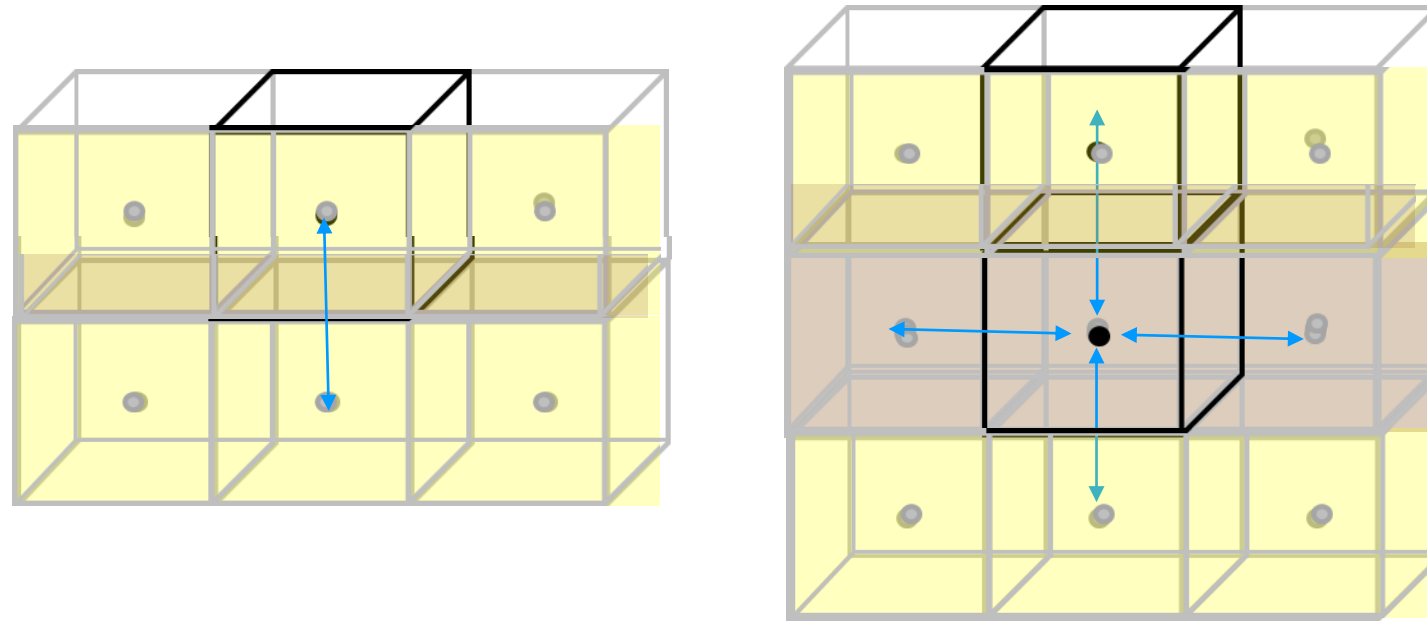


# Versneld rekenen in MODFLOW 6 (Dupuit-Forchheimer Leeft)



Andre Blonk

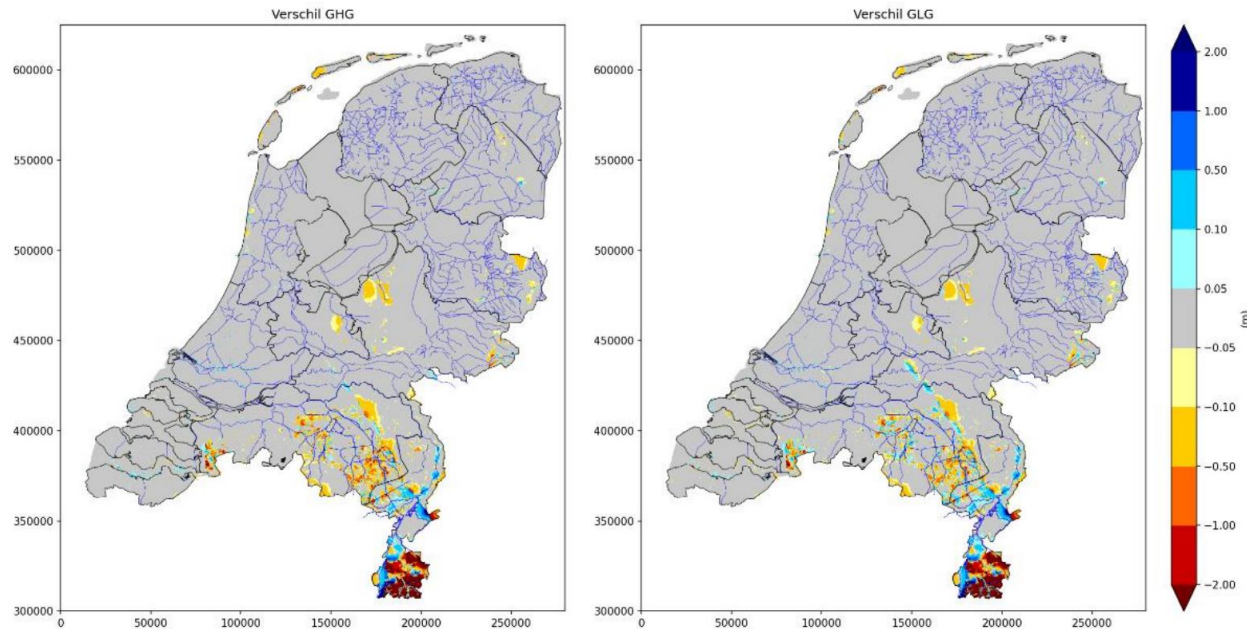
Hendrik Kok

Huite Bootsma

# Context: LHM benchmark

Tabel 4.2 Gealloceerd werkgeheugen en rekestijd voor MF2005 en MF6 LHM modellen zonder en met horizontale anisotropie.

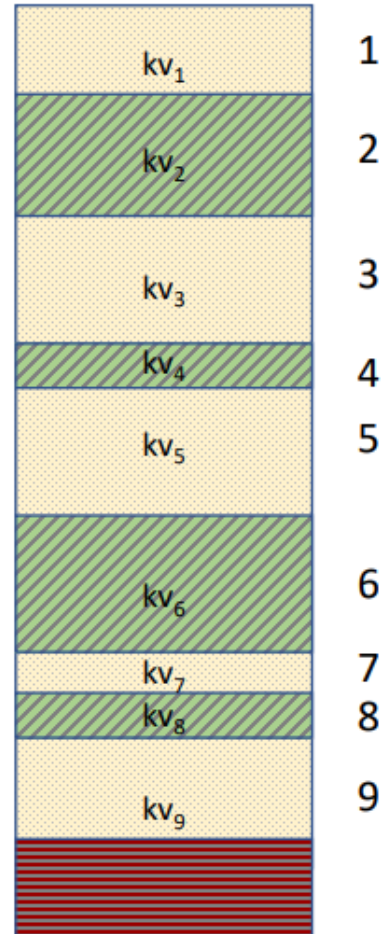
Model	Werkgeheugen (GB)	Rekestijd 1 modeljaar (uren)
MF2005-MS zonder ani (8 lagen)	10	3.13
MF2005-MS met ani (8 lagen)	10	3.35
MF6-MS zonder XT3D (15 lagen)	16	5.17
MF6-MS met XT3D (15 lagen)	23	18.02



# Versnelling: want halvering aantal lagen (+1)

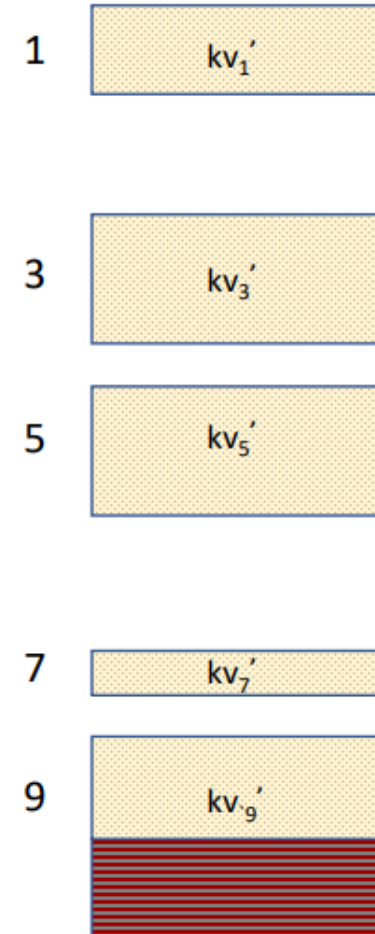
Mf6 model 9 laags met actieve c-lagen

$$c_2 = \frac{0.5 \times d_1}{kv_1} + \frac{d_2}{kv_2} + \frac{0.5 \times d_3}{kv_3}$$



Geen horizontale stroming in slechtdoorlatende lagen

Mf6 model 5 laags zonder actieve c-lagen



$$c_2 = \frac{0.5 \times d_1}{kv'_1} + \frac{0.5 \times d_3}{kv'_3}$$

En deze kiezen we constant en willekeurig bv  $kv'_9=0.001$   
De overige 4  $kv'$  waarden kunnen we met terug substitutie uitrekenen en kunnen  $< 0$  worden

# Hoe te doen met MODFLOW6

Vraagt om negatieve verticale doorlatendheden (k33)

MODFLOW6 laat dit niet toe, vanwege de parametercontroles

Bovendien moeten in een gestructureerd MODFLOW6 (DIS) model de boven- en onderkant van lagen aansluiten

Liever in Python via de imod-coupler: koppelt nu bijv. MODFLOW6 en MetaSWAP, maar principe is algemeen inzetbaar

Lost op het probleem van onaangesloten boven- en onderkanten op (via DISU)

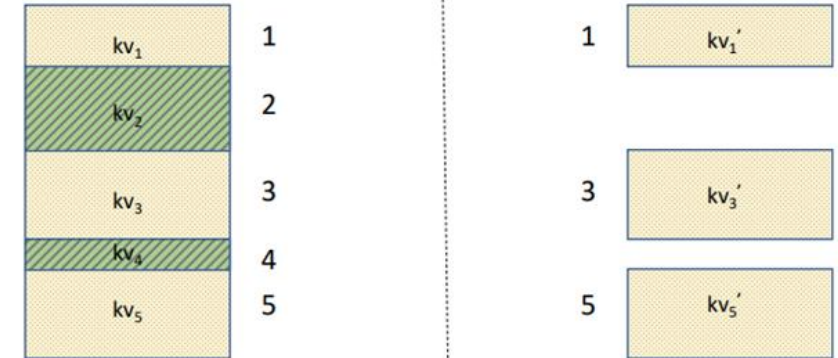
```
from xmipy import XmiWrapper

# Initialisatie
mf6 = XmiWrapper(
    lib_path="libmf6.dll",
    working_directory="ex01",
)
# Pas data in het geheugen aan
...

# Afsluiting
mf6.finalize()
```

# Benchmarks

periods	Dupuit (s)	3d (s)	ratio	Efficiency
5	5.9	8.7	1.48	0.89
10	16.3	22.4	1.37	0.82
50	121.5	181.7	1.50	0.90
100	159.0	231.6	1.46	0.87



Verhouding 3D versus Dupuit  
 $5 / 3 = 1.67$

Tabel 4.2 Gealloceerd werkgeheugen en rekentijd voor MF2005 en MF6 LHM modellen zonder en met horizontale anisotropie.

Model	Werkgeheugen (GB)	Rekentijd 1 modeljaar (uren)
MF2005-MS zonder ani (8 lagen)	10	3.13
MF2005-MS met ani (8 lagen)	10	3.35
MF6-MS zonder XT3D (15 lagen)	16	5.17
MF6-MS met XT3D (15 lagen)	23	18.02

$$5.17 / 3.13 = 1.65$$

# Benchmarks

Tabel 4.2 Gealloceerd werkgeheugen en rekentijd voor MF2005 en MF6 LHM modellen zonder en met horizontale anisotropie.

Model	Werkgeheugen (GB)	Rekentijd 1 modeljaar (uren)
MF2005-MS zonder ani (8 lagen)	10	3.13
MF2005-MS met ani (8 lagen)	10	3.35
MF6-MS zonder XT3D (15 lagen)	16	5.17
MF6-MS met XT3D (15 lagen)	23	18.02

$$5.17 / 3.13 = 1.65$$

Dus: gecorrigeerd voor lagen is MF6 even snel!

Extrapolatie, gegeven 0.87 efficiency

Lagen 3d	Lagen DF	Ratio	Versnelling
3	2	1.50	1.31
5	3	1.67	1.46
7	4	1.75	1.53
9	5	1.80	1.57
11	6	1.83	1.60
13	7	1.86	1.62
15	8	1.88	1.64
17	9	1.89	1.65
19	10	1.90	1.66
21	11	1.91	1.67
23	12	1.92	1.67
25	13	1.92	1.68
27	14	1.93	1.68
29	15	1.93	1.69
31	16	1.94	1.69

# Conclusie & verwachting

- MODFLOW6 “quasi-3d” is even snel als MODFLOW2005
- Vraagt wel om een aangepast model te schrijven, en te draaien met imod-coupler
- Gebruikt de onveranderde MODFLOW6 code
  
- 2021: Conceptuele uitwerking + Testen
- 2022: inbouwen imod-python & imod-coupler + Systematische benchmarks
- 2022: Test LHM MODFLOW6
- 2022: in iMOD release juni