

Curso: Semántica de Procesos

Docente: Pedro R. D'Argenio

Resumen (*Adaptado de [8]*)

Un proceso describe el comportamiento de un sistema. Un sistemas pueden ser una máquina, una partícula elemental, un protocolo de comunicación, etc. La teoría de procesos es el estudio de los procesos. Dos actividades principales de la teoría de procesos son el modelado y la verificación. El modelado es la actividad de representar procesos, mayormente por medio de estructuras matemáticas o por medio de expresiones en un lenguaje específico para la descripción de sistemas. La verificación es la actividad de demostrar propiedades sobre los procesos, por ejemplo, de que el comportamiento real del sistema es equivalente al comportamiento esperado. Por supuesto, esto solo es posible si existe un criterio bien definido que determine o no cuando dos procesos se comportan de manera semejante. Tal criterio constituye la *semántica* de una teoría de procesos. O más precisamente: constituye la semántica del concepto de igualdad utilizado por la teoría. Cuáles son los aspectos importantes del comportamiento de un sistema para un usuario en particular, depende del entorno en el cual se ejecuta el sistema y de los intereses del usuario particular (i.e. de las propiedades que este espera que el sistema cumpla o preserve).

En este curso estudiaremos:

- semánticas de procesos y las características que las diferencian;
- formas de caracterizar a estas semánticas (lógicas, escenarios de test, relaciones entre sistemas de transiciones, axiomatizaciones); y
- técnicas formales para la especificación de sistemas de transición y su relación con las semánticas y sistemas axiomáticos.

Contenido

1. **Prólogo:** Teoría de procesos: por qué y para qué. Cuatro dimensiones para la semántica de la concurrencia. Dominios de procesos. Nuestro contexto.
2. **Una introducción gentil.** Sincronización. Acciones y transiciones. El poder preventivo de las acciones internas. Un lenguaje básico para la representación de procesos. Semántica de transiciones y semántica operacional estructurada. Bisimulación fuerte. Congruencia. Bisimulación débil. El problema de la substitutividad. Congruencia observacional. Axiomas para procesos finitos. (*Ref.: [12, 1, 6]*)
3. **Semánticas Concretas: Trazas y Trazas Decoradas.** Caracterizaciones bajo testing, lógicas modales, relaciones entre grafos de procesos de las semánticas de trazas, trazas completas, fallas, trazas con fallas, conjuntos ready, trazas con conjuntos ready, futuros posibles. Variaciones de considerando comportamiento infinito e imagen finita. Relación entre las distintas semánticas. (*Ref.: [8]*)
4. **Semánticas Concretas: Simulaciones y Bisimulaciones.** Caracterizaciones bajo testing, lógicas modales, relaciones entre grafos de procesos de las semánticas de simulación, simulación con conjuntos ready, simulación con doble anidamiento, bisimulación, árboles, mundos posibles. Variaciones de considerando comportamiento infinito e imagen finita. Relación entre las distintas semánticas. (*Ref.: [8]*)
5. **Semánticas Concretas: Otros Conceptos.** Determinismo. Saturación. Axiomatizaciones sobre los operadores básicos. Axiomatizaciones con el operador de elección interna. Criterios para la elección de semánticas. Deadlock vs. terminación satisfactoria. (*Ref.: [8]*)

6. **Semánticas bajo abstracciones: Bisimulaciones.** Bifurcación y abstracción. Bisimulación de bifurcación. Bisimulación η . Bisimulación de demora. Congruencias. Axiomatizaciones. Lógicas modales. (Ref.: [9, 1, 6])
7. **Semánticas bajo abstracciones: Divergencia.** Procesos recursivos. Semántica de transiciones. Tipos de divergencia. Bisimulaciones débil con divergencia. Axiomatizaciones. (Ref.: [11])
8. **Semánticas bajo abstracciones: El espectro completo.** Escenarios de test. Nociones de observabilidad. Comportamiento observacional. Preordenes “may” y “must”. Caracterizaciones relacionales. (Ref.: [7])
9. **Especificación de sistemas de transiciones.** Signaturas, términos y sustituciones. Especificaciones de sistema de transiciones. El formato de regla *tyft/tyxt*. El formato como garantía de congruencia para la bisimulación. Congruencia de trazas. Otros formatos. Extensión conservativa operacional y extensión conservativa operacional sobre un preorden. Extensión conservativa en álgebras con desigualdades. Teoremas generales. (Ref.: [10, 5, 3, 4, 2])

Dictado y carga horaria

Semanalmente se dictarán 4 horas de teóricos. La modalidad radicará fuertemente en la lectura y comprensión de los textos citados con la exposición rotativa de estos trabajos por parte de los alumnos por lo cual la asistencia al curso será fundamental. Se estima una carga horaria de 60 horas presenciales de clase y 60 de trabajo individual.

Bibliografía

- [1] J.C.M. Baeten and W.P. Weijland. *Process Algebra*, volume 18 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1990.
- [2] B. Bloom, W.J. Fokkink, and R.J. van Glabbeek. Precongruence formats for decorated trace semantics. *ACM Transactions on Computational Logic*, 5(1):26–78, 2004.
- [3] B. Bloom, S. Istrail, and A.R. Meyer. Bisimulation can’t be traced. *Journal of the ACM*, 42(1):232–268, January 1995.
- [4] R.N. Bol and J.F. Groote. The meaning of negative premises in transition system specifications. *Journal of the ACM*, 43(5):863–914, 1996.
- [5] P.R. D’Argenio and C. Verhoef. A general conservative extension theorem in process algebras with inequalities. *Theoretical Computer Science*, 177(2):351–380, May 1997.
- [6] W.J. Fokkink. *Introduction to Process Algebra*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2000.
- [7] R.J. van Glabbeek. The linear time – branching time spectrum II (the semantics of sequential systems with silent moves). In E. Best, editor, *Proceedings CONCUR 93*, Hildesheim, Germany, volume 715 of *Lecture Notes in Computer Science*, pages 66–81. Springer, 1993.
- [8] R.J. van Glabbeek. The linear time – branching time spectrum I (the semantics of concrete, sequential processes). In J.A. Bergstra, A. Ponse, and S. Smolka, editors, *Handbook of Process Algebras*, pages 3–99. Elsevier Science Publishers, 2001.
- [9] R.J. van Glabbeek and W.P. Weijland. Branching time and abstraction in bisimulation semantics. *Journal of the ACM*, 43(3):555–600, May 1996.
- [10] J.F. Groote and F.W. Vaandrager. Structured operational semantics and bisimulation as a congruence. *Information and Computation*, 100(2):202–260, October 1992.
- [11] M. Lohrey, P.R. D’Argenio, and H. Hermans. Axiomatizing divergence. *Information and Computation*, 203(2):115–144, 2005.
- [12] R. Milner. *Communication and Concurrency*. Prentice-Hall International, 1989.