CIAA 2013 PROGRAM

JULY 15 – 19 HALIFAX

18th International Conference on Implementation and Application of Automata

CONTENTS

1. SCHEDULE

MONDAY July 15

- 1800--1930: Wine and cheese reception and registration (in SB 4th floor)

TUESDAY July 16 (all talks in room SB 255)

- 0800: REGISTRATION
- 0900: Welcome message
- 0930: Invited: "Automata for Codes", Helmut Jürgensen
- 1030: COFFEE break
- 1100: "Partial Word Automata", Chair: Stavros Konstantinidis
  Balkanski, Blanchet-Sadri, Kilgore and Wyatt
  "Approximate Matching between a Context-Free Grammar and a Finite-State Automaton",
  Yo-Sub Han, Sang-Ki Ko and Kai Salomaa
- 1200: LUNCH (and *registration*: 1300--1330)
- 1330: "Compressed Automata for Dictionary Matching", Chair: Kai Salomaa
  Tomohiro I, Takaaki Nishimoto, Shunsuke Inenaga, Hideo Bannai, Masayuki Takeda
  "Two-Pass Greedy Regular Expression Parsing",
  Niels Bjørn Bugge Grathwohl, Fritz Henglein, Lasse Nielsen, Ulrik Terp Rasmussen
  "Early Nested Word Automata for XPath Query Answering on XML Streams",
  Tom Sebastian, Denis Debarbieux, Olivier Gauwin, Joachim Niehren, Mohamed Zergaoui
- 1500: COFFEE break
- 1530: "Comparing Two-Dimensional One-Marker Automata to Sgraffito Automata",
  Daniel Prusa, Frantisek Mraz and Friedrich Otto Chair: Geraud Senizergues
  "Lambda-Confluence is Undecidable for Clearing Restarting Automata",
  Frantisek Mraz and Friedrich Otto

WEDNESDAY July 17

- 0800: REGISTRATION
- 0900: Invited: "Applications of Symbolic Finite Automata", Margus Veanes
- 1000: COFFEE break
- 1030: "Implementation Concepts in Vaucanson 2", Chair: Cezar Campeanu
  Demaille, Duret-Lutz, Lombardy and Sakarovitch
  "LALBLC a program testing the equivalence of dpda's",
  Patrick Henry and Géraud Sénizergues
  "Computing Weights",
  Houda Abbad and Eric Laugerotte
- 1200: LUNCH (and *registration*: 1300--1330)
- 1330: "Some Decision Problems Concerning NPDAs, Palindromes, and Dyck Languages",
  Oscar Ibarra and Bala Ravikumar Chair: Markus Holzer
  "Input-Driven Queue Automata: Finite Turns, Decidability, and Closure Properties",
  Martin Kutrib, Andreas Malcher, Carlo Mereghetti, Beatrice Palano, Matthias Wendlandt
  "Invertible Transducers, Iteration and Coordinates",
  Klaus Sutner
- 1500: COFFEE break
- 1530: "Deterministic counter machines and parallel matching computations",
  Stefano Crespi Reghizzi and Pierluigi San Pietro Chair: Yo-Sub Han

"Trimming Visibly Pushdown Automata",
Mathieu Caralp, Pierre-Alain Reynier and Jean-Marc Talbot
"On Palindromic Sequence Automata and Applications",
Md. Mahbubul Hasan, A. S. M. Sohidull Islam, M. Sohel Rahman, Ayon Sen

THURSDAY July 18

- 0800: REGISTRATION
- 0900: Invited: "Cover Languages and Implementation", Cezar Câmpeanu
- 1000: COFFEE break
- 1030: "Universal Witnesses for State Complexity of Basic Operations Combined with Reversal",
  Janusz Brzozowski and David Liu Chair: Andreas Maletti
  "On the Boundary of Regular Languages",
  Jozef Jirasek and Galina Jiraskova
  "More Robust Than Thought---Brzozowski's Minimization Algorithm"
  Markus Holzer and Sebastian Jakobi
- 1200: LUNCH (and *registration*: 1300—1330)
- 1330: Group Picture
- 1340: **Excursion and Banquet**

FRIDAY July 19

- 0800: REGISTRATION
- 0848: "Enhancing Approximations for Regular Reachability Analysis", Chair: Stavros Konstantinidis
  Aloïs Dreyfus, Pierre-Cyrille Heam and Olga Kouchnarenko
  "Generating small automata and the Cerny conjecture",
  Andrzej Kisielewicz and Marek Szykuła
  "Incomplete Transition Complexity of Basic Operations on Finite Languages",
  Eva Maia, Nelma Moreira and Rogério Reis
- 1000: COFFEE break
- 1030: "Hyper-Optimization for Deterministic Tree Automata", Chair: Rogerio Reis
  Andreas Maletti
  "A Completion Algorithm for Lattice Tree Automata",
  Thomas Genet, Le Gall, Axel Legay and Valérie Murat
  "A Uniformization Theorem for Nested Word to Word Transductions",
  Dmitry Chistikov and Rupak Majumdar
- 1200: LUNCH
- 1330: Business Meeting
- 1430: "On Length of Homing Sequences for Nondeterministic Finite State Machines",
  Kushik, Yevtushenko Chair: Nelma Moreira
  "Using Regular Grammars for Event-Based Testing",
  Fevzi Belli and Mutlu Beyazıt
- 1530: COFFEE break
- 1600: "Towards Nominal Context-Free Model-Checking" Chair: Stavros Konstantinidis
  Pierpaolo Degano, Gianluigi Ferrari and Gianluca Mezzetti
- 1630-1730: ***Sofware Demo Sessions***
  "Regular languages operations and simulations with Fado"
  Nelma Moreira and Rogério Reis
  "I-LaSer: implementing the maximality problem"
  Stavros Konstantinidis and Casey Meijer

2. PRACTICAL INFORMATION

SMU = Saint Mary's University

Coference room (July 16—19):
Room SB 255, in the Sobey Building—see map on next page

Wireless access: Those of you with Eduroam access can use it here at SMU, if your institution has already set you up. In any case, there will be wireless access via SMU accounts that will be given to you when you register.

Lunches: The conference fee includes four lunches at the Dockside Cafeteria—located on the main floor of Loyola Residence. Every noon during July 16-19 please go to that cafeteria (simply follow the crowd) and have your registration tag with you in order *to receive a lunch tickect* from Rose or Jenna who will be at the entrance of the cafeteria.

Excursion and Conference dinner: Thursday July 18 at 1340 outside of the Sobey Building (closest exit from the conference room).

City Buses: Please see the map on the next page for numbers of buses stopping at SMU. See also the page after the map for bus stops from hotels to SMU.

Dinnes and the Halifax Peninsula: The Halifax Peninsula is the location of old Halifax and the downtown, where one can find numerous eating establishments. In particular, you are encouraged to explore the area defined by the following **polygon**:
Lower Water Street St — South St — South Park St — Sackville
St — Brunswick St — Duke St — Lower Water Street St.
Some restaurants: a Mano (Ristorante), Bistro Le Coq (French), Bluenose II, Brussels Restaurant & Brasserie, ChaBaa (Thai), Cheelin Restaurant (Chinese), Curry Village (Indian), Ela (Greek), Estia, Gingerbread Haus Bakery (German and other snacks and sweets), Hamachi House (Japanese), Salty's, Seoul Restaurant (Korean), Talay Thai, The Bicycle Thief, The Economy Shoe Shop, The Five Fishermen, Your Father's Moustache.

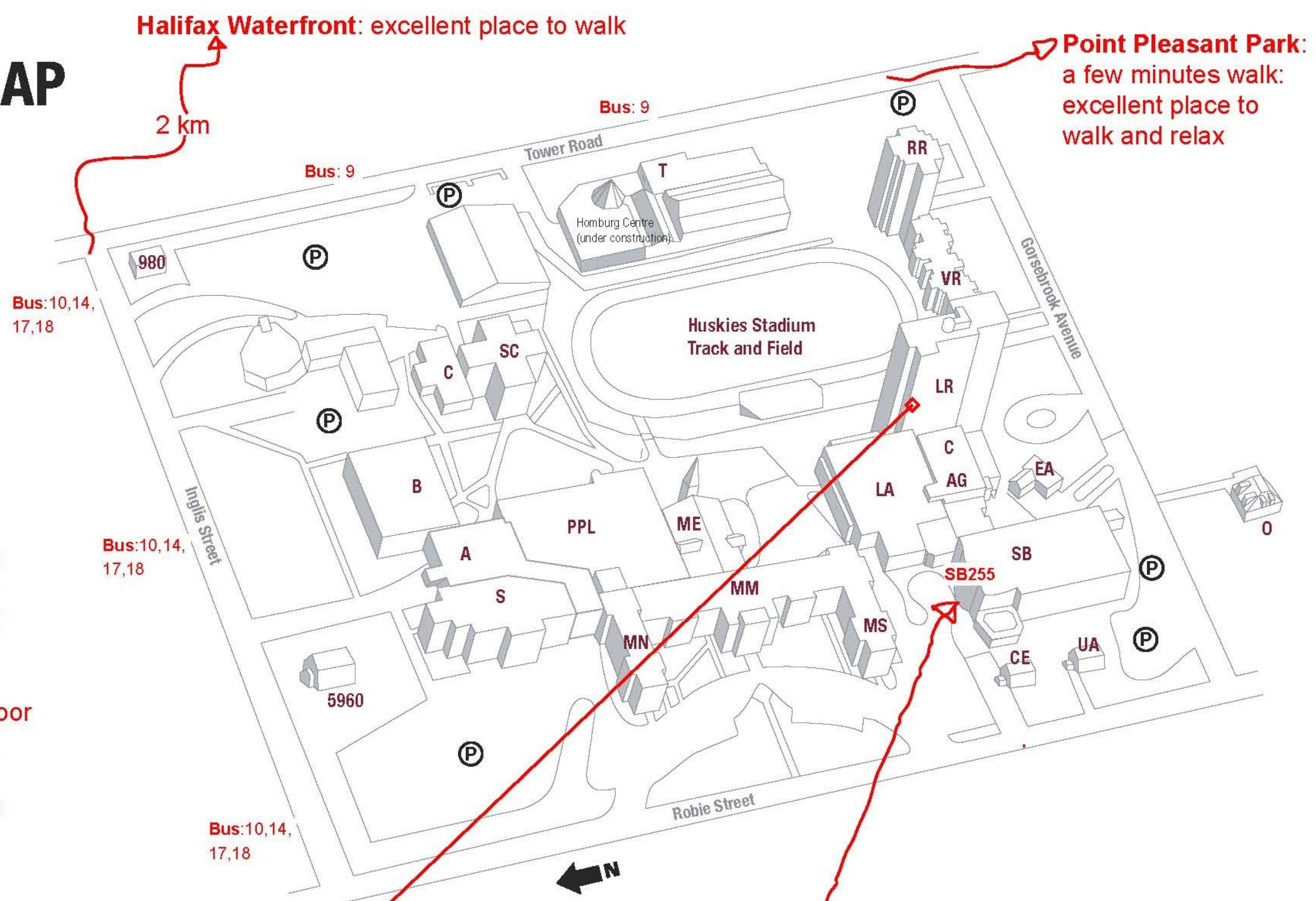
Halifax Attractions: The above **polygon** is an excellent place to explore: Pier 21, Alexander Keith's brewery, Maritime Museum of the Atlantic, Public Gardens.

Contact: Please send me an email message at any time: stavros@cs.smu.ca Most times during the conference I should have access to email via my cell phone.

CAMPUS MAP

SAINT MARY'S UNIVERSITY SINCE 1802

One University. One World. Yours.


Halifax Waterfront: excellent place to walk
2 km
Point Pleasant Park: a few minutes walk: excellent place to walk and relax
Bus: 9
Bus: 9
Tower Road
RR
T
Homburg Centre (under construction)
980
VR
Gorsebrook Avenue
Bus:10,14, 17,18
Huskies Stadium Track and Field
C
SC
LR
C
AG
EA
O
B
LA
PPL
ME
A
SB
SB255
MM
S
MS
MN
CE
UA
5960
Bus:10,14, 17,18
Inglis Street
Bus:10,14, 17,18
Robie Street
N

CIAA talks: room SB255 (enter Sobey Building as shown, and go left a few steps to find the room)

Wine&Cheese: SB 4th floor (enter Sobey Building as shown, go straight to elevator, take elevator to 4th floor, go left around the corner)

A	Atrium	LA	Loyola Academic Complex	Ⓟ	Parking	T	The Tower
AA	Alumni Arena	LR	Loyola Residence	PPL	Patrick Power Library	UA	Development/Alumni
AG	Art Gallery	ME	McNally East Wing	RR	Rice Residence	VR	Vanier Residence
B	Burke Building	MM	McNally Main	S	Science Building	980	TESL Centre
C	Cafeteria	MN	McNally North Wing	SB	Sobey Building CIAA	5960	Gorsebrook Research Institute for Atlantic Canada Studies/ CN Centre for Occupational Health & Safety
CE	Continuing Education	MS	McNally South Wing	SC	O'Donnell Hennessey Student Centre		
EA	External Affairs	O	The Oaks/International Activities				

All main buildings are wheelchair accessible and most are connected by tunnels or walkways.

Bus Routes to Saint Mary's University

Radisson Suite Hotel Halifax: Walk up the hill to Barrington Street. Cross the street and catch the **10** (Dalhousie) or the **14** (Mumford or Leiblin Park) to Saint Mary's.

The Lord Nelson Hotel: Walk left from the main entrance of the hotel to the corner of Spring Garden Road and South Park Street. Turn left at this intersection and you will see a bus stop right outside of Public Gardens Dentistry. Catch the **10** (Dalhousie) or the **14** (Mumford or Leiblin Park) to Saint Mary's. OR exit from the Lord Nelson onto Spring Garden Road and the bus stop will be directly in front of you.

Westin Nova Scotia: Walk out to Barrington Street. Cross the street and catch the **9** (Barrington to Point Pleasant) to Saint Mary's.

Commons Inn: Walk to Robie Street. Cross the street and go left. Continue walking on Robie Street past Tony's. You will see a bus stop ahead. Catch the **17** (Saint Mary's) or the **18** (Universities or Saint Mary's Only) to the university.

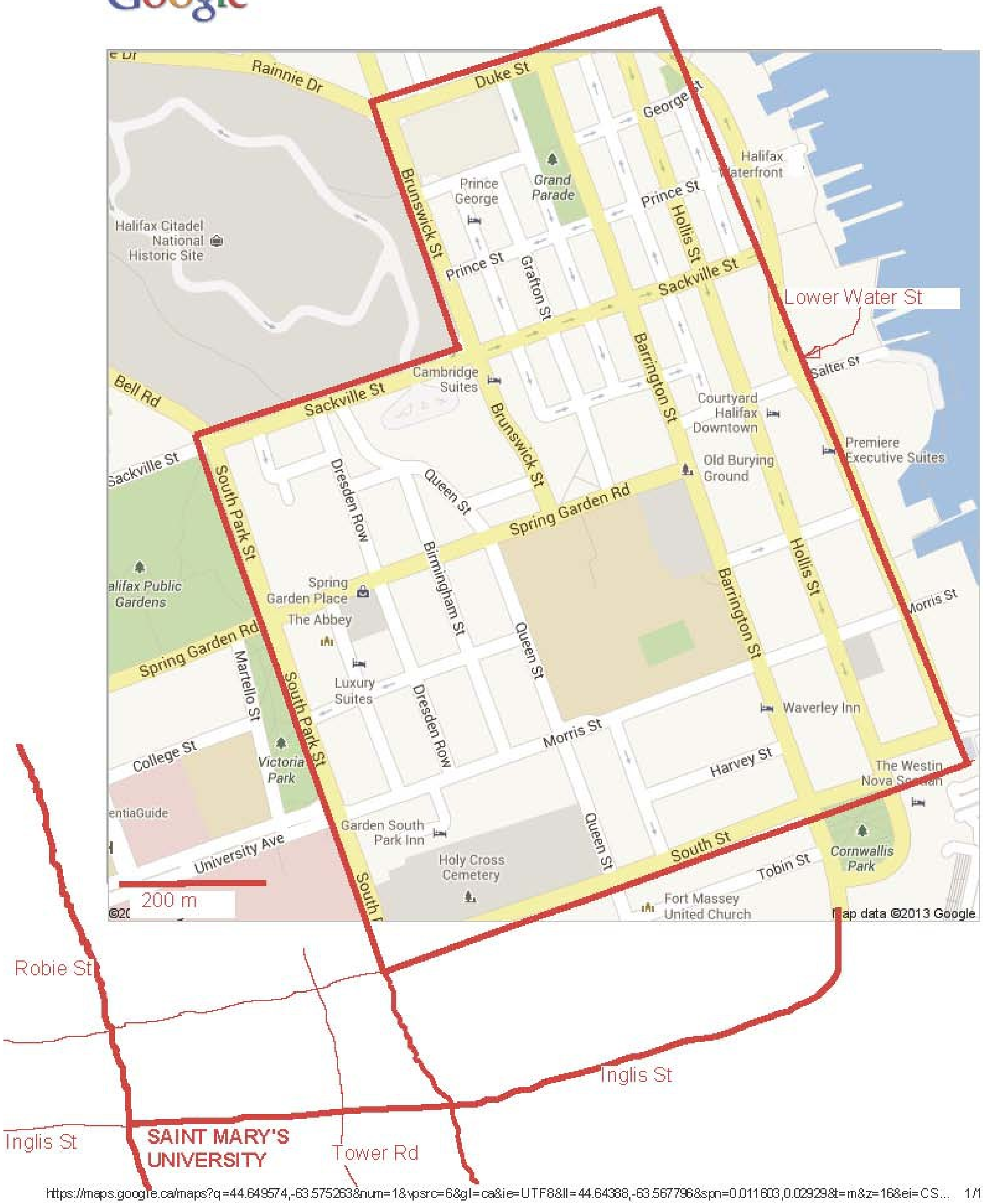
Courtyard by Marriott: Walk up the hill to Barrington Street. Cross the street and catch the **10** (Dalhousie) or the **14** (Mumford or Leiblin Park) to Saint Mary's.

Delta Halifax: Exit the hotel onto Barrington Street. Catch the **10** (Dalhousie) or the **14** (Mumford or Leiblin Park) from the Scotia Square terminal to Saint Mary's.

Four Points by Sheraton: Walk up the hill to Barrington Street. Cross the street and catch the **10** (Dalhousie) or the **14** (Mumford or Leiblin Park) to Saint Mary's.

Prince George: Walk down Prince Street to Barrington Street. Catch the **10** (Dalhousie) or the **14** (Mumford or Leiblin Park) to Saint Mary's.

Google


Rainnie Dr
Duke St
George St
Halifax Waterfront
Brunswick St
Prince George
Grand Parade
Prince St
Hollis St
Halifax Citadel National Historic Site
Prince St
Grafton St
Sackville St
Lower Water St
Barrington St
Salter St
Bell Rd
Cambridge Suites
Sackville St
Courtyard Halifax Downtown
Brunswick St
Sackville St
Old Burying Ground
Premiere Executive Suites
South Park St
Dresden Row
Queen St
Spring Garden Rd
Birmingham St
Halifax Public Gardens
Spring Garden Place
The Abbey
Barrington St
Hollis St
Morris St
Spring Garden Rd
Queen St
Martello St
South Park St
Luxury Suites
Dresden Row
Waverley Inn
College St
Victoria Park
Morris St
Harvey St
The Westin Nova Scotian
University Ave
Garden South Park Inn
Holy Cross Cemetery
Queen St
South St
Tobin St
Cornwallis Park
200 m
South Park St
Fort Massey United Church
Map data ©2013 Google
Robie St
Inglis St
Inglis St
SAINT MARY'S UNIVERSITY
Tower Rd

3. ABSTRACTS OF SOFTWARE DEMOS

.

Regular languages operations and simulations with Fado

Nelma Moreira and Rogério Reis

I-LaSer: implementing the maximality problem

Stavros Konstantinidis and Casey Meijer

Regular languages operations and simulations with FAdo

http://fado.dcc.fc.up.pt/

Departamento de Ciência de Computadores
Faculdade de Ciências da Universidade do Porto
Porto, Portugal

Computational and descriptional complexity of regular languages continues to be a major topic of research. Computational symbolic manipulation environments play an important role in assisting this research in several aspects. For instance:

- Little is known about the pratical performance of most classical finite automata and regular expressions algorithms and experimental results can give valuable information.
- Random generators are essential for testing combinatorial properties of regular langagues representations and average-case studies.
- Operational descriptional complexity rely heavly on the experimental search of family languages which provides lower bounds of complexity.
- Prototyping new algorithms benifices from available libraries with common data structures and standard algorithms.

The FAdo system aims to provide an open source extensible software library for the symbolic manipulation of regular languages and other formal languages. FAdo is implemented in Python and currently includes most standard operations for the manipulation of regular languages through finite automata (Python classes DFA, NFA, or GFA) and regular expressions (reex). Elementary regular language operations, such as, union, intersection, complementation, concatenation, iteration, and reverse are implemented for each class. Many combined operations for DFA have specialized algorithms. Witness families for several operational complexity lower bounds are defined and a canonical representation is supported. Several conversions between representations are implemented: NFA to DFA, NFA to reex, GFA to reex, reex to NFA, etc.. For DFA, several minimization (and hyperminimazation) algorithms are available: Moore, Hopcroft, Brzozowski, and some incremental algorithms. NFA minimization via universal automata is implemented but with the inherent inefficiency. Some support is provided for computing syntactic semigroups. There are several algorithms for language equivalence based on bisimulation-like algorithms. Finite languages can also be represented, by tries and AFA (acyclic finite automata). Exact and random generators for some classes of automata and regular expressions are also available. More detailed information on the models and the respective operations implemented can be found in the documentation file `http://fado.dcc.fc.up.pt/resources/Files/FAdo.pdf`.

In this presentation we will focus in a new implementation of (reduced) regular expressions modulo some algebraic properties: associativity (A), commutativity (C), and idempotence (I) of the disjunction; and associativity of concatenation (AC). This also includes extended regular expressions with intersection and complement. We will present new algorithms for equivalence of extended regular expressions and conversions to finite automata.

I-LaSer: implementing the maximality problem[1]

Stavros Konstantinidis and Casey Meijer

Department of Mathematics and Computing Science, Saint Mary's University,
Halifax, Nova Scotia, B3H 3C3 Canada
s.konstantinidis@smu.ca, dylanyoungmeijer@gmail.com

I-Laser is a software consisting of a web interface [6] and an implementation of several algorithms written in C++ and Python. These algorithms make extensive use of the FAdo library for finite state machines [1, 3]. I-Laser is currently capable of answering the following decision problem.

– ***The satisfaction question:*** Given the description of a regular language and the description of an independence property, decide whether the language satisfies the property.

In case the answer to the above question is negative, I-LaSer returns a pair of words violating the property. Examples of independence properties include prefix, suffix, bifix, infix, outfix, and solid codes, hypercodes, [5], and error-detecting languages for various cases of error situations [2].

In this talk we show our progress in an implementation for answering the following decision problem.

– ***The maximality question:*** Given the description of a regular language and the description of an independence property, decide whether the language is maximal with respect to that property.

In case the answer to the above question is negative, the implementation returns one or two words, each of which if added to the language (though not necessarily both), then the resulting language also satisfies the property. At the time of writing this extended abstract, the implementation of the maximality problem was not incorporated into I-LaSer, as some interface type programming issues remained to be resolved.

We note that, in general, the maximality problem is PSPACE-hard, and the algorithm used to decide it is of exponential time/space complexity in the worst case. For this reason, if the size of any of the constructed finite state machines involved in the algorithm exceeds 10^6 states, then the algorithm interrupts its execution with an appropriate message to the user.

The web interface allows the user to specify the files containing the descriptions of the regular language and the desired property. Regular languages can be described via NFAs using the Grail [4] or the FAdo format [3]. The issue of describing language properties is a central theme in this line of research. More details will be given in the talk. The reader is also referred to [2] for details.

[1] Research supported by NSERC.

References

1. A. Almeida, M. Almeida, J. Alves, N. Moreira and R. Reis. FAdo and GUItar: Tools for automata manipulation and visualization. In S. Maneth, *CIAA 2009. LNCS 5642*, 65–74. Springer-Verlag Berlin, 2009.
2. K. Dudzinski and S. Konstantinidis. Formal descriptions of code properties: decidability, complexity, implementation. *Intern. J. Foundations of Computer Science* **23**:1 (2012), 67–85.
3. FAdo. `http://fado.dcc.fc.up.pt/` Accessed on July 07, 2013
4. Grail+. `http://www.grailplus.org/` Accessed on July 07, 2013
5. H. Jürgensen and S. Konstantinidis. Codes. In [7], pp 511–607.
6. LaSer. `http://laser.cs.smu.ca/independence/` Accessed on July 07, 2013
7. G. Rozenberg and A. Salomaa (eds). *Handbook of Formal Languages, Vol. 1*. Springer-Verlag, Berlin, 1997.

4. SPONSORS

SAINT MARY'S UNIVERSITY SINCE 1802
One University. One World. Yours.

Microsoft Research

European Association for
Theoretical Computer Science