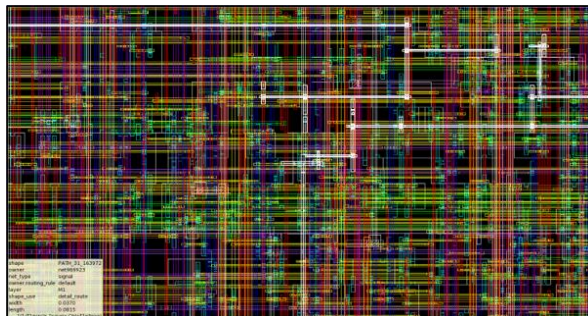


## Stadsontwerp op microschaal

De multinational Synopsys is een wereldspeler in het ontwerp van computerchips. Hun software berekent hoe miljoenen componenten samen zo handig mogelijk op een chip passen. Deelnemers aan de SWI gingen voor de Nederlandse tak van Synopsys op zoek naar een nieuwe manier om die puzzel te beter te leggen.

Om dit verhaal rond te krijgen, waren miljarden verkeersbewegingen nodig. Niet van mensen, maar van elektrische pulsen: beeldbellen, computersimulaties, tekstverwerking - al die klussen dirigeren eindeloze hoeveelheden informatie over computerchips. Zoom in op zo'n chip, en je ziet een netwerk van verbindingen en bestemmingen dat niet onderdoet voor een flinke metropool. Een SWI-opdracht over stadsontwerp op microschaal.



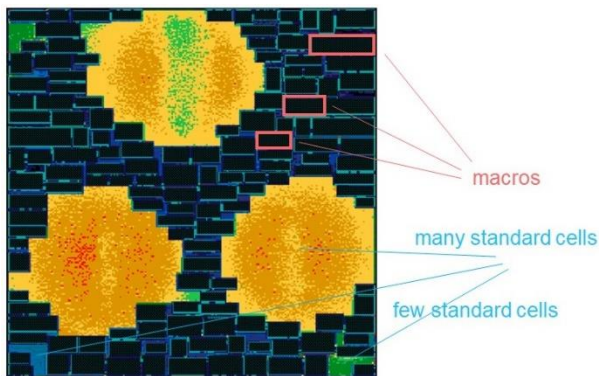
*Ontwerp van een computerchip. Bron: Synopsys*

### Maatwerk

Rekenklussen op standaardprocessoren zoals in je laptop zijn maar het topje van de digitale ijsberg. Ook in je auto, televisie, Wifi-router en wasmachine wemelt het van de computerchips.

Die worden steeds meer op maat gemaakt, vertelt toegepast wiskundige Anna Silvanus van Synopsys.

Op een chip van een paar vierkante millimeter zitten al snel miljoenen clusters van transistoren die samen bijvoorbeeld logische EN- of OF-poorten vormen, en honderden tot duizenden grotere elementen (*macro's*) voor bijvoorbeeld geheugenopslag. Voor optimale prestaties moeten die onderdelen slim worden geplaatst zodat er korte lijntjes zitten tussen knooppunten op de chip.



*Ontwerp voor een computerchip met grotere macro's (rechthoeken) en kleinere elementen. Bron: Synopsys*

### **Schuifpuzzel**

Chiponderdelen plaatsen is een puzzel met uiteenlopende regels. Een rekencel en een geheugenelement die vaak informatie uitwisselen moeten bijvoorbeeld zo dicht mogelijk bij elkaar staan, zodat de chip weinig tijd en energie kwijt is aan het heen-en-weer sturen van informatie. Zo zijn ook er regels voor de plaatsing van onderdelen, zodat die elkaar en de verbindingen ertussen niet in de weg zitten.

Juist dat plaatsingsprobleem is volgens Silvanus een taaie klus. Idealiter zou je die onderlinge afstanden tegelijkertijd met de andere randvoorwaarden moeten kunnen berekenen. In de

praktijk blijkt het moeilijk om in één keer volgens alle regels te optimaliseren.

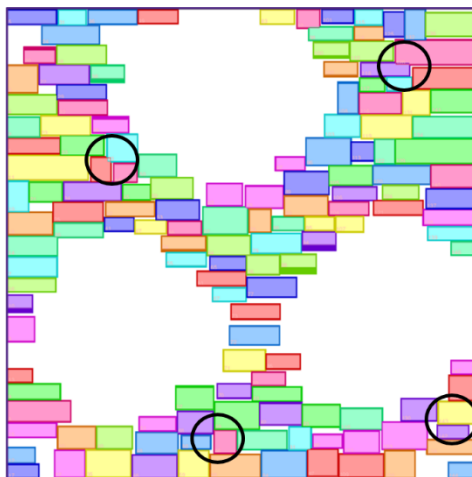
Silvanus: "Onze software maakt eerst een grove plaatsing die ongeveer klopt: onderdelen die veel samenwerken staan bij elkaar, maar sommige elementen kunnen elkaar nog overlappen." Twee onderdelen op dezelfde positie is op een echte chip natuurlijk niet mogelijk, dus gebruikt Synopsys een tweede algoritme dat alle componenten op kloppende posities zet. De uitdaging is daarbij om de oorspronkelijke plaatsing zo min mogelijk aan te tasten.

"Ons eigen algoritme is gebouwd door een collega die ondertussen ergens anders werkt", vertelt Silvanus. "Het werkt prima - meestal dan. Soms doet het erg lang over een configuratie, of geeft het matige oplossingen. We wilden het toch al uitbreiden, dus dacht ik: waarom vragen we de SWI niet of zij creatieve ideeën hebben?" Aan de SWI-deelnemers de uitdaging om het plaatsingsprobleem te kraken: een schuifpuzzel met miljoenen stukjes.

### **Standaardprobleem**

Voor promovendus Fabio Buccoliero aan de VU was de opdracht van Synopsys even schakelen. "Mijn eigen specialisme is veel theoretischer, en tijdens je promotie heb je natuurlijk meer dan een week de tijd om een probleem te kraken. Juist daarom deed ik mee aan SWI - ik wilde weleens een heel andere uitdaging."

Samen met collega-deelnemers leverde Buccoliero in de SWI-week niet één, maar drie mogelijke oplossingen voor het plaatsingsprobleem van Synopsys. Zo vonden de onderzoekers niet alleen zelfstandig uit hoe het bedrijf het plaatsingsprobleem zelf oplost, maar bedachten ze ook compleet andere aanpakken voor hetzelfde probleem.



*Ontwerp van een computerchip met overlappende elementen. Bron: Synopsys*

"Feitelijk is dit een heel klassiek probleem", legt Buccoliero uit: "Je hebt allerlei variabelen - de plaats en afmetingen van alle macro's - en een serie randvoorwaardes waar die aan moeten voldoen." Niet over elkaar heen liggen bijvoorbeeld, maar ook uitgelijnd zijn op een net raster, of voldoende ruimte om de macro's heen laten voor verbindingen. De standaardaanpak voor zulke problemen is een krachtpatser uit het computeronderzoek: *Mixed Integer Programming (MIP)*.

"Het is een techniek die hier enorm voor de hand ligt", zegt Buccoliero. "Toch was het goed dat ze ons niet aan het begin al vertelden hoe ze dit probleem zelf aanpakken, dan kom je zo vast te zitten aan één specifieke insteek."

### **In de buurt**

Voor MIP-problemen zijn softwarepakketten die oplossingen uitrekenen, als je het probleem maar goed weet te formuleren. Zo gezegd, zo gedaan, en de SWI-ers weten in een kleine week software op te leveren die vergelijkbare chipconfiguraties

uitreken als Synopsys. Om te zorgen dat het algoritme zich niet verslikt in de rekenklus van te plaatsen onderdelen, werkt de berekening van grofmazig naar steeds gedetailleerder.

Het rekenrecept van het SWI-team is nog wel trager dan het geoptimaliseerde algoritme van Synopsys zelf, maar dat is voor Anna Silvanus geen probleem: "Ons eigen MIP-pakket is op maat gebouwd voor ons probleem, natuurlijk is dat sneller." Zo rekent de software van Synopsys configuraties in een paar seconden door, waar de SWI-oplossing een uur nodig heeft. "Dat zit al erg in de buurt, dus ik kijk uit naar de details van hun aanpak om daarmee ons eigen algoritme te verbeteren!"

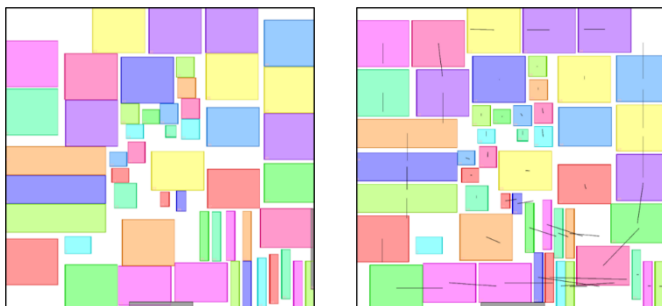
### Hebberig

Terwijl Team MIP de chip-onderdelen aanpakte met het breekijzer van *Mixed Integer Programming*, keken hun collega-wiskundigen naar heel andere manieren om de macro's op hun plaats te krijgen.

"Grappig genoeg is de techniek waarmee ze het verste kwamen, ongeveer de manier die wij gebruiken voor onze voorlaatste configuratie", zegt Anna Silvanus. Dat is de rekenslag waarbij onderdelen wel in de buurt komen van partners waarmee ze veel samenwerken, maar ook deels over elkaar heen komen te liggen.

'Hebberig' ('Greedy' in het Engels), noemen de SWI-ers deze aanpak: hun software kiest willekeurig een van de grotere componenten, plaatst die op zijn originele positie op de chip, en gaat door naar de volgende op de lijst. Als een macro wil landen waar een ander onderdeel al ruimte heeft ingepikt, duwt de software de nieuwkomer iets opzij. Buccoliero: "Zo blijf je dicht bij de originele configuratie, en krijg je toch snel een kloppende oplossing - als er al een oplossing mogelijk is."

De hebberige aanpak blijkt niet altijd een werkende oplossing te leveren. Niet vreemd natuurlijk, want misschien is een van



*Computerchip-ontwerp voor en na optimalisatieslag. In de rechterafbeelding zijn elementen verschoven van hun originele positie (begin zwarte lijn) naar een correcte plaatsing (eindpunt zwarte lijn in het midden van het element.) Bron: Synopsys*

de onderdelen die als eerste landt, juist de macro die alle andere onderdelen in de weg zit. Nog tijdens de studieweek weten de SWI-ers hun hebbelijke algoritme verder te verbeteren, bijvoorbeeld door de volgorde van plaatsing door elkaar te husselen en door sets van al geplaatste onderdelen als een blok te laten bewegen.

AI is het hebbelijke algoritme voor Synopsys niet zo interessant om de finale plaatsing te berekenen, het bedrijf is zoals veel opdrachtgevers onder de indruk van de snelheid waarmee SWI-ers werken. Silvanus: "Na de eerste dag van brainstormen lagen er vooral veel vragen op tafel, maar de volgende ochtend hadden ze al drie manieren uitgedacht om ons probleem aan te vliegen. Fantastisch hoe mensen met zulke verschillende wiskundige achtergronden zo snel meters maken."

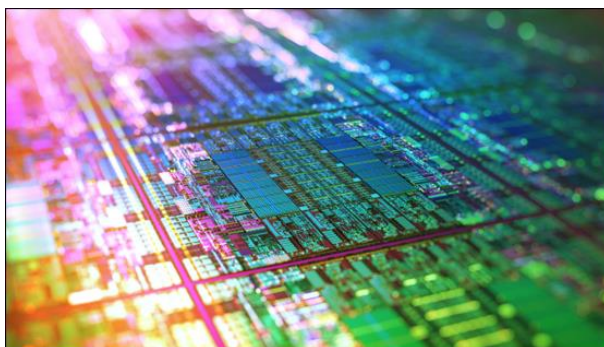
Buccoliero is het daarmee eens. Voor hem is de grootste winst van de SWI-week om een praktisch probleem te kraken met collega's uit andere takken van de wiskunde. "Heel interessant, en nuttig om zo ervaring op te doen met samenwerking met het bedrijfsleven." Die samenwerking leidt op de valreep tot wat Buccoliero een "vrij geniale aanpak" noemt: "ik mag dat zeggen, want ik heb het niet zelf bedacht!"

## Schudden voor gebruik

"We kregen de ingeving dat de meeste onderdelen op de chip al best dicht bij hun optimale positie liggen", herinnert Buccoliero zich. "Dus wat als we ze nou een beetje heen en weer schudden, alsof je bijna passende puzzelstukjes op hun plek laat vallen?"

Wiskundig krijgt dat schudden vorm door macro's die uit positie liggen iets heen en weer te trillen met een *Brownse beweging*, waarbij een deeltje willekeurig stapjes naar links, rechts, naar voren of naar achteren neemt. "Gemiddeld bewegen macro's zo niet ver van hun startpositie, terwijl overlappende stukjes door zachtjes schudden wel los van elkaar komen."

Aan het eind van de SWI-week is die Brownse-beweging aanpak nog niet ver gevorderd, maar de eerste aanzet trekt de aandacht van Synopsys. De moeite waard om verder uit te werken, denkt het bedrijf. "We hebben van SWI drie heel verschillende aanpakken gekregen", zegt Silvanus. "Dat was nooit gebeurd als we de deelnemers al vanaf het begin op ons eigen algoritme hadden gezet. Juist voor zo'n *out-of-the-box* aanpak als de Brownse beweging heb je als bedrijf geen tijd. Heel waardevol dat SWI die voor ons naar voren heeft gebracht."



Computerchip. Bron: Synopsys

