# Petri Nets and Mazurkiewicz Traces
# Partnership when Honeymoon is Forgotten

February 17, 2022

## Abstract

**History.** In the mid 1970-ties it became clear that free partially commutative monoids play a central role for the analysis and the understanding of concurrent systems. The foundations were developed in the 1973 JACM-paper by Robert Keller [1]. A few years later, in 1977 Antoni Mazurkiewicz published his seminal Aarhus technical report where he showed that, indeed, partial commutation describes the semantics of one-safe Petri nets perfectly [2]. He also introduced a graphical representation of elements in free partially commutative monoids. In his notation a "trace" is not a (firing) sequence of the net, but a labeled, directed and acyclic graph with an immediate visual understanding.

**Honeymoon.** A concurrent system has a visual representation (a one-safe Petri net), the executions have a visual representation as a trace. The set of all executions are recognized by an asynchronous automaton in the sense of Zielonka ([3]) with a purely algebraic semantics using syntactic congruences. Languages accepted by asynchronous automata have finite syntactic monoids. The Eldorado!

**Honeymoon is over.** Partnership remains: trace theory as a basic algebraic concept should not be forgotten.

**My lecture.** In my talk I will discuss the basic mathematics and extensions to notions of a semi-trace and/or a partial trace. These concepts can be applied to more general types of Petri nets, still the algebra can be expressed within the theory of free partially commutative monoids. It is a versatile tool connecting semantics and combinatorial algebra.

**My only friend, the end.** "Trace theory, may you stay forever young."[1]

## References

[1] R. M. Keller. Parallel program schemata and maximal parallelism I. Fundamental results. *Journal of the ACM*, 20(3):514–537, 1973.

[2] A. Mazurkiewicz. Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, Aarhus, 1977.

[3] W. Zielonka. Notes on finite asynchronous automata. *R.A.I.R.O. — Informatique Théorique et Applications*, 21:99–135, 1987.

---

[1]The Doors (1967), Bob Dylan (1974)