
Práctica 4: Procesos de Negocio

Métodos Formales para la Ingeniería del Software

Carmen González Ortega, Altair Bueno Calvente

2/12/2022

Contents

2.1 Utiliza el comando search para buscar estados del proceso a lo largo de su ejecución limitada a 100 unidades de tiempo en que no haya ningún token en el conjunto del atributo tokens. Explica el resultado.	2
Finalización en nodo final "n05" - estado 21	3
Finalización en nodo final "n08" - estado 1366	3
Finalización en nodo final "n25" - estado 1643	4
2.2 Utiliza el comando search para verificar si hay situaciones de bloqueo para ejecuciones del proceso antes del transcurso de 100 unidades de tiempo. Explica el resultado.	6

2.1 Utiliza el comando search para buscar estados del proceso a lo largo de su ejecución limitada a 100 unidades de tiempo en que no haya ningún token en el conjunto del atributo tokens. Explica el resultado.

```
search PROCESS =>* < o : Process | tokens: empty, gtime: T:Time,
Atts:AttributeSet >
```

```
Maude> search PROCESS =>* < o : Process | tokens: empty, gtime: T:Time, Atts:AttributeSet > .
search in BPMN-EX : PROCESS =>* < o : Process | Atts:AttributeSet,tokens: empty,gtime: T:Time > .

Solution 1 (state 21)
states: 22 rewrites: 272 in 0ms cpu (8ms real) (~ rewrites/second)
Atts:AttributeSet --> nodes: (start(id("n00"), id("f00")), end(id("n05"), id("f06")), end(id("n08"), id("f13")), end(
id("n25"), id("f28")), split(id("n04"), exclusive, id("f03"), (id("f04"), id("f05"), id("f06")), split(id("n10"),
exclusive, id("f08"), id("f09"), id("f10")), split(id("n13"), parallel, id("f29"), id("f17"), id("f21")),
split(id("n15"), parallel, id("f11"), id("f14"), id("f15"), id("f16")), split(id("n20"), exclusive, id("f22"), (
id("f23"), id("f24")), merge(id("n02"), exclusive, id("f01"), id("f04"), id("f02")), merge(id("n17"), parallel,
id("f15"), id("f17")), merge(id("n19"), parallel, id("f19"), id("f20")), merge(id("n23"),
exclusive, id("f25"), id("f26")), merge(id("n24"), parallel, id("f21"), id("f27")), id("f28")), task(
id("n01"), "Sign in", id("f00"), id("f01"), 2, empty), task(id("n03"), "Search products", id("f02"), id("f03"), 2,
empty), task(id("n06"), "Make an order", id("f05"), id("f07"), 2, empty), task(id("n07"), "Fill in feedback form",
id("f12"), id("f13"), 2, empty), task(id("n09"), "Check availability", id("f07"), id("f08"), 2, id("clerk")), task(
id("n11"), "Cancel order", id("f09"), id("f12"), 2, id("clerk")), task(id("n12"), "Pay for order", id("f14"), id(
"f29"), 2, empty), task(id("n14"), "Confirm order", id("f10"), id("f11"), 2, id("clerk")), task(id("n16"), id(
"Prepare parcel", id("f16"), id("f19"), 2, id("worker")), task(id("n18"), "Payment validation", id("f18"), id(
"f20"), 2, empty), task(id("n21"), "Deliver by drone", id("f23"), id("f25"), 2, id("drone")), task(id("n22"),
"Deliver by car", id("f24"), id("f26"), 2, id("car"), id("delivery man")), flows: (flow(id("f00"), 2), flow(id(
"f01"), 2), flow(id("f02"), 2), flow(id("f03"), 2), flow(id("f04"), 2), flow(id("f05"), 2), flow(id("f06"), 2),
flow(id("f07"), 2), flow(id("f08"), 2), flow(id("f09"), 2), flow(id("f10"), 2), flow(id("f11"), 2), flow(id("f12"),
2), flow(id("f13"), 2), flow(id("f14"), 2), flow(id("f15"), 2), flow(id("f16"), 2), flow(id("f17"), 2), flow(id(
"f18"), 2), flow(id("f19"), 2), flow(id("f20"), 2), flow(id("f21"), 2), flow(id("f22"), 2), flow(id("f23"), 2),
flow(id("f24"), 2), flow(id("f25"), 2), flow(id("f26"), 2), flow(id("f27"), 2), flow(id("f28"), 2), flow(id("f29"),
2)),resources: (resource(id("car"), 1, 1), resource(id("clerk"), 1, 1), resource(id("delivery man"), 1, 1),
resource(id("drone"), 1, 1), resource(id("worker"), 1, 1))
T:Time --> 14
```

Figure 1: Principio de ejecución, primer resultado

```
Solution 26 (state 1643)
states: 1644 rewrites: 43595 in 625ms cpu (2108ms real) (69752 rewrites/second)
Atts:AttributeSet --> nodes: (start(id("n00"), id("f00")), end(id("n05"), id("f06")), end(id("n08"), id("f13")), end(
id("n25"), id("f28")), split(id("n04"), exclusive, id("f03"), (id("f04"), id("f05"), id("f06")), split(id("n10"),
exclusive, id("f08"), id("f09"), id("f10")), split(id("n13"), parallel, id("f29"), id("f17"), id("f21")),
split(id("n15"), parallel, id("f11"), id("f14"), id("f15"), id("f16")), split(id("n20"), exclusive, id("f22"), (
id("f23"), id("f24")), merge(id("n02"), exclusive, id("f01"), id("f04"), id("f02")), merge(id("n17"), parallel,
id("f15"), id("f17")), merge(id("n19"), parallel, id("f19"), id("f20")), merge(id("n23"),
exclusive, id("f25"), id("f26")), merge(id("n24"), parallel, id("f21"), id("f27")), id("f28")), task(
id("n01"), "Sign in", id("f00"), id("f01"), 2, empty), task(id("n03"), "Search products", id("f02"), id("f03"), 2,
empty), task(id("n06"), "Make an order", id("f05"), id("f07"), 2, empty), task(id("n07"), "Fill in feedback form",
id("f12"), id("f13"), 2, empty), task(id("n09"), "Check availability", id("f07"), id("f08"), 2, id("clerk")), task(
id("n11"), "Cancel order", id("f09"), id("f12"), 2, id("clerk")), task(id("n12"), "Pay for order", id("f14"), id(
"f29"), 2, empty), task(id("n14"), "Confirm order", id("f10"), id("f11"), 2, id("clerk")), task(id("n16"), id(
"Prepare parcel", id("f16"), id("f19"), 2, id("worker")), task(id("n18"), "Payment validation", id("f18"), id(
"f20"), 2, empty), task(id("n21"), "Deliver by drone", id("f23"), id("f25"), 2, id("drone")), task(id("n22"),
"Deliver by car", id("f24"), id("f26"), 2, id("car"), id("delivery man")), flows: (flow(id("f00"), 2), flow(id(
"f01"), 2), flow(id("f02"), 2), flow(id("f03"), 2), flow(id("f04"), 2), flow(id("f05"), 2), flow(id("f06"), 2),
flow(id("f07"), 2), flow(id("f08"), 2), flow(id("f09"), 2), flow(id("f10"), 2), flow(id("f11"), 2), flow(id("f12"),
2), flow(id("f13"), 2), flow(id("f14"), 2), flow(id("f15"), 2), flow(id("f16"), 2), flow(id("f17"), 2), flow(id(
"f18"), 2), flow(id("f19"), 2), flow(id("f20"), 2), flow(id("f21"), 2), flow(id("f22"), 2), flow(id("f23"), 2),
flow(id("f24"), 2), flow(id("f25"), 2), flow(id("f26"), 2), flow(id("f27"), 2), flow(id("f28"), 2), flow(id("f29"),
2)),resources: (resource(id("car"), 1, 1), resource(id("clerk"), 1, 1), resource(id("delivery man"), 1, 1),
resource(id("drone"), 1, 1), resource(id("worker"), 1, 1))
T:Time --> 98
No more solutions.
```

Figure 2: Final de ejecución, último resultado

Para esta primera búsqueda se ha realizado un search con resultado un proceso tal que el conjunto de tokens esté vacío, es decir, se haya llegado a algún nodo de tipo end.

Nuestra especificación original de las reglas de reescritura es no terminante, ya que un proceso puede

entrar en un bucle infinito de ejecución aumentando el tiempo hasta el infinito. Para poder limitar la búsqueda de soluciones, se ha modificado la regla [tick] para que el reloj solo pueda alcanzar el valor de 100 ud de tiempo.

Como se puede observar, se encuentran 26 resultados, debido a que la tarea "Search products" está dentro en un bucle, haciendo que haya diversos estados de salida aunque sólo existan tres nodos finales. Sin embargo, podemos observar como todas estas soluciones acaban en en alguno de estos nodos finales.

Mostramos a continuación las salidas del comando `show path labels` para varios estados solución con comentarios adicionales.

Finalización en nodo final "n05" - estado 21

```
Maude> show path labels 21 .
start
tick
enter-task *** ("Sign in")
tick
exit-task
tick
merge-exclusive *** (inicio del bucle - 1 iteracion)
tick
enter-task *** ("Search products")
tick
exit-task
tick
split-exclusive *** (fin del bucle)
tick
end
```

Finalización en nodo final "n08" - estado 1366

```
Maude> show path labels 1366 .
start
tick
enter-task *** ("Sign in")
tick
exit-task
```

```
tick
merge-exclusive *** (inicio del bucle - 8 iteraciones)
tick
enter-task *** ("Search products")
tick
exit-task
tick
split-exclusive *** (fin del bucle)
tick
enter-task *** ("Make an order")
tick
exit-task
tick
enter-task *** ("Check availability")
tick
exit-task
tick
split-exclusive
tick
enter-task *** ("Cancel order")
tick
exit-task
tick
enter-task *** ("Fill in feedback form")
tick
exit-task
tick
end
```

Finalización en nodo final "n25" - estado 1643

```
Maude> show path labels 1643 .
start
tick
enter-task *** ("Sign in")
tick
exit-task
tick
```

```
merge-exclusive *** (inicio del bucle - 6 iteraciones)
tick
enter-task *** ("Search products")
tick
exit-task
tick
split-exclusive *** (fin del bucle)
tick
enter-task *** ("Make an order")
tick
exit-task
tick
enter-task *** ("Check availability")
tick
exit-task
tick
split-exclusive
tick
enter-task *** ("Confirm order")
tick
exit-task
tick
split-parallel
tick
enter-task *** ("Pay for order")
tick
enter-task *** ("Prepare parcel")
tick
exit-task
tick
exit-task
tick
split-parallel
tick
merge-parallel
tick
enter-task *** ("Payment validation")
tick
```

```

exit-task
tick
merge-parallel
tick
split-exclusive
tick
enter-task *** ("Deliver by drone")
tick
exit-task
tick
merge-exclusive
tick
merge-parallel
tick
end

```

2.2 Utiliza el comando search para verificar si hay situaciones de bloqueo para ejecuciones del proceso antes del transcurso de 100 unidades de tiempo. Explica el resultado.

```

search PROCESS =>! < o : Process | tokens: Tks:Set{Token}, gtime:
T:Time, Atts:AttributeSet > s.t. Tks:Set{Token} /= empty /\
mte(Tks:Set{Token}) + T:Time <= 100

```

```

=====
search in BPMN-EX : PROCESS =>! < o : Process | Atts:AttributeSet,tokens: Tks:Set{Token},gtime: T:Time > such that Tks:Set{Token}
  /= (empty).Set{Token} = true /\ T:Time + mte(Tks:Set{Token}) <= 100 = true .

No solution.
states: 1708  rewrites: 46080 in 628ms cpu (667ms real) (73359 rewrites/second)

```

Figure 3: Resultado del comando

Consideramos que un estado es de bloqueo si se cumplen las siguientes condiciones:

- No se pueden aplicar más reglas: El sistema no puede transitar
- Nos encontramos en un flujo, tarea o nodo no final (existen tokens por consumir en el conjunto de tokens)
- No hemos alcanzado la cota superior de 100ud de tiempo

Podemos observar como Maude no encuentra solución para la búsqueda, demostrando así que cualquier ejecución del modelo en 100ud de tiempo no se bloquea.

No obstante, esta verificación no nos proporciona información sobre el comportamiento del sistema más allá de dicha cota. Para solucionarlo, podemos crear una abstracción del sistema en el que retrasamos continuamente el tiempo global del sistema.