Abstract for Ph.D. dissertation:

Software Trajectory Analysis: An empirically based method for automated software process discovery

Pavel Senin
Collaborative Software Development Laboratory
senin@hawaii.edu

Recurrent behaviors are considered to be the basic building blocks of any human-driven goal-oriented process, reflecting the development of efficient ways for dealing with common tasks based on past performance. Thus, the ability to discover recurrent behaviors is utterly important for a bottom-up systematic study, modeling, and improvement of human-driven processes. In the context of software development, whose ultimate goal is the delivery of software, the ability to recognize recurrent behaviors enables the understanding, formal description, and effective guidance of evolving software processes. While a number of approaches for recurrent behavior discovery and software process modeling and improvement have been previously proposed, they typically built upon on-line intrusive techniques, such as observations and interviewing, therefore expensive, suffering from biases, and unwelcome by software developers.

In this exploratory study, I have developed and tested the idea of software process discovery via off-line analysis of software process artifacts. For this, I have prototyped and evaluated the Software Trajectory Analysis framework, which is built upon the definition of the “software trajectory” data type, that is a temporally ordered sequence of software artifact measurements, and a novel technique for temporal data classification, that enables the discovery and ranking of software trajectory characteristic patterns. By analogy with the notion of trajectory in Physics, which describes a projectile path in metric space, a software trajectory describes the software process and product progression in a space of chosen software metrics, whereas its recurrent structural patterns are related to the recurrent behaviors.

The claim of this dissertation is that (1) it is possible to discover recurrent behaviors off-line via systematic study of software artifacts, (2) the Software Trajectory Analysis framework provides an effective off-line approach to recurrent software process-characteristic behaviors discovery. In addition to the extensive experimental evaluation of a proposed algorithm for time series characteristic pattern discovery, three empirical case studies were carried out to evaluate the claim: two using software artifacts from public software repositories and one using the public dump of a Q&A web site. The results suggest that Software Trajectory Analysis is capable of discovering software process-characteristic recurrent behaviors off-line, though their sensible interpretation is sometimes difficult.