[i] \ \& \ (i < r_q + c_q)$$

В начальной конфигурации очередь  $Q_p$  пуста

$$p^2: \forall i \in \mathbb{N}_0 : out_p[i] \neq \perp \Rightarrow out_p[i] = in_q[i] \ \& \ (\ell_p > i - c_q)$$

В начальной конфигурации все значения  $out_p[i]$  равны  $\perp$

$$p^3: \ell_p \leq r_q$$

В начальной конфигурации верно  $\ell_p = 0 \leq 0 = r_q$

$q^0, q^1, q^2, q^3$  — аналогично



## Задача 2.2: начало решения

2.  $P_{BSWP}(\gamma) \ \&(\gamma \rightarrow \delta) \Rightarrow P_{BSWP}(\delta)$

Доказательство для действия  $\mathbf{R}_p$ : {очередь  $Q_p$  не пуста}

1.  $receive(\mathbf{pack}, w, i)$
2. Если  $out_p[i] = \perp$ :
  - 2.1  $out_p[i] := w$ ;
  - 2.2  $\ell_p := \max(\ell_p, i - c_q + 1)$ ;
  - 2.3  $r_p := \min(j \mid out_p[j] = \perp)$ ;

---

$p^0: \forall i \in \{0, 1, \dots, r_p - 1\} : out_p[i] \neq \perp$

Если  $out_p$  не изменяется, то и  $r_p$  не изменяется

Иначе по (2.3) после выполнения  $\mathbf{R}_p$  верно  $r_p = \min(j \mid out_p[j] = \perp)$

$p^1: \forall i \in \mathbb{N}_0 : (\mathbf{pack}, w, i) \in Q_p \Rightarrow w = in_q[i] \ \&(i < r_q + c_q)$

После выполнения  $\mathbf{R}_p$  в  $Q_p$  не появляются новые пакеты,  $in_q$  не изменяется и  $r_q$  не уменьшается

## Задача 2.2: начало решения

2.  $P_{BSWP}(\gamma) \ \&(\gamma \rightarrow \delta) \Rightarrow P_{BSWP}(\delta)$

Доказательство для действия  $\mathbf{R}_p$ : {очередь  $Q_p$  не пуста}

1.  $receive(\underline{\mathbf{pack}}, w, i)$
2. Если  $out_p[i] = \perp$ :
  - 2.1  $out_p[i] := w$ ;
  - 2.2  $\ell_p := \max(\ell_p, i - c_q + 1)$ ;
  - 2.3  $r_p := \min(j \mid out_p[j] = \perp)$ ;

---

$p^2$ :  $\forall j \in \mathbb{N}_0 : out_p[j] \neq \perp \Rightarrow out_p[j] = in_q[j] \ \&(\ell_p > j - c_q)$

После выполнения  $\mathbf{R}_p$ :

- ▶  $in_q$  и  $out_p[j]$  для всех  $j$ , кроме  $i$ , не изменяются
- ▶ Если перед (2)  $out_p[i] \neq \perp$ , то  $out_p$  и  $\ell_p$  не изменяются
- ▶ Если перед (2)  $out_p[i] = \perp$ , то:
  - ▶ Согласно  $p^1$ , после (1) верно  $w = in_q[i]$
  - ▶ Значит, после (2.1) верно  $out_p[i] = w = in_q[i]$
  - ▶ После (2.2)  $\ell_p$  может только увеличиться, и согласно правой части присваивания, верно  $\ell_p \geq i - c_q + 1$ , то есть  $\ell_p > i - c_q$

## Задача 2.2: начало решения

2.  $P_{BSWP}(\gamma) \ \&(\gamma \rightarrow \delta) \Rightarrow P_{BSWP}(\delta)$

Доказательство для действия  $\mathbf{R}_p$ : {очередь  $Q_p$  не пуста}

1.  $receive(\underline{\mathbf{pack}}, w, i)$
2. Если  $out_p[i] = \perp$ :
  - 2.1  $out_p[i] := w$ ;
  - 2.2  $\ell_p := \max(\ell_p, i - c_q + 1)$ ;
  - 2.3  $r_p := \min(j \mid out_p[j] = \perp)$ ;

---

$p^3$ :  $\ell_p \leq r_q$

После выполнения  $\mathbf{R}_p$ :

- ▶ Значение  $r_q$  не изменяется
- ▶ Если перед (2)  $out_p[i] \neq \perp$ , то и значение  $\ell_p$  не изменяется
- ▶ Если перед (2)  $out_p[i] = \perp$ , то после (2.2):
  - ▶ Из  $p^1$  следует  $i < r_q + c_q$
  - ▶ Значит,  $i - c_q + 1 < r_q + 1$ , то есть  $i - c_q + 1 \leq r_q$
  - ▶ Если  $\ell_p$  изменяется, то  $\ell_p = i - c_q + 1 \leq r_q$

## Задача 2.2: начало решения

2.  $P_{BSWP}(\gamma) \ \& (\gamma \rightarrow \delta) \Rightarrow P_{BSWP}(\delta)$

Доказательство для действия  $\mathbf{R}_p$ : {очередь  $Q_p$  не пуста}

1.  $receive(\mathbf{pack}, w, i)$
2. Если  $out_p[i] = \perp$ :
  - 2.1  $out_p[i] := w$ ;
  - 2.2  $\ell_p := \max(\ell_p, i - c_q + 1)$ ;
  - 2.3  $r_p := \min(j \mid out_p[j] = \perp)$ ;

---

$q^0$ :  $\forall i \in \{0, 1, \dots, r_q - 1\} : out_q[i] \neq \perp$

Значения  $r_q$  и  $out_q$  не изменяются

$q^1$ :  $\forall i \in \mathbb{N}_0 : (\mathbf{pack}, w, i) \in Q_q \Rightarrow w = in_p[i] \ \& (i < r_p + c_p)$

Значение  $r_p$  не уменьшается, значения  $Q_q$  и  $in_p$  не изменяются

$q^2$ :  $\forall i \in \mathbb{N}_0 : out_q[i] \neq \perp \Rightarrow out_q[i] = in_p[i] \ \& (\ell_q > i - c_p)$

Значения  $out_q$ ,  $in_p$  и  $\ell_q$  не изменяются

$q^3$ :  $\ell_q \leq r_p$

Значение  $\ell_q$  не изменяется,  $r_p$  — не уменьшается

## Задача 2.2: начало решения

$$2. P_{BSWP}(\gamma) \ \&(\gamma \rightarrow \delta) \Rightarrow P_{BSWP}(\delta)$$

**Действие  $S_p$ :**  $\{\ell_p < r_p + c_p\}$

1. Выбрать  $i \in \mathbb{N}_0$ :  $\ell_p \leq i < r_p + c_p$
2.  $send(\mathbf{pack}, in_p[i], i)$

**Действие  $L_p$**  {очередь  $Q_p$  не пуста}

1.  $receive(\mathbf{pack}, w, i)$

*Чтобы завершить решение, следует сделать то же самое и для этих двух действий*