Computation with Absolutely No Space Overhead

Lane Hemaspaandra¹ Proshanto Mukherji¹ Till Tantau²

¹Department of Computer Science University of Rochester

²Fakultät für Elektrotechnik und Informatik Technical University of Berlin

Developments in Language Theory Conference, 2003



315

The Model of Overhead-Free Computation The Standard Model of Linear Space Our Model of Absolutely No Space Overhead



The Model of Overhead-Free Computation The Standard Model of Linear Space Our Model of Absolutely No Space Overhead The Power of Overhead-Free Computation Palindromes Linear Languages Context-Free Languages with a Forbidden Subword Languages Complete for Polynomial Space



▲ 글 ▶ _ 글 남

The Model of Overhead-Free Computation The Standard Model of Linear Space Our Model of Absolutely No Space Overhead 2 The Power of Overhead-Free Computation Palindromes Linear Languages Context-Free Languages with a Forbidden Subword Languages Complete for Polynomial Space Limitations of Overhead-Free Computation 3 Linear Space is Strictly More Powerful



글 🖌 글 🗄

Standard Model Our Model

Outline

- The Model of Overhead-Free Computation
 The Standard Model of Linear Space
 Our Model of Absolutely No Space Overhead
- 2 The Power of Overhead-Free Computation
 - Palindromes
 - Linear Languages
 - Context-Free Languages with a Forbidden Subword
 - Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



▲ 글 ▶ ▲ 글 ▶ _ 글 날

Standard Model Our Model

Outline

- The Model of Overhead-Free Computation
 The Standard Model of Linear Space
 - Our Model of Absolutely No Space Overhead
- 2 The Power of Overhead-Free Computation
 - Palindromes
 - Linear Languages
 - Context-Free Languages with a Forbidden Subword
 - Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



Palindromes Linear Languages Forbidden Subword Complete Languages

Outline

- The Model of Overhead-Free Computation
 The Standard Model of Linear Space
 Our Model of Absolutely No Space Overhead
- The Power of Overhead-Free Computation
 Palindromes
 - Linear Languages
 - Context-Free Languages with a Forbidden Subword
 - Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



Palindromes Linear Languages Forbidden Subword Complete Languages

Outline

- The Model of Overhead-Free Computation
 The Standard Model of Linear Space
 Our Model of Absolutely No Space Overhead
- The Power of Overhead-Free Computation
 Palindromes
 - Linear Languages
 - Context-Free Languages with a Forbidden Subword
 - Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



Palindromes Linear Languages Forbidden Subword Complete Languages

Outline

- The Model of Overhead-Free Computation
 The Standard Model of Linear Space
 Our Model of Absolutely No Space Overhead
- 2 The Power of Overhead-Free Computation
 - Palindromes
 - Linear Languages
 - Context-Free Languages with a Forbidden Subword
 - Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



Palindromes Linear Languages Forbidden Subword Complete Languages

Outline

The Model of Overhead-Free Computation
The Standard Model of Linear Space
Our Model of Absolutely No Space Overhead

The Power of Overhead-Free Computation

- Palindromes
- Linear Languages
- Context-Free Languages with a Forbidden Subword
- Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



Strict Inclusion

Outline

- The Model of Overhead-Free Computation
 The Standard Model of Linear Space
 Our Model of Absolutely No Space Overhead
- 2 The Power of Overhead-Free Computation
 - Palindromes
 - Linear Languages
 - Context-Free Languages with a Forbidden Subword
 - Languages Complete for Polynomial Space
- Limitations of Overhead-Free Computation
 Linear Space is Strictly More Powerful



→ ★ 문 → 문 님

Further Reading

For Further Reading

- A. Salomaa. Formal Languages. Academic Press, 1973.
 - Smoothsort, an alternative for sorting in situ. Science of Computer Programming, 1(3):223–233, 1982.
- E. Feldman and J. Owings, Jr. A class of universal linear bounded automata. Information Sciences, 6:187–190, 1973.
- P. Jančar, F. Mráz, M. Plátek, and J. Vogel. Restarting automata. FCT Conference 1995, LNCS 985, pages 282–29



Further Reading

For Further Reading



🛸 A. Salomaa. Formal Languages. Academic Press, 1973.

E. Dijkstra. Smoothsort, an alternative for sorting in situ. Science of Computer Programming, 1(3):223-233, 1982.



Further Reading

For Further Reading



🛸 A. Salomaa. Formal Languages. Academic Press, 1973.

E. Dijkstra. Smoothsort, an alternative for sorting in situ. Science of Computer Programming, 1(3):223–233, 1982.

E. Feldman and J. Owings, Jr. A class of universal linear bounded automata. Information Sciences, 6:187–190, 1973.



Further Reading

For Further Reading



🛸 A. Salomaa. Formal Languages. Academic Press, 1973.

E. Dijkstra. Smoothsort, an alternative for sorting in situ. Science of Computer Programming, 1(3):223–233, 1982.

- E. Feldman and J. Owings, Jr. A class of universal linear bounded automata. Information Sciences. 6:187–190. 1973.
- P. Jančar, F. Mráz, M. Plátek, and J. Vogel. Restarting automata.

FCT Conference 1995, LNCS 985, pages 282–292, 1995.



문 / 문 / 문 물

Appendix

Appendix Outline





(日) (圖) (필) (필) (필)