Vercors – Verification of Concurrent Data Structures

Hedendaagse softwaretoepassingen stellen steeds hogere eisen aan de snelheid van de onderliggende hardware. Hardware is de laatste decennia wel veel sneller geworden, maar zo langzamerhand lijkt de grens bereikt te zijn: door fysieke beperkingen is het niet mogelijk om processoren nog sneller te laten werken.

Een voor de hand liggende oplossing is om *multi-core* processoren te gaan gebruiken, waarbij meerdere units tegelijkertijd berekeningen uitvoeren. Om optimaal gebruik te maken van zo'n processor moet de software die er op draait beschrijven welke berekeningen in parallel uitgevoerd kunnen worden. Het grote probleem van dit soort software is dat het erg foutgevoelig is; *alle* mogelijke interacties tussen de berekeningen moeten bekeken worden om zeker te weten dat een programma correct werkt. Voor een programmeur is het lastig om geen enkele interactie over het hoofd te zien, en zo te zorgen dat er nooit een fout optreedt.

Binnen de formele methoden wordt al jarenlang gewerkt aan methoden om over correctheid van programma's te kunnen redeneren. Zogenaamde programmalogica's kunnen gebruikt worden om te laten zien dat een programma zich in alle gevallen correct gedraagt. Voor sequentiële programma's zijn hiervoor krachtige *tools* beschikbaar, maar voor multithreaded programma's is dit nog steeds een open probleem. Desondanks is er in de afgelopen jaren veel vooruitgang geboekt, met name door het gebruik van een speciale programmalogica (*separation logic*), die het mogelijk maakt om op een natuurlijke manier te specificeren welke programmagedeeltes elkaar wel en niet kunnen beïnvloeden.

Marieke Huisman heeft deze logica al gebruikt om te redeneren over multithreaded programma's in de programmeertaal Java. Binnen het VerCors project wordt deze techniek verder uitgebreid en gegeneraliseerd, zodat deze ook voor andere programmeertalen gebruikt kan worden. Omdat andere programmeertalen andere technieken gebruiken om te zorgen dat gelijktijdige berekeningen op een correcte manier met elkaar synchroniseren, betekent dit dat de verificatiemethode ook met deze andere synchronisatietechnieken uitgebreid wordt.

Een specifieke toepassing van het project zijn datastructuren die in een multithreaded context gebruikt kunnen worden. Datastructuren zijn algemene structuren om gegevens in op te slaan. Omdat deze in verschillende programma's hergebruikt kunnen worden, is het van het grootste belang dat ze correct geïmplementeerd zijn. Binnen het VerCors project wordt het multithreaded gedrag van deze datastructuren gespecificeerd en geverifieerd. De specificaties worden daarbij op zo'n manier opgesteld dat ze zoveel mogelijk onafhankelijk zijn van het specifieke synchronisatiemechanisme dat gebruikt wordt. Dit zorgt er voor dat een datastructuurimplementatie makkelijk vervangen kan worden door een andere, bijvoorbeeld efficiëntere, implementatie, zolang deze maar over het gewenste gedrag beschikt.

Alle resultaten van het project worden opgenomen in een tool, waarmee het mogelijk wordt om automatisch correctheidseigenschappen van multithreaded programma's te verifiëren.