Optimal Scheduling for Industrial Robot Applications

Johan Ludde Wessén

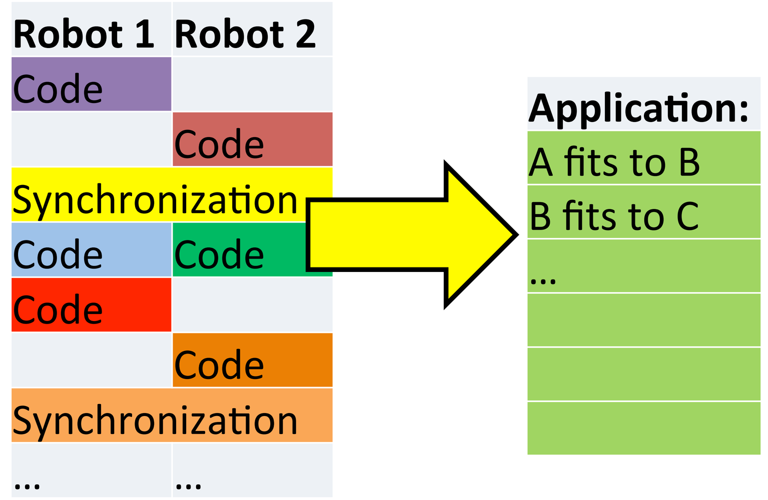
Description:

Industrial robots are typically used for cyclic, identical, jobs. Each job consists of the same sequence of tasks. One example is assembly production. In such an application, components are merged sequentially until the (sub-)assembly is completed. It is then fed forward to the next step of the production line. The aim is to achieve highest possible throughput. Manually creating such a program is feasible for one industrial robot, but very difficult to achieve for multiple robots. Creating the robot program using combinatorial optimization shows great promise, as it gives optimal solutions within reasonable time.

Background & Motivation

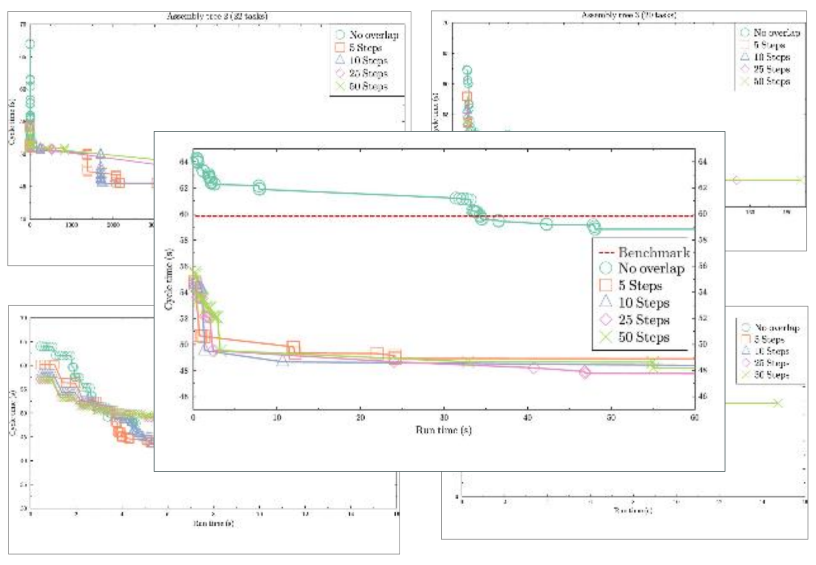
One type of robot application is *assemblies*, i.e. when one or more robots are merging several components into one assembly. Creating such program takes 6-8 weeks of skilled labor. Some products are only manufactured during a few months, before another product is assembled by the same robots. The throughput must be high, even when the re-programming time is short.

Research Goal & Questions



Today, industrial robots are programmed line-by-line for each robot, and are manually synchronized. The goal is to move into a higher level of abstraction, by declaring temporal, logical and geometrical rules of the robots and the application, as well as goals of the latter. A throughput optimal robot program is output, with provable robustness.

Methods & Preliminary Results



The rules of the assembly application – such as order of component, geometric relations, simulated movements etc. is mapped into a combinatorial optimization model, solved to throughput-optimality using Constraint Programming. The model is complex and have features from Vehicle Routing, Job-Shop Scheduling, as well as other constraints that stem from robot interactions. Optimal robot programs are generated in seconds to hours, depending on application characteristics.

Roadmap & Milestones

The research have thus far focused on finding optimal robot program, and provably complete error handling. Upcoming efforts will ensure that running a robot program will be robust and adaptive to changes.
The upcoming research aims to automatically making such robot program possible to execute with minimal human intervention.
One track might be to learn mapping between sensor streams and robots' states, as well as allowing the robot program to improve over time,

Overly simplified Pick and Place Application turned into Graph and finally Control Automaton.