



Divide-and-Conquer Determinization of Büchi Automata Based on SCC Decomposition



Yong Li, Andrea Turrini, Weizhi Feng,
Moshe Y. Vardi, and Lijun Zhang



34th International Conference on Computer Aided Verification (CAV'22)

主要联系人: 冯维直 邮箱: fengwz@ios.ac.cn

Background

非确定性Büchi自动机(NBAs)是接收无穷字串的有限自动机。

- NBAs是模型检测领域用来表示反应式和非终止式系统以及规约的重要模型。
- 某些场景下非确定性自动机不再适用,如概率验证和逻辑规约合成等问题,此时需要确定性自动机。
- 将NBAs转为等价的(即接收相同的 ω 正则语言)的确定性自动机往往是求解这些问题的必要操作。

Challenges

NBAs的确定化是自动机理论领域的一个基本问题,数十年来被广泛研究。

- 接收有穷字串的非确定性自动机(NFAs)确定化可使用子集构造法,但子集构造不适用于NBAs, NBAs确定化的构造更加复杂且具有挑战性。
- 随着输入自动机状态数的增加,确定化之后的自动机状态数往往会遇到状态爆炸的问题。

Approach

我们将分治(divide-and-conquer)思想应用于NBAs确定化中,提出一种基于强连通分量(SCC)分解的确定化算法,可将NBAs转为等价的DELAs(确定性Emerson-Lei自动机)。

- 现有算法关注于将NBA转为等价的确定性Rabin自动机(DRA)或者确定性Parity自动机(DPA);且一般直接对整个NBA结构进行确定化操作,当自动机状态数增大时,可扩展性往往较差。
- 我们将NBA转为DELAs,这种自动机具有更宽松的接收条件,从而可能得到状态数更小的确定化结果。
- 应用分治思想,不直接处理整个NBA,而是将输入NBA中的SCC结构分为三种类型, IWC(inherently weak), DAC(deterministic accepting)和NAC(nondeterministic accepting)然后利用run DAGs分别计算状态读入字串的后继,通过run DAGs每一层的状态集合构造等价DELAs的状态, DELAs可以较容易地转为DRA或DPA。

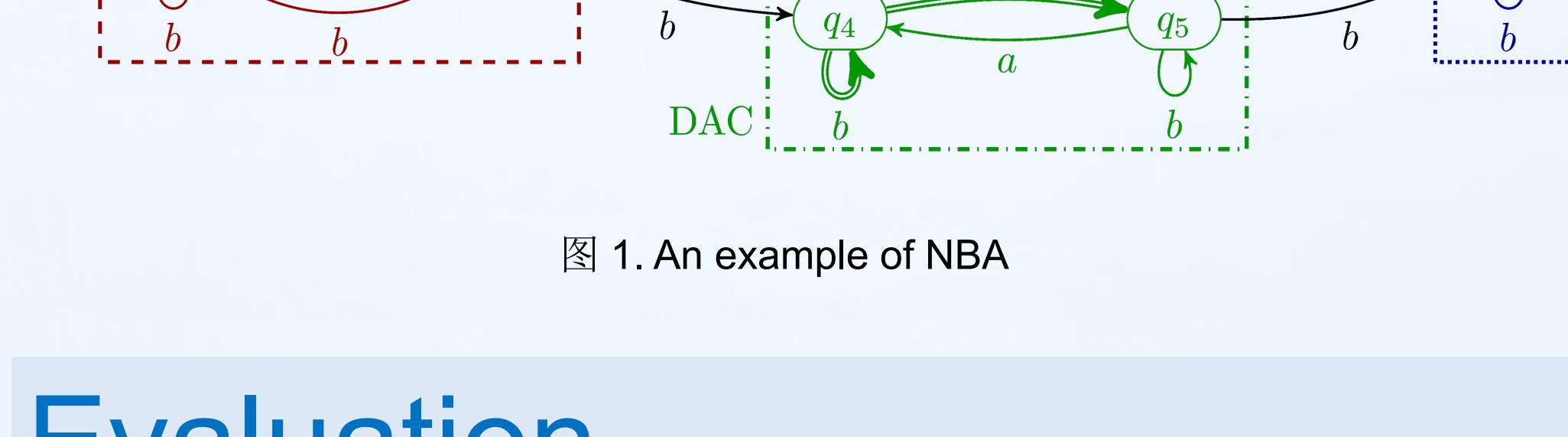


图 1. An example of NBA

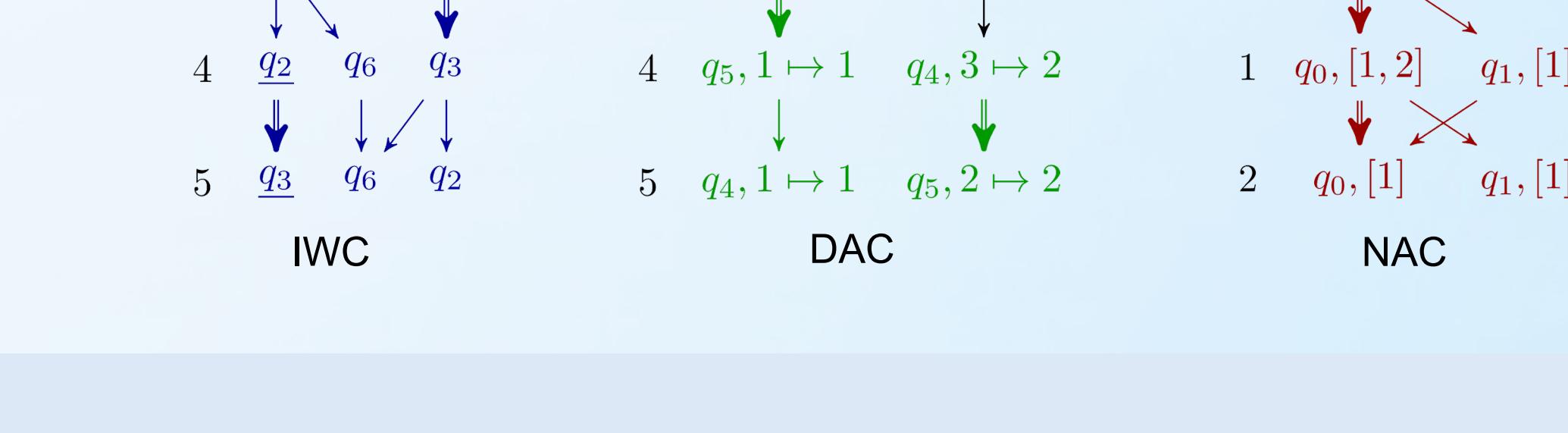


图 1 中三种SCC所对应的run DAG:

Evaluation

我们将算法实现在工具COLA中,并和state-of-the-art的工具SPOT以及OWL进行比较,实验证明COLA在时间和生成自动机大小上具有更好的效果。

实验设置

- 让所有工具都生成DPAs,限制时间10分钟,比较15913个自动机的运行结果。

部分实验结果

- 图2展示了不同工具求解所有自动机例子的运行时间, COLA可求解15903个例子, SPOT可求解15862个例子, OWL可求解15749个例子;
- 图3展示了生成DPAs的状态数;
- COLA相比其他工具可以在更短时间内求解更多例子,且生成相对更小的DPAs。
- COLA link: <https://github.com/liyong31/COLA>

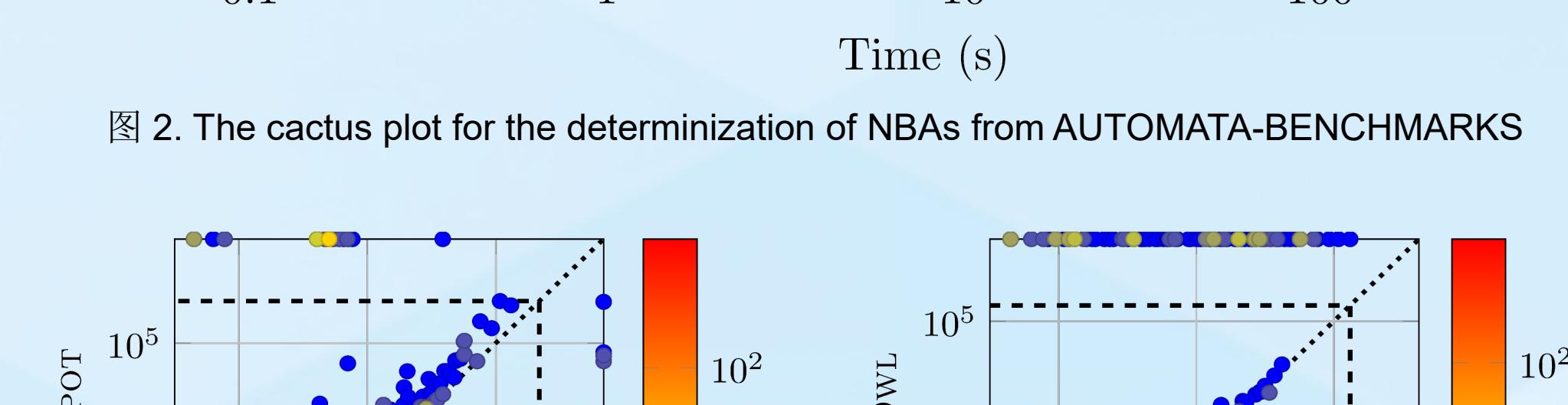


图 2. The cactus plot for the determinization of NBAs from AUTOMATA-BENCHMARKS

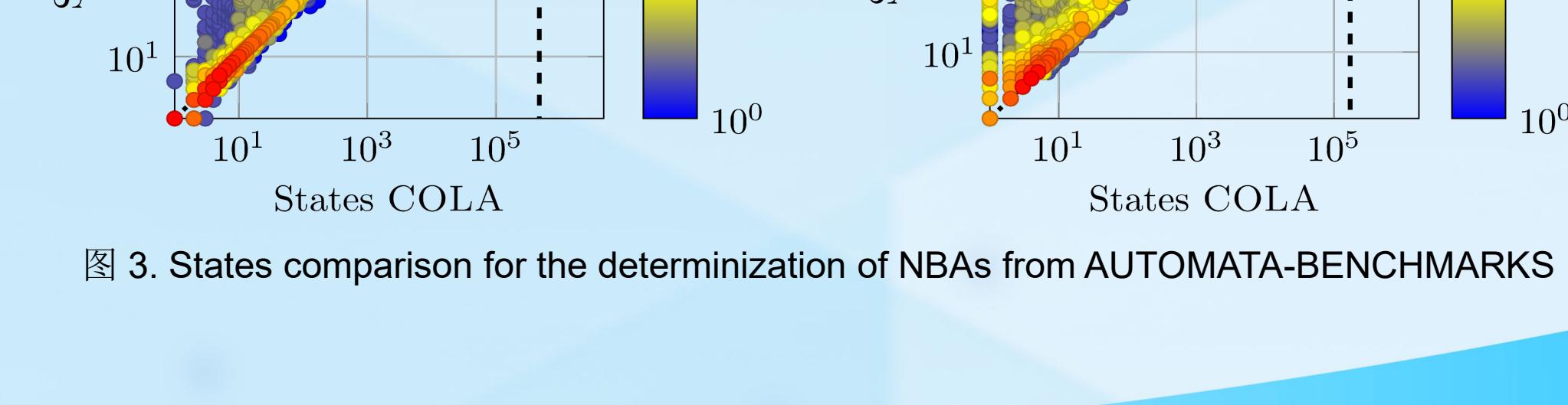


图 3. States comparison for the determinization of NBAs from AUTOMATA-BENCHMARKS