DOI 10.34185/1562–9945–5–166–2026–09
УДК 004.75

Л.І. Цвіркун, Ю.А. Миронов

КЕРУВАННЯ АКТИВАЦІЄЮ РОБОЧИХ ПРОЦЕСІВ ЗА УМОВ БЛОКУВАННЯ ЧЕРЕЗ ОБМЕЖЕНІ РЕСУРСИ БАГАТОРАЗОВОГО ВИКОРИСТАННЯ В РОЗПОДІЛЕНИХ СИСТЕМАХ ВИКОНАННЯ СПРЯМОВАНИХ АЦИКЛІЧНИХ ГРАФІВ

Анотація. Досліджено керування активацією робочих процесів у відкритій стохастичній системі виконання САГ із обмеженим ресурсом багаторазового використання. У дискретно–подійному моделюванні порівняно дві архітектури отримання ресурсів – блокування за схемою «спочатку виконавець» та атомарне отримання ресурсів – і п'ять політик активації. За високого навантаження керування WIP зменшує p95 часу відгуку на 23–36% лише тоді, коли очікування токена утримує виконавця. За атомарного отримання цей виграш зникає. DTR з урахуванням САГ не показує практично значущої загальної переваги над простішими засобами керування.
*Ключові слова**:** розподілені системи робочих процесів; виконання САГ; активація робочих процесів; незавершена робота в системі; ресурси багаторазового використання; ліцензії на програмне забезпечення; блокування ресурсів; затримка у хвості розподілу; дискретно–подійне моделювання; хмарні обчислення.*

Постановка проблеми. У відкритій системі виконання САГ завдання можуть одночасно потребувати обчислювального виконавця та скінченного ресурсу багаторазового використання, наприклад токена плаваючої програмної ліцензії. На поведінку системи впливають дефіцит такого ресурсу та порядок його отримання. Якщо завдання спочатку займає виконавця і лише після цього очікує токен, недоступність токена тимчасово вилучає виконавця з обчислювального пулу. У роботі цей механізм розглядається як зовнішній ефект блокування виконавця. За атомарного отримання завдання запускається лише за одночасної доступності виконавця й токена, тому очікування токена не споживає обчислювальну потужність.

Звідси виникає питання керування активацією цілих робочих процесів: чи може обмеження активного WIP зменшити хвіст наскрізного часу відгуку, не знижуючи пропускну здатність, і чи потрібна для цього детальна інформація про позицію майбутнього попиту на ресурс у САГ.

Аналіз останніх досліджень і публікацій. Дослідження систем виконання робочих процесів охоплюють моделювання планувальників, динамічне керування ресурсами та вибір політик під час виконання [1–4]. WfCommons і WRENCH підтримують відтворювані дослідження та моделювання систем керування робочими процесами [5, 6]. Роботи з обмеженнями програмних ліцензій розглядають їх як явний ресурс плануван-

© Цвіркун Л.І., Миронов Ю.А., 2026
Copyright for this paper by its authors. Use permitted under License CC BY 4.0.
ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

ня: Qiao et al. – в онлайн–плануванні робочих процесів [7], Gulhan et al. – через прогнозування використання ліцензій у HPC [8], Noaman et al. – у backfilling для задач CAE з ліцензійними обмеженнями в гібридній хмарі [9].

Концептуально близькі дослідження з керування навантаженням (Workload Control, WLC) показують, що керування допуском може послаблювати негативний вплив блокування, а взаємодія кількох обмежених ресурсів із правилами допуску та диспетчеризації істотно впливає на продуктивність [10–12]. Водночас простіші засоби безпосереднього керування навантаженням іноді досягають рівня складніших схем [13].

Невирішеними залишаються питання: чи виникає відповідний ефект у відкритій стохастичній САГ–системі саме через утримання виконавця під час очікування ресурсу багаторазового використання; чи зберігається він за атомарного отримання ресурсів; та чи забезпечує прогноз структурної близькості майбутнього ресурсного попиту практично значущий додатковий виграш.

Мета дослідження – експериментально визначити умови, за яких керування активацією робочих процесів зменшує час відгуку у хвості розподілу у відкритій САГ–системі з обмеженим ресурсом багаторазового використання, та оцінити граничну цінність урахування структури САГ.

Для конкретизації мети сформульовано три дослідницькі гіпотези, що перевіряють механізм блокування, межу ефекту за архітектурою отримання ресурсів та граничну цінність урахування структури САГ.

H1 – механізм блокування. Чи зменшує керування активацією/WIP p95 наскрізного часу відгуку без зменшення пропускної здатності за значного навантаження на ресурс багаторазового використання, коли очікування цього ресурсу займає потужність виконавців?

H2 – межа за архітектурою отримання ресурсів. Чи зникає або стає практично незначним цей виграш, коли отримання ресурсу багаторазового використання є атомарним і неблокувальним?

H3 – гранична цінність урахування САГ. Чи забезпечує DTR з урахуванням САГ практично значущу перевагу над простішими налаштованими засобами керування WIP або обсягом роботи, що потребує токенів?

Викладення основного матеріалу дослідження.

Модель системи та навантаження. Робочі процеси надходять у відкриту систему та перебувають у відкладеній черзі FIFO або в активній множині $A(t)$. Відкладені процеси не споживають виконавців чи токенів; після активації процес не повертається до відкладеної черги. На рис. 1 показано розмежування між відкладеною чергою, контролером активації, активною множиною САГ і незмінним планувальником готових завдань. Активна система має C слотів виконавців і L токенів ресурсу багаторазового використання.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

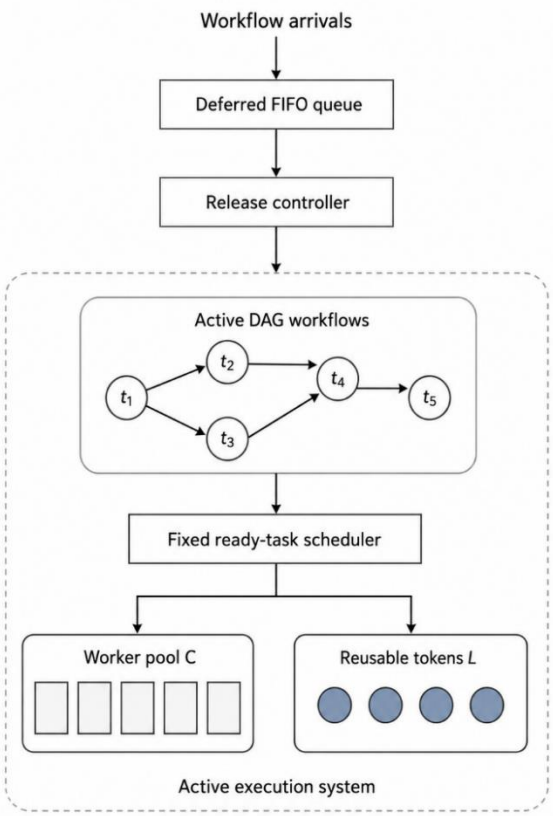


Рисунок 1 – Архітектура системи керування активацією робочих процесів

Для класів робочих процесів k із інтенсивністю надходження λ_k, очікуваною роботою виконавців $B_{C,k}$ та роботою ресурсу $B_{L,k}$ координати запропонованого навантаження визначаються за формулою (1):

$$\rho_C = \frac{\sum_k \lambda_k B_{C,k}}{C}, \ \rho_L = \frac{\sum_k \lambda_k B_{L,k}}{L}. \quad (1)$$

Ці показники характеризують робоче навантаження відносно номінальних місткостей і не трактуються як достатні умови стійкості повної САГ–системи.

Структуру семивершинного шаблону робочого процесу задає формула (2):

$$v_1 \to \{v_2, v_3\} \to v_4 \to \{v_5, v_6\} \to v_7, \quad (2)$$

Є три класи з єдиною операцією, що потребує токена, у v_1, v_4 або v_7. Змішане навантаження вибирає класи з однаковою ймовірністю; однорідний контрольний сценарій використовує середню позицію. Середній час обслуговування звичайного завдання дорівнює 1, токенового – 2, тому $B_C = 8$ і $B_L = 2$; у моделі використано $C = 12$ слотів і $L = 2$токени.

Часи обслуговування незалежні та мають розподіл (3):

$$S_v \sim Erlang\left(2, \frac{\mu_v}{2}\right). \quad (3)$$

Надходження пуассонівські. Основні змішані режими навантаження: $(\rho_C, \rho_L) = (0.3333, 0.50)$, $0.60, 0.90$ та $0.6267, 0.94$; однорідний контроль використо-

вує 0.60,0.90. Планувальник готових завдань незмінний для всіх політик і використовує ключ (ready time, workflow ID, task ID) та порядок $v_1 < \cdots < v_7$.

Архітектури отримання ресурсів. За блокування «спочатку виконавець» (рис. 2а) токенове завдання спочатку займає виконавця; якщо токен недоступний, воно очікує у черзі FIFO, продовжуючи утримувати виконавця. За атомарного отримання (рис. 2б) завдання диспетчеризується лише за одночасної доступності виконавця й токена.

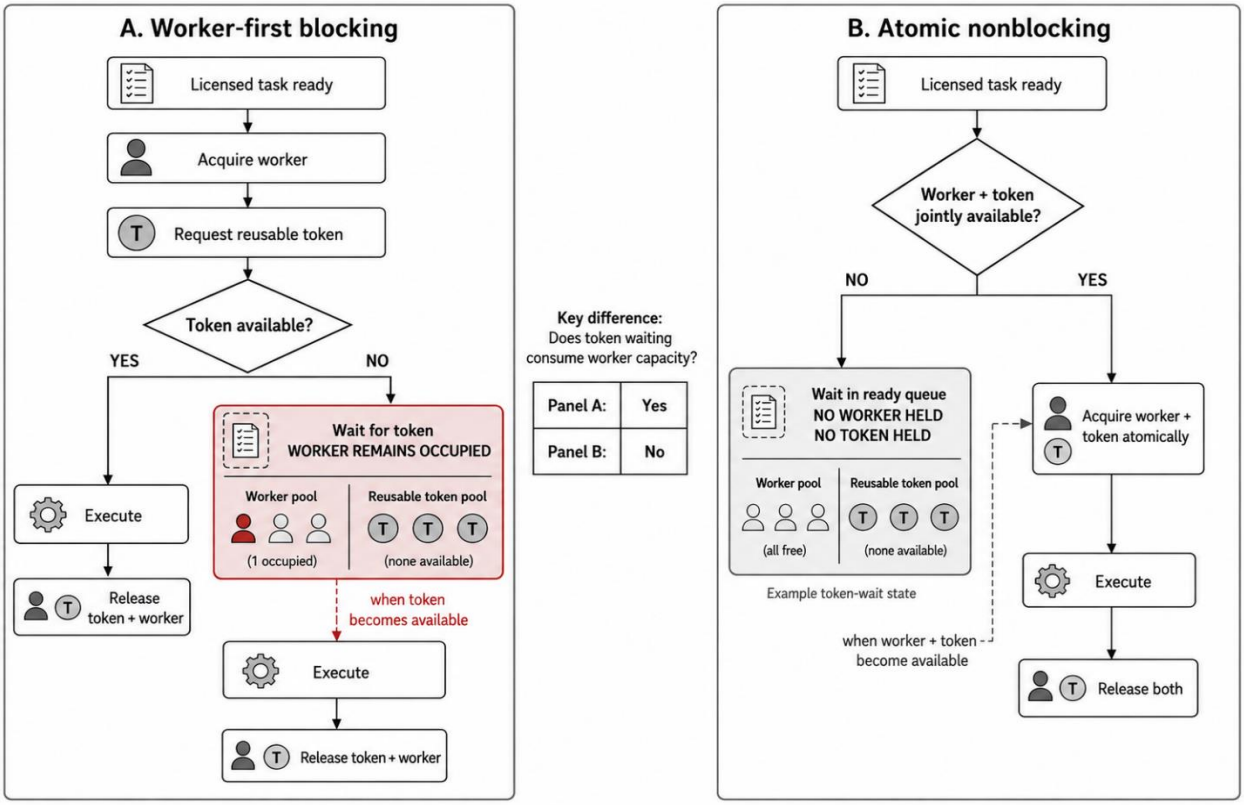


Рисунок 2 – Блокування за схемою «спочатку виконавець» порівняно з атомарним отриманням ресурсів

Політики активації. B0 негайно активує кожен процес, тобто $T_i^{admission} = T_i^{arrival}$.

Політика B1 активує новий робочий процес лише за виконання умови (4):

$$|A(t)| < N_{\max}, \quad (4)$$

в експерименті взято $N_{\max} = 12$.

Для B2a невиконаний обсяг токенової роботи активної множини та кандидата визначається за формулою (5):

$$W_L(t) = \sum_{j \in A(t)} \sum_{v \in U_j^L(t)} l_v \hat{h}_{jv}(t), \qquad W_i(t) = \sum_{v \in U_i^L(t)} l_v \hat{h}_{iv}(t). \quad (5)$$

Для B2a порогове значення задається як $K_W = \kappa_W L \bar{H}$, де $\bar{H} = 2$, та $\kappa_W = 4$. Кандидат B2a активується за виконання умови (6):

$$W_L(t) + W_i(t) \le K_W \quad (6)$$

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

B2b є політикою з позиційним вагуванням, натхненою підходом WLC. Рівень завдання для B2b визначається рекурсією (7):

$$r_v = \begin{cases} 1, & pred(v) = \varnothing, \\ 1 + \max\limits_{u \in pred(v)} r_u, & \text{otherwise.} \end{cases} \quad (7)$$

Для заданої топології $r_v = 1,2,2,3,4,4,5$, а позиційно зважений тиск робочого процесу j визначається за формулою (8):

$$P_j^{WLC}(t) = \sum_{v \in U_j^L(t)} \frac{l_v \hat{h}_{jv}(t)}{r_v}. \quad (8)$$

Кандидат B2b активується за виконання умови (9):

$$P_A^{WLC}(t) + P_i^{WLC}(t) \le K_{WLC}, \qquad K_{WLC} = \kappa_{WLC} L\bar{H}, \quad (9)$$

В експерименті узято $\kappa_{WLC} = 2$.

B3 – discounted time–to–resource (DTR). DTR дисконтує майбутню токенову роботу відповідно до структурної відстані до її готовності. Прогноз залишкової тривалості обслуговування незавершеного токенового завдання визначається за формулою (10)

$$\hat{h}_{iv}(t) = \begin{cases} \dfrac{a_{iv}(t) + \mu_v}{1 + \dfrac{2a_{iv}(t)}{\mu_v}}, & \text{якщо } v \text{ виконується,} \\ \mu_v, & \text{якщо обслуговування } v \text{ ще не розпочато,} \end{cases} \quad (10)$$

де $a_{iv}(t)$ – уже спостережена тривалість його обслуговування.

Структурна оцінка для попередника *u* задається рекурсією (11):

$$F_{iu}(t) = \begin{cases} 0, & \text{якщо } u \text{ завершено,} \\ \dfrac{a_{iu}(t) + \mu_u}{1 + \dfrac{2a_{iu}(t)}{\mu_u}}, & \text{якщо } u \text{ виконується,} \\ \mu_u, & \text{якщо } u \text{ готове або заблоковане в очікуванні токена,} \\ \mu_u + \max\limits_{p \in Pred(u)} F_{ip}(t), & \text{якщо } u \text{ ще не готове за залежностями САГ,} \end{cases} \quad (11)$$

де $max\varnothing = 0$.

Структурну відстань до готовності токенового завдання визначаємо за формулою (12):

$$\hat{d}_{iv}(t) = \begin{cases} 0, & \text{якщо } v \text{ готове, заблоковане, виконується або } \mathrm{Pred}(v) = \emptyset, \\ \max\limits_{u \in \mathrm{Pred}(v)} F_{iu}(t), & \text{інакше.} \end{cases}$$

Тиск DTR для робочого процесу *i* визначається за формулою (13):

$$P_i(t; h) = \sum_{v \in U_i^L(t)} l_v \, \hat{h}_{iv}(t) \exp\left(-\frac{\hat{d}_{iv}(t)}{h}\right), \quad (13)$$

Сумарний тиск DTR активної множини визначається за формулою (14):

$$P_A(t;h) = \sum_{j \in A(t)} P_j(t;h). (14)$$

За $A(t) \neq \varnothing$ кандидат у черзі FIFO активується тоді й лише тоді, коли виконується умова (15):

$$P_A(t;h) + P_i(t;h) \leq \theta Lh. (15)$$

Для порожньої активної множини перший кандидат активується для забезпечення прогресу; перший елемент черги FIFO не обходиться. Після активації диспетчеризація доводиться до фіксованої точки, після чого стан переоцінюється. В експерименті узято $h = 8$, $\theta = 1$.

DTR регулює активацію цілих робочих процесів. Порядок готових завдань задає незмінний планувальник. Рис. 3 описує алгоритм, який реалізує умову активації DTR за формулою (15), використовуючи тиск кандидата за формулою (13) і сумарний тиск активної множини за формулою (14).

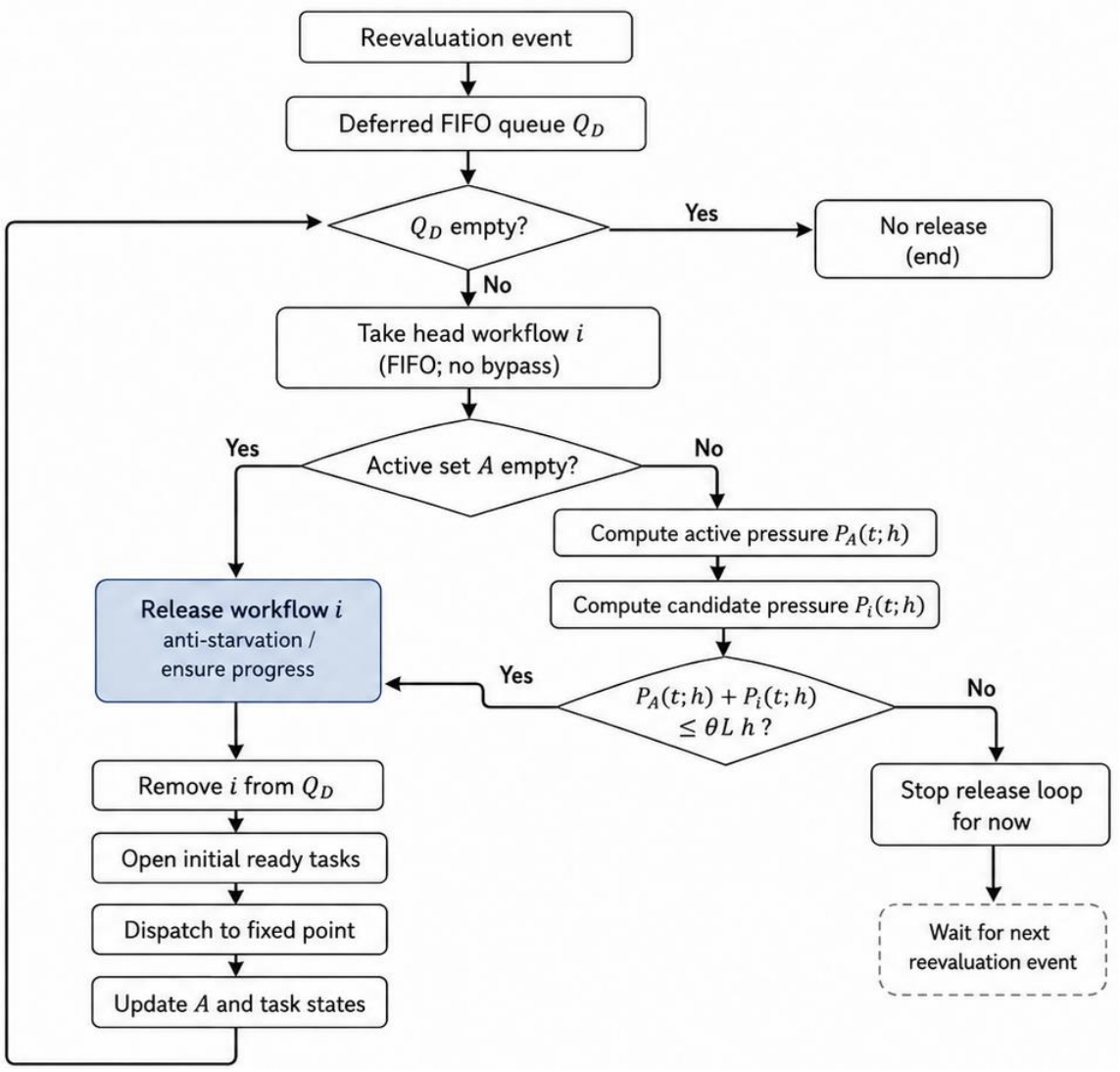


Рисунок 3 – Керування активацією робочих процесів за алгоритмом DTR

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

Експериментальний дизайн і статистичний аналіз. Підтверджувальна серія експериментів містить $8 \times 5 \times 20 = 800$ прогонів: для змішаного раннього/середнього/пізнього токена досліджено блокувальну й атомарну архітектури при $\rho_L = 0.50,0.90,0.94$; для однорідного середнього токена – обидві архітектури при $\rho_L = 0.90$. У кожній комірці порівнювали B0, B1, B2a, B2b і B3/DTR із 20 реплікаціями. За методом спільних випадкових чисел для різних політик повторно використовували однакові екзогенні траси надходжень, класів і часів обслуговування.

На початку кожної реплікації відсікали $N_w = 8000$ еквівалентів надходжень; вимірювання виконували на фіксованому горизонті $N_m = 30000$ еквівалентів надходжень. Відповідні часові горизонти становлять $T_w = N_w/\bar{\lambda}$ і $T_m = N_m/\bar{\lambda}$.

Статистичне підтвердження завершення прогріву на цій точці та асимптотичної збіжності p95 або p99 не встановлювалося; цільовою величиною є парний ефект політики на заданому скінченному горизонті, а точні стаціонарні квантилі не оцінюються. Після припинення надходжень систему спорожнювали для отримання повних наскрізних спостережень, але пропускну здатність і часові коефіцієнти використання обчислювали лише на спільному вимірювальному інтервалі.

Наскрізний час відгуку робочого процесу і визначається за формулою (16):

$$R_i = T_i^{completion} - T_i^{arrival}. \quad (16)$$

Для політики *P* відносно базової політики *B* парний відносний ефект для p95 визначається за формулою (17):

$$\delta_{P,B} = \frac{p95_P - p95_B}{p95_B}. \quad (17)$$

Від'ємний ефект означає поліпшення. 95% інтервали обчислювали парним непараметричним бутстрепом із 10 000 повторних вибірок; одиницею статистичного висновування є незалежна реплікація. Наперед зафіксована межа практичної значущості ± 5

Критерій еквівалентності пропускної здатності вимагає, щоб увесь 95% інтервал її відношення знаходився в інтервалі $[0.98,\ 1.02]$.

Результати перевірки гіпотез. Результати підтверджувальної кампанії далі розглянуто послідовно відповідно до H1–H3, зберігаючи p95 наскрізного часу відгуку як основну підтверджувальну метрику.

H1. За блокування «спочатку виконавець» керування активацією практично не впливає на p95 при $\rho_L = .50$, але дає великий виграш за високого навантаження. Таблиця 1 і рис. 4 показують зменшення приблизно на 27–29% при змішаному $\rho_L = .90$, на 34–36% при $\rho_L = .94$ та на 23–28% в однорідному сценарії $\rho_L = .90$.

Таблиця 1

Парний відносний ефект для p95 порівняно з B0 за блокування за схемою «спочатку виконавець»

Сценарій	B1	B2a	B2b	B3/DTR
Змішаний, $\rho_L = .50$	−0.08% [−0.12, −0.04]	−0.09% [−0.13, −0.05]	−0.05% [−0.10, −0.01]	−0.07% [−0.11, −0.04]
Змішаний, $\rho_L = .90$	−29.18% [−30.32, −28.06]	−28.75% [−29.83, −27.70]	−26.96% [−28.00, −25.93]	−28.62% [−29.71, −27.51]
Змішаний, $\rho_L = .94$	−36.35% [−38.3, −34.5]	−34.92% [≈−36.9, ≈−33.0]	−34.28% [≈−36.4, ≈−32.2]	−35.78% [≈−37.7, ≈−33.9]
Однорідний, токен у середині, $\rho_L = .90$	−27.62%	−27.23%	−23.18%	−26.98%

Значення – середні значення відносних ефектів на рівні реплікацій; у квадратних дужках наведено 95% довірчі інтервали парного бутстрепу.

Глобальна оболонка 95% меж відношення пропускної здатності повністю лежить усередині наперед заданої смуги еквівалентності $0.98, 1.02$.

За високого токенового навантаження ці результати підтримують H1 без практично виявлюваної втрати пропускної здатності.

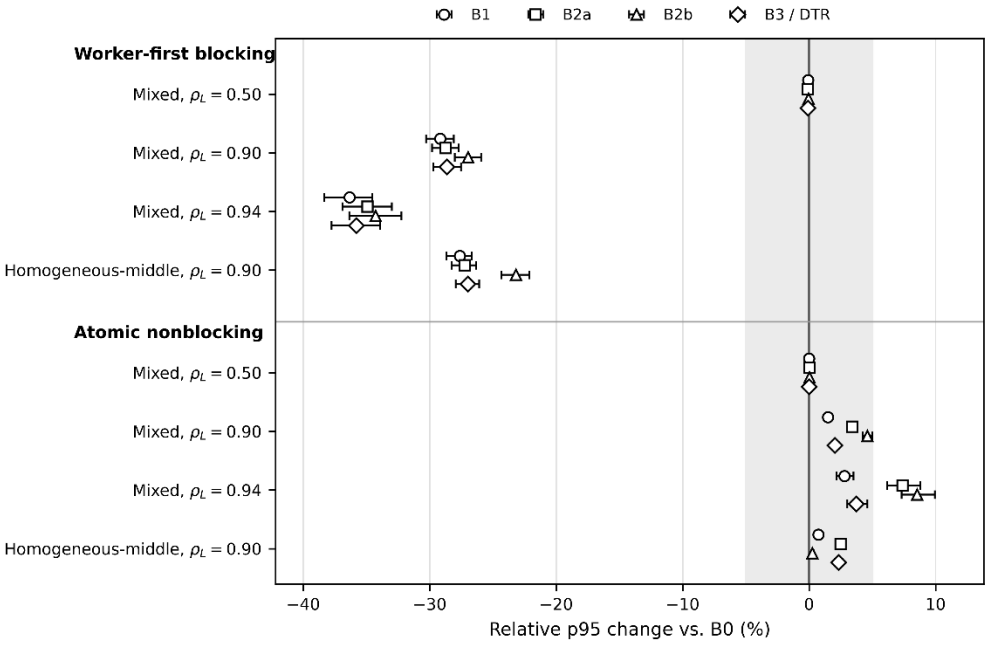


Рисунок 4 – Відносний ефект для p95 порівняно з необмеженою активацією

H2. За атомарного отримання великий виграш зникає. Для DTR ефекти p95 відносно B0 у трьох узгоджених високонавантажених сценаріях становлять $+2.05$–$+3.74$ і

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

залишаються в межах ±5%. Контрасти між архітектурами, що становлять приблизно 29–40 відсоткових пунктів, наведено в Таблиці 2 і на рис. 5.

Таблиця 2

Контрасти DTR між архітектурами отримання ресурсів

Узгоджене навантаження	Ефект DTR за блокування	Ефект DTR за атомарного отримання	Блокування − атомарне отримання
Змішане, $\rho_L = .90$	−28.62%	+2.05%	−30.67 в.п. [−31.87, −29.47]
Змішане, $\rho_L = .94$	−35.78%	+3.74%	−39.52 в.п. [−41.88, −37.41]
Однорідне, токен у середині, $\rho_L = .90$	−26.98%	+2.33%	−29.32 в.п. [−30.43, −28.28]

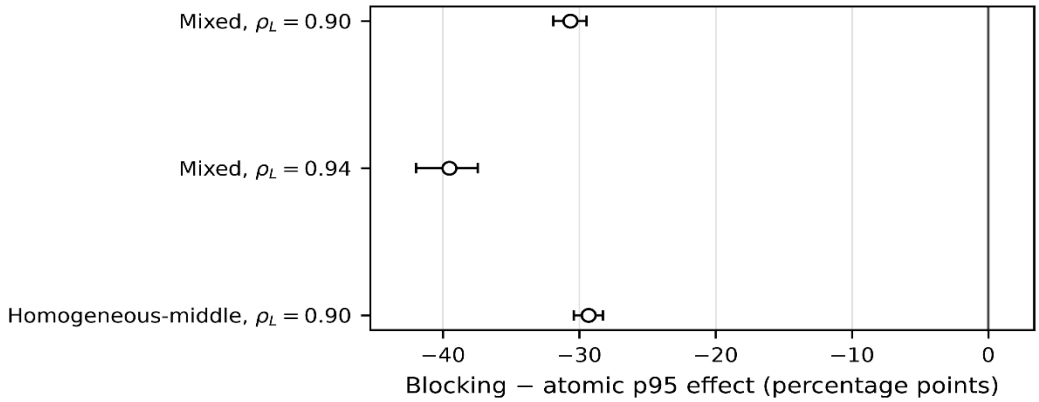


Рисунок 5 – Межа за архітектурою отримання ресурсів

За змішаного атомарного $\rho_L = .94$ B2a і B2b погіршують p95 приблизно на 7.4–8.5%, тоді як для B1 і DTR погіршення становить приблизно 2.8–3.7%. Цей результат підтримує H2: великий виграш виникає за можливості токенового очікування утримувати виконавця.

H3. Практична перевага DTR над сильнішими простими контролерами не підтверджується. Таблиця 3 та рис. 6 показують, що в трьох високонавантажених блокувальних сценаріях DTR приблизно на 0.8–0.9% повільніший за B1, відрізняється від B2a не більш як на 1.3% за модулем, а його найбільша перевага над B2b становить 4.94%, тобто не перевищує наперед заданої межі 5%.

Таблиця 3

Ефект DTR для p95 відносно простіших засобів керування
в блокувальних сценаріях із високим навантаженням

Сценарій	проти B1	проти B2a	проти B2b
Змішаний, $\rho_L = .90$	+0.79% [0.54, 1.03]	+0.18% [−0.20, 0.56]	−2.27% [−2.65, −1.92]
Змішаний, $\rho_L = .94$	+0.90% [0.37, 1.44]	−1.30% [−2.03, −0.59]	−2.26% [−2.84, −1.71]
Однорідний, токен у середині, $\rho_L = .90$	+0.89% [0.60, 1.19]	+0.35% [0.10, 0.57]	−4.94% [−5.30, −4.51]

Додатні значення означають, що DTR повільніший за порівнювану політику. Практична перевага, сформульована в H3, не підтверджується.

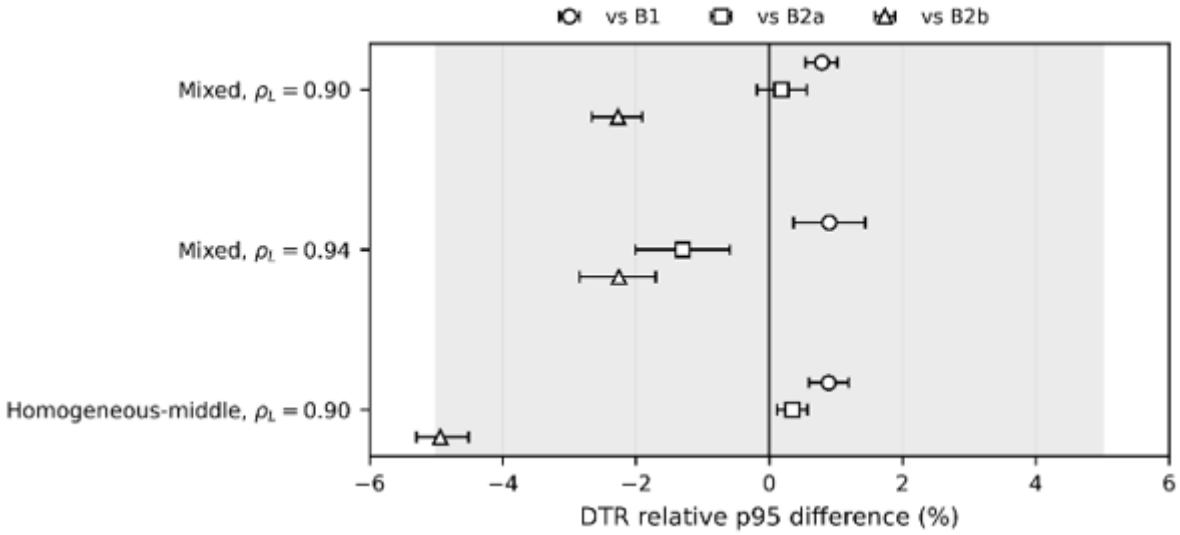


Рисунок 6 – DTR порівняно з простішими засобами керування

Вторинна p99–метрика дає той самий якісний висновок. У блокувальних сценаріях високого навантаження зменшення для політик керованої активації становлять приблизно 29–37%, тоді як атомарні ефекти DTR залишаються малими й додатними. p99 є вторинною метрикою і не замінює p95 як основну підтверджувальну кінцеву точку.

Механізм додатково характеризують метрики стану в Таблиці 4 та на рис. 7.

В обох високонавантажених режимах DTR переносить частину очікування до етапу допуску, зменшує активний і загальний WIP та частку потужності виконавців, зайняту заблокованими токеновими завданнями, тоді як використання токенів практично не змінюється. Точні значення наведено в Таблиці 4. Ці дані узгоджуються з механізмом зменшення шкідливого утримання виконавців, але не становлять повної причинної декомпозиції.

Обговорення результатів і загрози валідності. H1 стосується режимів, у яких очікування одного дефіцитного ресурсу споживає інший; результат не узагальнюється

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

на довільні обмеження допуску. За блокування за схемою «спочатку виконавець» токенове очікування зменшує доступний пул виконавців; контроль WIP обмежує кількість робочих процесів, здатних одночасно створювати цей стан. Саме тому за низького токенового навантаження ефект майже відсутній, а за високого стає великим.

Таблиця 4

Метрики стану для B0 і DTR за змішаного блокування
за схемою «спочатку виконавець»

Метрика	$\rho_L = .90$ B0	$\rho_L = .90$ DTR	$\rho_L = .94$ B0	$\rho_L = .94$ DTR
Середній p95 часу відгуку	41.91	29.77	78.45	49.35
Середній p99 часу відгуку	59.73	40.18	103.69	65.05
Середній активний WIP	14.77	9.06	25.97	10.11
Середній відкладений WIP	0.00	3.32	0.00	8.80
Середній загальний WIP	14.77	12.37	25.97	18.91
Частка потужності виконавців у блокуванні	0.219	0.194	0.270	0.235
Коефіцієнт використання токенів	0.89934	0.89934	0.94065	0.94063
Середня затримка допуску	0.00	3.68	0.00	9.35
Середній час в активній системі	16.40	10.06	27.59	10.75
Пропускна здатність у вимірювальному інтервалі	0.90025	0.90025	0.94012	0.94018

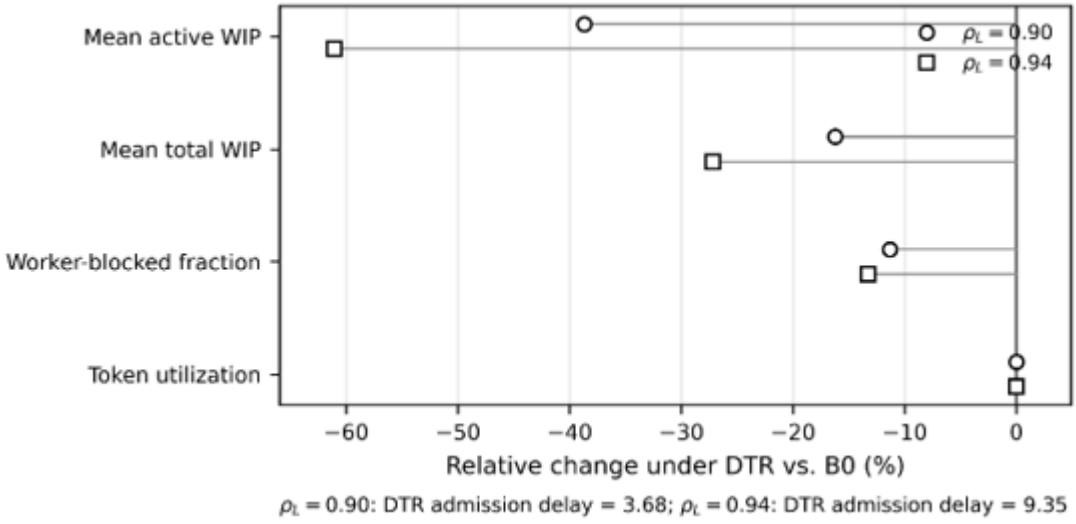


Рисунок 7 – Метрики стану, що характеризують механізм

Атомарний експеримент ізолює роль утримання виконавця: місткості й навантаження зберігаються, а завдання без доступного токена не займає виконавця. Зникнення великого виграшу та виразні контрасти між архітектурами підтримують висновок, що ефект виникає через зв'язування ресурсів; самого дефіциту токенів для нього недостатньо. За відсутності такого зв'язування надмірне обмеження активації може лише додати затримку допуску, що видно для B2a/B2b при атомарному $\rho_L = .94$.

Для дослідженої топології та режимів навантаження DTR не дає практично значущого виграшу, який виправдовував би додаткову структурну інформацію. B1 та B2a відновлюють практично той самий основний виграш. Основний результат визначається обсягом активної роботи та семантикою отримання ресурсів; точніший структурний прогноз не забезпечує додаткової практично значущої переваги. Це не доводить, що керування з урахуванням структури САГ не може бути корисним для інших топологій, кількох типів ресурсів або інших розподілів тривалості.

Основними обмеженнями є синтетична семивершинна топологія; один тип ресурсу багаторазового використання; $C = 12$, $L = 2$; часи обслуговування з розподілом Erlang–2; відсутність відмов, повторних спроб, еластичного масштабування, мережевої конкуренції та деталей конкретного ліцензійного сервера. Результати отримано за допомогою дискретно–подійного моделювання; промислове розгортання не оцінювалося.

Параметри контролерів було налаштовано на незалежних трасах до початку підтверджувального етапу, але висновки залишаються умовними щодо цих налаштувань. Інтерпретація хвостових квантилів обмежена заданим скінченним горизонтом.

Інженерна рекомендація стосується архітектури отримання ресурсів. За можливості дефіцитну обчислювальну потужність слід виділяти лише тоді, коли може бути задоволена вимога до обмеженого ресурсу багаторазового використання. Якщо блокування за схемою «спочатку виконавець» неминуче, першим засобом протидії доцільно розглядати просте обмеження активного WIP або токенової роботи; складніше керування з урахуванням структури САГ потребує окремих доказів додаткового практичного виграшу.

Висновки. Керування активацією робочих процесів істотно зменшує хвостовий час відгуку в дослідженій відкритій САГ–системі за тієї умови, що очікування обмеженого ресурсу багаторазового використання утримує дефіцитну потужність виконавця. За високого токенового навантаження контрольовані політики зменшують p95 приблизно на 23–36% без практично виявлюваної втрати пропускної здатності; за низького навантаження ефект практично відсутній.

Атомарне отримання ресурсів усуває великий виграш: ефекти DTR залишаються в межах ±5%, а контрасти між блокувальною та атомарною архітектурами становлять приблизно 29–40 відсоткових пунктів. Це підтримує H2 і визначає зв’язування виконавця з ресурсом як межу дії досліджуваного механізму.

DTR не демонструє загальної практично значущої переваги над B1 або B2a; практична перевага, сформульована в H3, не підтверджується. Коли блокування неможливо усунути архітектурно, просте керування активним WIP або незавершеною токеновою роботою забезпечує більшу частину спостережуваного виграшу без додаткової інформаційної складності DTR.

Data Availability Statement (Заява про доступність даних). Дані дослідження доступні за запитом. Пошта для контактів: Myronov.Y.A@nmu.one.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

ЛІТЕРАТУРА

1. Beránek J., Böhm S., Cima V. Analysis of workflow schedulers in simulated distributed environments // The Journal of Supercomputing. 2022. Vol. 78, no. 13. P. 15154–15180. DOI: https://doi.org/10.1007/s11227–022–04438–y.

2. McDonald J., Dobbs J., Wong Y. C., Ferreira da Silva R., Casanova H. An exploration of online–simulation–driven portfolio scheduling in Workflow Management Systems // Future Generation Computer Systems. 2024. Vol. 161. P. 345–360.
DOI: https://doi.org/10.1016/j.future.2024.07.005.

3. Bhattarai R., Pritchard H., Ghafoor S. Dynamic Resource Management for Elastic Scientific Workflows using PMIx // 2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). IEEE, 2024. P. 686–695.
DOI: https://doi.org/10.1109/IPDPSW63119.2024.00131.

4. Tsvirkun L., Myronov Y. Challenges and Specificities of Adopting Continuous Integration within Scalable Cloud Environments // 2023 IEEE 18th International Conference on Computer Science and Information Technologies (CSIT), Lviv, Ukraine, 19–21 October 2023.
IEEE, 2023. P. 1–4. DOI: https://doi.org/10.1109/CSIT61576.2023.10324010.

5. Coleman T., Casanova H., Pottier L., Kaushik M., Deelman E., Ferreira da Silva R. WfCommons: A framework for enabling scientific workflow research and development // Future Generation Computer Systems. 2022. Vol. 128. P. 16–27.
DOI: https://doi.org/10.1016/j.future.2021.09.043.

6. Casanova H., Ferreira da Silva R., Tanaka R., Pandey S., Jethwani G., Koch W., Albrecht S., Oeth J., Suter F. Developing accurate and scalable simulators of production workflow management systems with WRENCH // Future Generation Computer Systems. 2020. Vol. 112. P. 162–175. DOI: https://doi.org/10.1016/j.future.2020.05.030.

7. Qiao Q., Chen L., Cai H., Zhang H., Yao Z. An online workflow scheduling algorithm considering license limitation in heterogeneous environment // Concurrency and Computation: Practice and Experience. 2023. Vol. 35, no. 2. e7452.
DOI: https://doi.org/10.1002/cpe.7452.

8. Gulhan A. B., Akbulut G. G., Amritkar A., Sampson J., Honavar V. G., Focht A., Pavloski C., Kandemir M. T. License Forecasting and Scheduling for HPC // Proceedings of the 2023 31st International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS). IEEE Computer Society, 2023. P. 1–8. DOI: https://doi.org/10.1109/MASCOTS59514.2023.10387539.

9. Noaman M., Dasgupta S., Gerndt M. Deadline Miss Minimization Scheduling for License–Constrained CAE Jobs in Hybrid Cloud Infrastructure // Job Scheduling Strategies for Parallel Processing: 28th International Workshop, JSSPP 2025, Milan, Italy, June 3–4, 2025, Revised Selected Papers / eds. D. Klusáček, J. Corbalán, G. P. Rodrigo. Cham : Springer, 2026. Lecture Notes in Computer Science. Vol. 16210. P. 327–347. DOI: https://doi.org/10.1007/978–3–032–10507–3_17.

10. Thürer M., Ma L., Stevenson M. Workload Control order release in general and pure flow shops with limited buffer size induced blocking: an assessment by simulation // International Journal of Production Research. 2021. Vol. 59, no. 8. P. 2558–2569.

DOI: https://doi.org/10.1080/00207543.2020.1735667.
11. Thürer M., Stevenson M. Order release, dispatching and resource assignment in multiple resource–constrained job shops: an assessment by simulation // International Journal of Production Research. 2022. Vol. 60, no. 12. P. 3669–3681.
DOI: https://doi.org/10.1080/00207543.2021.1930240.
12. Calle M., Gonzalez–R P. L., Andrade J. L., Leon–Blanco J. M., Canca D. Workload control in dual–resource constrained flexible job shops: a simulation analysis // International Journal of Production Research. 2025. Vol. 63, no. 20. P. 7363–7385.
DOI: https://doi.org/10.1080/00207543.2025.2496976.
13. Fernandes N. O., Thürer M., Stevenson M. Direct Workload Control: simplifying continuous order release // International Journal of Production Research. 2022. Vol. 60, no. 4. P. 1424–1437. DOI: https://doi.org/10.1080/00207543.2020.1857451.

REFERENCES

1. Ber Beránek, J., Böhm, S., & Cima, V. (2022). Analysis of workflow schedulers in simulated distributed environments. *The Journal of Supercomputing, 78*(13), 15154–15180. https://doi.org/10.1007/s11227–022–04438–y
2. McDonald, J., Dobbs, J., Wong, Y. C., Ferreira da Silva, R., & Casanova, H. (2024). An exploration of online–simulation–driven portfolio scheduling in workflow management systems. *Future Generation Computer Systems, 161,* 345–360.
https://doi.org/10.1016/j.future.2024.07.005
3. Bhattarai, R., Pritchard, H., & Ghafoor, S. (2024). Dynamic resource management for elastic scientific workflows using PMIx. *In 2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)* (pp. 686–695).
IEEE. https://doi.org/10.1109/IPDPSW63119.2024.00131
4. Tsvirkun, L., & Myronov, Y. (2023). Challenges and specificities of adopting continuous integration within scalable cloud environments. *In 2023 IEEE 18th International Conference on Computer Science and Information Technologies (CSIT)* (pp. 1–4).
IEEE. https://doi.org/10.1109/CSIT61576.2023.10324010
5. Coleman, T., Casanova, H., Pottier, L., Kaushik, M., Deelman, E., & Ferreira da Silva, R. (2022). WfCommons: A framework for enabling scientific workflow research and development. *Future Generation Computer Systems, 128,* 16–27.
https://doi.org/10.1016/j.future.2021.09.043
6. Casanova, H., Ferreira da Silva, R., Tanaka, R., Pandey, S., Jethwani, G., Koch, W., Albrecht, S., Oeth, J., & Suter, F. (2020). Developing accurate and scalable simulators of production workflow management systems with WRENCH. *Future Generation Computer Systems, 112,* 162–175. https://doi.org/10.1016/j.future.2020.05.030
7. Qiao, Q., Chen, L., Cai, H., Zhang, H., & Yao, Z. (2023). An online workflow scheduling algorithm considering license limitation in heterogeneous environment. *Concurrency and Computation: Practice and Experience, 35*(2), e7452. https://doi.org/10.1002/cpe.7452
8. Gulhan, A. B., Akbulut, G. G., Amritkar, A., Sampson, J., Honavar, V. G., Focht, A., Pavloski, C., & Kandemir, M. T. (2023). License forecasting and scheduling for HPC. *In Pro-*

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

ceedings of the 2023 31st International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS) (pp. 1–8). IEEE Computer Society. https://doi.org/10.1109/MASCOTS59514.2023.10387539
9. Noaman, M., Dasgupta, S., & Gerndt, M. (2026). Deadline miss minimization scheduling for license–constrained CAE jobs in hybrid cloud infrastructure. In D. Klusáček, J. Corbalán, & G. P. Rodrigo (Eds.), Job scheduling strategies for parallel processing: 28th International Workshop, JSSPP 2025, Milan, Italy, June 3–4, 2025, revised selected papers *(Lecture Notes in Computer Science, Vol. 16210,* pp. 327–347). Springer. https://doi.org/10.1007/978–3–032–10507–3_17
10. Thürer, M., Ma, L., & Stevenson, M. (2021). Workload control order release in general and pure flow shops with limited buffer size induced blocking: An assessment by simulation. *International Journal of Production Research, 59*(8), 2558–2569.
https://doi.org/10.1080/00207543.2020.1735667
11. Thürer, M., & Stevenson, M. (2022). Order release, dispatching and resource assignment in multiple resource–constrained job shops: An assessment by simulation. *International Journal of Production Research, 60*(12), 3669–3681.
https://doi.org/10.1080/00207543.2021.1930240
12. Calle, M., Gonzalez–R, P. L., Andrade, J. L., Leon–Blanco, J. M., & Canca, D. (2025). Workload control in dual–resource constrained flexible job shops: A simulation analysis. *International Journal of Production Research, 63*(20), 7363–7385.
https://doi.org/10.1080/00207543.2025.2496976
13. Fernandes, N. O., Thürer, M., & Stevenson, M. (2022). Direct workload control: Simplifying continuous order release. *International Journal of Production Research, 60*(4), 1424–1437. https://doi.org/10.1080/00207543.2020.1857451

Received 11.06.2026.
Accepted 13.08.2026.
Published 30.10.2026

Workflow activation control under blocking caused by limited reusable resource in distributed dag execution systems

Analysis of recent research and publications. Distributed DAG execution systems may contain tasks that require both ordinary compute capacity and a finite reusable resource, such as a floating software–license token. Existing research addresses workflow scheduling, dynamic resource management, simulation–based workflow–system evaluation, and scheduling under software–license constraints. Workload Control studies also show that limiting the amount of work admitted to an active system can mitigate blocking and interactions among multiple constrained resources. What remains insufficiently characterized is whether a corresponding benefit in an open stochastic DAG execution system is specifically caused by a task retaining compute capacity while waiting for a second resource, and whether knowledge of downstream DAG structure provides a practically meaningful additional benefit for workflow–release control.

Research objective. The study aims to determine when workflow activation control reduces tail end–to–end response time in the presence of a finite reusable resource; to identify the architectural boundary of this effect by comparing blocking and atomic resource acquisi-

tion; and to evaluate the marginal value of a DAG–aware discounted time–to–resource (DTR) activation heuristic relative to simpler active–WIP and token–work controls.

Presentation of the main research material. A discrete–event model of an open execution system is used with $C = 12$ *worker slots,* $L = 2$ *reusable–resource tokens, and seven–task DAG workflows whose single token–requiring operation occurs at an early, middle, or late DAG position. Two resource–acquisition semantics are compared. Under worker–first blocking, a token–requiring task first acquires a worker and continues to retain that worker while waiting for a token. Under atomic nonblocking acquisition, the task is dispatched only when both resources are simultaneously available. Five activation policies are evaluated: unlimited activation B0, an active–WIP cap B1, outstanding token–work control B2a, a position–weighted WLC–inspired policy B2b, and DAG–aware DTR B3. The confirmatory experiment comprises eight scenarios, five policies, and 20 paired replications per scenario, for 800 simulation runs in total, using common random numbers across policies. The primary endpoint is workflow p95 end–to–end response time. Paired 95% intervals are estimated using a nonparametric bootstrap, with a prespecified ±5% practical–significance criterion and a throughput–equivalence interval of* $0.98, 1.02$.

Under high token load and worker–first blocking, controlled activation reduces p95 by approximately 23–36% relative to unlimited activation while throughput remains practically unchanged; at low token load, the effect is near zero. Under atomic acquisition, DTR effects remain within the ±5% practical region, and paired blocking–versus–atomic DTR contrasts are approximately 29–40 percentage points. DTR does not show a generally practically meaningful advantage over the stronger simpler controls. State metrics are consistent with reductions in active and total WIP and in worker capacity held by token–blocked tasks, while token utilization remains essentially unchanged.

Conclusions. The observed tail–latency benefit arises from coupling reusable–resource scarcity to compute capacity when a waiting task retains a worker. Where architecture permits, compute resources should therefore be allocated only when the reusable–resource requirement can be satisfied. Where worker–first blocking cannot be eliminated, simple active–WIP or outstanding token–work control captures most of the available benefit. The hypothesis that DAG–aware DTR provides a practically meaningful general performance advantage is not supported by the confirmatory experiment. Under the tested conditions, the additional DAG–structural information provides only limited marginal performance benefit.

Keywords: distributed workflow systems; DAG execution; workload activation; work in process; reusable resources; software licenses; resource blocking; tail latency; discrete–event simulation; cloud computing.

ISSN 1562-9945 (Print)
ISSN 2707-7977 (Online)

Цвіркун Леонід Іванович – професор кафедри інформаційних технологій та комп'ютерної інженерії, Національний технічний університет «Дніпровська політехніка», Дніпро, Україна.
ORCID: https://orcid.org/0000–0002–5568–5516.
Миронов Юрій Анатолійович – аспірант кафедри інформаційних технологій та комп'ютерної інженерії, Національний технічний університет «Дніпровська політехніка», Дніпро, Україна.
ORCID: https://orcid.org/0009–0006–0675–8033.

Tsvirkun Leonid – Professor of Department of Information Technology and Computer Engineering, Dnipro University of Technology, Dnipro, Ukraine.
ORCID: https://orcid.org/0000–0002–5568–5516.
Myronov Yurii – Post–graduate student of the Department of Information Technology and Computer Engineering, Dnipro University of Technology, Dnipro, Ukraine.
ORCID: https://orcid.org/0009–0006–0675–8033.