Errata for PhD Thesis: "Rigorous Simulation: Its Theory and Applications"

December 5, 2016

1. **Page 12**
   "0 to n" should be "0 to $n-1$"

2. **Page 23 and 106**
   "all behavior" should be "all behaviors"

3. **Page 34**
   The column labels "Trad" and "Rig" in Table 2.2 refer to the first and second column below the "Num" label, respectively.

4. **Page 44**
   The `create`s provide initial values for the state variables `x` and `x'`, as well as for `x''`. The latter can be considered as redundant, given the equation `x'' = -x`. However, Acumen currently does not recognize this, and requires all variables of a model to be declared and given values at initialization.

5. **Page 46**
   "When x is greater than 1" should be "When x is greater than or equal to 1".

6. **Page 49**
   Throughout the dissertation, a translation refers to a mapping between the terms of two languages.

7. **Page 52**
   "a sequence $\langle g_i \rangle^{i \in n}$" should be "a conditional if / then / else".

8. **Page 54**
   All occurrences of $x > 0$ should be $x < 1$.

9. **Page 55**
   MicroAcumen is intended as the smallest possible subset of MiniAcumen that has at least the same expressivity. By defining the semantics for a minimal core language, the definition of the semantics becomes small, and thus easier to analyze.

10. **Page 56**
    $\mathbb{M}^{\mu}$ should be $\mathbb{U}$ in Definition 4.2.5.

11. **Page 65**
    The names "r" and "b" in the rules Real and Bool should be interchanged.

12. **Page 78**
    Definition 5.5.1 should be as follows: A *value enclosure* $d \in \mathbb{D} \subset \mathcal{P}_{\text{closed}}(\mathbb{S})$ is a representation of a non-empty closed subset of $\mathbb{S}$ and comes with a decidable inclusion predicate $d_1 \subseteq d_2$.

13. **Page 75**
    This definition of a hybrid system formalizes the informal notion used in Section 2.1. The definition is similar to the languages described in Section 2.1, but simpler. For example, unlike hybrid automata, hybrid systems (as per Definition 5.4.1) do not explicitly identify locations (parts of the flow map that correspond to different continuous dynamics), and do not explicitly specify transitions between locations. This makes hybrid systems close to the Acumen language, where locations do not need to be explicitly specified in the syntax.

14. **Page 74**
    $\mathbb{E} \times \mathbb{E}^n$ should be $\mathbb{E} \times \mathbb{E}^{\mathsf{N}}$ in Definition 5.3.9.

15. **Page 76**
    $\langle e_i \rangle^{i \in n}$ should be $\langle e_i \rangle^{i \in \mathsf{N}}$ in Definition 5.4.3.

16. **Page 88**
    "rule rule" should be "rule".

17. **Page 90**
    The execution of a model using enclosures can be seen as an abstract interpretation [CC] that makes it possible to answer questions about a model, such as whether a particular state is reachable, even though the exact problem is undecidable [HKPV].

18. **Page 91**
    In the example in subsection 5.8.3, the initial state $x = [0]$ should be $x = [\frac{1}{2} .. \frac{3}{4}]$.

19. **Page 98**
    "Two kinds of interpreters that are" should be "Two kinds of interpreters are".

20. **Page 102**
    "July 2017" should be "July 2012".

21. **Page 112**
    The definition of the contractor given in subsection 6.8.5 is incorrect, and should be as follows. Let $\mathbb{IR}^n$ be the set of interval boxes, that is, the set of n-dimensional real interval vectors. Given an interval box $x \in \mathbb{IR}^n$ and a constraint $K : \mathbb{R}^n \to \mathbb{B}$ the contractor is a function $C : \mathbb{IR}^n \to \mathbb{IR}^n$ that satisfies the following two properties:

    - $C(x) \subseteq x$,
    - $C(x) \cap z = x \cap z$,

    where z is the set $\{v \in \mathbb{R}^n \mid K(v)\}$ associated with K.

22. **Page 142**
    Continuous systems are modeled by differential equations in Acumen. Guaranteed solutions to differential equations are provided by two different integrators. The first is based on the function interval Picard operator (Section 6.8.3) and the second on Taylor series methods (Section 6.8.4).

23. **Page 156**
    Acumen's parser is implemented using the Scala Standard Parser Combinator library [MPO].

24. **Page 182**
    Definition B.0.4 uses *lambda notation* to define an anonymous function. For example, $\lambda X.\ \text{closure}(\bigcup X)$ can equivalently be written as $X \mapsto \text{closure}(\bigcup X)$.

25. **Page 185**
    The current release of Acumen can be found at `https://bitbucket.`

org/effective/acumen-dev/downloads/2016_11_17_Acumen.zip.

26. **Page 188**
    The horizontal line can be disregarded.

27. **Page 188**
    "now Publishers" should be "Now Publishers" in reference [CPPSV].

28. **Page 193**
    Reference [MU] should be as follows: Shota Matsumoto and Kazunori Ueda. Hyrose: A symbolic simulator of the hybrid constraint language HydLa. *Computer Software*, 30(4):18–35, 11 2013. https://www.jstage.jst.go.jp/article/jssst/30/4/30_4_18/_article

29. **Page 197**
    Reference [UMT] should be as follows: Kazunori Ueda, Shota Matsumoto, Akira Takeguchi, Hiroshi Hosobe and Daisuke Ishii, et. al. HydLa : A High-Level Language for Hybrid Systems. In *Proceedings of the Second Workshop on Logics for System Analysis*, 2012, pp.3–17. http://www.ls.cs.cmu.edu/LfSA12/LfSA12.pdf

Bibliography

[CC] P. Cousot and R. Cousot. Static determination of dynamic properties of programs. In *Proceedings of the Second International Symposium on Programming*, pages 106–130. Dunod, Paris, France, 1976.

[HKPV] Thomas A Henzinger, Peter W Kopke, Anuj Puri, and Pravin Varaiya. What's decidable about hybrid automata? In *Proceedings of the twenty-seventh annual ACM symposium on Theory of computing*, pages 373–382. ACM, 1995.

[MPO] Adriaan Moors, Frank Piessens, and Martin Odersky. Parser combinators in Scala. Technical report, Department of Computer Science, KU Leuven, 2008.