

Spezifikation und Lösen von Dienstplanproblemen mit dem MiniZinc-Solver

Fachgebiet: Computer Perception and Virtual Reality
Betreuer: Prof. Dr. Jürgen Eckerle
Experte: Dr. Federico Flückiger (Generalsekretariat EFD)
Industriepartner: Optor AG, Ittigen

Optimale Dienstpläne bringen Firmen einen erheblichen Wettbewerbsvorteil. Deshalb wird in diesen Bereich viel Geld und Zeit investiert. In der Praxis werden die Probleme mit Parametern und Rahmenbedingungen formal beschrieben und durch einen Solver gelöst oder optimiert. Um die Verwendung dieser komplexen Anwendungen zu vereinfachen, haben wir eine formale Sprache entwickelt, welche die deklarative Constraintprogrammierung um objektorientierte Ansätze erweitert.

Ausgangslage

Wer optimal arbeitet und plant, spart Zeit und Geld. Kein Wunder legen viele Firmen ein Augenmerk auf die Optimierung ihrer Dienst- und Produktionspläne. Ressourcen wie Mitarbeiter sind oft nur exklusiv einsetzbar, was zu Abhängigkeiten, so genannten Constraints, führen kann. Interessanterweise wird diese Planung bis heute in vielen Unternehmen noch von Mitarbeitenden erledigt, obwohl moderne Computersysteme in der Lage wären, solche Optimierungen selbständig oder durch Menschen begleitet durchzuführen.

Um Dienstpläne automatisch zu berechnen, können so genannte Solver eingesetzt werden, die mit Hilfe von ausgeklügelten Algorithmen Lösungen für die spezifizierten Probleme suchen. In der Literatur werden solche Probleme Constraint Satisfaction Problems oder lineare Programmierungsprobleme genannt. Oft sind diese Softwarelösungen sehr komplex und haben eine steile Lernkurve, was zusätzliche Kosten und Aufwände verursacht.

Ziele

Das Ziel ist es, die Verwendung von Solver für die Optimierung und Planung zu vereinfachen. Dazu braucht es geeignete Hilfsmittel, die eine formale Beschreibung von Optimierungsproblemen unterstützen. Es soll ein neuer Ansatz verfolgt werden, der die Formulierung von Constraints in ressourcenabhängigen Problemen vereinfacht und somit näher an eine natürliche

Problembeschreibung bringt. Die Lernkurve wird dadurch verkleinert und der Aufwand für die Erstellung und Modifizierung von Dienstplanproblemen verringert sich.

Idee und Lösungsansatz

Im Rahmen dieser Bachelor Thesis ist eine Sprache entwickelt worden, welche technische Endbenutzer bei der Spezifizierung von Ressourcenplanungsproblemen unterstützt. Die deklarative Constraintprogrammierung wurde um objektorientierte Ansätze erweitert und ermöglicht, auf Basis eines selbst definierten Modells, Rahmenbedingungen zur Problemlösung zu definieren. Constraintprobleme können so agiler und komfortabler deklariert werden, was zu einer natürlicheren Problembeschreibung führt.

Die Sprache und modulare Architektur der Implementation wurde anhand von exemplarischen Referenzproblemen aus dem Bereich der Dienstplanberechnung konstruiert. Obwohl das Hauptaugenmerk auf der Dienstplanung lag, wurden klassische Optimierungsprobleme zur Konsultation beigezogen um Entscheidungen zu validieren. Zur Lösung der Problemspezifikationen, wurde eine Anwendung in Ruby implementiert, welche die entwickelte Sprache parsen, interpretieren und einem Solver zur Lösung übergeben kann.

Fazit

Die Verschmelzung von deklarativen und imperativen Konstrukten stellte eine Herausforderung dar. Die Entwicklung der Sprache und die dazugehörige Implementation haben aber gezeigt, dass die Idee der Erweiterung der Constraintprogrammierung um objektorientierte Ansätze funktioniert. Die entstandene Lösung bringt durch eine natürlichere Beschreibung eine wesentliche Verbesserung in der Modellierung, Lesbarkeit und Handhabbarkeit einer Problemspezifikation.

Die in der Arbeit verwendeten Beispiele besitzen zwar noch nicht die Komplexität realer Dienstplanprobleme, liefern jedoch einen Proof-Of-Concept der formalen Sprache. Offen bleibt, ob und inwieweit sich diese Konzepte in der Praxis bewähren werden.



Nino Müller
nino.mueller@gmail.com



Christoph Schnyder
christoph.schnyder@gmail.com

