----------------------- REVIEW 1 ---------------------

SCORE: 2 (accept)

\*\* Comments

[A1] It took me a while to understand the definition of m-TS ("w contains no consecutive subsequence w′ ∈ E^{m+1}"); please rephrase, and note that E^{m+1} does not seem to be defined.

[A2] I understand that space is limited, but having a picture of an example of a pushdown transducer and a pushdown automaton (e.g. Examples 4, 6, 18…) could ease the reader's understanding, notably considering the audience of ICTAC is rather wide.

[A3] It seems the exact complexity of the two main problems is not shown (only the upper bound is given), unless I missed an implicit reasoning.

\*\* typos

Please add page numbers, and line numbers (package lineno), to ease reviewing.

p.1

- has its origin => have their origin

- has been one => have been one

- Player~II

p.6

- there exist an infinite number => there exists an infinite number

p.19

- an 2m-DPDA => a 2m-DPDA

p.10

- by a NRPDA => by an NRPDA

- by a NPDA => by an NPDA

p.16

- showm => showed/shown

- alphabels => alphabets

----------------------- REVIEW 3 ---------------------

SCORE: 1 (weak accept)

Evaluation

The paper is technically very dense and to my taste somewhat overformalistic.

To a large extent it consists of definitions, which are painstakingly built one over another. I realize that this is difficult to avoid, given the selected topic, which requires a lot of things: games, automata, data languages, push-down transformers, etc. Still, I think that the authors could have done better job in making the paper more readable. I am also not so sure about the 'visible' version of rpda -- it seems ad-hoc, since it basically requires the automaton to reveal its internal structure, but admittedly similar constrains occur in the literature, e.g. [1].

Some (minor) comments:

[B1] p.4., middle of page: "contains no consecutive" -> "contains no"

[B2] p.5., just before example 4: "we assume that no run of PDT reaches an ID whose stack is empty" -- how can this be assumed? What is the mechanism to ensure it?

[B3] p.5, in definition 5 you do not say that Sigma must contain tau. Or maybe it does not, because Sigma is strictly partitioned between input and output symbols.

[B4] p.6, end of page: "by definition, a play rho is winning for Player I iff (rho,w) is a run of A for some w" -- I suppose this must be an accepting run, not just any run?

[B5] Overall, this stack of definition was difficult to get through and is not particularly logically organized, as to me. The acceptance condition for automata is game-theoretic by nature, but the authors just throw a bulk of formal definitions, from which one can read out players and parity conditions and so on, only if one knows about them. And then, at some point later, they still introduce games and just drop briefly: btw. a word is recognized if it comes from a play with Player I wins. This is a strange approach for presenting such things.

[B6] p7, Lemma 9: the defining clause does not depend on rho

[B7] p7, corollary 11: what is Gamma'?

[B8] p18, conclusions: "showm" -> "shown".