シャフル表現による Web システム動作系列の記述

阿部 真也
地方独立行政法人東京都立産業技術研究センター
abe.shinya@iri-tokyo.jp

概要　システムの動作テストにおいて，実行時のイベント系列が，開発者の意図する系列であるかを判定するのは重要である．イベント系列の記述には，一般に正規表現が用いられる．ところが，各々のイベントが非同期並行的に実行されるシステムでは，正規表現の記述能力では不十分であることが知られている．そこで本稿では，シャフル表現による非同期イベント系列の記述法を提案し，複数のユーザが非同期並行的にアクセスする Web システムの記述例を与える．

キーワード　Web システム，動作テスト，非同期並行イベント，シャフル表現

1　はじめに

システムの動作テストにおいて，実行時に発生したイベント系列が，開発者の意図するイベント系列であるかを判定するのは重要である．動作テストは自動化が望ましいが，そのためには，イベント系列のルールを，開発者が形式的に記述しなければならない．イベント系列の記述には，一般に正規表現が用いられる．その理由として，記述が容易であること，テスト自動化ツールによる解析が容易なことが挙げられる．

ところが，各々のイベントを非同期並行的に実行するシステムでは，正規表現の記述能力では不十分であることが知られている [1, 3, 3, 4]．たとえば，セマフォによる同期機構の動作を記述するには，現在の状態を保持する必要があるが，正規表現では数を記憶できないため，記述が困難である．また，イベント系列の並行性は，正規表現であっても記述可能であるが，多くの和を用いた表現になり簡潔ではない．

そこで筆者らは，シャフル表現 [5] を用いた非同期イベント系列の記述法 [6] を提案し，代表的な非同期問題に対する記述例を与えた．本稿では，これを発展させ，複数のユーザが非同期並行的にアクセスする Web システムの記述例を与える．

2　シャフル表現

ここでは，シャフル，シャフル閉包，シャフル表現，シャフル言語について述べる．

定義 2.1 シャフル

Σ を有限記号集合，λ を空語とする．このとき，シャフルを次のように定義する．

- 全ての $u \in \Sigma^*$ について，$u \odot \lambda = \lambda \odot = \{u\}$
- 全ての $u, v \in \Sigma^*, a, b \in \Sigma$ について，$au \odot bv = a(u \odot bv) \cup b(au \odot v)$

また，言語 L_1, L_2 について，シャフルを次のように定義する．

$$L_1 \odot L_2 = \bigcup_{u \in L_1, v \in L_2} u \odot v$$

定義 2.2 シャフル閉包

Σ を有限記号集合，λ を空語とする．このとき，言語 $L \subset \Sigma^*$ について，シャフル閉包を次のように定義する．ただし，$L^{\otimes 0} = \{\lambda\}$，$L^{\otimes i} = L^{\otimes i-1} \odot L$ である．

$$L^{\otimes} = \bigcup_{i=0}^{\infty} L^{\otimes i}$$

定義 2.3 シャフル表現

空集合 ϕ，空語 λ，記号 a はシャフル表現である．S_1，S_2 をシャフル表現とすると，積 S_1S_2，和 $S_1|S_2$，閉包 S_1^*，シャフル $S_1 \odot S_2$，シャフル閉包 $S_1^{\otimes}$ もシャフル表現である．

定義 2.4 シャフル言語

シャフル表現 S_1，S_2 から生成されるシャフル言語 $L(S)$ を次のように定義する．$L(\phi) = \phi$，$L(\lambda) = \{\lambda\}$，$L(a) = \{a\}$，$L(S_1S_2) = L(S_1)L(S_2)$，$L(S_1|S_2) = L(S_1) \cup L(S_2)$，$L(S_1^*) = (L(S_1))^*$，$L(S_1 \odot S_2) = L(S_1) \odot L(S_2)$，$L(S_1^{\otimes}) = (L(S_1))^{\otimes}$．

3　基本的な動作モデルの記述

ここでは，シャフル表現を用いて，基本的な動作モデルの記述例を与える．

モデル 1 キュー

初期状態が空のキューがある．このキューに対して，enqueue イベント（末尾にデータを追加する）や dequeue イベント（先頭のデータを取り出す）が非同期並行的に

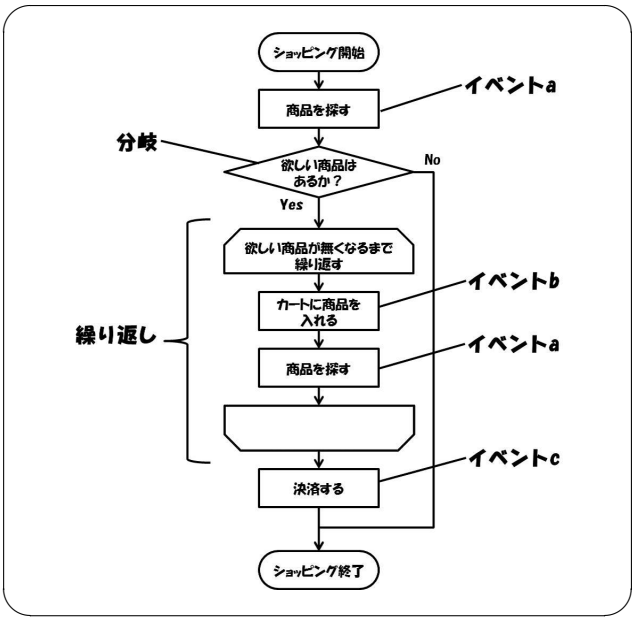

図 1 ショッピングサイトのフロー

実行されることを考える．このとき，キューにデータが 1 つ以上存在するときのみ，dequeue が可能でなければならないという実行条件を与えるとする．

enqueue イベントを e，dequeue イベントを d とすると，この実行条件の形式的記述は次のようになる．

キューの形式的記述

$$(ed)^{\otimes}$$

モデル 2 共有バッファ

高々1 つのデータを格納できる共有バッファがある．このバッファに対して，read イベント（データを読み込む）や write イベント（データを書き込む）が非同期並行的に実行されることを考える．このとき，read と write あるいは write と write が同時に起こってはいけないという実行条件を与えるとする．

read イベントの開始を r，read イベントの終了を r’，write イベントの開始を w，write イベントの終了を w’ とすると，この実行条件の形式的記述は次のようになる．

共有バッファの形式的記述

$$((rr')^{\otimes}|(ww'))^{*}$$

4 Web システムの記述

ここでは，複数のユーザが非同期並行的にアクセスする Web システムの記述例を与える．例として，ショッピングサイトを取り上げる．

ユーザアクセスのフローを図 1 に示す．このフローは，3 つのイベントと，分岐及び繰り返しから成っている．まず，ユーザは，商品を探す（イベント a）．欲しい商品がなかった場合は，ショッピングを終了する．欲しい商品があった場合は，それをカートに入れる（イベント b）．さらに，欲しい商品が無くなるまで，イベント a，b を繰り返す．最後に，決済（イベント c）を行ってショッピングを終了する．

このフローを正規表現やシャフル表現を用いて記述する．複数のユーザによる同時アクセスを認めない場合は，正規表現によって次のように記述できる．

同時アクセスを認めない場合

$$a|(a(ba)^{*}c)$$

同時アクセスを認める場合は，シャフル表現によって次のように記述できる．

同時アクセスを認める場合

$$(a|(a(ba)^{*}c))^{\otimes}$$

このように，同時アクセスを認めない場合は正規表現の記述能力で十分であるが，非同期並行的な同時アクセスを認める場合は，シャフル表現が必要になる．

5 おわりに

本稿では，シャフル表現を用いた非同期イベントの記述法を提案し，複数のユーザが非同期並行的にアクセスする Web システムの記述例を与えた．今後は，シャフルオートマトン [7] をテスト自動化ツールに組み込むことで，非同期並行プログラムの動作テスト容易化を図りたい．

参考文献

[1] Shaw, A. C. : Software description with flow expression, IEEE Trans. Software Eng., Vol.SE-4, No.3, pp.242-254, 1978.

[2] 西谷泰昭，伊藤貴康:同期機構を持つ並行プロセスの記述能力について―フロー表現の拡張とその記述能力―，電子情報通信学会論文誌，Vol.J64-D, No.9, pp.831-838, 1981.

[3] 西谷泰昭，伊藤貴康:同期制御された正規表現の記述能力，電子情報通信学会論文誌，Vol.J64-D, No.12, pp.1089-1096, 1981.

[4] Garg, V. K. and Raggunath, M. T. : Concurrent Regular Expressions and their Relationship to Petri Nets, Theoretical Computer Science, 96:285-334, 1992.

[5] Jedrzejowicz, J. and Szepietowski, A. : Shuffle languages are in P, Theoretical Computer Science, 250:31-53, 2001.

[6] 阿部真也：シャフル表現による非同期イベント系列の記述，情報処理学会第 73 回全国大会，pp.237-238(1), 2011.

[7] Jedrzejowicz, J. : Structural Properties of Shuffle Automata, Grammars, Vol.2, No.1, pp.35-51(17), 1999.