

MEMO BvtH/M08.042
Datum 12-06-2008
Auteur(s) Bas van 't Hof en Nils van Velzen
Onderwerp Koppeling van WAQUA aan COSTA en OpenDA

Documentinformatie

Versie	Auteur	Datum	Opmerkingen	Review
0.1	BvtH	09-06-2008	Concept	CvV, JD
1.0	BvtH	12-06-2008	Aangepaste versie na revisie	
1.1	CVV	29-08-2008	Uitgebreide versie opgenomen koppeling met OpenDA	
Bestandslocatie:		/home/vortech/Memos		

1 Inleiding

In de change c80111 is een koppeling gemaakt tussen WAQUA en COSTA, waarbij verschillende routines van WAQPRO zijn gebruikt om COSTA objecten in te richten. Het feit dat de koppeling met COSTA hierdoor onderdeel is geworden van SIMONA, heeft een aantal voordelen:

- COSTA is voortaan niet meer met een oudere versie van WAQPRO gekoppeld, maar met de nieuwste;
- Alle COSTA-methoden kunnen eenvoudig worden toegepast op WAQUA.
- Gebruikers kunnen kennis maken met de mogelijkheden van COSTA.
- de mogelijkheid ligt open om COSTA de huidige Kalman-functionaliteiten te laten vervangen.

In de change c84342 is WAQUA ook aan OpenDA gekoppeld. Het OpenDA systeem bestaat uit een Java definitie van data assimilatie componenten en een aantal data assimilatie algoritmen. COSTA implementeerd voor de meeste componenten ook (een belangrijk gedeelte van de) OpenDA Java interface. Het is daarom mogelijk om COSTA componenten zonder aanpassingen ook in OpenDA te gebruiken. In het kader van de change c84342 is een koppeling gerealiseerd tussen het COSTA component van het WAQUA/TRIWAQ model zoals

gerealiseerd in change c80111 en OpenDA. Door gebruik te maken van deze koppeling is het momenteel mogelijk om het Ensemble Kalman filter en het Ensemble Square Root filter van OpenDA te combineren met het COSTA gekoppelde WAQUA/TRIWAQ model.

2 Gebruik van de COSTA-WAQUA koppeling: waqcosta.pl

Het gebruik van de nieuwe koppeling van WAQUA aan COSTA wordt op een zeer eenvoudige wijze aangeboden aan de gebruiker: er is een PERL-script `waqcosta.pl` gemaakt, waarmee de berekening -in plaats van met `waqpro.pl`- kan worden opgestart.

Het script `waqcosta` kent de volgende command line opties:

<code>-back</code>	(optional)	Run in background (y) or not (n)	default [N]
<code>-debug</code>	(default)	Debug program (y) or not (n)	default [N]
<code>-nmdbg</code>	(optional)	Nmdbg input file	
<code>-nmode</code>	(optional)	Number of RRSQRT-modes	default [1]
<code>-obsfile</code>	(mandatory)	Observations file (sqlite format)	
<code>-precision</code>	(default)	Precision (single or double)	default [SINGLE]
<code>-runid</code>	(mandatory)	Run identification	
<code>-tstart</code>	(mandatory)	Begin time (should correspond to TSTART in siminp-file)	
<code>-tstep</code>	(mandatory)	Time step (should correspond to TSTEP in siminp-file)	
<code>-tstop</code>	(mandatory)	End time (should correspond to TSTOP in siminp-file)	

3 Gebruik van de COSTA-openDA koppeling: waqopenda.pl

Het gebruik van de koppeling tussen COSTA en openDA is ongeveer gelijk aan de koppeling met COSTA. Het OpenDa filter programma met het WAQUA/TRIWAQ model wordt opgestart met behulp van het PERL-script `waqopenda.pl`.

Het script `waqcosta` kent de volgende command line opties:

<code>-back</code>	(optional)	Run in background (y) or not (n)	default [N]
<code>-damethod</code>	(default)	Data assimilation method to use (enkf/ensr)	
<code>-debug</code>	(default)	Debug program (y) or not (n)	default [N]
<code>-nmdbg</code>	(optional)	Nmdbg input file	
<code>-nmode</code>	(optional)	Number of ensemble members	default [1]
<code>-obsfile</code>	(mandatory)	Observations file (sqlite format)	
<code>-precision</code>	(default)	Precision (single or double)	default [SINGLE]
<code>-runid</code>	(mandatory)	Run identification	

4 Beperkingen van de huidige koppeling

De nieuwe versie van WAQPRO is volledig compleet: met de testbank is gecontroleerd dat alle functionaliteit nog beschikbaar is en dat er geen verschillen optreden in de resultaten.

Beperkingen treden alleen op wanneer WAQUA wordt gedraaid binnen COSTA:

- Er kan alleen sequentieel gerekend worden. Parallele berekening en domein decompositie zijn niet beschikbaar voor WAQUA-in-COSTA.
- Er kunnen alleen waterstandsmetingen worden verwerkt.
- WAQUA-in-COSTA is alleen beschikbaar voor Linux.
- Voorlopig is alleen het RRSQRT-filter beschikbaar voor WAQUA-in-COSTA. Berekeningen zonder filter kunnen dus nog niet binnen COSTA worden gedraaid, en ook steady state filters kunnen nog niet binnen COSTA worden gebruikt.
- Er is alleen nog getest met single precision berekeningen. Het is niet bekend of double precision ook werkt.
- Het ruismodel wordt (nog) van de SDS-file gelezen: de ruismodellen van COSTA kunnen nog niet worden gebruikt.
- Het circular buffer gebeuren ('smoothing van de randvoorwaarden') wordt nog niet door COSTA aangeboden.

Dezelfde beperkingen traden ook op in de pilot-versie.

5 Veranderingen in de SIMONA-programmatuur

De volgende aanpassingen zijn doorgevoerd in de SIMONA-programmatuur:

- Het nieuwe systeem `costa` is toegevoegd in de directory `src/extern`. Het bestaat uit de volgende onderdelen:
 - enkele include-files
 - enkele dynamische bibliothekende COSTA-bibliotheken zijn verkregen door COSTA te installeren via het commando `./configure --disable-mpi`, omdat anders de parallele faciliteiten van COSTA en die van WAQUA conflicten veroorzaakten.
- De executable `costawb.exe`
- Het script `obs2sql`

Hiermee kunnen metingen worden vertaald van het OBS2SDS-formaat naar het SQLITE3-formaat. COSTA verwerkt metingen alleen als ze zijn opgegeven in dit formaat. De conversie werkt alleen als het programma `sqlite3` is geïnstalleerd.

- Het nieuwe systeem `costadam`, in de directory `src/kalman`, met enkele lege COSTA-functies, is nodig om WAQPRO te kunnen linken zoals men gewend is.
- Het nieuwe systeem `waqcta`, in de directory `src/kalman`, biedt de faciliteiten om WAQUA-in-COSTA te kunnen gebruiken:
 - Het script `waqcosta.pl` is hierboven beschreven. Dit is het enige waar de gebruiker mee wordt geconfronteerd.
Dit script vult wat informatie in in het configuratiebestand `rrsqr_t_waqua.xml`, en start daarna de executable `costawb.exe` op.
 - Een template voor het configuratiebestand `rrsqr_t_waqua.xml`, waarin een COSTA-modelklasse wordt ingericht door de juiste routines op te geven, en het gebruik van het RRSRQT-algoritme wordt ingesteld.
 - Het script `make_waqcta_so.pl`, dat van de statische SIMONA-libraries een dynamische library `libwaqcosta.so` maakt. Dit script hoeft alleen maar te worden aangeroepen wanneer er veranderingen zijn opgetreden in de programmatuur.
 - Een aantal routines, die nodig zijn om de genoemde COSTA-modelklasse in te richten. Deze routines zijn in SIMONA's fortran-stijl geschreven, maar roepen wel veelvuldig COSTA functies aan.
- Kleine aanpassingen in WAQPRO:
 - Herstructurering, zodat op een hoog nivo verschillende calls herkenbaar zijn: sommige voor Kalman filters (die in WAQUA-in-COSTA niet worden aangeroepen) en andere calls, die uiteraard wel worden gebruikt in WAQUA-in-COSTA.
Naast enkele `tools`- en `coclib`-routines, worden 10 `waqpro`-routines en 4 `waqgen`-routines aangeroepen door WAQUA-in-COSTA. Alleen aanpassingen aan het gebruik van deze routines hebben dus een gevolg in de koppeling.
Ongeveer 65 `waqpro`-routines worden niet gebruikt in WAQUA-in-COSTA. De functionaliteit van deze routines is of zal moeten worden overgenomen door COSTA.
 - Enkele routines zijn uitgebreid zodat informatie desgewenst met COSTA kan worden uitgewisseld. Het gaat hier vooral om de routine `waskre`, die de informatie-uitwisseling verzorgt tussen het Kalman-filter en WAQUA. In deze routines (`waskpt`, `waskre` en `waskv1`) zijn calls opgenomen naar COSTA. Deze komen nergens anders voor in WAQPRO.

6 Testresultaten

De nieuwe versie van `waqpro` doorloopt de modellen-testbank zonder verschillen met de uitgangsversie.

Er zijn twee testen uitgevoerd met WAQUA-in-COSTA, met het csm8 model en het kust3d-model. Deze testen zijn afkomstig uit de modellentestbank. De test met het kust3d-model moest eerst worden aangepast, omdat COSTA alleen waterstandsmetingen kan verwerken.

De resultaten van WAQUA-in-COSTA zijn vergeleken met die van waqpro. Deze vergelijking wordt gerapporteerd in de Tabellen 1 en 2. De overeenkomst tussen de resultaten met het csm8-model is overduidelijk. De overeenkomst voor de resultaten met het kust3d-model wordt voldoende geacht.

De test-invoer voor deze testen is toegevoegd aan de testen `kust3d.kalman` en `csm8_rr.kalman` in de modellentestbank. Deze invoer wordt niet automatisch gedraaid door de testbank, maar de tests kunnen wel eenvoudig worden herhaald door een ervaren gebruiker.

99%-verschillen voor het CSM8 model			
Field	kalman	0kalman	kalman-0kalman
	costa-waqpro	costa-waqpro	waqpro
qx	9.906250	0	16406.312500
qy	7.312500	0	12526.759766
hu	0.000000	0	0.005798
hv	0.000004	0	0.007462
sep	0.000004	0	0.007489
up	0.000001	0	0.003238
vp	0.000001	0	0.003116
zwl	0.000003	0	0.007361

Tabel 1: *Vergelijking van resultaten van WAQUA-in-COSTA met die van waqpro. De resultaten van de 0kalman-berekening (gebruik van Kalman filter waarbij onzekerheden gelijk zijn aan 0) zijn identiek. De resultaten voor een echte Kalman-berekening zijn nagenoeg identiek: de verschillen (links) zijn veel kleiner dan de effecten van het Kalman filter (rechts).*

7 Aanpassingen aan het ontwerp

Voor deze melding was een ontwerp gemaakt in memo 07058. De aanpak die daarin wordt beschreven komt niet overeen met de manier waarop uiteindelijk de koppeling tussen WAQUA en COSTA in deze melding tot stand is gebracht. Het probleem met het ontwerp zoals dat is gepresenteerd in memo 07058 is, dat daarin is uitgegaan van een *integratie* van de pilot-versie van WAQUA-in-COSTA. Voor deze integratie is daarbij sterk de nadruk komen te liggen op de tekstuele vergelijking van verschillende versies van de WAQUA-routines. De pilot-versie is echter gebaseerd op een zeer oude versie: `Kalmina0507`.

Sinds deze versie zijn er veel veranderingen in WAQPRO opgetreden, ook op het vlak van de uitwisseling van informatie tussen WAQUA en het Kalman filter. Dit was noodzakelijk vanwege andere ontwikkelingen in WAQUA, zoals de uniformering van TRIWAQ en WAQUA,

99%-verschillen voor het KUST3d model			
Field	kalman costa-waqpro	Okalman costa-waqpro	kalman-Okalman waqpro
atr	104960	0	557094400
adtr	104960	0	557094400
qprev	68925	0	6477284
fltr	3336	0	17218056
rffi	616	0	2065274
qy	266.843628	0	695.762863
qx	49.079323	0	417.628174
dtr	13.949951	0	14.584581
ctr	1.660156	0	4504.009766
vp	0.073453	0	0.199690
rp	0.069839	0	0.415035
up	0.031502	0	0.146311
hu	0.013638	0	0.034770
hv	0.008085	0	0.022516
sep	0.007012	0	0.031458
zkpol	0.004294	0	0.009920
zwl	0.004294	0	0.015508
gro	0.002937	0	0.651257
zcuru	0.002116	0	0.034529
zcur	0.002091	0	0.060052
zcurv	0.001847	0	0.050990
vicow	0.001719	0	0.111704
dfl	0.001698	0	0.244503
zkcur	0.001213	0	0.015508
wphys	0.000170	0	0.000442
w	0.000091	0	0.000244
zcurwp	0.000017	0	0.000186
zcurw	0.000016	0	0.000054
czv	0.000006	0	0.000016
czu	0.000004	0	0.000015

Tabel 2: *Vergelijking van resultaten van WAQUA-in-COSTA met die van waqpro. De resultaten van de Okalman-berekening (gebruik van Kalman filter waarbij onzekerheden gelijk zijn aan 0) zijn identiek. De resultaten voor een echte Kalman-berekening zijn nagenoeg identiek: de verschillen (links) zijn veel kleiner dan de effecten van het Kalman filter (rechts).*

en het nieuwe barrier-schema. Zo zijn de Kalman Tabellen sterk krachtiger gemaakt, waardoor WAQUA veel geschikter is geworden voor een koppeling met COSTA dan het eerder was.

Bij het volgen van de aanpak zoals beschreven in memo 07058 (tekstueel integreren), werden een aantal van deze verbeteringen ongedaan gemaakt, om zodanig een WAQPRO-versie te krijgen die meer lijkt op de uitgangsversie Kalmina0507. Het was uiteraard niet de bedoeling om de ontwikkeling van de afgelopen 3 jaar terug te draaien voor de huidige melding.

8 Gebruikte compilers

Bij het uitvoeren van deze melding is er vertraging opgetreden als gevolg van problemen met de gebruikte compilers. Op het netwerk van VORtech bleek het onmogelijk om COSTA te compileren met de Intel Fortran compiler versie 9.0, de compiler die wordt gebruikt voor SIMONA. De reden was dat de compiler een foutmelding gaf wanneer hem werd gevraagd een dynamische library aan te maken. Op de computer van Nils van Velzen lukte dat wel, en alle andere beschikbare versies (versie 9.1, 10.0 en 10.1) konden bij VORtech dynamische libraries aanmaken. Het is niet uitgezocht hoe het kwam dat deze ene functionaliteit van de compiler kapot was, en waarom het defect optrad bij VORtech.

Enige tijd is doorgewerkt met versie 9.0 van de compiler. Toen er diagnostische uitvoer nodig was vanuit COSTA om fouten op te sporen (waarvan sommige in COSTA zelf zaten), bleek dit onwerkbaar. De (zeer tijdsintensieve) werkcyclus was op dat moment het volgende: Bas van 't Hof las de uitvoer voor aan Nils van Velzen, die elders extra debug-code maakte, compileerde, bibliotheken opstuurde, waarna Bas van 't Hof weer kon runnen en uitvoer voor kon lezen.

Op dat moment is de gehele code overgegaan op versie 10.0. Dit was op dat moment de nieuwste versie bij VORtech. Versie 10.1 is pas in de loop van de melding geïnstalleerd. Hiermee is een tijdelijk prototype (op een branch) afgemaakt. Bij de integratie van dit prototype naar de trunk is overgegaan op versie 10.1. Op dat moment is ook een aantal wijzigingen doorgevoerd aan het make-systeem en de `general-tools` van SIMONA, zodat SIMONA kan functioneren met een andere Intel Fortran compiler dan degene die op een computer als default is geïnstalleerd. De instelling `INTELDIR` in het bestand `Settings.inc` bepaalt nu welke compiler wordt gebruikt.

De nieuwe versie is daarna gecompileerd met beide compilers (versie 9.0 en versie 10.1). De resultaten met compiler versie 9.0 van de nieuwe SIMONA-versie waren identiek aan die van de trunk-versie. De gehele modellen-testbank kon worden verwerkt met de code die door compiler versie 10.1 is gecompileerd, maar er traden wel verschillen op. In een klein aantal gevallen waren deze verschillen groot. Binnen deze melding is het daarom niet verstandig om over te gaan op de nieuwe versie van de compiler. Er is niet met de systeemtesten vastgesteld of er nog andere problemen optraden.

De versie die was gecompileerd met compiler versie 9.0 kon echter niet functioneren binnen COSTA, dat immers met een andere compiler was gecompileerd. De resultaten in de tabellen 1 en 2 zijn verkregen door de vergelijking van de code die is gecompileerd met compiler versie 10.1. Aan het einde van de melding is COSTA nog eenmaal door Nils van Velzen gecompileerd

met compiler versie 9.0, waarna is vastgesteld dat de COSTA-testen nog doorlopen kunnen worden. Er is niet meer in detail naar de resultaten gekeken.

Voorlopig kan verder worden gewerkt met versie 9.0 van de compiler. Het probleem met het maken van dynamische bibliotheken streedt niet op binnen SIMONA, omdat deze niet worden gemaakt binnen SIMONA.

9 Andere verbeteringen in de code

- De routine `spotfe`, die in SIMONA bij het systeem `lapack` hoorde, is geen echte `lapack`-routine. Het is een door Martin Verlaan aangepaste versie van de `lapack`-routine `spotf2`.

Omdat COSTA zijn eigen `lapack` gebruikt, moest deze routine worden verplaatst: de lokatie `src/waquapublic/lapack/spotfe.f` is daarvoor gekozen. Daarbij is de routine aan de SIMONA-normen aangepast en voorzien van aanpassingen voor variabele precisie. Hierdoor is het nu ook mogelijk om domein-decompositie in double precision te combineren met Kalman filtering.

- Een fout in de SIMONA-interpolatie tool is hersteld, waardoor er geen fout meer optreedt wanneer het LIGPNT-array toevallig helemaal aan het einde van de INFOR-tabel staat.
- Een fout is hersteld bij het afsluiten van parallele berekeningen, waardoor COEXEC ervan op de hoogte raakt dat er errors zijn opgetreden, ook als alle processen ondanks deze errors door kunnen rekenen tot het einde.