

Apps kunnen het gebruik van analytische methoden nieuw leven inblazen

KEES VAN IMMERZEEL

Analytische formules worden in de hydrologische praktijk spaarzaam gebruikt. Een belangrijke reden hiervoor is dat de wiskundige formules intimerend zijn. Daarnaast vergt het vertalen van deze formules naar een spreadsheet of bijvoorbeeld een m-file in MATLAB veel kennis en doorzettingsvermogen. Het kan gemakkelijker, door analytische methoden in de vorm van apps toe te voegen aan de gereedschapskist van iedere hydroloog. Niet coderende hydrologen kunnen er dan direct mee aan de slag dankzij een gebruikersvriendelijke user interface. Coderende hydrologen kunnen zo de resultaten van hun werk met zoveel mogelijk collega's delen. Dit artikel beschrijft als voorbeeld een app gemaakt op basis van de Shiny package in R.

Essay

Inleiding

'Snelle oudjes gaan MATLAB' was de titel van een artikel van Kees Maas en Theo Olsthoorn in *Stromingen* 3 (1997). "Wat leven we toch in een heerlijke tijd!" verzuchtten de auteurs. Daarbij doelden zij op "het schitterende gereedschap" dat MATLAB kan zijn voor iedere hydroloog.

Ik moet toch constateren dat de meeste hydrologen dit schitterende gereedschap al meer dan twee decennia links laten liggen. Waarom is dat?

De prijs misschien? Inderdaad kan dat een rol spelen, want MATLAB is nog steeds prijzig. Dat werd ook al geconstateerd in het artikel van Kees Maas en Theo Olsthoorn. Maar intussen zijn er diverse gratis open source alternatieven voor MATLAB.

Waarschijnlijker is dat veel hydrologen geen affiniteit hebben met programmeren of scripten. Het vinden van een passende analytische formule voor een grondwaterprobleem en vervolgens het vertalen van dat probleem naar een MATLAB-code is dan een lastig te nemen hobbel.

Een website met analytische oplossingen zou dan ook bij voorkeur niet moeten linken naar scripts of spreadsheets maar naar apps.

Apps

Onder een app versta ik hier een applicatie die draait op een webserver. De webserver doet dus ook het rekenwerk. De applicatie is via een browser benaderbaar en werkt op diverse platformen (Windows, iOS) en apparaten (computer, mobiel, iPad).

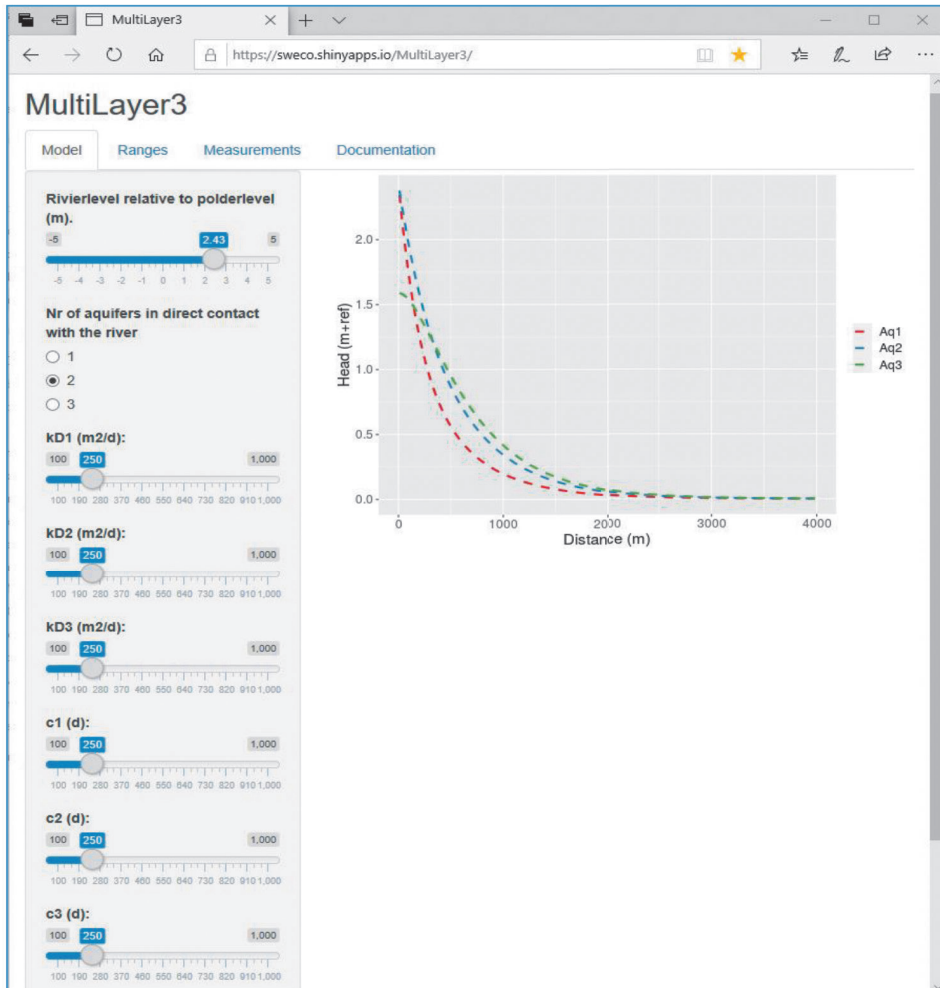
Spreadsheets, scripts en apps

Waarom is een app in het gebruik laagdrempeliger dan een spreadsheet of een script?

Een script vraagt kennis van de desbetreffende taal. Daarnaast moet de script taal in de juiste versie worden geïnstalleerd, inclusief de juiste versies van de bibliotheken waarvan het script afhankelijk is.

Spreadsheets zijn laagdrempeliger, maar daarin gaat het overzicht makkelijk verloren door een veelheid aan cel-verwijzingen en formules. Fouten zijn dan snel gemaakt en moeilijk op te sporen.

Laten we eerlijk zijn: in de praktijk zwerven scripts en spreadsheets her en der in verschillende versies op het netwerk of op je computer. Dat bevordert het gebruik ervan niet.



Afbeelding 1 Screenshot van de app "MultiLayer3".

Bovenstaande nadelen gelden niet voor een app. Via een webadres krijgt iedereen, waar je ook bent en welk apparaat je ook gebruikt, toegang tot de laatste versie van de app.

Als voorbeeld wordt in hier de app "MultiLayer3" gebruikt. Dit is de link naar de app: <https://sweco.shinyapps.io/MultiLayer3/>. Afbeelding 1 toont een screenshot van deze app.

Afbeelding 1 laat zien dat de parameters die worden gebruikt in de app door middel van een "schuif" in te stellen zijn binnen vooraf begrensde waarden. De app is reactief; daarmee wordt in dit geval bedoeld dat het effect van een verstelling van een schuif op het berekeningsresultaat (de grafiek met berekende stijghoogten) direct zichtbaar is. Daardoor ontstaat "gevoel" voor het relatieve belang van de verschillende parameters voor het berekeningsresultaat. De techniek hierachter ("reactive programming") is verwerkt in de "Shiny" package van R (<https://shiny.rstudio.com/>). Deze site toont overigens meer voorbeelden van de mogelijkheden die de techniek van "reactive programming" biedt (<https://shiny.rstudio.com/gallery/>).

Voor R zijn een meerdere *packages* beschikbaar die de vertaling van analytische formules naar programmacode vergemakkelijken. Afbeelding 2 toont ter illustratie een stukje code in MATLAB en de corresponderende code in R. Het voorbeeld laat zien dat de R-code nagenoeg even compact is als de code in MATLAB.

```
function A = sysmat(kD,c)
% De functie sysmat(kD,c) stelt de systeem-matrix samen.
% kD en c zijn kolomvectoren!
n = length(kD);
a = 1./(kD.*c);
b = 1./(kD.*[c(2:n);inf]);
A = diag(a+b)-diag(a(2:n),-1)-diag(b(1:n-1),1);
```

```
# Create system matrix from kD and c values
sysmat <- function(kD, c) {
  n <- length(kD)
  a <- 1 / (kD * c)
  b <- 1 / (kD * c(c[2:n], Inf))
  di_a <- rbind(0, cbind(diag(a[2:n]), 0))
  di_b <- cbind(0, rbind(diag(b[1:(n - 1)]), 0))
  A <- diag(a + b) - di_a - di_b
  return(A)
}
```

Afbeelding 2 Voorbeeld van code in MATLAB (boven) en de corresponderende code in R (onder).

Aan de app "MultiLayer3" is tenslotte een pagina toegevoegd ("Documentation") met daarin verwijzingen naar achtergronddocumenten.

Open source

Analytische methoden kunnen alleen nieuw leven in worden geblazen als deze worden vertaald naar apps. Gezien de veelheid van methoden en de hoeveelheid werk die hiermee is gemoeid, is het niet realistisch te veronderstellen dat dit op korte termijn gebeurt.

Dat hoeft ook niet. Het maken van apps kan een gezamenlijke inspanning zijn van de hydrologische wereld. Daarbij helpt het als we zoveel mogelijk bij elkaar kunnen "spieken". Broncode moet daartoe worden gedeeld. GitHub is daarvoor de uitgelezen plek. De broncode van de app "MultiLayer3" vindt u hier: <https://github.com/KeesVanImmerzeel/MultiLayer3>.

Summary Apps can revitalize the use of analytical methods.

In the hydrological practice, analytical formulas are rarely used. An important reason for this is that mathematical formulas are intimidating. Furthermore, the conversion of these formulas to a spreadsheet or for example an m-file in MATLAB, requires a lot of knowledge and perseverance. Adding apps to the toolbox of every hydrologist can revitalize the use of analytical methods. Hydrologists that are skilled in scripting can now share their work with non-scripting colleagues by creating a user friendly app accessible via a web address. As an example, this article describes a reactive app based on the "Shiny" package of the R-language (<https://github.com/KeesVanImmerzeel/MultiLayer3>).

Auteurs

KEES VAN IMMERZEEL

Sweco

kees.vanimmerzeel@sweco.nl