
De potentie van Matlab als standaard

Een inventarisatie

Ingeleid door H. Boukes

In grote lijnen is onze hydrologisch instrumentarium wel bekend. Van oudsher hebben we analytische formules, en sinds de jaren tachtig hebben we de beschikking over numerieke rekenprogramma's. De jaren negentig hebben de optimalisatie-programmatuur daar aan toegevoegd en de kwaliteitsprogramma's toegankelijk gemaakt.

De laatste tijd zien we regelmatig de naam 'Matlab' terugkeren in artikelen in Stromingen. Matlab is een pakket met wiskundige routines waarmee het relatief eenvoudig wordt om ingewikkelde wiskundige bewerkingen te programmeren. Dit kunnen analytische formules zijn, maar ook numerieke modellen kunnen geprogrammeerd worden. De resultaten kunnen netjes gevisualiseerd worden.

Er zijn mensen die lyrisch over het pakket zijn, en mensen die het niet zien zitten. Als redacteur van STROMINGEN heb ik willen onderzoeken wat nu de status is van Matlab: is het de nieuwe toekomst of een vluchtige modegril? Daartoe zijn zes hydrologen benaderd waarvan we het idee hebben dat ze hier iets zinnigs over te melden hebben.

Inleiding

Het gebeurt regelmatig in de techniek dat het niet de beste oplossing is die tot standaard wordt verheven. In de tachtiger jaren ontwikkelde Philips het Video-2000-systeem, dat in alle opzichten superieur was aan de systemen van andere maatschappijen. Inmiddels werken al onze apparaten volgens het VHS-systeem van Sony. Ook in de jaren tachtig en negentig was het besturingssysteem van Apple Computers vele malen beter dan MS-DOS en zelfs Windows, maar wilde toch iedereen een 'IBM-compatible' machine hebben.

Dit soort ontwikkelingen lijkt nogal storend, maar op een gegeven moment is het feit dat er voor een standaard gekozen wordt belangrijker dan welke standaard er gekozen wordt. Zodra namelijk de massa achter een bepaald product gaat staan, worden verdere ontwikkelingen gebundeld langs de weg van die standaard, en worden alle problemen die die standaard in zich heeft, geleidelijk opgelost.

Binnen de hydrologische modelwereld zijn we nog niet toe aan één standaard. Het RIVM rekent met LGM, RIZA met MLAEM, Alterra met SIMGRO, Iwaco met TRIWACO, en de programma's MicroFEM en MODFLOW worden op meerdere plaatsen toegepast.

Regelmatig lijkt het ontbreken van een standaard een erg onhandige situatie te zijn. Wie het ene model wil gebruiken met de gereedschappen van een ander, merkt dat conversie erg lastig is. Vanwege het budget nemen we dan maar genoeg met sub-optimale oplossingen, of een enkele keer zelfs ondeugdelijke oplossingen.

In het buitenland is in mijn waarneming de situatie veel overzichtelijker. MODFLOW blijkt internationaal toch eigenlijk de enige standaard te zijn, wat er toe leidt dat juist langs de MODFLOW-poot veel ontwikkelingen hebben plaats gevonden. Die ontwikkelingen hebben enerzijds betrekking op de pre- en postprocessing, anderzijds op koppelingen met stromings- en kwaliteitsprogramma's. In de nieuwste ontwikkelingen ontstaan koppelingen tussen de andere rekenprogramma's en de MODFLOW-afgeleiden MT3D en RT3D.

Is het echt wenselijk dat er twee landelijke modellen zijn, te weten LGM van het RIVM en NAGROM van RIZA, die beide niet aansluiten bij de overige in Nederland gangbare rekenmethoden? In hoeverre is het op deze manier waarschijnlijk dat er geprofiteerd gaat worden van de potentie van die modellen, bijvoorbeeld om voor kleinschaliger studies de randvoorwaarden te zetten?

En is het nu wel zo wenselijk dat voor al die programma's kalibratie-functionaliteit geschreven moet worden, dat er koppelingen naar MT3D gemaakt moeten worden, die qua waterbalans per definitie nooit volledig kloppen?

Persoonlijk heb ik altijd gevonden dat we ons in Nederland moeten voegen naar de wereldstandaard, niet zozeer omdat dat het beste programma is, maar om de ontwikkelingsinspanningen te bundelen. Ik denk zelfs wel eens dat we door onze eigenzinnigheid kansen hebben laten lopen om onze positie als best gemodelleerde land uit te dragen. In het kader van onze op overeenstemming gerichte polder-mentaliteit wijzen we dan liever op de veelkleurigheid van ons modellenpalet, dan op de inefficiëntie ervan. Aan de andere kant: er is de afgelopen jaren toch ook niet erg veel fout gegaan. Alle programma's op zich deugen wel, en waarom zou je je dan inspannen om een andere situatie te bereiken?

Er zijn twee redenen dat ik denk dat nu het moment gekomen is om nog eens te overwegen of de situatie niet veranderd zou moeten worden.

Ten eerste speelt de verwatering van de wereldstandaard. MODFLOW blijkt zelf niet meer de eenheid die het jarenlang geweest is. De makers menen nieuwe opties te moeten toevoegen, maar de consistentie met eerdere toevoegingen wordt daarbij op het spel gezet. Bovendien bestaat er geen eenduidige handleiding meer van het programma. Alle informatie is verknipt over vele rapporten. De gebruiker mag hieruit zijn eigen puzzel samenstellen. Daar komt bij dat het bedieningsgemak van het oorspronkelijke MODFLOW dermate gering is, dat commerciële partijen pre- en postprocessors aanbieden, die onderlinge uitwisselbaarheid verhinderen. Ook internationaal ontstaat er een leemte in de invulling van de standaard op het gebied van hydrologisch modelleren.

De tweede reden om de huidige situatie te toetsen aan de gewenste, is de beschikbaarheid van Matlab, een programmeer- en rekenomgeving vol voorgeprogrammeerde wiskundige modules. Met dit pakket is het relatief eenvoudig om allerlei soorten rekenmodellen te maken, dus ook hydrologische. Op meerdere plaatsen zijn al Matlab-modules geschreven die in grote lijnen hetzelfde kunnen als het huidige MODFLOW, maar ook eindige-elementen-berekeningen en analytische uitwerkingen horen tot de toegankelijke mogelijkheden. Daarmee voegt Matlab inhoudelijk dus niets toe. Dat kunnen we met onze verzameling hydrologische programma's nu ook al. Het is echter wel zo dat Matlab de wiskundige ontwikkelingen volgt. Als er nieuwe ontwikkelingen zijn in de vertaling van wiskunde naar de

computer kan de hydroloog deze met Matlab gelijk implementeren in de actuele berekeningen. We hoeven daarvoor niet meer te wachten op de bereidwilligheid van de leverancier van hydrologische programmeurs. Op den duur kan Matlab dus meer.

Een tweede voordeel is het enorme potentieel aan Matlab-gebruikers. Het geeft ons mogelijkheden om met allerlei vakmensen te communiceren over hun manieren van rekenen. Wellicht dat we daar heel veel van kunnen leren. Daarvoor kijk ik bijvoorbeeld in de richting van meteorologen, die grootschalige instationaire 3-D stromingmodellen gebruiken voor de voorspellingen van luchtbewegingen.

Een derde voordeel is dat al onze manieren van modelleren weer onder één noemer gebracht kunnen worden. Of je nu een landelijk model maakt of een bronbemaling doorrekent, je moet allemaal door een Matlab-conversie heen om het model van data te voorzien. Op die manier verlagen we de barrières tussen landelijke, regionale en lokale modellen. Zeker als we afspraken maken over data-opslag en verwerking kunnen we meer dan nu van elkaars kennis profiteren.

Tot zover de kansen. Bedreigingen? Ten eerste de prijs van Matlab, voor een beetje bedrijf geen probleem, maar voor een kleine jongen als ikzelf (waarvan er steeds meer lijken te komen) en erg hoge horde.

Ten tweede de hydrologische Matlab-programma's. Die zijn er namelijk nog niet, of eigenlijk alleen hap-snap. Als het ons lukt om daar structuur in aan te brengen, en tot een eenheid te komen, dan zijn er mogelijkheden om tot een nieuwe standaard te komen. Zo'n standaard moet dan op zijn minst in de basis gratis beschikbaar zijn.

Op voorwaarde van samenwerking en coördinatie lijken er dus hele mooie dingen mogelijk. De redactie van STROMINGEN heeft gemeend dat dit een voldoende belangrijke vraag is om er een onderzoekje naar in te stellen. Daarom zijn zes prominente hydrologen gevraagd om antwoord te geven op een drietal vragen.

Reacties op de inleiding

Willem-Jan Zaadnoordijk (IWACO):

Bij de keuze voor een rekenprogramma gaat het vaak niet om de mogelijkheden ervan, maar om de conversies en hulpprogramma's. Ondanks dat Windows op dit moment de standaard is, bestaat er nog steeds een groep Apple-Macintosh-gebruikers, en blijkbaar voorziet ook Linux in een behoefte. Het gaat zeker ook om de gebruikersschil. Als je je dat realiseert is MODFLOW al veel minder standaard en kun je hoogstens stellen dat voor standaard rekenwerk veel MODFLOW gebruikt wordt. Het heeft een aantal belangrijke tekortkomingen. De belangrijkste is de starre geometrie, daarnaast ook topsysteem en zoet-zout. Er zijn overigens niet-standaard versies waarin sommige problemen verholpen zijn... Ik zet ook mijn vraagtekens bij het rendement van de externe MODFLOW ontwikkelingen als je ziet dat de USGS alleen eigen ontwikkelingen meeneemt naar volgende versies.

Verder vraag ik me af wat de vraag is. Wat moet er standaard en openbaar worden? Gaat het nu om een public domain TRIWACO of alleen een public domain FLAIRS (het centrale rekenprogramma)?

Het belang van een standaard gebruikersschil lijkt me groter dan van een standaard rekenprogramma. Het gaat om import- en exportmogelijkheden, voor- en nabewerkings-opties en dergelijke.

We hebben vraagtekens bij de potentie van Matlab. Is Matlab (of de GNU-kloon Octave) een geschikt programma om een goede modelleeromgeving te maken, die algemeen toepasbaar is en voldoende flexibel is? De opzet van een geohydrologische simulator in Matlab vraagt zoveel kennis en implementatie dat men toch afhankelijk is van een beperkt aantal programmeurs. En die programmeurs zullen weer de nieuwe opties aan moeten spreken. Ook voor Iwaco is de prijs van Matlab een hinderpaal. En hoe gaat het met de ontwikkelingen op GIS- en GUI-gebied?

Je kunt met Matlab misschien makkelijker gebruik maken van ontwikkelingen op andere vakgebieden en speciaal de wiskunde, maar kun je de buitenlandse hydrologische wereld voldoende meekrijgen om van de hydrologische ontwikkelingen te blijven profiteren?

Kick Hemker (VU Amsterdam en MicroFEM):

De inleiding bevat een te simpele kijk op het ontstaan en de voordelen van standaarden. Juist omdat er ook een MacOS van Apple bestond (naast de standaard dus) is er iets ontwikkeld wat op het beste van de bestaande besturingssystemen leek. De ontwikkeling van standaard software treedt niet op *omdat* de massa er achter staat, maar omdat het niet kan achterblijven bij de ontwikkelingen van vergelijkbare producten. De ontwikkeling van software is juist gebaat bij (een zekere mate van) diversiteit. In die zin wijst het grote aantal programma's in Nederland er op dat we juist heel ver zijn. Natuurlijk is er ook een grens: een programma moet wel minstens door een heel instituut worden ondersteund om levensvatbaar te zijn/blijven.

Het probleem is dus niet dat er te veel programma's zijn, maar dat je de informatie niet makkelijk kunt uitwisselen. Alle programma's hebben hun eigen bestaansrecht, dus ook hun eigen sterke en zwakke kanten. Je kunt niet alle sterke kanten in één programma combineren. Het kiezen voor één standaard programma lost wel het conversie-probleem op, maar leidt ook tot versraling. Ook tot inefficiëntie, als een ander programma beter geschikt zou zijn om een probleem op te lossen.

MODFLOW is zeker niet 'de enige standaard', maar MODFLOW is wereldwijd gewoon het meest gebruikte programma (eigenlijk de meest gebruikte methode, want er zijn verschillende programma's). Als eindige differentie over de hele wereld veel meer gebruikt wordt dan eindige elementen, moet je niet zeggen dat eindige differentie 'de enige standaard' is.

De problemen bij het overnemen van randvoorwaarden worden overschat. Randvoorwaarden van een numeriek model mogen best een beetje fout zijn. Het inzoomen op bestaande modellen heeft in het algemeen ook weinig zin en leidt zelfs gemakkelijk tot fouten. Waterbalansen zijn heel nuttig, maar ze hoeven in een model niet per definitie te kloppen: een voldoende nauwkeurige benadering is goed genoeg. Met de analyse uit de inleiding dat we ons moeten voegen naar de 'wereldstandaard' ben ik het dus niet eens.

De differentiatie van MODFLOW ontstaat 'vanzelf' (door marktwerking) als er een hele grote groep gebruikers van een programma bestaat. Bij MODFLOW wordt dat bovendien in de hand gewerkt doordat de rekencode openbaar is. Een dergelijke ontwikkeling kun je dan niet voorkomen.

Het argument dat de doorvertalingen van wiskundige ontwikkelingen sneller gaan, is heel erg verzocht. Ik wil 'de hydroloog' wel zien, die de wiskundige ontwikkelingen volgt en deze direct in Matlab implementeert. De meeste hydrologen willen niet zelf programmeren. Twintig jaar geleden was het ook al mogelijk om je eigen programma te schrijven. Of je doet iets zelf, of je wacht tot anderen het voor je doen. De andere vakgebieden waarop Matlab wordt gebruikt zijn heel interessant, maar 'de hydroloog' zal daar niet veel van leren, wat hij beroepsmatig kan toepassen.

Het lijkt mij heel nuttig om bij deze discussies steeds voldoende onderscheid te maken tussen het gebruik van programma's (en de daarbij mogelijke rekenmethoden) en het gebruik en de vorm (het formaat) van benodigde gegevens. Matlab is handig voor het uitvoeren van berekeningen. Een programma voor het uitvoeren van hydrologische berekeningen bestaat echter maar voor een klein deel uit een rekenkern; de rest is de schil (user-interface) die het de gebruiker mogelijk maakt om allerlei andere dingen met een model te doen dan rekenen. Deze schil is bij professionele programma's vaak 80%-90% van het programmeerwerk.

Rien Pastoors en Karel Kovar (RIVM):

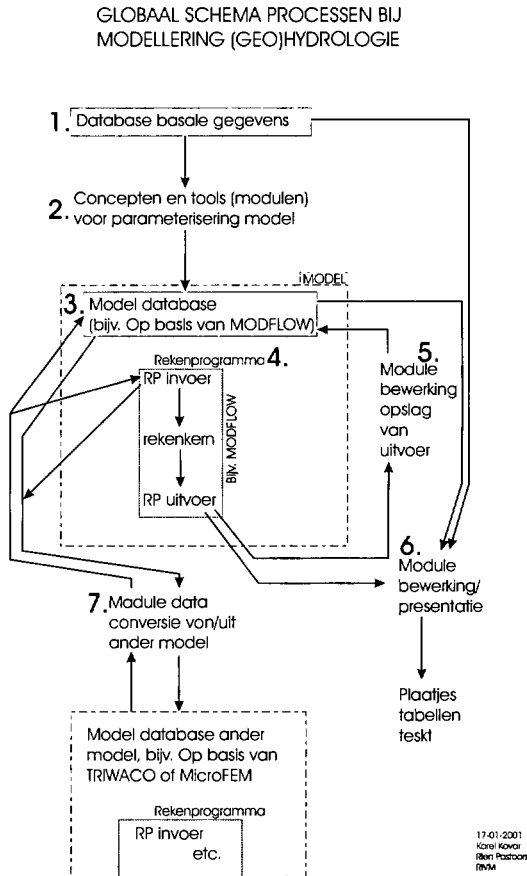
Wij zijn van mening dat het relatief grote aantal rekenprogramma's in dit land (LGM, MicroFEM, NAGROM, SIMGRO, TRIWACO) op het gebied van (geo)hydrologie als positief moet worden beschouwd. Wij twijfelen er niet aan dat elk rekenprogramma zijn eigen voordelen (conceptuele opties) heeft en dat het een onbegonnen werk zou zijn om de voordelen van alle rekenprogramma's in één enkel nieuw rekenprogramma in te bakken. Daarnaast, en dat mag filosofisch klinken, komt uit de 'confrontatie' en het coëxisteren van al die rekenprogramma's vast iets positiefs tevoorschijn. Noem het maar de vruchten van uitdagingen en van die veelkleurigheid. Neen, het streven naar een enkel super-rekenprogramma komt ons niet alleen als onwenselijk maar ook als niet-realistisch over. Om een parallel te trekken: wij willen immers ook de biodiversiteit in stand houden?!

Maar toch kan en moet veel beter! We vermoeden dat de kern van het in de inleiding beschreven probleem voortkomt uit de moeite/inspanning die hydrologen meemaken (zelf of bij anderen) bij de bouw en toepassing van modellen, in het bijzonder als er een aantal verschillende modellen (ieder met eigen database) in het spel zijn.

Een antwoord op de drie vragen kunnen wij pas na een relatief lange inleiding geven. Wij vinden dat de vragen niet voldoende houvast (richting) geven voor de oplossing van het eigenlijke probleem. Daarom geven wij naast een antwoord op de vragen ook *vier suggesties voor de verbetering van het modelleringsproces*.

BEGRIP 'REKENSTANDAARD'

Onzes inziens zou je het woord 'rekenstandaard' niet moeten gebruiken! Standaard waarvoor? Voor een programmeertaal? Dat denken wij niet. Of voor een modelomgeving in de ruime zin? Daar lijkt het het meeste op. Als wij het begrip 'rekenstandaard' in de vragen vervangen door 'standaard voor rekenprogrammaschil' of 'modelleerstandaard' (niet 'modelstandaard!'), dan zouden wij op die vragen al een gericht antwoord kunnen geven.



Modelminded als wij zijn, hebben wij het modelleerproces eerst in een schema gezet. Het is hierbij zeker nodig consistent onderscheid maken tussen de begrippen rekenprogramma en model. Een model is een rekenprogramma gevuld met invoerdata, dienend voor een gegeven probleemstelling. Het 'modelleerproces' bestaat in principe uit de volgende zeven elementen:

- 1 *databases met basale gegevens*: de informatie over het (geo)hydrologische systeem (k, kD, c, diepten, etc.) en overige randvoorwaarden (waterstanden, neerslag, etc.), opgeslagen in een aantal databases, zoals REGIS, TOP10Vector, WIS, BIS (bodemkaart), DINO (voorheen OLGA) en KNMI;
- 2 *faciliteiten voor de parameterisering van modelinvoer*: dit zijn bijvoorbeeld procedures en algoritmen voor de bepaling van c-waarden, kD-waarden en de relatie tussen grond- en oppervlaktewater. De faciliteiten bestaan uit concepten (methoden) en tools (software);

- 3 *modeldatabase*: deze database bevat de modelgegevens gemaakt op basis van het gehanteerde rekenprogramma. Bijvoorbeeld, voor MODFLOW bevat dit de invoer- en uitvoerparameters op basis van een FDM-grid;
- 4 *rekenprogramma*: een door middel van een computertaal weergegeven (stelsel van) van fysische of empirische relaties/wetten. In ons geval betreft dit meestal de wet van Darcy en de wet van behoud van massa, aangevuld met specifieke aspecten gebonden aan de gebruikte oplossingstechniek (FEM, FDM, AEM, etc.). Een rekenprogramma bevat ook onderdelen voor het lezen van de invoer en het aanmaken van de uitvoer;
- 5 *module voor bewerking en opslag van modeluitvoer*: de resultaten van een rekenprogramma zijn vaak later (in een bewerkte vorm) voor andere processtappen nodig;
- 6 *module voor bewerking en presentatie*: hiermee wordt de modelinvoer en modeluitvoer naar een presentabele vorm omgezet;
- 7 *module dataconversie van / uit ander model*: voor de conversie van gegevens (invoer rekenprogramma) naar de invoer voor het gehanteerde rekenprogramma, of andersom vanuit het gehanteerde rekenprogramma naar een ander type rekenprogramma.

Rondom de eigenlijke rekenprogramma's (LGM, MicroFEM, NAGROM, SIMGRO, TRIWACO) is vaak een gebruikers-interface gebouwd. Deze schil (modelleromgeving) kan de elementen 2 t/m 7 bevatten. Het gaat hier te ver om de daadwerkelijke inhoud daarvan hier per geval in kaart te brengen.

VIER SUGGESTIES VOOR DE VERBETERING VAN HET MODELLEERPROCES

Wij denken dat het modelleerproces langs volgende wegen efficiënter zou kunnen worden gemaakt:

- a punt 1 in figuur: maak de basale gegevens algemeen beschikbaar. Dit betreft databases, zoals REGIS, TOP10 Vector, WIS, BIS (bodemkaart), DINO- en KNMI-gegevens;
- b punt 2 in figuur: er is overeenstemming nodig over mogelijke concepten voor modelschematisering. Het is echter niet de bedoeling dat er voor één proces (parameter) slechts één concept moet worden gebruikt. Het type concept is afhankelijk van de probleemstelling en ook van de modelschematisering. Voorbeelden van concepten zijn de algoritmen/procedures voor de bepaling van c-waarden, de nuttige grondwateraanvulling, en de relatie tussen grond- en oppervlaktewater;
- c punt 2 in figuur: tools moeten beschikbaar zijn (worden gemaakt) voor de parameterisering, met andere woorden voor het aanmaken van modelinvoer bij een gegeven modelschematisering. Een dergelijk gereedschap is doorgaans een rekenprogramma waarin een bepaald concept (zie punt b hiervoor) verwerkt is. Op dit moment gebruikt elk van de genoemde rekenprogramma's een eigen formaat voor de in- en uitvoer van gegevens. Hier is enige winst te boeken door in alle rekenprogramma's (eventueel naast eigen in- en uitvoerformaat) een algemeen bruikbare formaat te hanteren. Sowieso zou dit voor ruimtelijk (x/y vlak) georiënteerde gegevens handig zijn. Het formaat zou bijvoorbeeld HDF kunnen zijn, of CDF of GRIDASCII (Arc/INFO);
- d punt 7 in figuur: de problemen van de overdracht van modelgegevens tussen twee rekenprogramma's kunnen op korte termijn relatief eenvoudig worden opgelost door de model-schil van een module te voorzien dat de conversie verzorgt van de complete invoer van rekenprogramma X naar de complete invoer van rekenprogramma Y.

Is het gewenst om in hydrologisch Nederland op zoek te gaan naar een (nieuwe) rekenstandaard?

Willem-Jan Zaadnoordijk (IWACO):

Afspraken over data-opslag en verwerking zijn veel belangrijker om de samenwerking te bevorderen en dubbel werk te voorkomen. De meeste tijd zit in modelleren en niet in programmaontwikkeling.

Ik denk dat de behoefte aan een standaard rekenprogramma op zich niet bestaat. Zolang je een programma hebt wat je conceptuele modellen aan kan is het goed. De keuze wordt dan veeleer bepaald door de efficiëntie waarmee je dat kan doen en dus de schil.

Ik moet daarbij denken aan een uitspraak van Frank Schwarz op het 1e analytische elementen-congres is 1994 in Indianapolis. Hij zag het nut van de discussie analytische elementen – eindige elementen – eindige differenties niet zo. Hij voorzag dat de gemiddelde hydroloog in de toekomst niet zou weten wat er onder de motorkap van zijn simulatiesysteem zou zitten. De gebruikersschil is veel meer bepalend in de efficiëntie van het modelleren, waarbij naast een efficiënte aansturing van de simulatiemogelijkheden van het rekenprogramma ook een overzichtelijke dataopslag verzorgd moet worden.

Ik zou het zeer toejuichen om eindelijk die starre rechthoeken te verlaten.

Kick Hemker (VU-Amsterdam en MicroFEM):

Wanneer dataconversie het probleem is, moet je dat niet willen oplossen met een (nieuwe) rekenstandaard.

André Blonk (Tauw):

Het kiezen van een nieuwe rekenstandaard is naar mijn idee alleen zinvol als men bereid is om ervaringen met elkaar te delen. In Nederland is de cultuur nog niet aanwezig om ervaringen en programma's vrij met elkaar uit te wisselen. Als ideaal zie ik een gereedschapskist waar een ieder vrij gebruik van kan maken. Programma's mits goed gedocumenteerd kunnen door een ieder daaraan worden toegevoegd. Het probleem wat dan ontstaat is waarschijnlijk het handhaven een standaard en de kwaliteit. Het gebruik van een programma als Matlab kan zo'n zekere wildgroei in toom houden.

Wim de Lange (RIZA):

Ja, mits het een Open Software standaard is voor iedereen beschikbaar en bruikbaar, gratis en waaraan iedereen kan bouwen en verbeteren en waaraan een kwaliteitsborgingsorganisatie wordt gekoppeld. Maar, dit heeft echter niet direct invloed op de overdraagbaarheid van modeldata zoals dat in de inleiding wordt beoogd! Daarvoor moet een directe en tweezijdige link met de basisdata worden gelegd, anders dreigen geïnterpreteerde parameterwaarden een eigen leven te gaan leiden. Voor de verbinding tussen modellen en data is de ontwikkeling voor een standaard raamwerk modellen reeds jaren geleden op gang gebracht

en daarvan worden nu reeds implementaties ontwikkeld. Het gaat dus eerder om de stekkers dan om het rekenhart.

Omdat het rekenhart van analytische elementen veel inspanning vergt en door verschillende mensen wordt gebouwd is inmiddels al drie jaar gelden door alle ontwikkelaars van AEM-achtige modellen besloten om aan een dergelijke standaard (en organisatie) te gaan werken. Het ontwerp kan ook FDM- en FEM-typen modellen aan en er kunnen allerhande schillen van gebruikers-interfaces aan worden gekoppeld. Voor mij is het essentieel dat iedereen met zijn eigen tools gebruik kan gaan maken van de bibliotheek van modules; dus ook Matlab-gebruikers, maar ook C++, FORTRAN- of PYTHON-gebruikers. Matlab zou voor een zeker aantal gebruikstypen geschikt kunnen zijn, maar vergeet niet hoeveel inspanning het kost om gebruikersinterfaces te maken die uiteindelijk zorgen of het werk gedaan kan worden of niet. Bestaande bewezen interfaces zijn zeer kostbaar voor gebruikers in de praktijk, evenals goede, flexibele en accurate ondersteuning van de makers ervan.

Paul Torfs (WU):

Ik wil *ten zeerste en met de grootst mogelijke klem* pleiten tegen een mogelijke overdreven standaardisering. Het feit dat er twee of meerdere landelijke grondwatermodellen bestaan is *een zegen*. Wanneer er maar één model zou bestaan, zou dit noodzakelijkerwijs alleen maar de waarheid kunnen opleveren, en is kritiek haast onmogelijk. En het is alleen kritiek die ons vooruit kan helpen.

Een vergelijkbare vraag wat mij betreft: waarom twee wiskundige leermethoden ontwikkelen voor het basisonderwijs, waarom niet alle energie in één methode stoppen, die wordt er dan toch beter van. Nee natuurlijk, omdat gebrek aan vergelijkingen kritiek onmogelijk maakt, en dus uiteindelijk tot een slechter resultaat leidt.

Een dergelijke schadelijke drang tot uniformiteit bestond vroeger ook in het communistische oostblok : waarom energie verspillen door het ontwikkelen van meerdere typen auto's, als we nu alle energie stoppen in een type, moet dit noodzakelijk het beste worden. De gevolgen zijn bekend. Het is precies op die terreinen waar men daar meerdere ontwikkelingen naast mekaar gedoogde (bijvoorbeeld op het gebied van militaire vliegtuigen) dat men successen boekte.

Rien Pastoors en Karel Kovar (RIVM):

Voorzover daarmee een 'standaard voor rekenprogrammaschil' of 'modellerstandaard' wordt bedoeld, *luidt het antwoord 'neen'*. De bestaande rekenprogramma's hebben hun eigen bestaansrecht. De verbeteringen moeten worden gezocht in de sfeer van (a) beschikbaarstelling van basale gegevens, (b) beschikbaarstelling van een reeks kiesbare concepten voor de parameterisering aan de hand van de basale gegevens, en (c) beschikbaarstelling van tools (software) voor daadwerkelijke uitvoering van de parameterisering.

Heeft Matlab naar uw mening de potentie om op hydrologisch gebied uit te groeien tot een nieuwe rekenstandaard?

Willem-Jan Zaadnoordijk (IWACO):

Ik zie niet wat Matlab als voordeel zou hebben voor een rekenprogramma boven bijvoorbeeld Fortran of C++ met alle bibliotheken en sources die (vrijelijk) beschikbaar zijn. Voor de grafische mogelijkheden denk ik dat er teveel in Matlab geprogrammeerd moet worden om het een succesvol project te maken.

Omdat je met een dergelijk initiatief aan een GNU-achtige constructie denkt, lijkt het me voor de hand liggend om aan te sluiten bij bestaande GNU-programma's. Dus Octave in plaats van Matlab. Je zoekt een basis die niet te veel overhead geeft bij alle mogelijkheden die het heeft, en of deze programma's daarin een juiste balans hebben?

Matlab verdient volgens de reactie van diverse collega's wel meer aandacht in de hydrologische opleidingen en praktijk, maar dan denk men aan analytische toepassingen, het boek van Gijs Bruggeman.

Kick Hemker (VU-Amsterdam en MicroFEM):

Uit de inleiding wordt niet duidelijk hoe je je moet voorstellen dat er met één standaard sterk uiteenlopende problemen efficiënt worden berekend. Op z'n minst moeten er dan allerlei (tientallen of honderden?) Matlab-programma's worden gemaakt. Voor analytische modellen lijkt me dat wel mogelijk. Bij numerieke programma's neemt de gebruikers-interface om verschillende redenen een veel belangrijker plaats in. Wanneer voor alle Matlab-rekenprogramma's ook nog interfaces moeten worden gemaakt, dan wordt het een geweldig grote klus. Het is bekend dat met Matlab veel mogelijk is, maar kun je er ook volwaardige interfaces mee bouwen?

André Blonk (Tauw):

Ik ben zelf een fanatieke Fortran-gebruiker en ik kan alle problemen welke ik tegenkom snel in een Fortran-programma vertalen. Door het jarenlang bouwen en verzamelen van subroutines bestaat het meeste werk dan nog uit assemblage. Veel hydrologen missen naar mijn idee deze handigheid en ik geloof dan ook dat een programma als Matlab als rekenstandaard onmisbaar is. Het vak hydrologie kan namelijk niet goed worden uitgeoefend als de aangeleverde of geproduceerde data niet in de juiste vorm kunnen worden gekneed. Dit laatste geldt met name ook voor de presentatie van de resultaten.

Wim de Lange (RIZA):

Nee, het voldoet niet aan de vrije beschikbaarheid, het leidt tot te traag rekenen met grote modellen en het is en blijft maar een van de beschikbare tools voor de modelleur. Toevallig is nu Matlab in de picture, maar dat als norm stellen is net zo een misser als het stellen dat FORTRAN de programmeertaal voor eeuwig moet worden twintig jaar geleden. Ook

bestaande software moet niet opnieuw moeten worden ontwikkeld. Voor het probleem dat in de inleiding wordt aangesneden gaat het om de structuur, het raamwerk dat verschillende routines/modules verbindt, dat zorgt voor data-transmissie en -transformatie. Elke ontwikkelaar/modelleur moet zijn eigen en nieuwe tools kunnen gebruiken.

Paul Torfs (WU):

Matlab is een middel, een hulpmiddel, vaak een handig hulpmiddel. Wanneer je handig met Matlab bent, kun je er ontzettend handig mee werken. Wanneer je handiger met een spreadsheet bent, is Matlab een vloek. Een spreadsheet is tegenwoordig zo uitgebreid dat het ook een algemene programmeertaal, dus er is niks wat je met Matlab kunt doen wat je niet met een spreadsheet kunt doen, en omgekeerd.

Wat goed is aan Matlab (en overigens grotendeels ook aan spreadsheets) is, in vergelijking met algemene programmeertalen als C, C++, Pascal enz, is :

- er is veel vooraf 'ingebakken': bewerkingen met vectoren en matrices, maken van grafieken kost een enkele lijn, en veel lijnen in programmeertalen;
- er is onmiddellijke feedback, omdat het een interpreter is (dit in tegenstelling tot een compiler-linker omgeving), zodat de gevolgen van een kleine verandering *onmiddellijk* te zien zijn, wat het leerproces in het algemeen sterk versnelt.

Wat slecht is:

- omdat vector en matrix bewerkingen zo makkelijk zijn, bestaat de neiging om alles in die termen te vertalen, *ook* als dit tegen de geest van het probleem in is.
- een interpreter *blijft* trager en is *minder zuinig* met geheugen dan een compiler.
- men is te weinig kritisch ten opzichte van de toolboxes die aanwezig zijn. Zo is de Solver in Excel een prachtige optimizer, maar zeker niet de enige, en zeker niet de eerst aangewezene in een heleboel problemen.

Ook hier geldt de opmerking die bij het vorige punt gemaakt is: alleen door het gebruik van meerdere hulpmiddelen kunnen we kritisch tegenover deze hulpmiddelen blijven staan.

Een overdreven en verkeerd uniformiseringsbeleid werkt contraproductief.

Rien Pastoors en Karel Kovar (RIVM):

Waarschijnlijk neen. Ook al zou een modellerschil op basis van Matlab worden uitgewerkt, blijven de eerder genoemde punten a (basisdata) en b (concepten parameterisering) als probleem bestaan. Die oplossing voor die punten is onzes inziens belangrijker (nuttiger voor modelleur en 'zijn/haar' organisatie) dan de uniformering van datastromen door middel van Matlab, HDF, etc. Los daarvan staat of Matlab platform-onafhankelijk is (op PC maar ook op UNIX?), wat zonder meer een eis moet zijn.

Welke rol ziet u hierin voor uw bedrijf/organisatie?

Willem-Jan Zaadnoordijk (IWACO):

Een zelfde rol als we nu hebben: relevante ontwikkelingen beschikbaar maken in modelleergereedschap.

Kick Hemker (VU-Amsterdam en MicroFEM):

In eerste instantie alleen meedenken over het juiste antwoord op vraag 1 en vraag 2.

André Blonk (Tauw):

De beschikbaarheid van programma's is in het belang van elke hydroloog. Zeker gezien het eerder aangegeven ideaalbeeld en onze programmeerervaringen zullen wij een actieve bijdrage willen en kunnen leveren.

Wim de Lange (RIZA):

Zie ook antwoord op vraag 1. RIZA duwt al jaren in de richting van overdraagbaarheid van data tussen modellen en van databases van en naar modellen (zie AQUEST, Standaard raamwerkmodellen, Vloeiend modelleren in Waterbeheer, et cetera). In feite is NAGROM ook een poging om beschikbare regionale data die reeds in modelparameters vertaald zijn beschikbaar te stellen als omgeving voor detailmodellen. We zijn nu bezig om een gestandaardiseerde manier te ontwikkelen tussen de basisdata (in REGIS) en de parameterwaarden in het model. In feite is de topsysteem module MONA een algemeen bruikbare (dus ook voor MODFLOW, MicroFEM, TRIWACO, SIMGRO etc.) standaardisering van verwerking van topsysteem data in modelparameters. MONA is openbaar en voor iedereen beschikbaar maar de routines zijn nog niet echt toegankelijk; een ieder die daaraan wil bijdragen is welkom. Dit geldt ook al enige tijd voor NAGROM en hoogstwaarschijnlijk binnen afzienbare tijd voor alle RIZA-modellen.

Paul Torfs (WU):

Ik denk dat een universiteit *bij uitstek* een plaats is waar men meerdere modellen en hulpmiddelen moet blijven gebruiken, toetsen en testen, omdat het precies de taak van een universiteit is om kritisch met kennis en middelen om te springen (hoewel daar soms in bestuurslagen boven mij anders over wordt gedacht).

Rien Pastoors en Karel Kovar (RIVM):

Geen rol met betrekking tot Matlab. Wél actief meewerken aan de beschikbaarstelling van basisgegevens, en aan de ontwikkeling van concepten en tools voor de parametrisering van modelinvoer (ook door kalibratie).