

NOTITIE

Onderwerp Uittesten allocatie
Project NHI koppelen Bommelerwaard met hoofdwatersysteem
Opdrachtgever Deltares
Projectcode 139538
Status Concept 01
Datum 26 juni 2024
Referentie 139538/24-009.218
Auteur(s) Mevrouw dr. ir. J. van Houwelingen

Gecontroleerd door De heer drs. W.M. Zijderwijk
Goedgekeurd door De heer drs. W.M. Zijderwijk
Paraaf



Bijlage(n) -

Aan Deltares Timo Kroon
Edwin Snippen

Kopie

1 INLEIDING

Achtergrond

Binnen het TKI155 oppervlaktewatermodule project wordt een oppervlaktewatermodule ontwikkeld voor een vereenvoudigde, gelumpte benadering voor het Nederlandse oppervlaktewatersysteem. Het modelconcept zal zowel op regionaal als landelijk schaalniveau in samenhang kunnen worden toegepast met MODFLOW en MetaSWAP en desgewenst D-Hydro en kunnen dienen als vervanger van bestaande software (MOZART/DM, SIMRES). De software zal tijdens de ontwikkeling in verschillende gebieden (cases) toegepast worden. Deze toepassingen hebben tot doel om de ontwikkelde software en pre- en postprocessing tooling te testen.

Witteveen+Bos werkt als partner mee aan deze TKI en voert momenteel de case Bommelerwaard uit waarin een regionaal gelumpte (Ribasim) model gekoppeld met Modflow 6 wordt opgezet. Deltares heeft Witteveen+Bos gevraagd de Bommelerwaard te koppelen aan het Landelijk Hydrologisch Model (LHM) en deze te testen bij een lage rivierafvoersituatie waarbij de verdringingsreeks geldt. Uit de testberekeningen moet blijken in hoeverre het gekoppelde model tijdens een lage rivierafvoer (aanvoersituatie) de watervraag en toewijzing vanuit regionale model goed wordt doorgegeven.

Doel

Het doel van deze studie is het uittesten van de allocatieprincipes die zijn ingebouwd in de Ribasim software.

Leeswijzer

In voorliggende notitie zijn de methodiek en resultaten (+problemen) gepresenteerd van de uitgevoerde testen met betrekking tot het thema allocatie. Voor achtergrondinformatie rondom de gebiedsbeschrijving en de ideeën voor het conceptuele model wordt verwezen naar de uitgangspuntennotitie [ref. 1]. **De gecreëerde schematisaties zijn op enkele punten versimpeld ten opzichte van de originele ideeën in de uitgangspuntennotitie. Daarnaast beperken de testcases zich tot het testen van tekortsituaties waarbij allocatie van toepassing is.** Dit is gedaan omdat de focus in deze fase van de softwareontwikkeling meer ligt op het uittesten van de principes die zijn ingebouwd in de software en niet zo zeer om het opzetten van een schematisatie waarmee de werkelijke situatie goed kan worden nagebootst.

2 ALLOCATIE

Met behulp van de allocatiefunctie kunnen watervragen (in tijden van tekort) worden afgehandeld in volgorde van prioriteit. Uiteindelijk zullen er in Ribasim drie verschillende allocatieknopen (gebruikers) toepasbaar zijn, namelijk:

- UserDemand:
Poogt om zijn volledige vraag te onttrekken aan het gekoppelde basin totdat het basin leeg is of het minimale niveau (waarvoor ingesteld) is bereikt. Deze knoop kan worden toegepast om bijvoorbeeld onttrekkingen (industrie, beregening, etc.) op te drukken.
- LevelDemand:
Kent een minimaal en maximaal waterstand toe aan het gekoppelde basin. Er ontstaat een watervraag met gegeven prioriteit indien het water zakt onder de toegekende minimale waterstand. Boven de maximale waterstand gaat de knoop als een bron fungeren voor de watervragen binnen het subnetwerk. Indien aan peilhandhaving een prioriteit kan worden toegekend ten opzichte van andere watervragen, dan kan deze knoop worden toegepast. Het enige wat een LevelDemand dus in feite doet is invloed uitoefenen op andere watervragers, de LevelDemand kan niet worden gebruikt als sturing.
- FlowDemand:
Kent een (non-consuming) afvoer toe aan een verbindingsknoop (zoals een pomp of een tabulated rating curve) met gegeven prioriteit. Hiermee zou bijvoorbeeld een vraag voor doorspoeling kunnen worden opgedrukt.

Daarnaast kunnen gebieden worden aangemerkt vanuit waar de watervragen moeten worden voorzien (de bronnen).

Sinds Ribasim versie 2024.4.0 zijn de UserDemand en LevelDemand beschikbaar.

Door toepassing van de allocatiefunctie wordt per allocatietijdstap bekeken hoeveel watervraag er vanuit de verschillende gebruikers binnen het subnetwerk van een bron is. Ook wordt er gekeken hoeveel water er beschikbaar is via de bron. Indien het aanbod lager is dan de totale watervraag treedt allocatie in werking, waarbij de watervraag naar prioriteit wordt afgewerkt via een optimalisatiealgoritme. Afhankelijk van lokale waterbeschikbaarheid, prioriteit etc. kan de hele of alleen een deel van de gealloceerde watervraag worden gerealiseerd.

3 METHODIEK

Voor het uitvoeren van de testen in relatie tot allocatie is een basismodel opgesteld van de Bommelerwaard gekoppeld aan het hoofdwatersysteem (Maas) in Ribasim versie 2024.4.0. Het model is handmatig opgesteld in Qgis met behulp van de Ribasim plugin. Er is een statisch model opgesteld om deze eerste testen met allocatie overzichtelijk te houden.

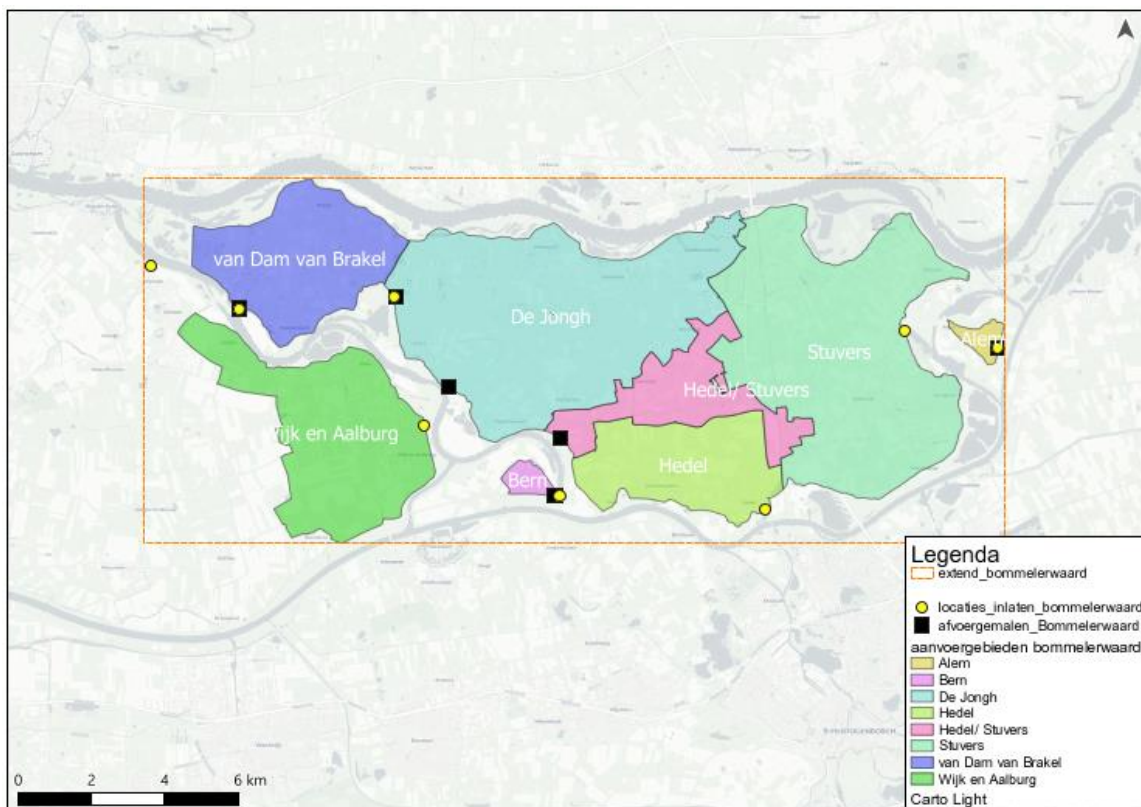
Schematisatie Maas

De aanvoergebieden van de Bommelerwaard zijn alleen gekoppeld aan de Maas, omdat alle inlaten aan de Maas liggen. Het overige buiten water is niet meegenomen. De schematisatie voor de Maas is aangeleverd door Hydrologic. Om te beginnen is dit model eerst omgezet naar de juiste Ribasim versie. Sinds de oplevering van dit Maasmodel zijn er enkele breaking changes doorgevoerd in de source code van Ribasim. De aanvoer via de flow boundary bovenstreams is ingesteld op 25 m³/s. Benedenstreams is de level boundary ingesteld op 0,5 m +NAP. Dit komt overeen met de afvoer ten tijde van een droge situatie.

Schematisatie Bommelerwaard

In afbeelding 3.1 zijn de aanvoergebieden aangegeven met in- en uitlaatvoorzieningen. De aanvoergebieden Wijk en Aalburg, Bern en Alem zijn niet opgenomen in de schematisatie. Alle aanvoergebieden lozen water op de Maas en/of laten water vanuit de Maas in tijden van wateroverschot/tekort. De aanvoergebieden Hedel en Stuvers voeren af op Hedel/Stuvers, welke afvoert richting de Maas. Voor de Bommelerwaard is een schematisatie opgesteld op aanvoergebiedsniveau. Dit wil zeggen dat elk aanvoergebied is geschematiseerd door middel van één basinknoop. In afbeelding 3.2 is de basisschematisatie van het model weergegeven.

Afbeelding 3.1 Aan- en afvoergebieden van de Bommelerwaard met aanduiding van de inlaatvoorzieningen en afvoergemalen



Het profiel van elk basin is opgesteld op basis van twee niveaus, waarbij is aangenomen dat het bakje 1 m diep is (streefpeil en de bodem op streefpeil -1 m). Voor de oppervlakte van de basins is aangenomen dat 5 % van het totale oppervlak van het aanvoergebied bestaat uit oppervlaktewater (bij streefpeil). Op de bodem is de oppervlakte de helft van het oppervlak bij streefpeil. Het streefpeil in het basin komt overeen met het zomerpeil van het benedenstroomse peilgebied.

Uit een andere pilot kwam naar voren dat de sturing soms niet wil werken als het setpoint van de sturing hetzelfde is als het initiële peil. De initiële conditie van de basins is daarom voor de zekerheid gekozen op het zomerpeil van het benedenstroomse peilgebied + 1 cm.

Het model is opgesteld voor een tekort situatie. Dit is nagebootst door alleen een verdampingspost

(3 mm/dag = $3.5 \cdot 10^{-8}$ m/s) toe te voegen aan het model. In werkelijkheid zal de tekort situatie ook mede worden veroorzaakt door infiltratie. De aanvoergebieden Hedel, Stuffers en Hedel/Stuffers zijn met elkaar verbonden via een tabulated rating curve, waarbij er niet/nauwelijks afvoer is tot het streefpeil. De aanvoergebieden zijn in de praktijk verbonden via stuwtjes. De tabulated rating curves zorgen ervoor dat de aanvoergebieden Hedel en Stuffers niet 'leegstromen' richting Hedel/Stuffers, waardoor de aanvoerpomp continue aan moet staan.

Koppeling Maas en Bommelerwaard (toevoegen sturing)

De Maas en Bommelerwaard wisselen water uit via inlaten en afvoergemalen. In het model zijn deze toegevoegd als pomp met discrete control. De pompen worden gestuurd op basis van twee levels (5 cm onder streefpeil en 5 cm boven streefpeil). Een aanvoergemaal slaat aan als het water zakt onder streefpeil - 5 cm en slaat af als deze boven streefpeil + 5 cm komt. Een afvoergemaal slaat aan als het water stijgt tot boven streefpeil + 5 cm en slaat af als deze zakt onder streefpeil - 5 cm. De capaciteit van de pompen komt overeen met de capaciteit in de praktijk.

Issue I: Sturing op drie levels (min, setpoint, max)

In eerste instantie was sturing op drie niveaus toegevoegd aan het model (waarbij setpoint = streefpeil). Dit blijkt vrij snel convergentie issues op te leveren, de tijdstap wordt zeer klein. Met de tweetrapssturing lijkt dit probleem verholpen.

Toevoegen van allocatie

Tot slot is aan elke aanvoergebied en aan de bovenstroomse- en benedenstroomse zijde van de Maas een allocatieknoop toegevoegd waarmee watervragen kunnen worden opgedrukt aan het model. De bovenstroomse aanvoer via de FlowBoundary en de UserDemand aan de benedenstroomse zijde van de Maas geven de gelegenheid om een gecontroleerde tekortsituatie te maken voor het tussenliggende gebied. De benedenstroomse LevelBoundary drukt alleen een waterstand op, maar fixeert niet de afvoer. Door deze UserDemand (#58) de hoogste prioriteit te geven, kan de afvoer benedenstrooms van de aanvoergebieden worden 'gefixeerd'.

Issue II

Bij toevoeging van allocatie moet het netwerk nog opeenvolgende nummering hebben. Dit probleem was voorheen ook een probleem in de andere modules van Ribasim, maar de oplossing is nog niet doorgevoerd voor het allocatie principe.

Dit is gemeld bij Deltares en zal in een toekomstige versie worden verholpen.

Issue III

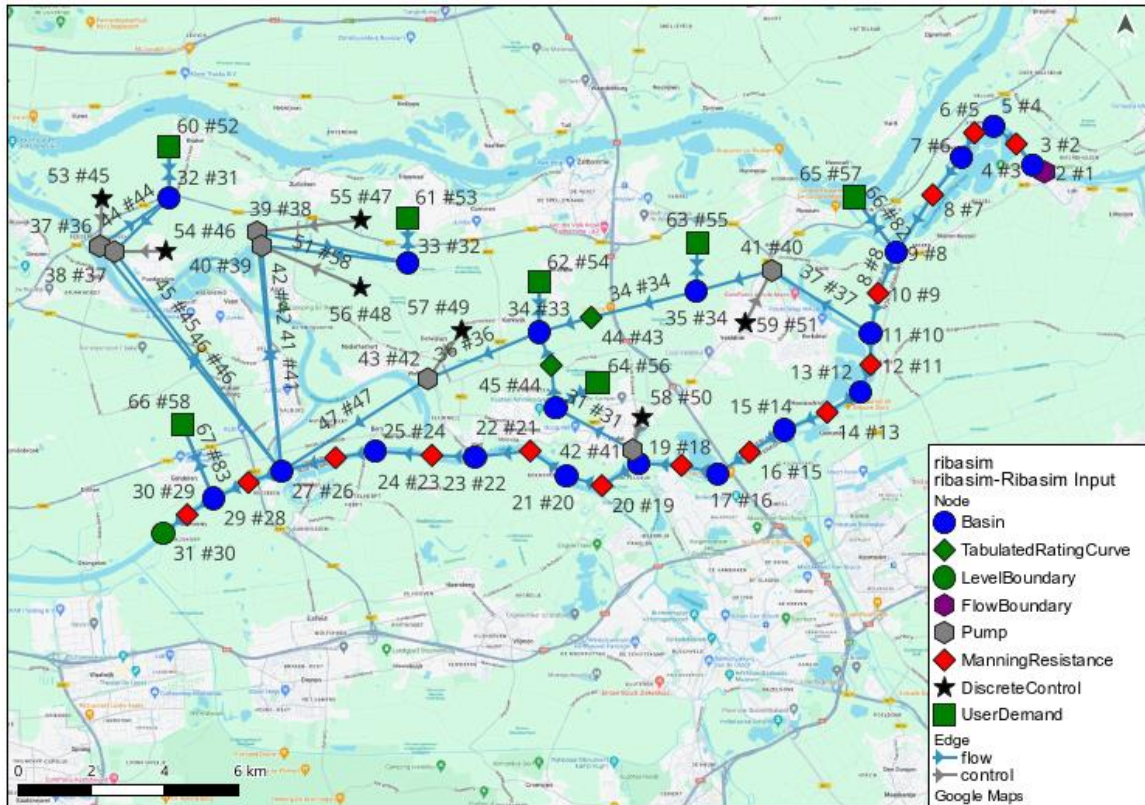
Bij toevoeging van allocatie moeten alle knopen in het netwerk worden toegekend aan een subnetwerk voor de verdeling van de watervraag. In de edges daarentegen moet alleen het subnetwerk-ID worden toegekend aan de source edge. Deze werkwijze is nog niet heel duidelijk beschreven in de documentatie.

Dit is bekend bij Deltares en wordt verholpen in een nieuwe versie van de documentatie.

Solver settings

De simulaties worden gedraaid met het QNDF algoritme (incl. adaptieve tijdstappen). De resultaten worden met een interval van 10 min. opgeslagen. Het update interval voor allocatie is ingesteld op de default waarde van 1 dag. Dit betekent dat per dag wordt bekeken hoeveel water er gevraagd wordt en hoeveel water er beschikbaar is om te verdelen, waarop de verdeling wordt aangepast.

Afbeelding 3.2 Basisschematisatie van de Bommelerwaard gekoppeld aan het hoofdwatersysteem



Dit basismodel wordt aangepast in de verschillende testen door bijvoorbeeld de prioriteiten en watervragen van de allocatieknopen aan te passen.

4 RESULTATEN

In tabel 4.1 is een totaal overzicht gegeven van de uitgevoerde testen. In de paragrafen hieronder wordt per test het resultaat gepresenteerd. Ook zijn de casenummers toegevoegd aan de tabel, waarmee de settings van de testcase kunnen worden teruggevonden in het logboek.

Tabel 4.1 Overzicht van de uitgevoerde testcases in Ribasim versie 2024.4.0.

Test #	Kenmerk	Case	Vraag	Leerpunt
0	referentie	0	werkt het model zonder allocatie?	runt het model zit sturing correct in het model
1	allocatie (in)actief	1,2	staat allocatie aan/uit?	modelcomponent allocatie is aan/uit te zetten
2	tekort	3	wat doet allocatie bij een tekort?	wordt de gebruiker gekort indien de watervraag hoger is dan de aanvoer?
3a	vraag bovenstrooms van sturing	16,17,18	wat doet allocatie bij een vraag bovenstrooms van een gestuurd object?	wordt de watervraag bovenstrooms van een gestuurd object herkend? Kan de vraag worden gerealiseerd vanuit de source?

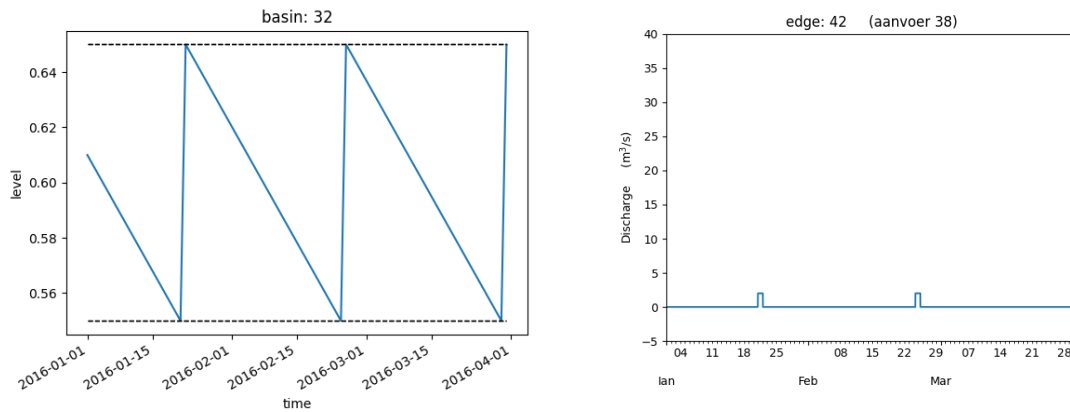
Test #	Kenmerk	Case	Vraag	Leerpunt
3b	min_level	7 t/m 15	wat is de invloed van de setting min_level?	stopt de abstractie door de watervrager als min_level wordt bereikt?
4a	prioriteit gelijk	5,6	wat doet allocatie bij een tekort, wanneer prioriteit gelijk is? (beide users in het hoofwatersysteem)	wordt het water eerlijk verdeeld over de watervragers?
4b	prioriteit gelijk	21,22	wat doet allocatie bij een tekort, wanneer prioriteit gelijk is? (1 user in hoofwatersysteem, 1 user in aanvoergebied)	wordt het water eerlijk verdeeld over de watervragers?
4c	prioriteit gelijk	27,28	wat doet allocatie bij een tekort, wanneer prioriteit gelijk is? (users in verschillende aanvoergebieden)	wordt het water eerlijk verdeeld over de watervragers?
5a	prioriteit verschillend	19,20	wat doet allocatie bij een tekort, wanneer prioriteit niet gelijk is? (beide users in het hoofwatersysteem)	wordt de watervrager met de hoogste prioriteit eerst gerealiseerd?
5b	prioriteit verschillend	23,24	wat doet allocatie bij een tekort, wanneer prioriteit niet gelijk is? (1 user in hoofwatersysteem, 1 user bovenstrooms van sturing)	wordt de watervrager met de hoogste prioriteit eerst gerealiseerd?
5c	prioriteit verschillend	29,30	wat doet allocatie bij een tekort, wanneer prioriteit gelijk is? (users in verschillende aanvoergebieden)	wordt de watervrager met de hoogste prioriteit eerst gerealiseerd?
6	LevelDemand tekort-overschot	(runs#2) 5,6,7	worden de vragen en overschotten van de leveldemand gezien?	worden de vragen en overschotten van de leveldemand gezien en wordt de vraag gerealiseerd?
7	LevelDemand i.c.m. UserDemand	(runs#2) 4	wat doet allocatie als een LevelDemand een overschot heeft en er een UserDemand aanwezig is?	wordt het overschot gebruikt voor de (lokale) vraag?

Test 0: Referentie

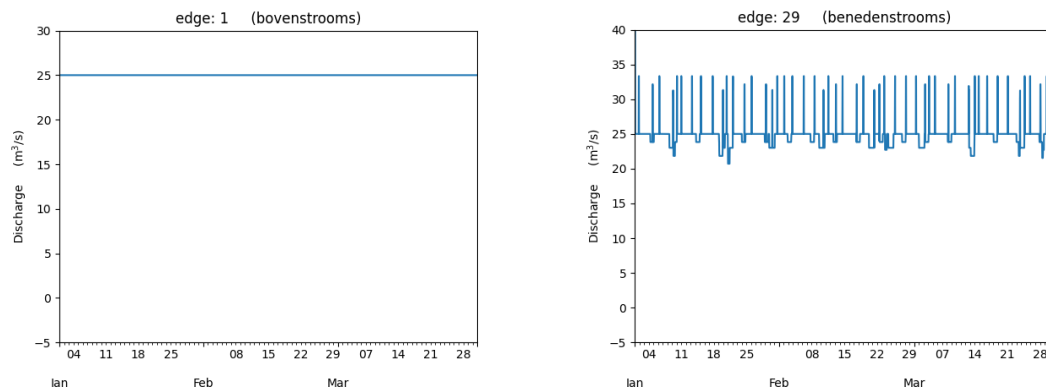
Deze test is uitgevoerd om te checken of het model draait zonder allocatie en of te checken of de opgelegde sturing naar behoren werkt. De UserDemand knopen en source zitten in de schematisatie, maar allocatie is uitgeschakeld in de .toml file.

In afbeelding 4.1 is ter illustratie de waterstand en de aanvoer via de pomp van één aanvoergebied weergegeven. In afbeelding 4.2 is de bovenstroomse aanvoer via de FlowBoundary en de Maas afvoer benedenstrooms van de aanvoergebieden weergegeven. Door verdamping/infiltratie daalt de waterstand in de aanvoergebieden. De pompen slaan aan als de waterstand onder streefpeil - 5 cm zakt. De pompen gaat weer uit als de waterstand boven streefpeil + 5 cm is. De benedenstroomse afvoer fluctueert een klein beetje door het aan- en afslaan van de pompen.

Afbeelding 4.1 Ter illustratie de waterstand van basin 32 en de aanvoer vanuit de Maas (edge 38) via het inlaatgemaal (pomp 37)



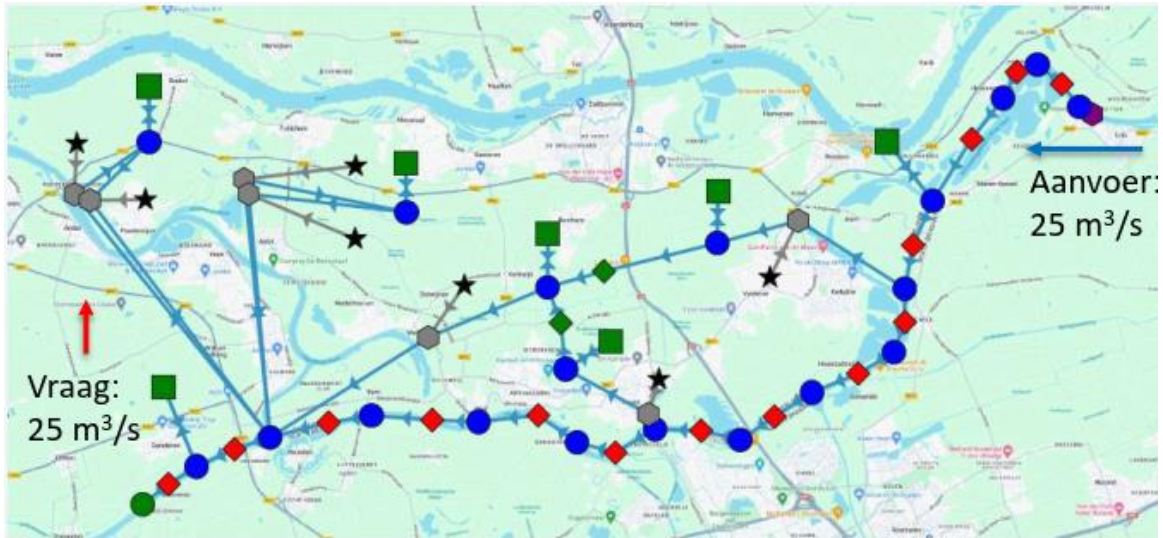
Afbeelding 4.2 Bovenstroomse aanvoer via FlowBoundary en Maas afvoer benedenstrooms van aanvoergebieden



Test 1: Allocatie (in)actief

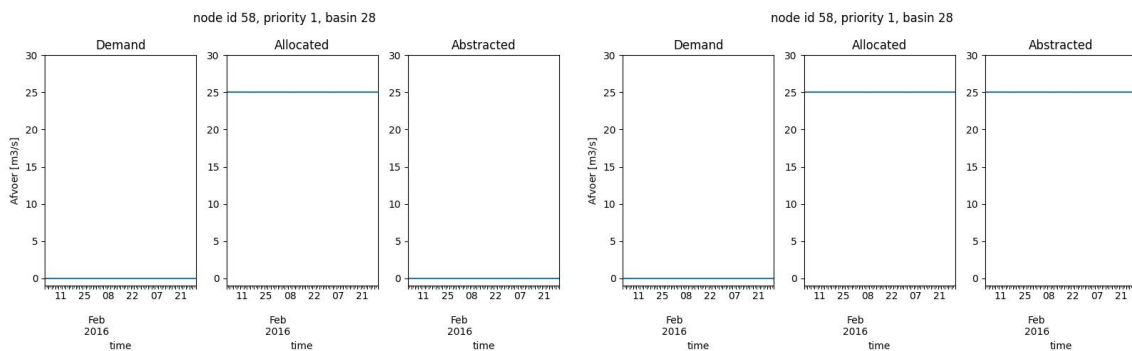
In deze test is allocatie geactiveerd in de .toml file. Bij de UserDemand knopen is een vraag opgedrukt, maar alle knopen zijn op inactief gezet (a). Daarnaast is een som uitgevoerd waarbij de benedenstroomse UserDemand op de Maas (#58) is geactiveerd met een watervraag van 25 m³/s met prioriteit 1 (b), zie afbeelding 4.3. Deze UserDemand zal in de testcases hierna vaak worden gebruikt om een gecontroleerde tekortsituatie te genereren.

Afbeelding 4.3 Opzet testcase 1

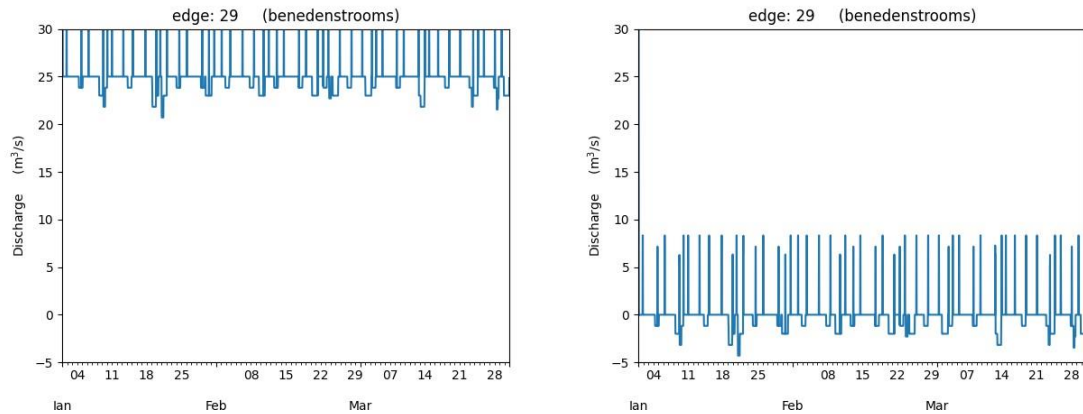


In afbeelding 4.4 is een overzicht gegeven van de allocatieresultaten. In afbeelding 4.5 is de Maasafvoer benedenstrooms van de aanvoergebieden gepresenteerd. Door activatie van de UserDemand neemt de afvoer benedenstrooms met $25 \text{ m}^3/\text{s}$ af. Dit betekent dat de vraag inderdaad wordt bediend. Het fluctuerende patroon door het in- en uitschakelen van de pompen blijft exact hetzelfde. Kijkend naar de allocatieresultaten lijkt de administratie in de .arrow file niet helemaal op orde. In beide gevallen zou de Demand op $25 \text{ m}^3/\text{s}$ moeten staan, deze staan op 0. Bij de inactieve UserDemand (a) zouden Allocated en Abstracted op 0 moeten staan. Bij activatie van de UserDemand (b) zouden daarnaast Allocated en Abstracted beide op $25 \text{ m}^3/\text{s}$ uit moeten komen.

Afbeelding 4.4 Overzicht van de allocatieresultaten voor de UserDemand benedenstrooms op de Maas (#58). Links: allocatieknopen inactief (a), rechts: allocatie #58 geactiveerd (b)



Afbeelding 4.5 Maasafvoer benedenstrooms van aanvoergebieden (links: allocatie inactief (a), rechts: allocatie (#58) geactiveerd (b))



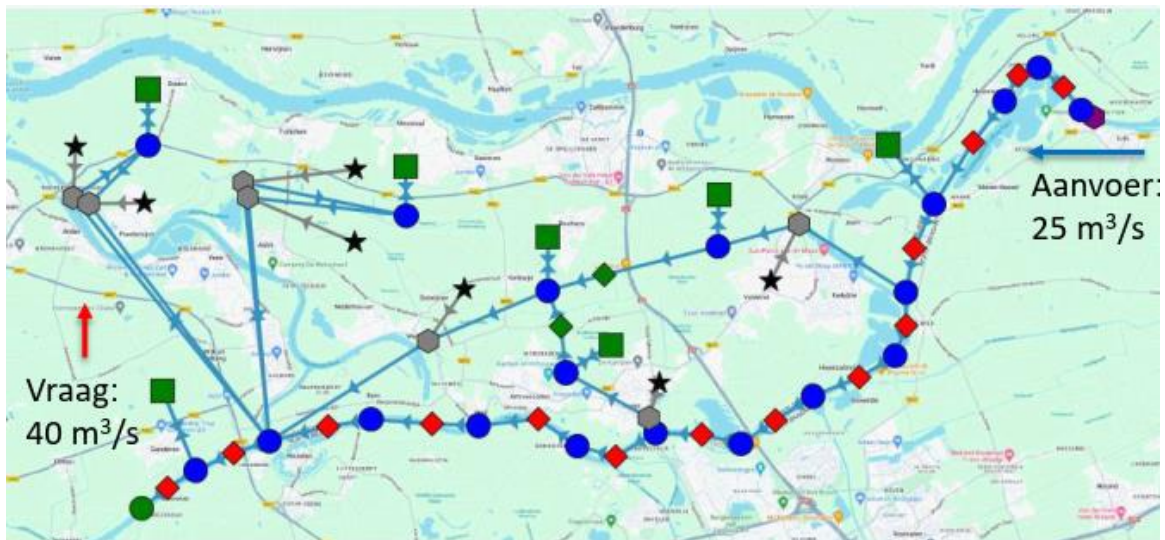
Issue IV: Administratie .arrow file

De administratie van de allocatieresultaten in de .arrow file lijkt niet helemaal te kloppen.

Test 2: Tekort

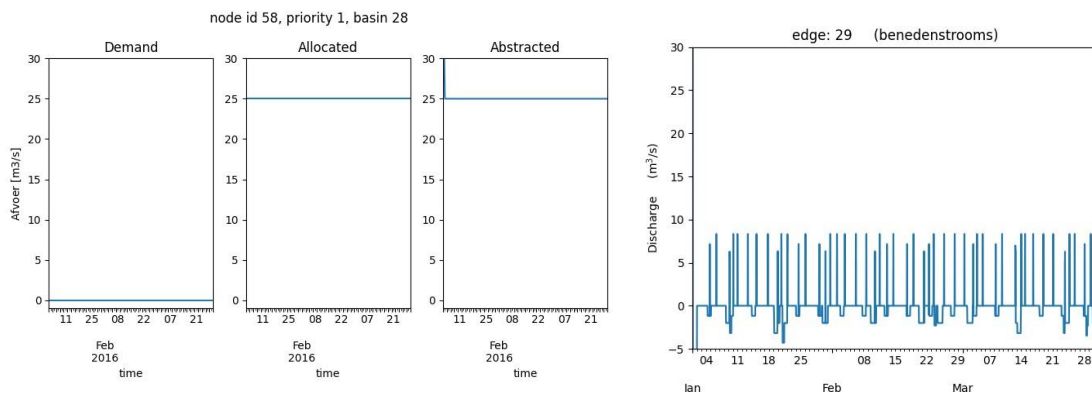
Deze test is vergelijkbaar opgebouwd als de vorige test, alleen is de vraag nu verhoogd naar 40 m³/s (zie afbeelding 4.6), zodat de vraag hoger is dan de aanvoer.

Afbeelding 4.6 Opzet testcase 2



De vraag kan inderdaad niet worden bediend, er wordt maar 25 m³/s gerealiseerd (zie afbeelding 4.7). De benedenstroomse afvoer is gelijk aan testcase 1.

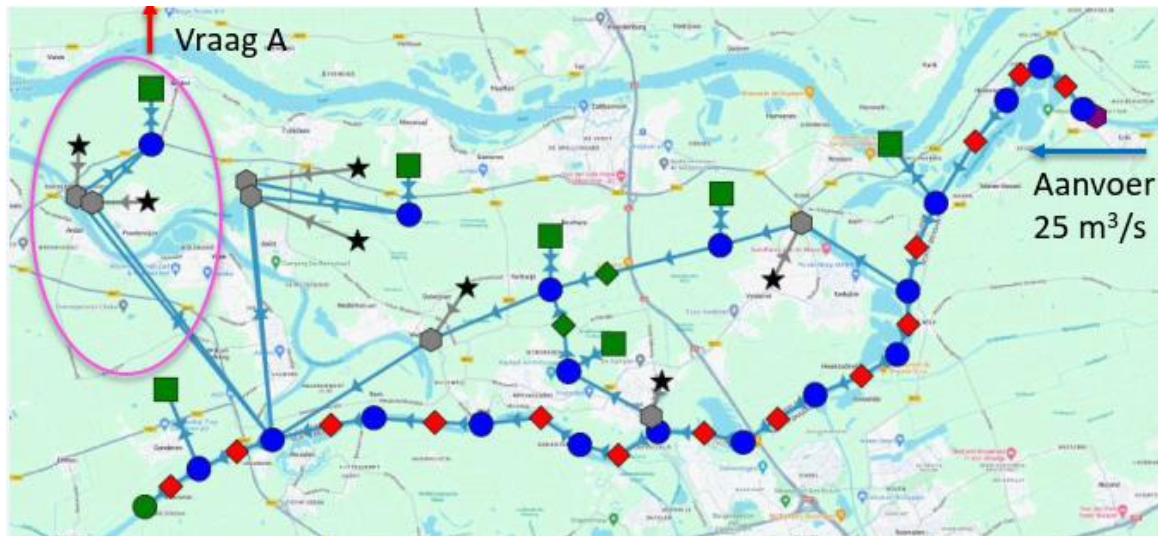
Afbeelding 4.7 Resultaten testcase 2



Test 3: Invloed min_level

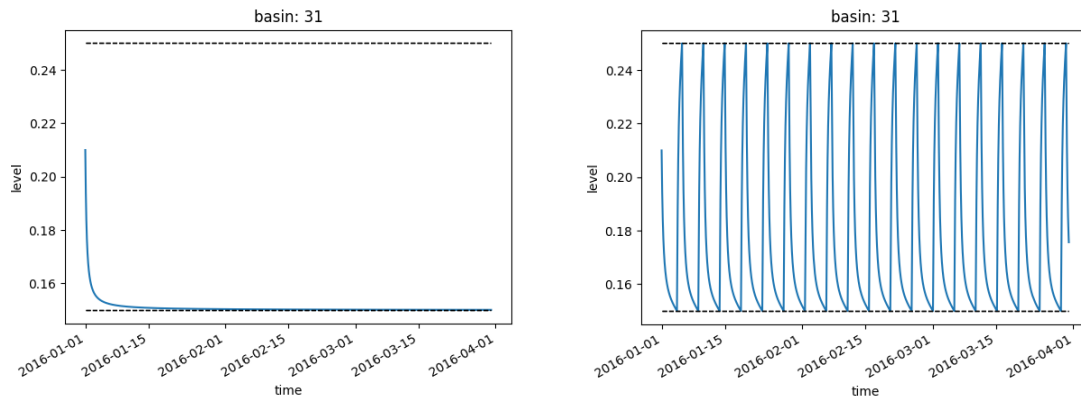
In deze testcase wordt de invloed de min_level instelling voor allocatie op de uitvoering van het allocatieprincipe en de sturing van de pomp bekeken. Dit is gedaan door de verdamping (waterverbruiker) aan- en uit te zetten. De limiterende factor voor de aanvoer van water wordt gecreëerd door de pompcapaciteit. Voor het betreffende aanvoergebied is de pompcapaciteit 2,29 m³/s. De vraag (A) is ingesteld op 2 m³/s.

Afbeelding 4.8 Opzet testcase 3 met een watervraag in een aanvoergebied bovenstrooms van een gestuurd object (pomp)

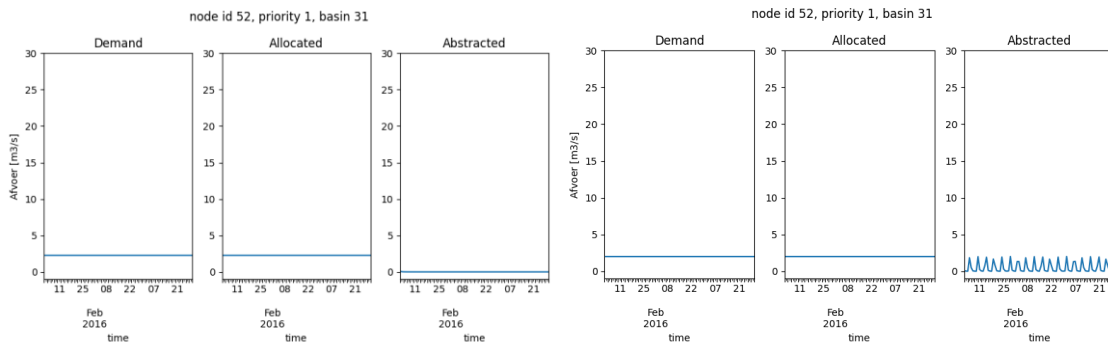


In afbeelding 4.9 en afbeelding 4.10 zijn de resultaten weergegeven van deze testcase. Wanneer de UserDemand de enige verbruiker is van water en het min_level is ingesteld op streefpeil -5 cm (zelfde aanslagpeil pomp) dan stopt de abstractie op min_level. De pomp wordt echter nog niet getriggerd om water aan te voeren, aangezien het water niet onder streefpeil -5 cm zakt. Het waterniveau blijft dus gelijk aan min_level. Bij inschakeling van verdamping wordt de pomp wel getriggerd en fluctueert de waterstand tussen streefpeil -5 cm en + 5 cm.

Afbeelding 4.9 Waterstand in het aanvoergebied met activatie van min_level op streefpeil - 5 cm. Links: verdamping is uitgeschakeld, rechts: verdamping ingeschakeld



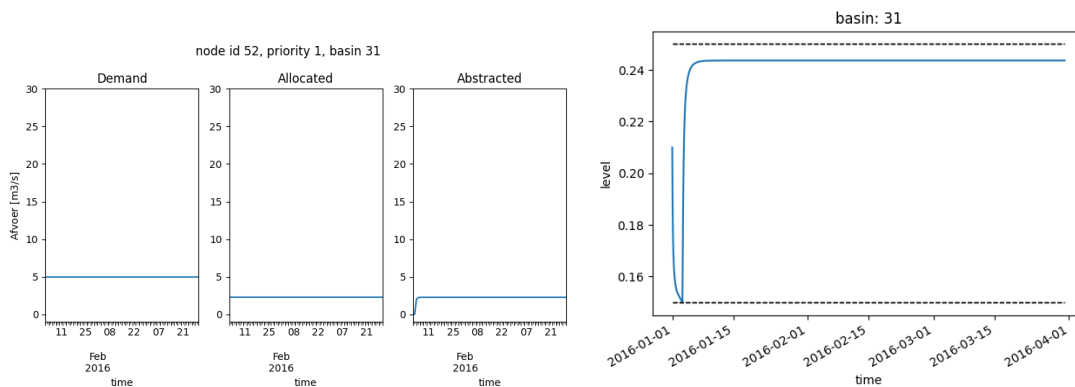
Afbeelding 4.10 Allocatieresultaten in het aanvoergebied met activatie van min_level op streefpeil - 5 cm. Links: verdamping uitgeschakeld, rechts: verdamping ingeschakeld



Wat verder opvalt is dat de gerealiseerde abstractie door de UserDemand fluctueert. Er kan niet continu aan de vraag worden voldaan, ondanks dat de opgedrukte vraag kleiner is dan de capaciteit van het gemaal. Dit komt, omdat deze elke keer wordt afgebroken op het moment dat min_level wordt bereikt.

Als de vraag groter is dan de capaciteit van de pomp, dan kan een deel van de vraag worden gerealiseerd doordat de pomp continue aan blijft staan. Er ontstaat een evenwichtssituatie waarbij de waterstand stabiel wordt (zie afbeelding 4.11).

Afbeelding 4.11 Resultaten testcase 3, wanneer watervraag groter is dan de pompcapaciteit



Om de setting `min_level` in de volgende testen geen beperkende factor te laten zijn, zal deze in het vervolg worden ingesteld op het bodemniveau van het basin van het aanvoergebied. Zodoende wordt de sturing van de pomp niet gedwarsboomd.

Test 4: prioriteit gelijk

De verdeling van watervraag over verschillende gebruikers met dezelfde prioriteit is getest aan de hand van een drie verschillende configuraties (zie afbeelding 4.12 tot en met afbeelding 4.14):

a. twee watervragers in het hoofdwatersysteem

$A + B > \text{aanvoer}$, de prioriteit van A en B is 1. Twee varianten doorgerekend, waarbij de grootte van de watervraag is verwisseld tussen de gebruikers ($A > B$ en $B > A$);

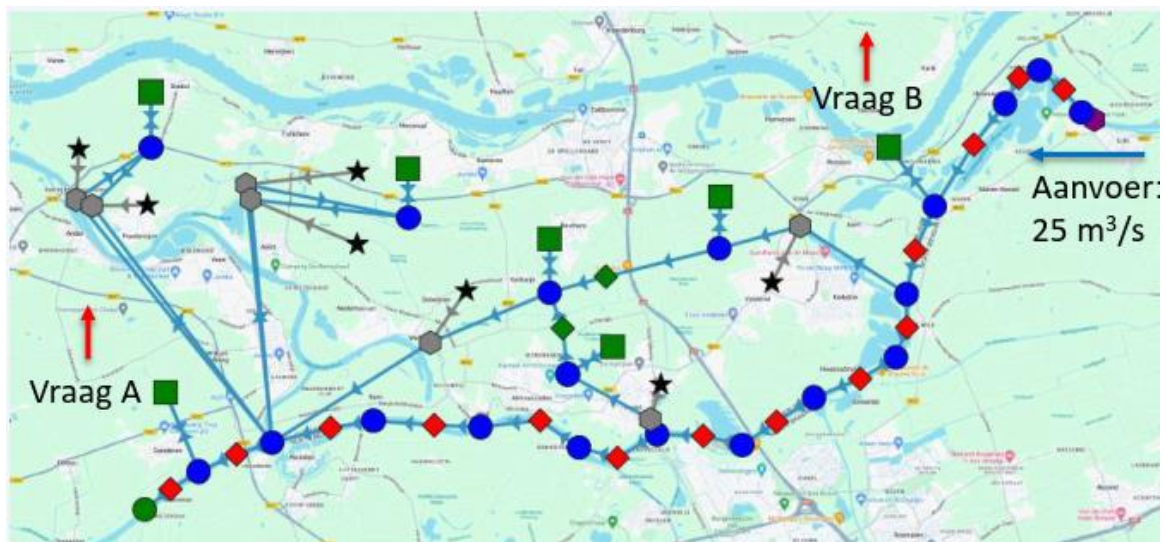
b. één watervrager in het hoofdwatersysteem (B) en één watervrager in een aanvoergebied (C)

$B + C > \text{aanvoer} - A$, de prioriteit van B en C is 2. De watervraag van B en C is $2 \text{ m}^3/\text{s}$. A (prioriteit 1) wordt gebruikt om een tekortsituatie te creëren van respectievelijk 2 en $3 \text{ m}^3/\text{s}$. Verdamping is uitgeschakeld voor een eerlijke vergelijking;

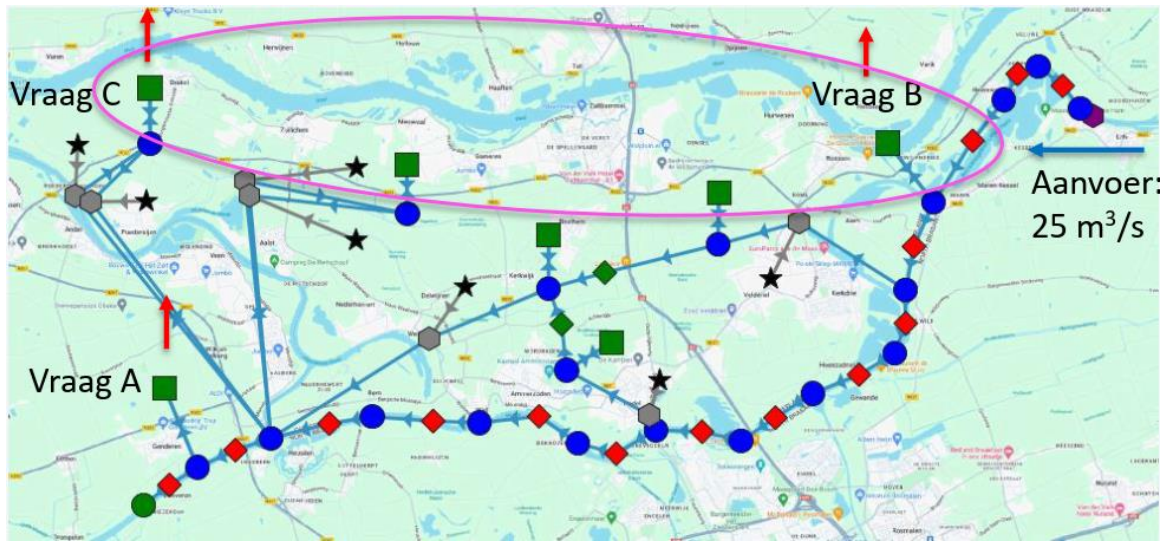
c. twee watervragers in verschillende aanvoergebieden

$B + C > \text{aanvoer} - A$, de prioriteit van B en C is 2. De watervraag van B en C is $1 \text{ m}^3/\text{s}$. A (prioriteit 1) wordt gebruikt om een tekortsituatie te creëren van respectievelijk 1 en $1,5 \text{ m}^3/\text{s}$. Verdamping is uitgeschakeld voor een eerlijke vergelijking.

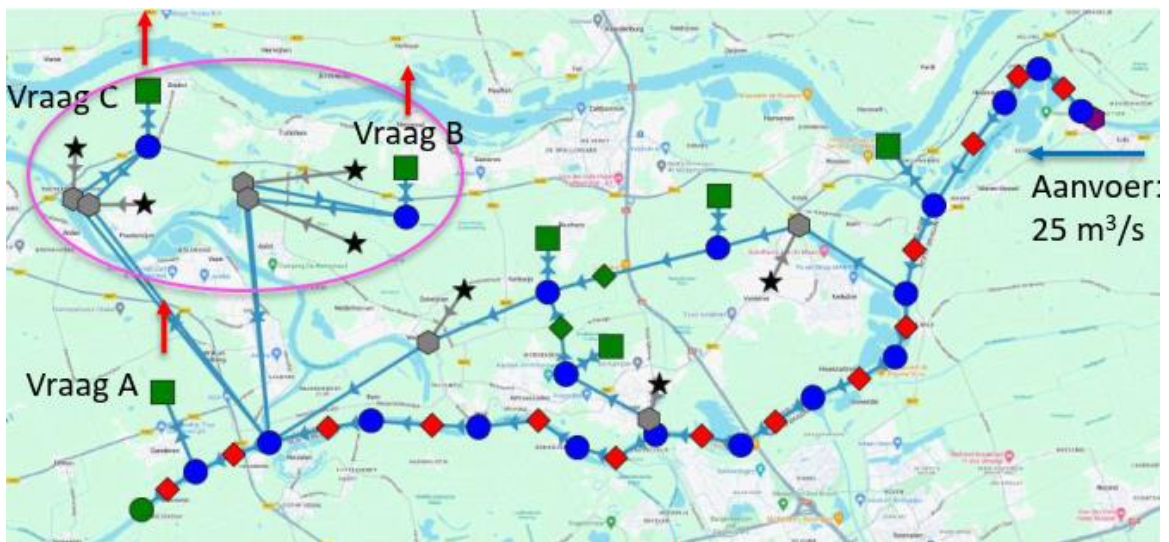
Afbeelding 4.12 Configuratie a: twee watervragers in het hoofdwatersysteem (beneden- (A) en bovenstrooms (B))



Afbeelding 4.13 Configuratie b: twee watervragers, één in het hoofdwatersysteem (B) en één in een aanvoergebied bovenstrooms van een gestuurd object (C)

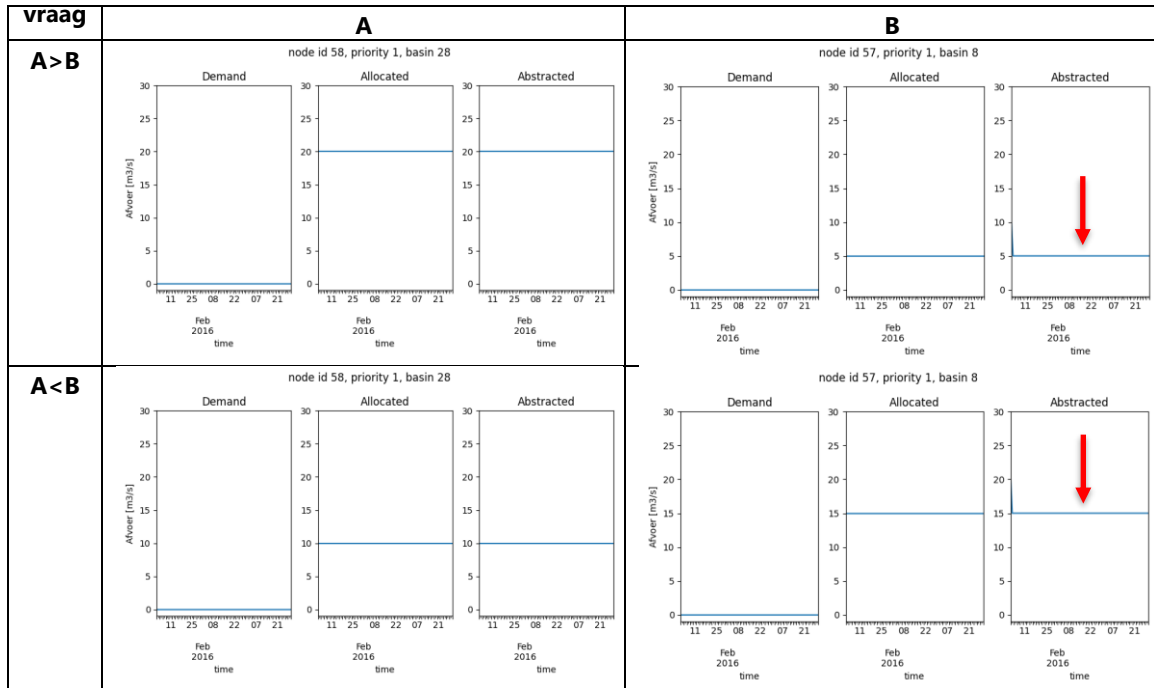


Afbeelding 4.14 Configuratie c: twee watervragers, beiden in een aanvoergebied bovenstrooms van een gestuurd object (B en C)



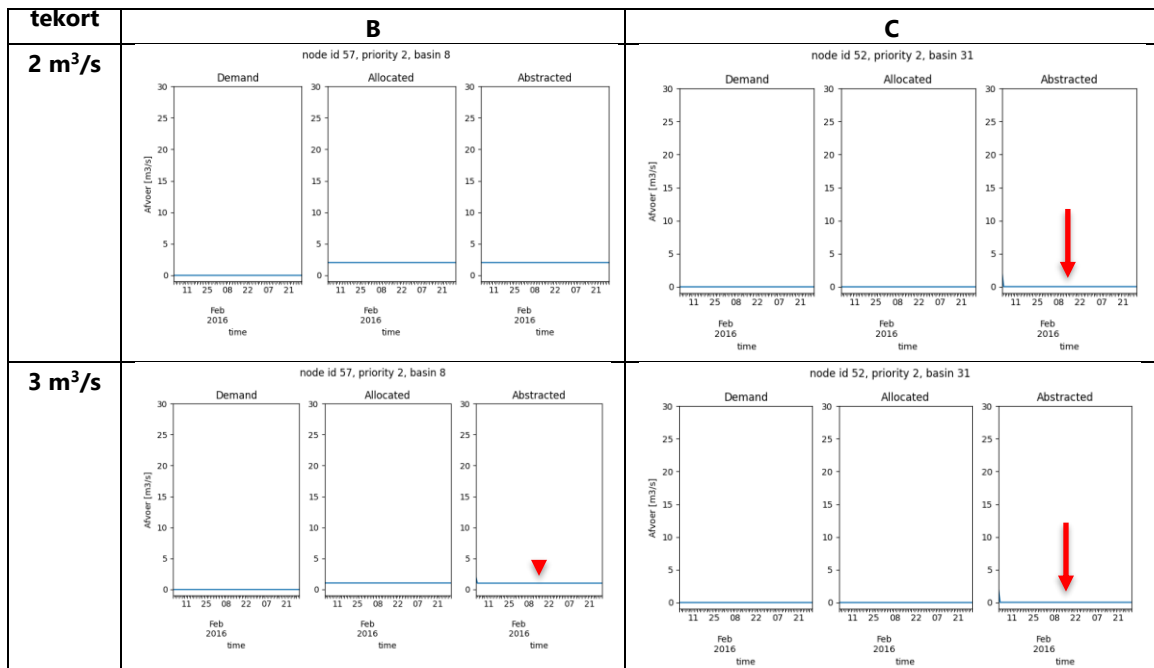
In afbeelding 4.15 zijn de resultaten van configuratie a weergegeven. De totale watervraag van de gebruikers is 30 m³/s, terwijl de aanvoer 25 m³/s is. In beide situaties lijkt het tekort te worden opgelost door de bovenstroomse abstractie (B) te korten. Bij gelijke prioriteit kan worden verwacht dat de abstractie voor beide gebruikers in dezelfde mate wordt gekort. Dit is niet het geval.

Afbeelding 4.15 Allocatieresultaten van configuratie a: twee watervragers in het hoofwatersysteem



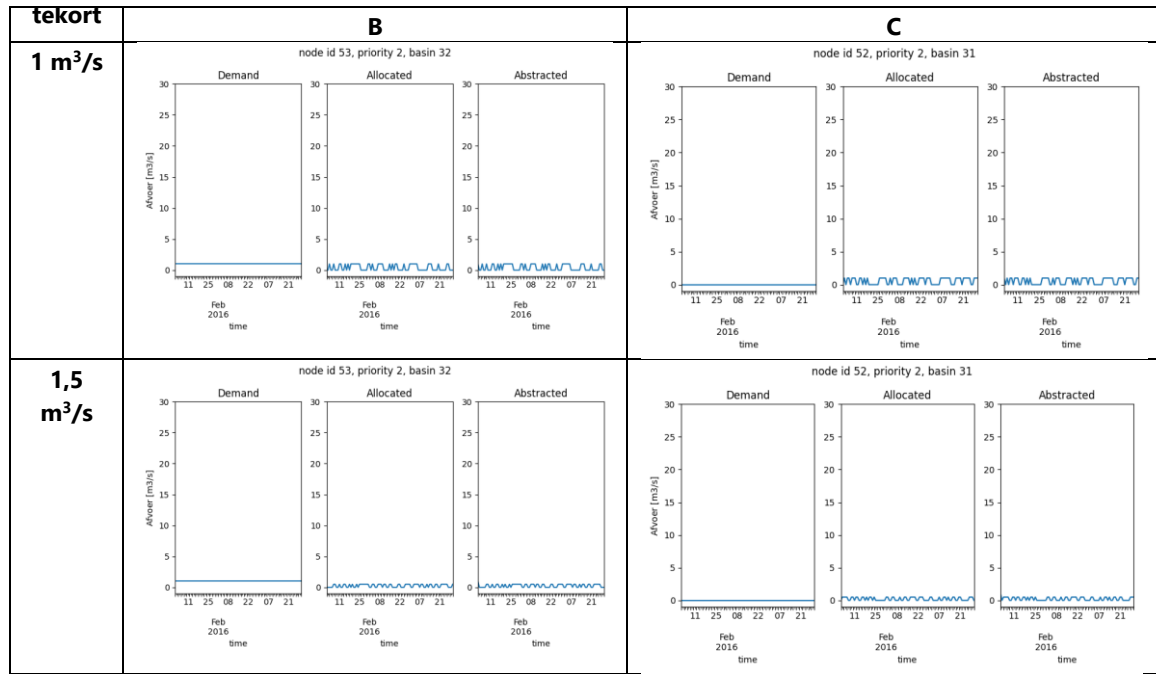
In afbeelding 4.16 zijn de resultaten van configuratie b weergegeven. De totale watervraag is $4 \text{ m}^3/\text{s}$. Terwijl het aanbod 2 of $1 \text{ m}^3/\text{s}$ is: een tekort van respectievelijk 2 en $3 \text{ m}^3/\text{s}$. Ondanks de gelijke prioriteit, lijkt in beide situaties het tekort te worden opgelost door de abstractie van de gebruiker bovenstrooms van de pomp (C) het meeste te korten.

Afbeelding 4.16 Allocatieresultaten van configuratie b: een watervrager in het hoofwatersysteem (B) en een watervrager in een aanvoergebied bovenstrooms van een gestuurde pomp (C)



In afbeelding 4.17 zijn de resultaten van configuratie c weergegeven. De totale watervraag is 2 m³/s. Het aanbod is 1 of 0,5 m³/s: een tekort van respectievelijk 1 en 1,5 m³/s. De gerealiseerde abstractie voor beide gebruikers lijkt te fluctueren. Op het moment dat de ene gebruiker wel water krijgt toebedeeld, krijgt de andere gebruiker dit niet.

Afbeelding 4.17 Allocatieresultaten van configuratie c: twee watervragers in twee verschillende aanvoergebieden bovenstrooms van een gestuurde pomp



Issue V: bovenstroomse abstractie wordt gekort

Bij gebruikers met gelijke prioriteit in hetzelfde subnetwerk is het wenselijk dat het water eerlijk wordt verdeeld over de verschillende gebruikers. Dit is niet het geval. Onder eerlijke verdeling zijn nog verschillende opties denkbaar:

- 1 het beschikbare water wordt eerlijk verdeeld over het aantal gebruikers;
- 2 het beschikbare water wordt naar verhouding (linear) verdeeld afhankelijk van de grootte van de vraag van elke gebruiker;
- 3 geen lineaire verdeling maar bijvoorbeeld een kwadratische verdeling.

Issue VI: allocatie en gerealiseerde abstractie fluctueren

Het lijkt erop dat er om en om aan de vraag wordt voldaan i.p.v. een continue verdeling.

Test 5: Verschillende prioriteit

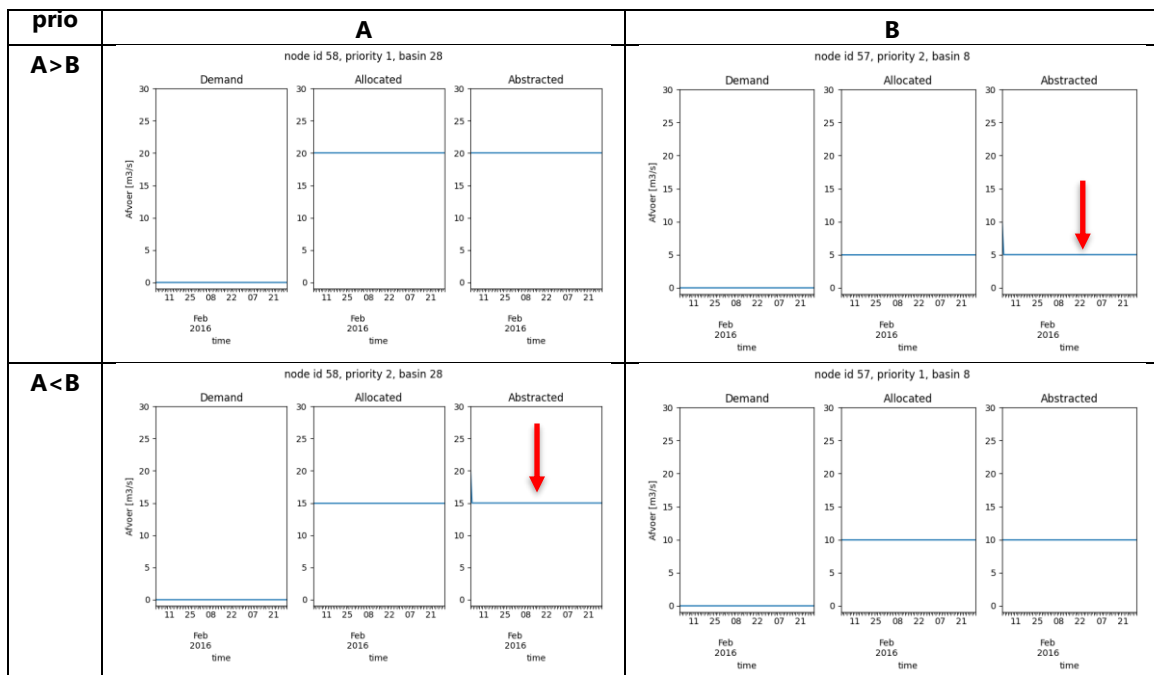
De verdeling van watervraag over verschillende gebruikers met verschillende prioriteit is ook getest aan de hand van een drie verschillende configuraties (zie afbeelding 4.12 tot en met afbeelding 4.14):

- a. twee watervragers in het hoofwatersysteem**
De watervraag van A en B is respectievelijk 20 en 10 m³/s. Twee varianten zijn doorgerekend, waarbij de prioriteit is verwisseld tussen de gebruikers (A>B en B>A);
- b. één watervrager in het hoofwatersysteem (B) en één watervrager in een aanvoergebied (C)**
De watervraag van B en C is 2 m³/s. Twee varianten doorgerekend, waarbij de prioriteit is verwisseld tussen de gebruikers (B>C en B<C). A (prioriteit 1) wordt gebruikt om een tekortsituatie te creëren van respectievelijk 2 m³/s. Verdamping is uitgeschakeld voor een eerlijke vergelijking;
- c. twee watervragers in verschillende aanvoergebieden**

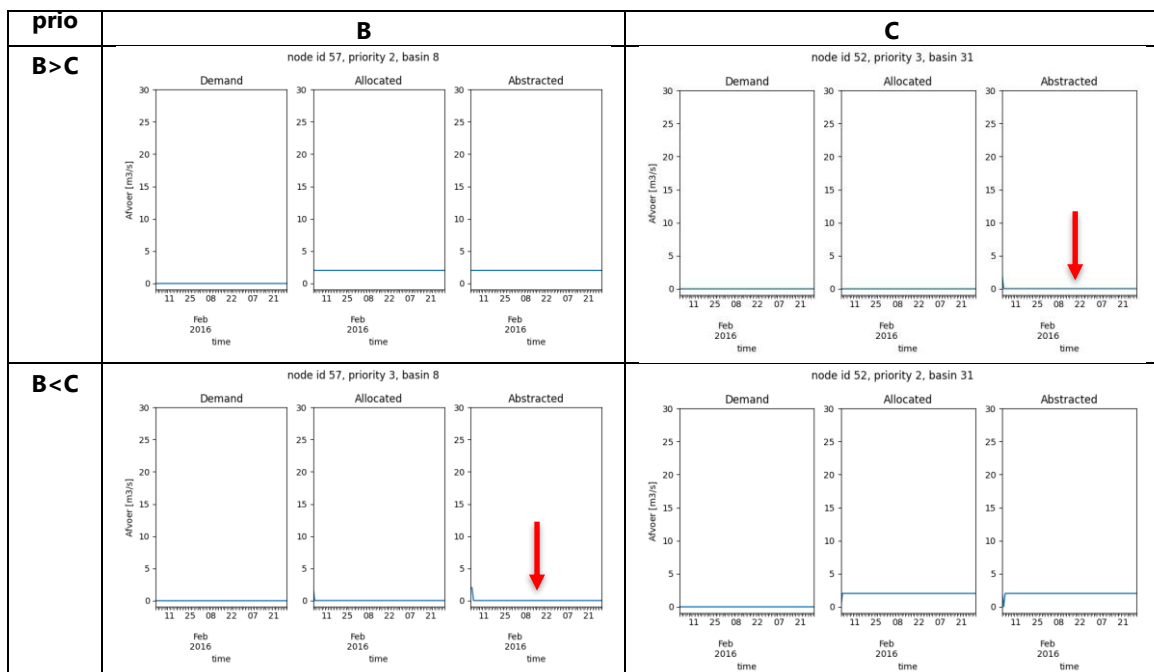
De watervraag van B en C is 1 m³/s. Twee varianten doorgerekend, waarbij de prioriteit is verwisseld tussen de gebruikers (B>C en B<C). A (prioriteit 1) wordt gebruikt om een tekortsituatie te creëren van respectievelijk 1 m³/s. Verdamping is uitgeschakeld voor een eerlijke vergelijking.

In afbeelding 4.18 en afbeelding 4.19 zijn de allocatieresultaten weergegeven van configuratie a en b. Uit beide resultaten blijkt dat de volgorde van prioriteit goed wordt afgewerkt. De abstractie van de gebruiker met de laagste prioriteit wordt in alle situaties gekort.

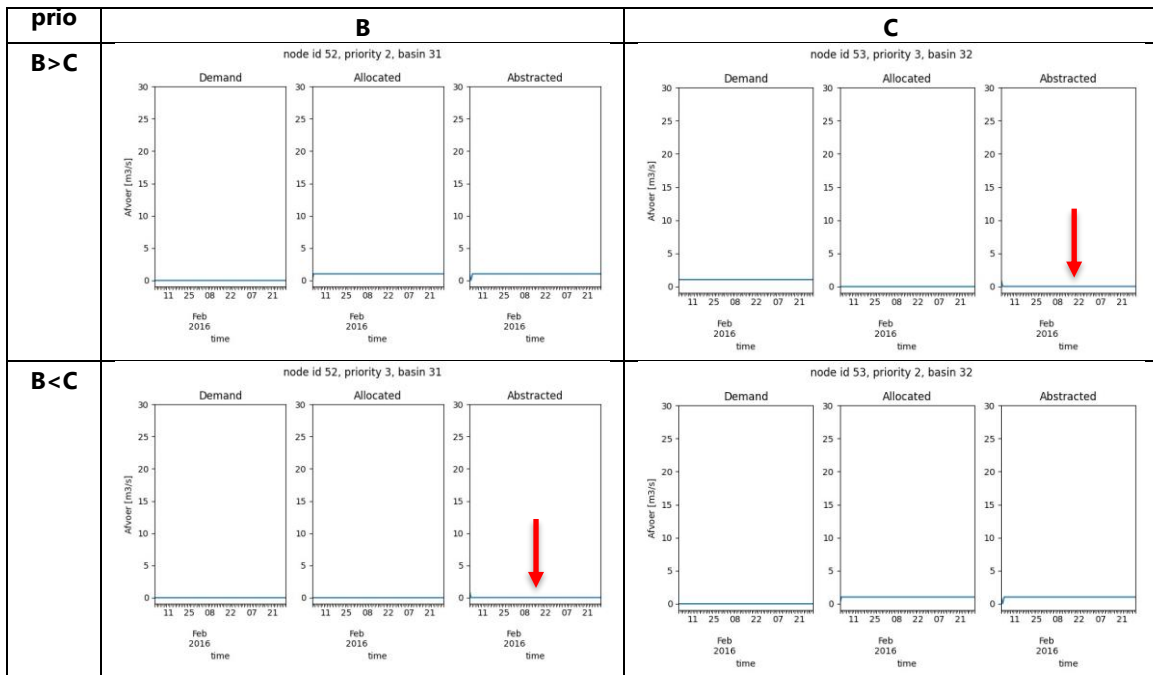
Afbeelding 4.18 Allocatieresultaten van configuratie a: twee watervragers in het hoofdwatersysteem



Afbeelding 4.19 Allocatieresultaten van configuratie b: een watervrager in het hoofdwatersysteem (B) en een watervrager in een aanvoergebied bovenstrooms van een gestuurde pomp (C)



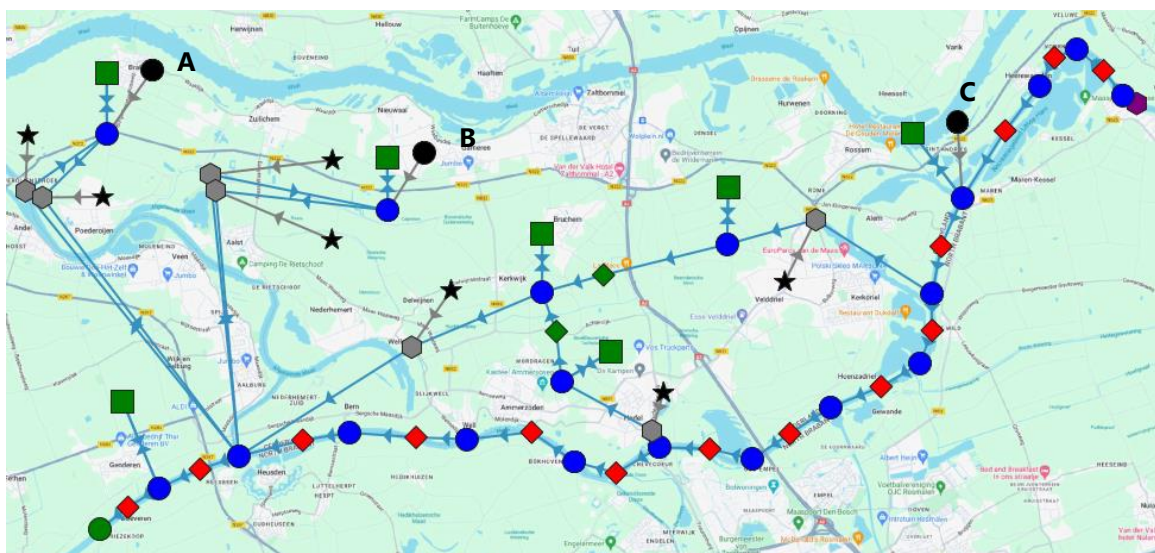
Afbeelding 4.20 Allocatieresultaten van configuratie b: twee watervrager in aanvoergebieden bovenstrooms van een gestuurde pomp (B en C)



Test 6: LevelDemand vraag/overschot

Om de LevelDemand knoop te testen, zijn LevelDemands toegevoegd aan enkele basins in de schematisatie (zie afbeelding 4.21). Aan een LevelDemand kan een min_level en max_level worden meegegeven. Wanneer de waterstand boven max_level uitkomt, ontstaat een overschot, wat kan worden gebruikt door andere watervragers binnen het subnetwork. Beneden min_level ontstaat een vraag.

Afbeelding 4.21 Schematisatie met toevoeging van enkele LevelDemand knopen



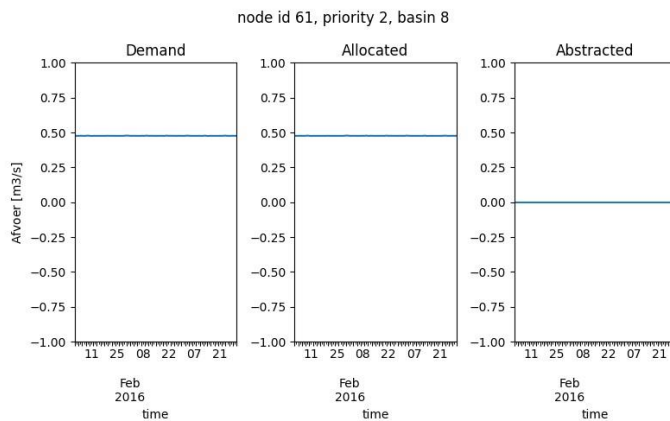
In eerste instantie is er getest of er inderdaad watervragen en wateroverschotten worden berekend. Dit is gedaan m.b.v. verschillende configuraties. Er is kunstmatig een situatie gemaakt, waarbij er $1 \text{ m}^3/\text{s}$

beschikbaar is voor het allocatiealgoritme, door bovenstrooms 25 m³/s als randvoorwaarde op te drukken en benedenstrooms 24 m³/s te onttrekken met een UserDemand met hogere prioriteit.

a. watervraag

Het min_level en max_level van C worden ingesteld op een waterstand die hoger is dan voorkomt in de refentiesituatie (0,6 en 0,7 m), zo moet er een watervraag ontstaan. Er ontstaat inderdaad een watervraag, maar deze lijkt niet gerealiseerd te worden, terwijl hier voldoende water voor beschikbaar is.

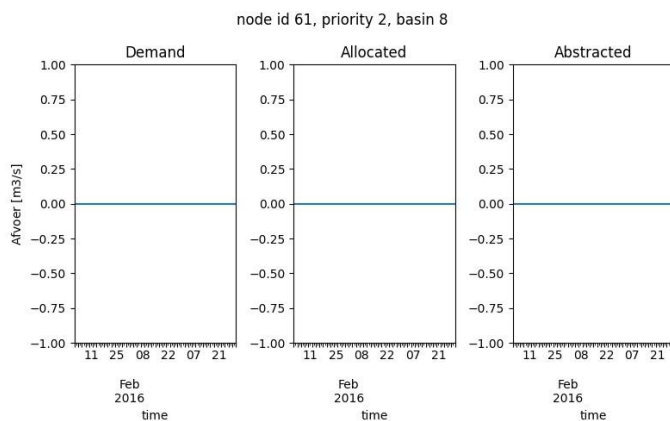
Afbeelding 4.22 Allocatieresultaat LevelDemand, configuratie a



b. wateroverschot

Het min_level en max_level van C worden ingesteld op een waterstand die lager is dan voorkomt in de refentiesituatie (0,3 en 0,4 m), zo moet er een wateroverschot ontstaan. Dit wateroverschot is niet zichtbaar in de resultaten, ook niet als negatieve vraag.

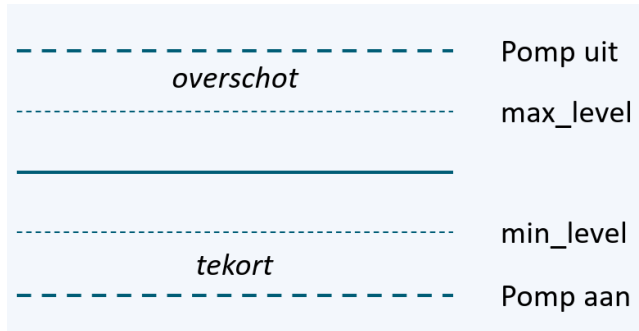
Afbeelding 4.23 Allocatieresultaat LevelDemand, configuratie b



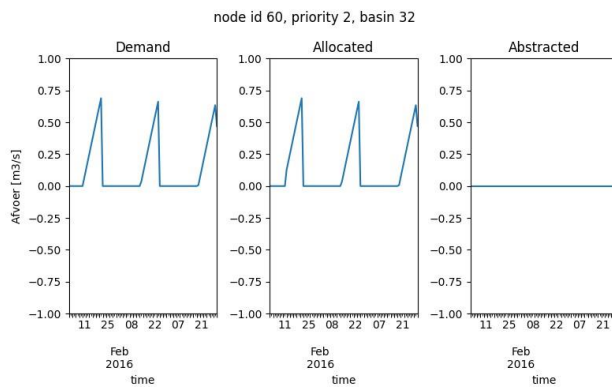
c. alternerend wateroverschot-watervraag

Dit is getest in een aanvoergebieden bovenstrooms (A en B) van een gestuurde pomp. Het min- en max_level van de LevelDemand is ingesteld zodat dit binnen de grenzen valt van de sturing van de pomp (zie afbeelding 4.24). Zo kan er in het gebied tijdelijk sprake zijn van een overschot en een tekort. De watertekorten worden inderdaad gevonden. Het overschot wordt niet gezien, ook niet als negatieve vraag.

Afbeelding 4.24 Sturing pomp en min-, max_level van de LevelDemand



Afbeelding 4.25 Allocatieresultaat LevelDemand, configuratie c



Issue VII: missende min_level en max_level waarde

In de manual staat: 'If both are missing, LevelDemand won't have any effects on allocation.'

In de praktijk geeft de simulatie een error bij missende waarden. Het is dus niet mogelijk om op deze manier LevelDemand knopen te (de)activeren.

Issue VIII: overschot wordt niet gezien

Overschot is niet terug te vinden in het resultaat, ook niet als negatieve demand.

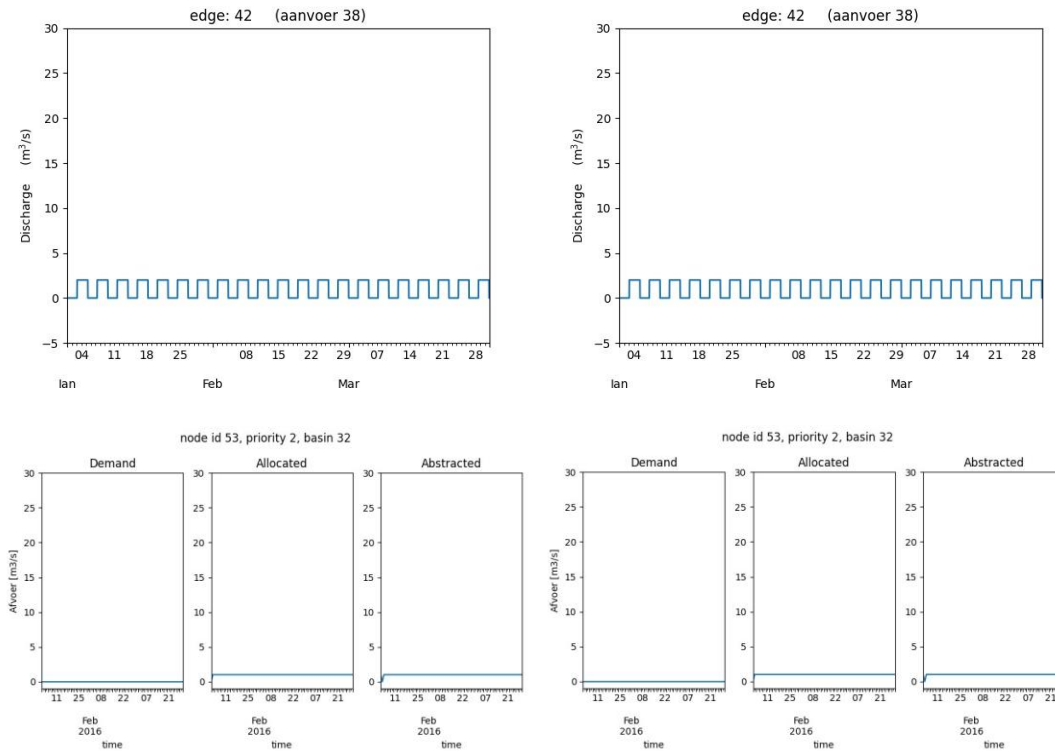
Deltares/Ribasim issue #1547

Test 7: LevelDemand in combinatie met UserDemand

Met deze test wordt gekeken of lokaal het overschot van een LevelDemand wordt gebruikt om de vraag van de UserDemand (deels) op te lossen. Door de min- en max_level van de LevelDemand op de onderste helft van het peilregime (sturing pomp) in te stellen, wordt er herhaaldelijk een overschot gecreëerd. De LevelDemand kan juist worden uitgeschakeld door de min- en max_level gelijk te kiezen aan de onder- en bovengrens van de pomp, aangezien de waterstand niet of nauwelijks hierbuiten zal komen.

De vraag van de UserDemand wordt in beide gevallen gerealiseerd. De aanvoer via de pomp is gelijk gebleven. Het lijkt er dus op dat het overschot van de LevelDemand nergens aan bijdraagt.

Abbeelding 4.26 Allocatieresultaten voor de combinatie LevelDemand-UserDemand. Links: LevelDemand ingeschakeld, Rechts: LevelDemand uitgeschakeld



Issue IX: overschot wordt niet gebruikt om lokaal een vraag op te lossen
Deltares/Ribasim issue #1545

5 CONCLUSIE

Met het uitvoeren van de testcases voor allocatie in Ribasim (versie 2024.4.0) zijn samengevat de volgende bevindingen gedaan:

- bij gelijke prioriteit in hetzelfde wordt de bovenstroomse gebruiker gekort;
- bij gelijke prioriteit met een watervraag bovenstrooms van een gestuurd object, wordt de gebruiker bovenstrooms van het object eerder gekort;
- bij verschillende prioriteit wordt inderdaad de gebruiker met de laagste prioriteit gekort;
- de watervraag van de gebruiker wordt ook gezien over gestuurde objecten, de realisatie is (logischerwijs) ook afhankelijk van de capaciteit die kan worden aangevoerd over het object;
- de watervraag van LevelDemand knopen wordt niet gerealiseerd;
- het overschot van LevelDemand knopen wordt niet gezien en gebruikt.

Afgezien van enkele issues, onduidelijkheden en (nu nog) ontbrekende functionaliteiten die nog moet worden opgelost, is het wenselijk dat bij gelijke prioriteit van verschillende gebruikers een eerlijke verdeling wordt berekend.

6 VOORUITBLIK/AANBEVELINGEN

In deze fase van de ontwikkeling is het van belang om veel simpele testmodellen te bouwen en op te nemen in de testbank om basisfunctionaliteiten uit te testen. Ook moet goed worden gecontroleerd of updates binnen het rekenhart ook zijn doorgevoerd in de Qgis plugin.

Het model en de logboeken zullen worden gedeeld met de projectgroep/Deltares. Mogelijk kan dit (of een vergelijkbaar) model in de testbank van Ribasim worden opgenomen, zodat bij nieuwe versies van de Ribasim software de tests kunnen worden herhaald.

7 REFERENTIES

- 1 Witteveen+Bos (2023). Uitgangspunten notitie koppeling Bommelerwaard met hoofdwatersysteem.