Bisimulation Indexes Induced by Metrics on Actions

In the last chapter we introduced an approximate version of bisimilarity, namely, near bisimilarity, which can serve as a formal tool for describing a kind of approximate correctness of concurrent programs. In this chapter we intend to loosen the concept of bisimilarity from another direction. It is well known that bisimulation expresses the equivalence of processes whose external actions are identical. The condition of possessing the same actions is quite rigorous, and sometimes we meet two processes that fail to fit this condition but are still more or less bisimilar in the sense that whenever a process makes an action, another process can make an action different from but very similar to the action the first process made. The purpose of this chapter is to provide some mathematical tools that are suitable for describing this kind of approximate bisimilarity.

In the previous chapters we always consider a general topology on either actions or agents. But in this chapter we presume that a metric on actions describes a certain distance (and on the other hand, similarity) between actions. Metrics are a special kind of topological structures, and each metric naturally induces a topology called metric topology on the underlying set. From the presumed metric on actions, bisimulation indexes of agents are induced. Based on the concept of bisimulation indexes, the notion of $\lambda$-bisimilarity is introduced. This notion provides us with a continuous spectrum of equivalences that equate processes with different degrees, and it is also very suitable for describing approximate correctness (of course, a new kind of approximate correctness different from that we dealt with within the framework of near bisimulations) of concurrent programs. Since a metric is presumed on actions and a bisimulation index is derived from this metric, the approach adopted in this chapter is also intentional, like near bisimulation.

In this chapter we generalize some of the classical properties of bisimulation in the setting of a bisimulation index. In particular, we give a Hennessy-Milner logical characterization of $\lambda$-bisimilarity. An objection to $\lambda$-bisimilarity is that it is not preserved by parallel Composition. We met a similar situation in our consideration of near bisimilarity. The objection is mainly caused by the requirement of the precise match of the input port and output port in the communication rule $\text{Com}_3$: $\lambda$-bisimilarity allows us to simulate an action of an agent by a different (but similar) action of
another agent, and this obviously does not agree with the requirement regarding the exact match of input and output ports in the communication rule. Fortunately, this objection can be overcome by introducing an approximate communication rule. An indispensable part of this chapter is the three examples presented at the end. As applications of λ-bisimilarity, in these examples we study the behaviors of clocks and show that in contrast to classical bisimilarity, λ-bisimilarity allows us to relate precise clocks with unreliable clocks.

This chapter consists of seven sections. Section 6.1 introduces the concept of bisimulation index in general labeled transition systems and its various properties, especially properties related to operations of transition systems. In Section 6.2 we establish a Hennessy-Milner logical characterization of bisimulation indexes. Some fundamental properties of strong and weak bisimulation indexes are elaborated in Sections 6.3 and 6.4. These properties generalize almost all important properties of classical bisimilarity except substitutivity under Composition. To recover the substitutivity of λ-bisimilarity under Composition, an approximate rule is introduced in Section 6.5. Replacing the original communication rule in CCS by the approximate one, we obtain a modified calculus in which λ-bisimilarity is substitutive under Composition. Real-time systems are being intensively studied in computer science. Two important formal models of real-time systems are timed CCS, which may be used to provide semantics for timed LOTOS, and real-time ACP. Sections 6.6 and 6.7 give two examples in timed CCS and an example in real-time ACP to illustrate how bisimulation indexes can be applied to describe the approximate behaviors of processes. As mentioned before, I hope the concepts and results developed in this chapter (and others) can be used to characterize the relationship between specifications of programs and their approximate implementations. So, more examples need to be developed.

### 6.1 Bisimulation Indexes in Transition Systems

Recall that a metric space is a pair \((X, \rho)\) in which \(X\) is a nonempty set and \(\rho\) is a mapping from \(X \times X\) into \([0, \infty]\) such that

1. \(\rho(x, y) = 0\) if and only if \(x = y\),
2. \(\rho(x, y) = \rho(y, x)\), and
3. \(\rho(x, z) \leq \rho(x, y) + \rho(y, z)\) for any \(x, y, z \in X\).

If (1) is weakened by \((1)'\rho(x, x) = 0\) for each \(x \in X\), then \(\rho\) is called a pseudo-metric, and if (3) is strengthened by \((3)'\rho(x, z) \leq \max\{\rho(x, y), \rho(y, z)\}\) for any \(x, y, z \in X\), then \(\rho\) is called an ultrametric or is said to be non-Archimedean (see [Engelking 1977], Section 4.1).