Naar 'Big Biology': Snelle Verificatie van Grootschalige Simultane Systemen

Ph.D. Thesis
Vrije Universiteit Amsterdam, 2012

Elżbieta Krępska
e.krepska@vu.nl

Sammenvatting

Executeerbare modellen vormen het belangrijkste instrument van de systeembiologie: ze worden gebruikt voor het eenduidig coderen van ons begrip van complexe biologische processen en ze maken het mogelijk experimenten uit te voeren die anders technisch onmogelijk of onethisch zouden zijn. Om model-gebaseerde voorspellingen te kunnen doen over biologie, moeten modellen een getrouwe weergave zijn van de natuur. Hiertoe moeten modellen gecontroleerd worden, d.w.z., worden vergeleken met het beschikbare biologische bewijsmateriaal.

In dit proefschrift onderzoeken we *hoe grote discrete modellen van biologische systemen geverifieerd kunnen worden*. Dergelijke systemen bestaan doorgaans uit een groot aantal simultane kleine onderdelen. De belangrijkste uitdaging bij het verifiëren hiervan is de toestandsexplosie: de exponentiële groei van de toestandsruimte bij het vergroten van het aantal simultane componenten in het systeem. Daarom is de belangrijkste focus van dit proefschrift de schaalbaarheid: het vermogen om systemen met een zeer grote toestandsruimte te behandelen. Om schaalbaarheid te bereiken gebruiken we deels bestaande technieken, maar stellen we ook enkele nieuwe voor op het gebied van grootschalig rekenen en model verificatie.

In hoofdstuk 2 bestuderen we *Monte Carlo simulaties*. In deze methode wordt een systeem geanalyseerd door het uitvoeren en onderzoeken van een groot aantal simulaties; in dit geval wordt een populatie van organismen nagebootst tijdens het ondergaan van het bestudeerde biologische proces. We passen deze benadering toe bij ons model over de bepaling van het celtype tijdens de vorming van een vulva (het orgaan dat eitjes produceert) in de worm *C. elegans*; de grootte van de toestandsruimte van dit model is in de orde van 2^{715}. Voor elk van de 64 genetische verstoringen binnen ons model voerden we 5000 simulaties uit. Naast agressieve optimalisatie van individuele simulaties paralleliseerden we de Monte Carlo experimenten op een rekencluster van computers. Op een gedistribueerde machine met 256 processorkernen resulteerde dit in een vermindering van de benodigde tijd voor de volledige verzameling van controle-experimenten tot minder dan een uur.

Waar simulaties mogelijk niet alle uithoeken van een toestandsruimte kunnen bereiken, zijn formele methoden, d.w.z. het analyseren van een model in de vorm van een computerprogramma, wel in staat alle toestan-

den of trajecten in een systeem te controleren. Een dergelijke methode is *abstracte interpretatie*, die het mogelijk maakt om een eigenschap van een systeem te bewijzen door het interpreteren van slechts de relevante delen hiervan voor de betreffende eigenschap. In hoofdstuk 3 introduceren we BioCheck, een efficiënte procedure voor het bewijzen van stabilisatie (het bereiken van een unieke vast punt) van systemen. Deze applicatie bewijst stabilisatie door de constructie van de globale liveness eigenschap uit een keten van kleinere liveness eigenschappen, die snel zijn aan te tonen. BioCheck bereikt schaalbaarheid door het alleen lokaal doorzoeken van de toestandsruimte op kleine delen van het systeem in plaats van het systeem als geheel. We gebruikten dit om stabilisatie van een 3-D topologie van $200 \times 500 \times 5$ zoogdier-huidcellen te bewijzen; de toestandsruimte van dit model bevat 2^{26mln} bereikbare toestanden.

In hoofdstukken 4 en 5 behandelen we een toestandsruimte als een grote ijle graaf, die moet worden verdeeld over meerdere machines om de toestandsexplosie te beperken; een belangrijk gevolg van deze aanpak is dat de verificatie-algoritmen moeten worden geparalleliseerd. Hoofdstuk 4 introduceert HIPG, een hoog niveau raamwerk voor het schrijven van deze gedistribueerde graafalgoritmen. De kerngedachte in HIPG is dat een gebruiker een graafalgoritme uitdrukt door het definiëren van de data opgeslagen door een knoop en de methoden die geëexecuteerd kunnen worden op een knoop. Het raamwerk maakt het mogelijk om naadloos methoden op een knoop van de graaf uit te voeren, zowel lokaal als op een andere machine. HIPG paralleliseert de graafapplicatie automatisch en zorgt voor de details van de uitvoering op een gedistribueerde machine.

Met behulp van HIPG implementeerden we SPINJADI, een gedistribueerde enumeratieve model checker, waarin een toestandsruimte gaandeweg wordt onderzocht: het begint met een lege graaf, verkent het systeem in de oorspronkelijke toestand, haar opvolgers, de opvolgers van de opvolgers, en zo verder, totdat er een bug gevonden wordt of tot de toestandsruimte is uitgeput. Eigenschappen van oneindige executies worden gaandeweg gecontroleerd met behulp van een cykeldetectie-algoritme door Brim *et al*. Met behulp van SPINJADI hebben we twee mutual exclusion protocollen en een biologisch model van T-cel activatie tijdens een immuunrespons onderzocht.

In hoofdstuk 5 introduceren we TSCCDC, een efficiënt gedistribueerd algoritme voor het vinden van terminaal sterk verbonden componenten (TSCC's) in grote grafen. In de biologie komen TSCC's overeen met toestanden van terminale differentiatie (wanneer een cel stopt met zijn specialisatie), of met stationaire toestanden. TSCCDC is een parallel verdeel-en-heers graafalgoritme: met behulp van bereikbaarheidberekeningen is een graaf te splitsen in vier onafhankelijke subgrafen die niet kunnen worden 'overgestoken' via SCC's, en die zo parallel recursief kunnen worden doorgezocht. Ons algoritme was als enige in staat een realistisch voorbeeldmodel te verwerken: een model van menselijke hematopoëtische (bloed) cellen, met een grootte in de orde van 2^{34} knopen en zijden.

Samengevat, de bijdragen van dit proefschrift zijn als volgt. Hoofdstuk 2 introduceert een nieuwe benadering van het modelleren in de biologie, die we gebruiken voor het modelleren van de vulva ontwikkeling in *C. elegans*—dit model bevat twee eerder gepubliceerde, maar niet gemodelleerde hypo-

thesen over het proces. Hoofdstuk 3 bevat een nieuwe schaalbare techniek om de stabilisatie van gelijktijdige systemen aan te tonen. In hoofdstuk 4 ontwerpen en implementeren wij een nieuw raamwerk om het schrijven van gedistribueerde graafalgorithmen te vereenvoudigen; bovendien wordt dit raamwerk gebruikt voor het implementeren van een gedistribueerde verdeel-en-heers enumeratieve model checker. Hoofdstuk 5 introduceert een nieuw efficiënt gedistribueerd algoritme voor het vinden van terminaal sterk verbonden componenten in een grote graaf.