



ÜLKEMİZE UYGUN HİDROLOJİK, SU KALİTESİ VE EKOLOJİK MODELLERİN GELİŞTİRİLMESİ PROJESİ



HİDROJEOLOJİK MODÜL

HİDRO-YAS MODELİ

17 Aralık 2019

Ankara

Prof. Dr. Ahmet DOĞAN
Mohsen Mahmoody VANOLYA
Yıldız Teknik Üniversitesi

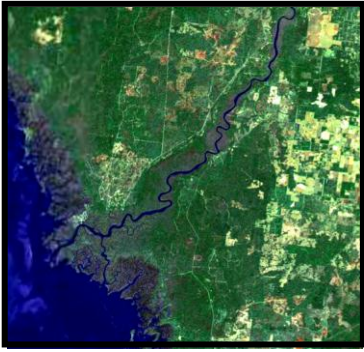


İÇERİK

- Hidrojeolojik Modül
- Denklemler
- Sınır Şartları
- Akış Yazılımı
- Taşıma Modeli
- Taşıma Yazılımı

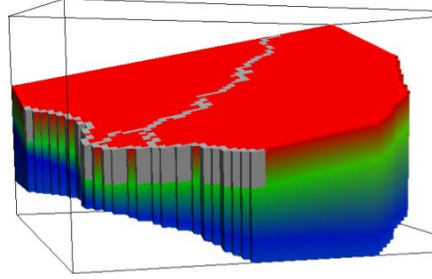


Hidrojeolojik Modül:



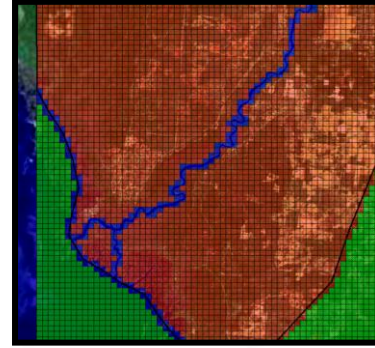
Gerçek Sistem

Problemler ve çözüm süreçleri?



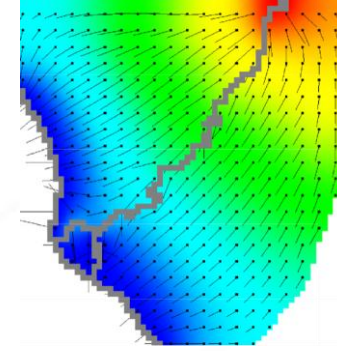
Kavramsal Model

Kullanıcının hidrojeolojik yorumu?



Girdiler

Grid boyutu, zaman adımı, Parametre Tahmini, Sınır Şartlarının İfadesi

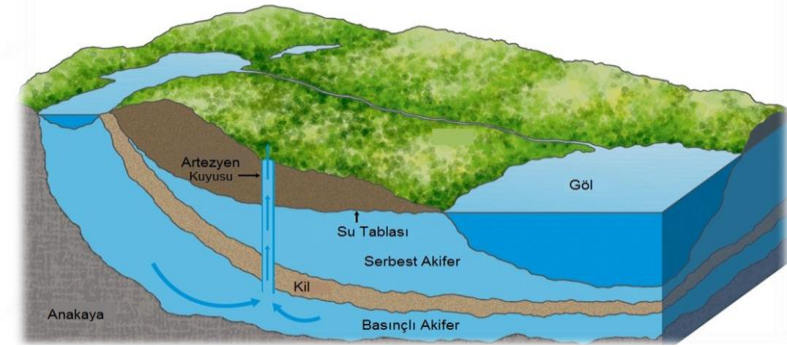
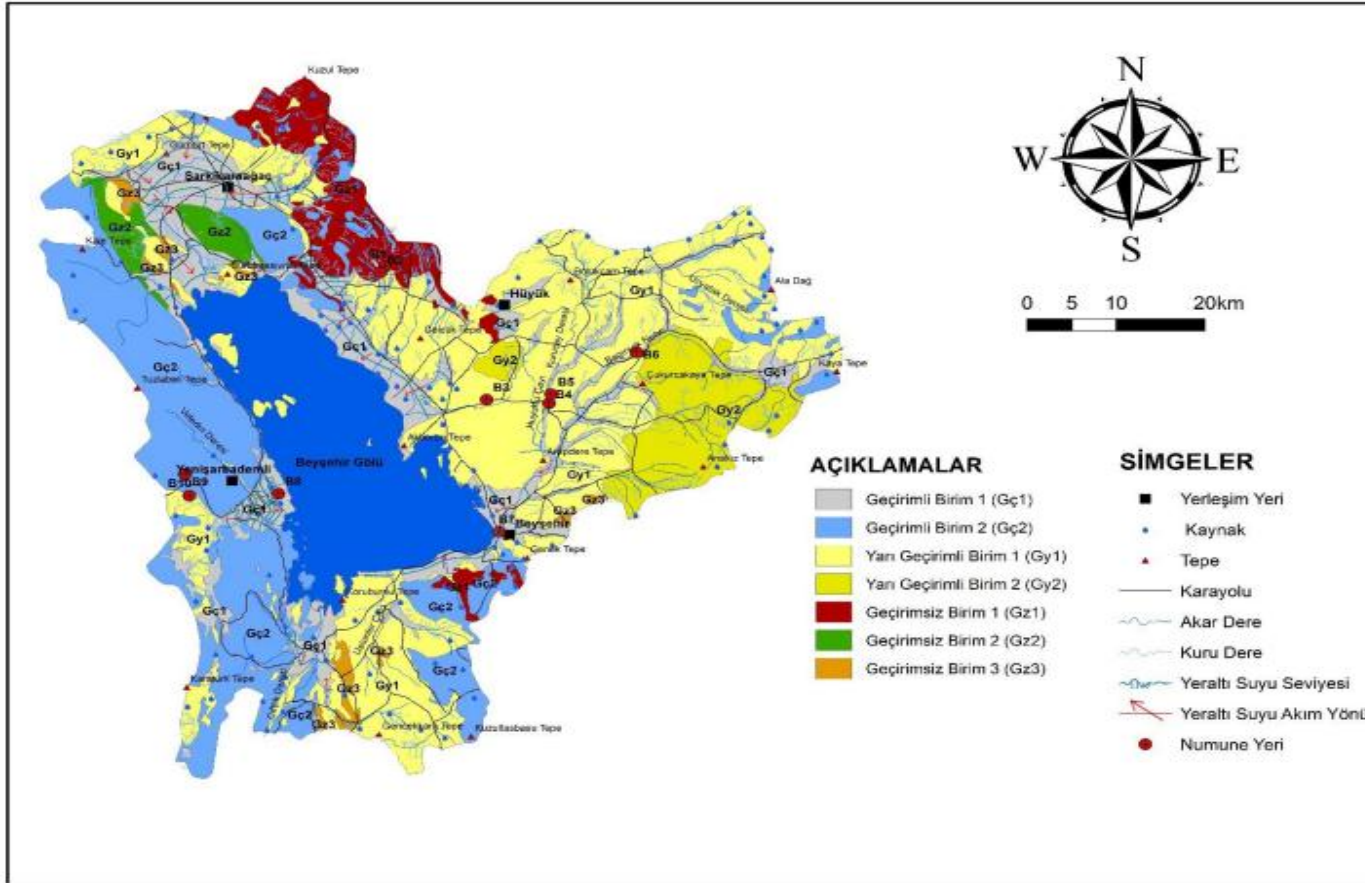


Çıktılar

Sistemin hidrolojik veya insan kaynaklı etkiye karşı tepkisi

Hidrojeolojik Modül: YAS Geoloji ve Hidrolik kombinasyonudur

Suyun Depolaması ve İletmesi



Hidrojeolojik Birimler

Akifer: Su taşır ve su iletir

Çakıl, kum, kumtaşı, karstik kireçtaşı vbg.

Akitard: Su taşır ve suyu az iletir

Killi ince kum, Killi siltli kum, Silt vbg.

Akiklöd: Su taşır ve suyu iletmez

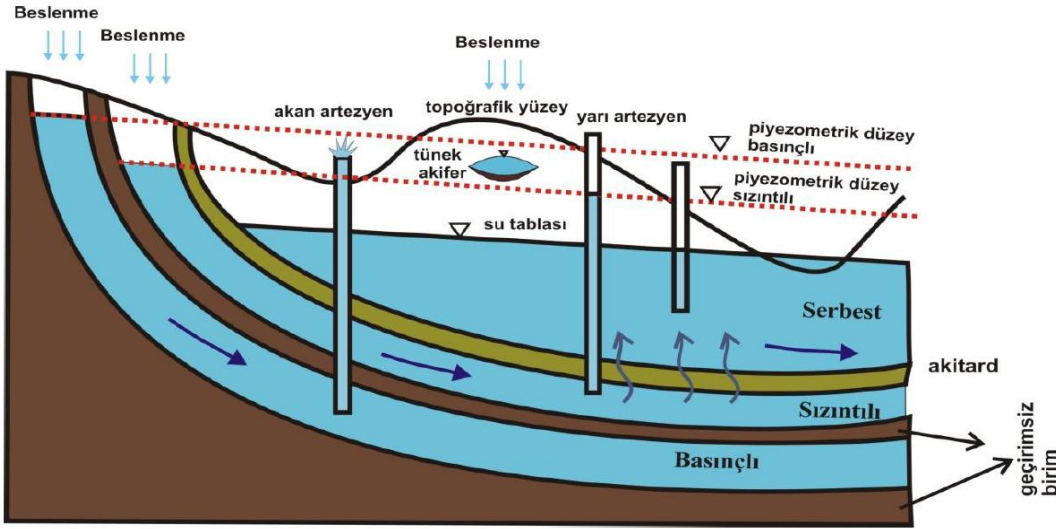
Kil, Siltli Kil.

Akifüj: Suyu ne taşır ne de iletir

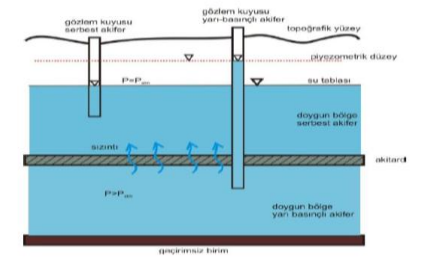
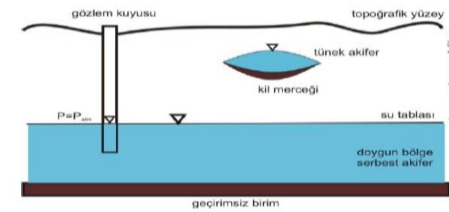
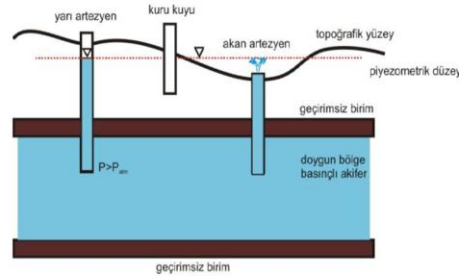
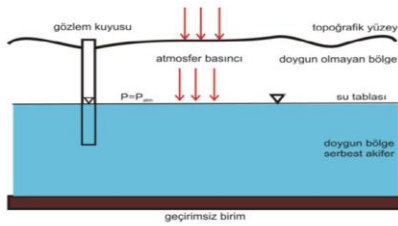
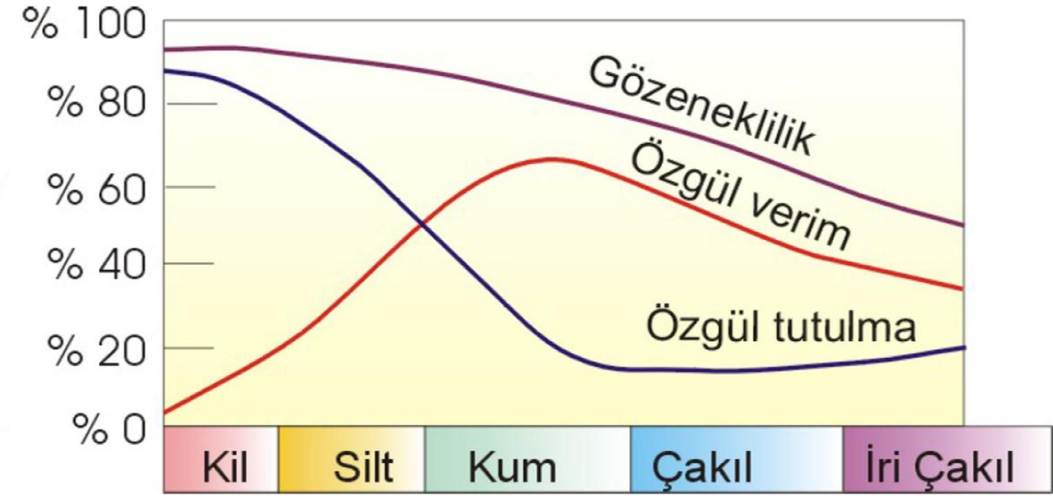
Granit, Şist, Diyabaz vbg.

Hidrojeolojik Modül:

Kısmi Su Döngüsü



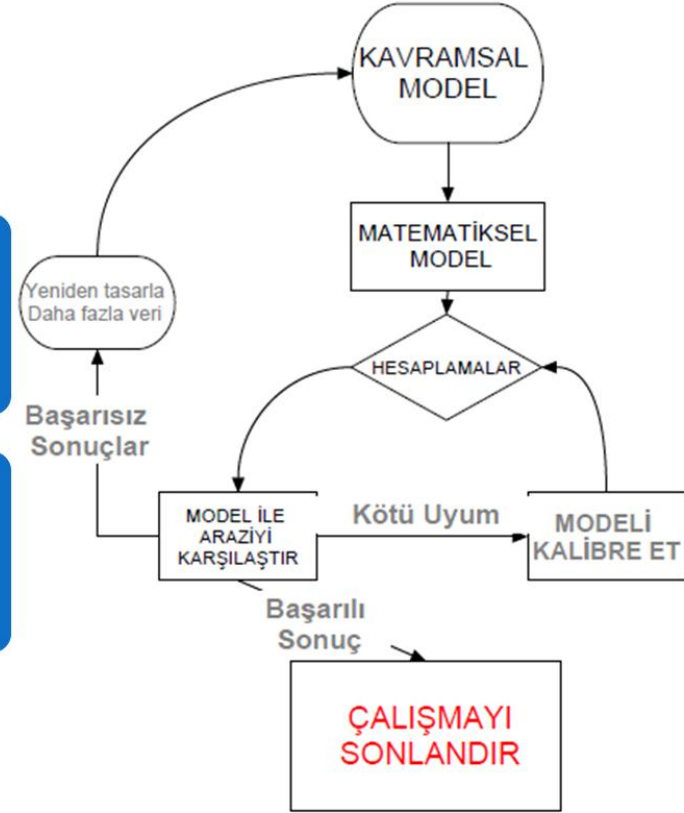
Suyu olan Yapının Özellikleri





Hidrojeolojik Modül:

MODEL AŞAMALARI





Hidrojeolojik Modül:

▼ TOB : SYGM

▼ MODELLEME PROJESİ

Yeni Proje Aç Kaydet

Aktif Proje : Gediz

Ön İşlem Araçları Akım Sınır Koşulları Taşınım Modeli

Zaman Tanımı Model Gridi Oluştur Çözüm Parametreleri

Su Kalitesi
Hidrojeolojik (Su Kutlesi)
Hidrodinamik
Hidrojeoloji
Model dis
Referans Akifer
Gediz_kuyular
Nehir
Demirkopru
Nehir
yas_yukari_gediz
Havza
Hidrolojik İşlem Birimi
Altlik Haritalar
Google Maps

ARAÇLAR

Ön Hazırlıklar

- Zaman Dilimleri ve Stress Periodları
- Katmanlar ve Sınırlar
- Grid Boyutu ve Yerleşimleri
- Sınır Şartları
- Taşınım ve Kirlilik Modellemesi
- Çözüm Yöntemleri



YAS Denklemleri

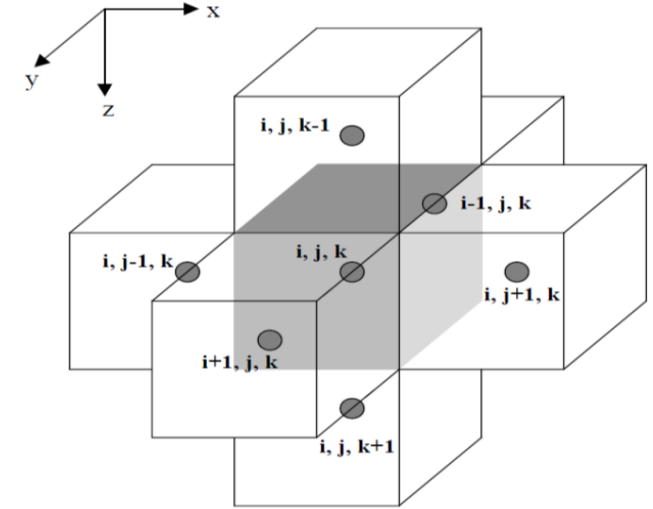
Akım Denklemi

$$\frac{\partial}{\partial x} \left(K_x \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left(K_y \frac{\partial h}{\partial y} \right) + \frac{\partial}{\partial z} \left(K_z \frac{\partial h}{\partial z} \right) = S_s \frac{\partial h}{\partial t}$$

Taşıma Denklemi

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial x} \left(D_{xx} \frac{\partial C}{\partial x} \right) + \frac{\partial}{\partial y} \left(D_{yy} \frac{\partial C}{\partial y} \right) + \frac{\partial}{\partial z} \left(D_{zz} \frac{\partial C}{\partial z} \right) - \frac{\partial}{\partial x} (v_x C) - \frac{\partial}{\partial y} (v_y C) - \frac{\partial}{\partial z} (v_z C)$$

Sonlu-hacim Yöntemi





YAS Modeli ve Denklemler

Girdi →

YAS Sistemi
 $f(x_i, t)$

→ Çıktı

Beslenme
Doğal Boşalım
Yeraltı suyu
Kullanımı

Sistemi Karakterize Eden Parametreler
1- Hidrostratigrafik Yapı (akifer/akitard)
2- Sistemin Geometrisi
3- Sınır Koşulları (İç/Dış)
4- Başlangıç Koşulları
5- Hidrolik Parametreler

Yük
Düşüm
Akış hızı

Kütle

Konsantrasyon



YAS Denklemleri

Sonlu-Hacim Yöntemiyle YAS Akış- Denklem Takımlarının Elde Edilmesi

$$\begin{aligned} & CN_{i-1/2,j,k}^{m+1} \left(\frac{H_{i,j,k}^{m+1} - H_{i-1,j,k}^{m+1}}{dx_i} \right) - CN_{i+1/2,j,k}^{m+1} \left(\frac{H_{i+1,j,k}^{m+1} - H_{i,j,k}^{m+1}}{dx_i} \right) \\ & + CN_{i,j-1/2,k}^{m+1} \left(\frac{H_{i,j,k}^{m+1} - H_{i,j-1,k}^{m+1}}{dy_j} \right) - CN_{i,j+1/2,k}^{m+1} \left(\frac{H_{i,j+1,k}^{m+1} - H_{i,j,k}^{m+1}}{dy_j} \right) \\ & + CN_{i,j,k-1/2}^{m+1} \left(\frac{H_{i,j,k}^{m+1} - H_{i,j,k-1}^{m+1}}{dz_k} \right) - CN_{i,j,k+1/2}^{m+1} \left(\frac{H_{i,j,k+1}^{m+1} - H_{i,j,k}^{m+1}}{dz_k} \right) \\ & + \frac{(Q_e)_{i,j,k}^{m+1}}{dx_i dy_j dz_k} = SS_{i,j,k}^{m+1} \frac{H_{i,j,k}^{m+1} - H_{i,j,k}^m}{t^{m+1} - t^m} \end{aligned}$$

$$[A][H]=[RHS]$$



HİDROYAS Sınır Şartları

Ön İşlem Araçları

Akım Sınır Koşulları

Taşıyım Modeli



Katman İçİ Özellikleri



Besleme



Kuyu



Sabit Su Seviyesi



Genel Su Seviyesi



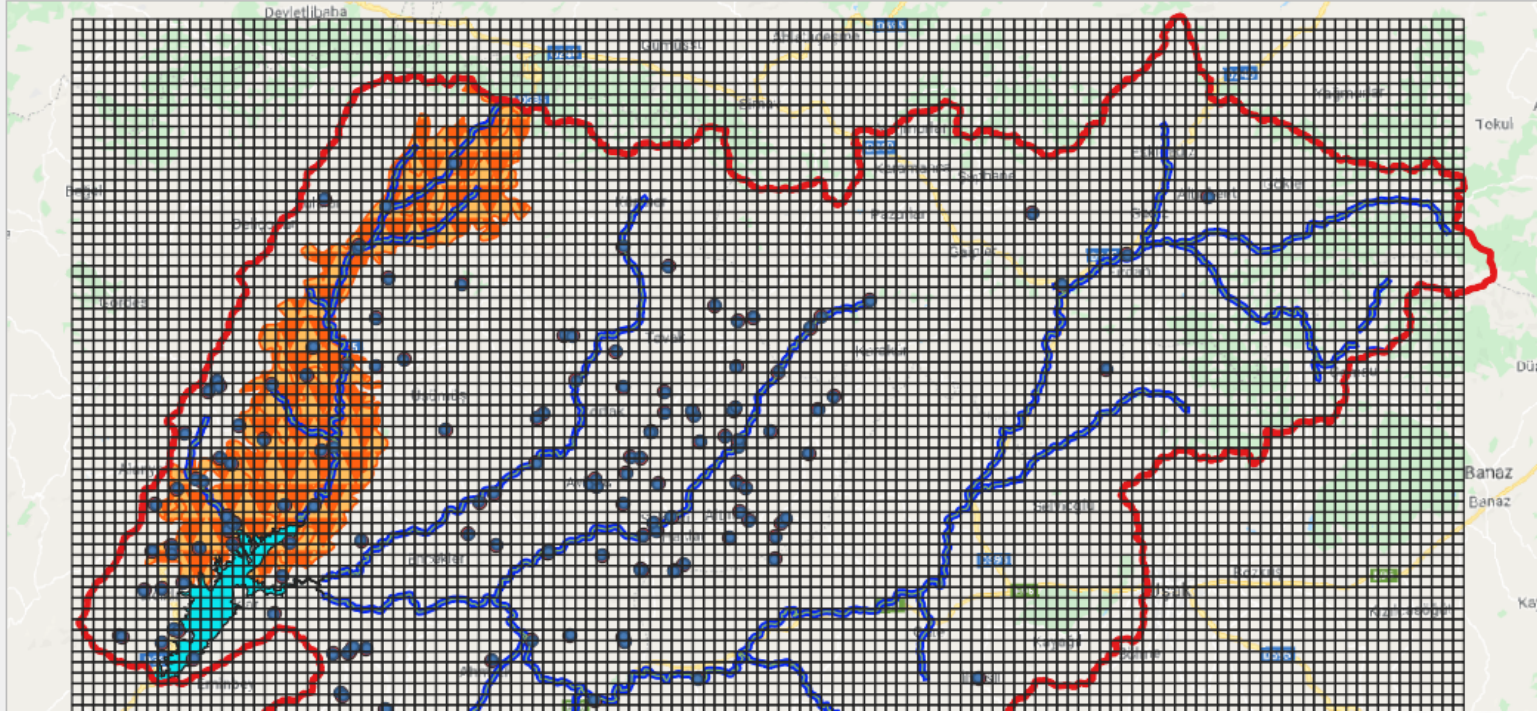
Pınar



Buharlařma

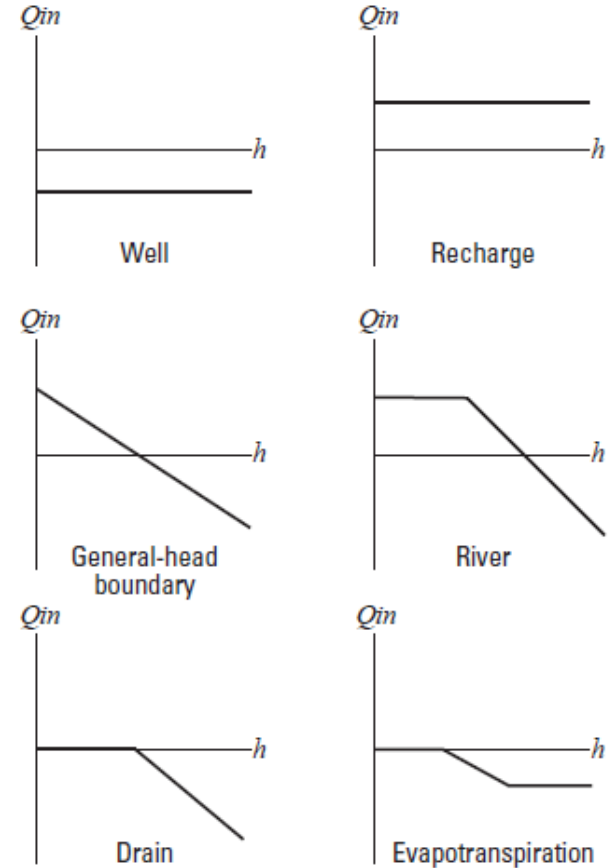


Nehir



$$Q_e = -C(H_{ext} - H) + Q$$

Modele giren veya çıkan sınır şartına baęlı akımlar





Denklem Setinin (Matris) Çözümü: Preconditioned Conjugate Gradient Method {PCGM}

- $[A] \{x\} = \{b\}$ burada $[A]$ $n \times n$ simetrik, pozitif-belirli matristir
- $A = M + N$ burada $[M]$ Jacobi preconditionin matris olup $[A]$ 'nın köşegen elemanlarından elde edilir.
- $Mx^{k+1} = Mx^k + b - Ax^k$
- Burada k iterasyon indisidir. Eğer ilk denklem takımında $(b - Ax^k)$ residual r^k ise, k^{th} iterasyon adımında $s^k = x^{k+1} - x^k$ olur
- Yeni x değerleri $x^{k+1} = x^k + s^k$ şeklinde hesaplanır.
- $Ms^k = r^k \Rightarrow s^k = M^{-1}r^k$

PCGM için aşağıdaki algoritma takip edilir:

$$r^0 = b - Ax^0$$

$$s^k = M^{-1}r^k$$

$$p^k = \begin{cases} s^k & \text{for } k = 0 \\ s^k + \beta^k p^{k-1} & \text{for } k > 0 \end{cases}$$

$$x^{k+1} = x^k + \alpha^k p^k$$

$$r^{k+1} = r^k - \alpha^k A p^k$$

burada

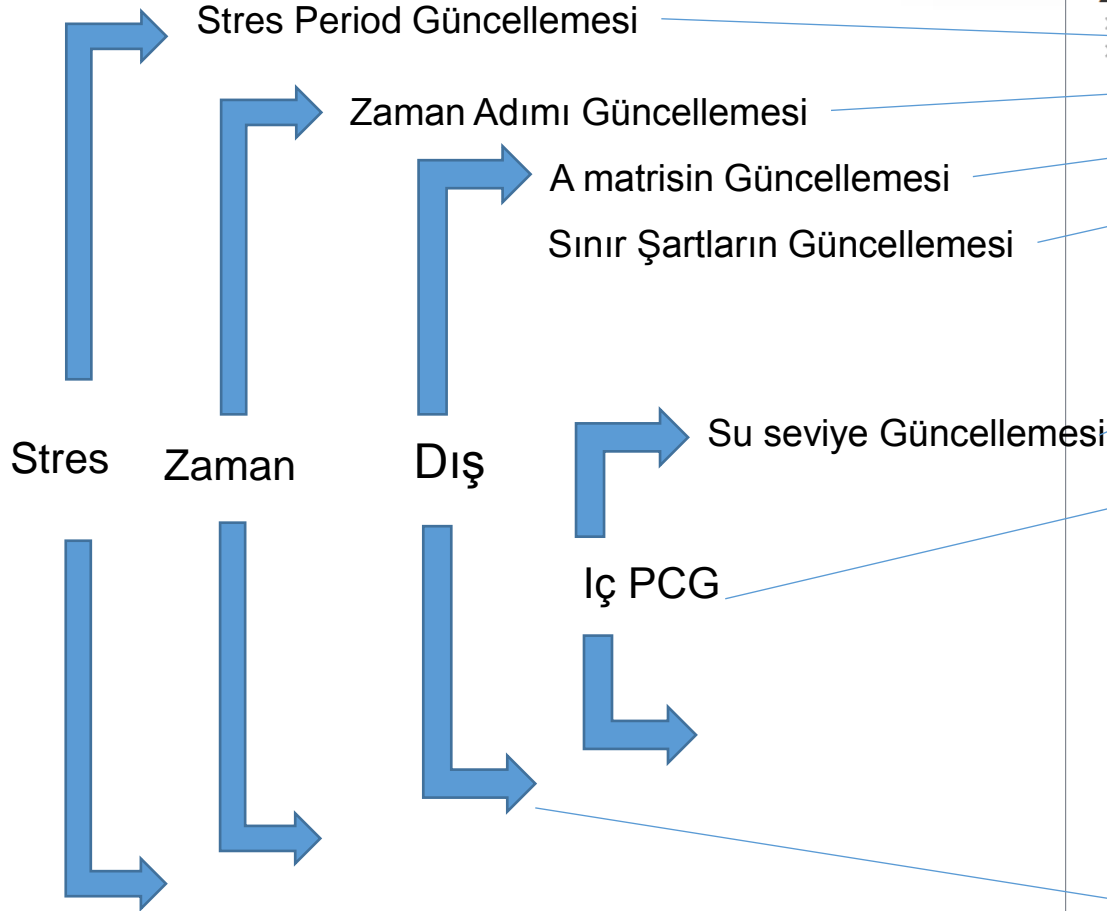
$$\beta^k = \frac{\{s^k\}^T r^k}{\{s^{k-1}\}^T r^{k-1}}$$

$$\alpha^k = \frac{\{s^k\}^T r^k}{\{p^k\}^T A p^k}$$



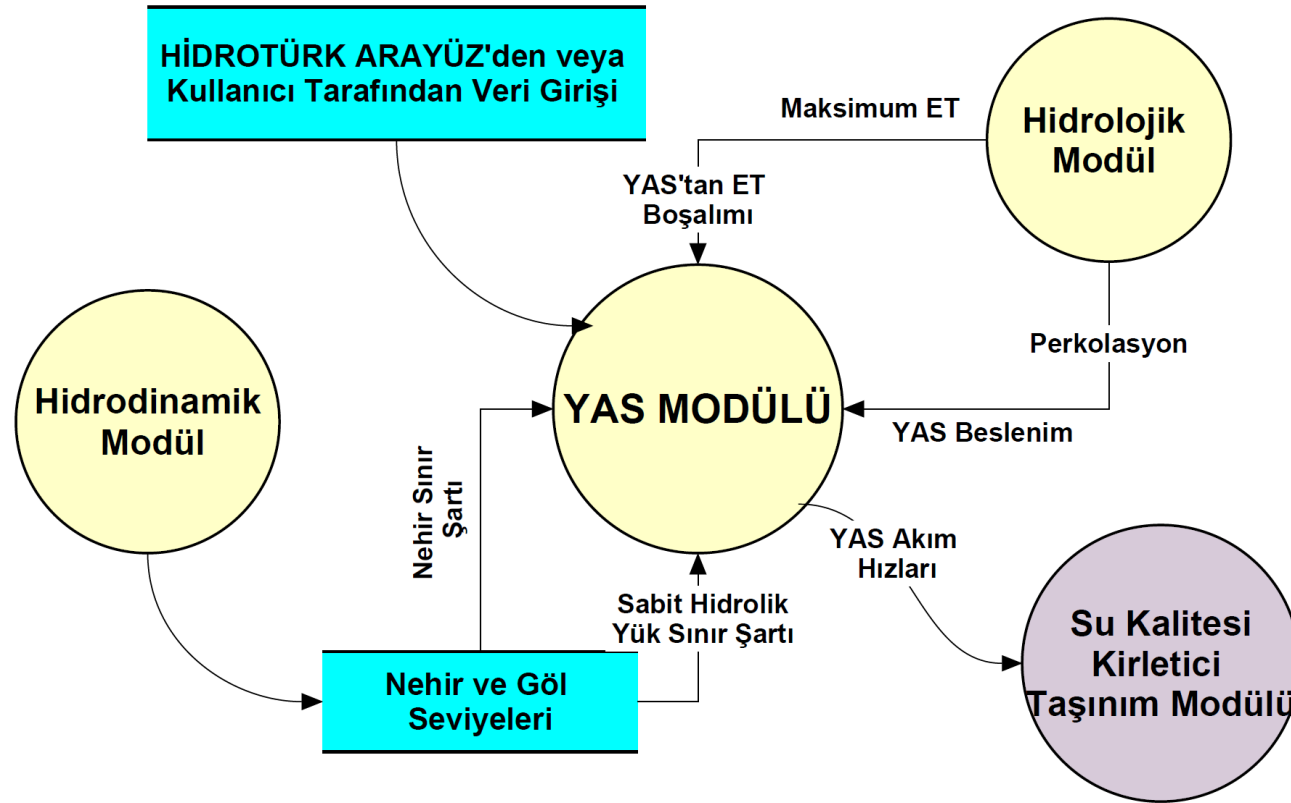
Akış Yazılımı

```
D:\HİDROTURK\HİDROTURK_HİDROYAS\YAS30102019 - Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\HİDROTURK\HİDROTURK_HI
Project explorer Editor - D:\HİDROTURK\HİDROTURK_HİDROYAS\YAS30102019\AKIM.py
AKIM.py*
76 ZBOT=(YASIN['ZBOT'].values+0.0).reshape(SHP)
77 for KPER in range(1,MPER+1):
78     for KSTP in range(1,KSTP1[KPER-1]+1):
79         for OITER in range(1,O_ITER+1):
80             P1,P2,RHS,QEXIN,QEXOUT,QXCEL[KTS-1],XXM1,Adiag,A,Cx,Cy,Cz,dzh
81             dzh=(dzh).reshape(NUMCEL)
82             QEX[KTS-1,:,0]=QEXIN.sum(axis = 1, skipna = True)
83             QEX[KTS-1,:,1]=QEXOUT.sum(axis = 1, skipna = True)
84             QEX[KTS-1,:,2]=QEX[KTS-1,:,0]+QEX[KTS-1,:,1]
85 #
86             RHS=(RHS+P2*XXK[:,O_ITER-1])
87             if "SABITH" in SINVER:
88                 [I,J,K,CEL_SHS,HXS]=SINVER['SABITH']
89                 for ii in range(len(CEL_SHS)):
90                     XXX[CEL_SHS[ii]-1,O_ITER-1]= HXS[ii,KPER-1]
91                     A[CEL_SHS[ii]-1,CEL_SHS[ii]-1]=1
92                     RHS[CEL_SHS[ii]-1]=HXS[ii,KPER-1]
93             XX1=XXK[:,O_ITER-1]
94             XX2,IITER,STROLD,NJ[O_ITER],inCong=YASCOZ(SHP,IBOUND,SINVER,A,
95             if inCong==0:
96                 if O_ITER==1:
97                     print("Solution did not coverage")
98                     XXX[:,O_ITER]=XX2
99                     break
100                 else:
101                     DELTA[:,O_ITER]=XX2-XX1
102                     DAMPC[O_ITER]=DAMP(M(COZ,Hdif_max,NJ,O_ITER,DAMPC,CHGLim
103                     DAMPC[O_ITER]=0.4
104                     XXX[:,O_ITER]=XXX[:,O_ITER-1]+DAMPC[O_ITER]*DELTA[:,OITE
105                     if O_ITER==O_ITER:
106                         print("Solution did not coverage")
107                         XXX[:,O_ITER]=XX2
108                         break
109                 else:
110                     if O_ITER==1:
111                         print("Solution covered")
112                         XXX[:,O_ITER]=XX2
113                     elif IITER==1:
114                         print("Solution covered")
115                     else:
116                         XXX[:,O_ITER]=XX2
117                         DELTA[:,O_ITER]=XX2-XX1
118                         DAMPC[O_ITER]=DAMP(M(COZ,Hdif_max,NJ,O_ITER,DAMPC,CHGLim
```





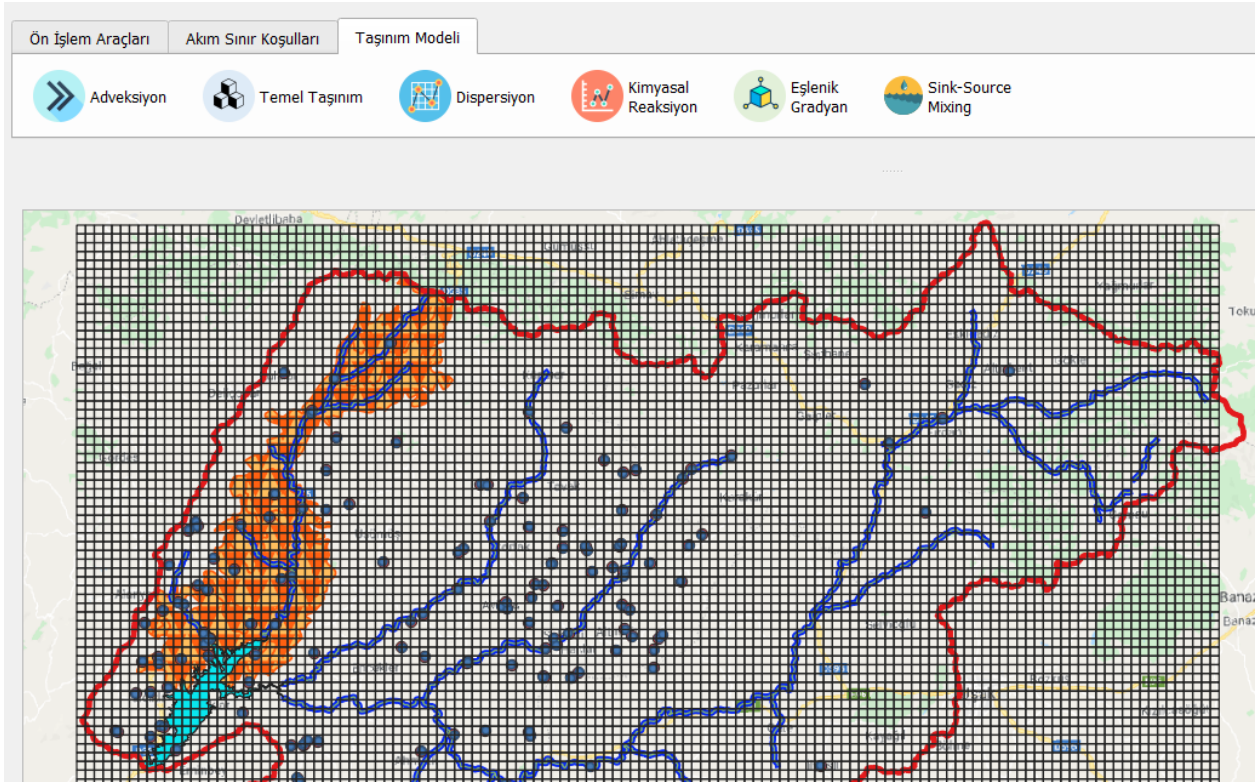
Akış Yazılımı-Veri Alışverişi ve İntegrasyonu





TAŞIMA Modeli

$$R \frac{\partial(\theta C)}{\partial t} = \frac{\partial}{\partial x_i} \left(\theta D_{ij} \frac{\partial C}{\partial x_j} \right) - \frac{\partial}{\partial x_i} (q_i C) + q_s C_s + \sum_{k=1}^N R_k$$

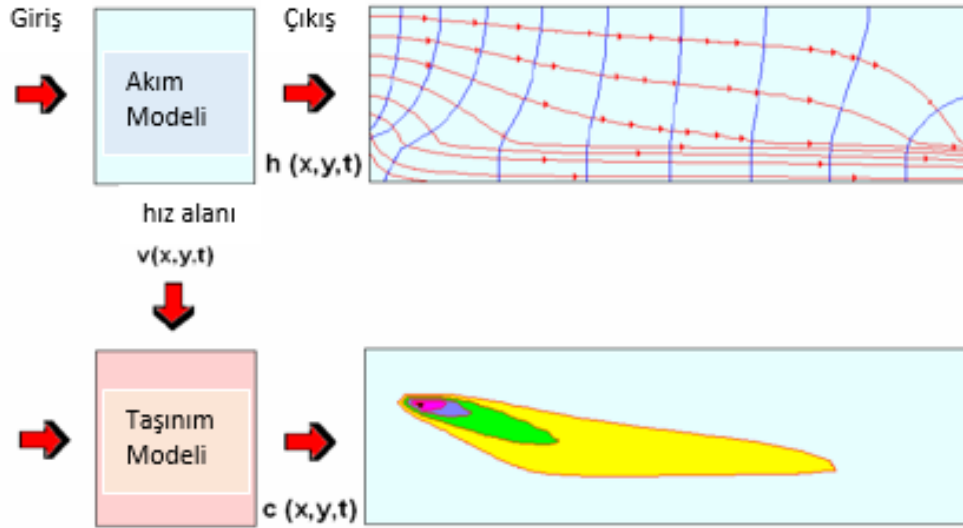


```
D:\HİDROTURK\HİDROTURK_HİDROYAS\YAS30102019 - Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
Project explorer Editor - D:\HİDROTURK\HİDROTURK_HİDROYAS\YAS30102019\TASIM.py
YAS30102019
  _pycache_
  YASIN
  ADV.py
  AKIM.py
  AMAT.py
  Balance.py
  Balance.xlsx
  BTN.py
  CRP.py
  DAMPING.py
  DSP.py
  FTL.py
  GCG.py
  HİDROYAS.py
  LMT.py
  MATRED.py
  PCG.py
  SINHESAP1.py
  SINIRSART.py
  SSM.py
  TASIM.py
  yasin1.xlsx
AKIM.py* TASIM.py*
3 Created on Sat Sep 28 20:16:59 2019
4 @author: mvano
5 """
6 import numpy as np
7 import pandas as pd
8 from FTL import FTLM
9 from BTN import BTNM
10 from GCG import GCGM
11 def TASIM(nammt, file_in, YASIN, MT, ZAMVER, HEAD, QFCEL, QXCEL,
12 SHP=np.max(YASIN["K"].values), np.max(YASIN["I"].values):
13     namfile="MT3D_"+nammt+".NAM"
14     NAM = open(namfile, 'w')
15     NAM.write('{}\n'.format("# Nam file for MT3DMS linking
16     NAM.write('{:10}' '{} ' {:15}\n'.format("LIST 0 "
17     file=nammt+".btn"
18     NAM.write('{:10}' '{} ' {:15}\n'.format("BTN 0 "
19     NCOMP, MCOMP=BTNM(file, MT, YASIN, ZAMVER)
20     if MT.loc["ADV", "deger"]==1:
21         NAM.write('{:10}' '{} ' {:15}\n'.format("ADV 0
22         file=nammt+".adv"
23         from ADV import ADVM
24         ADVM(file)
25     if MT.loc["DSP", "deger"]==1:
26         NAM.write('{:10}' '{} ' {:15}\n'.format("DSP 0
27         file=nammt+".dsp"
28         from DSP import DSPM
29         DSPM(file, YASIN)
30     if MT.loc["SSM", "deger"]==1:
31         NAM.write('{:10}' '{} ' {:15}\n'.format("SSM 0
32         file=nammt+".ssm"
33         from SSM import SSMM
34 #         SSMM(file, file_in, YASIN, QXCEL)
35     if MT.loc["CRP", "deger"]==1:
36         NAM.write('{:10}' '{} ' {:15}\n'.format("RCT 0
37         file=nammt+".rcm"
38         from CRP import CRPM
39         CRPM(file, file_in, YASIN, ZAMVER, NCOMP, MCOMP)
40     file=nammt+".gcg"
41     NAM.write('{:10}' '{} ' {:15}\n'.format("GCG 0 "
42     GCGM(file, file_in)
43     NAM.write('{:10}' '{} ' {:15}\n'.format("FTL 0 "
44     file=nammt+".ftl"
45     FTLM(file, YASIN, ZAMVER, HEAD, QFCEL, QXCEL)
```




TAŞIMA Yazılım

YAS modeli ile taşınım modeli arasındaki ilişki



```
D:\HİDROTURK\HİDROTURK_HİDROYAS\YAS30102019 - Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
Project explorer Editor - D:\HİDROTURK\HİDROTURK_HİDROYAS\YAS30102019\FTL.py
YAS30102019
  _pycache_
  YASIN
  ADV.py
  AKIM.py
  AMAT.py
  Balance.py
  Balance.xlsx
  BTN.py
  CRP.py
  DAMPING.py
  DSP.py
  FTL.py
  GCG.py
  HİDROYAS.py
  LMT.py
  MATRED.py
  PCG.py
  SINHESAP1.py
  SINIRSART.py
  SSM.py
  TASIM.py
  yasin1.xlsx
AKIM.py*
TASIM.py*
FTL.py*
CRP.py
BTN.py
1 "Created on Thu Nov 22 14:09:02 2018 @author: mvano"
2 " This is Flow Transport Link File (FTL) for MT3D"
3 " LINK-MT3DMS (LMT) PACKAGE FOR YAS-HİDROTURK"
4 " Version support for MT3DMS-530"
5
6 import numpy as np
7 import pandas as pd
8 def FTLM(file,YASIN,ZAMVER,HEAD,QFCEL,QXCCEL):
9     SHP=np.max(YASIN["K"].values),np.max(YASIN["I"].valu
10     FTLf = open(file,'a')
11     NLAY,NROW,NCOL=SHP
12     PERLEN=ZAMVER['PERLENGTH'].values
13     MPER=len(ZAMVER)
14     KTS=1
15     KSTP1=ZAMVER.loc[:, 'KSTP'].values
16     if ZAMVER.at[1, 'DURUM']==0:
17         KSTP1[0]=1
18         MPER=1
19         NTTS=1
20         Dt=np.zeros(NTTS+1)
21         Mtime=np.zeros(NTTS+1)
22         Time_Flag=0# KARARLI AKIM
23         PERLEN=[1]
24         Dt[KTS]=1
25         Mtime[KTS]=Mtime[KTS-1]+Dt[KTS]
26     else:
27         NTTS=sum(KSTP1)
28         Time_Flag=1
29         Dt=np.zeros(NTTS+1)
30         Mtime=np.zeros(NTTS+1)
31         for i in range(1,MPER+1):
32             for j in range(KSTP1[i-1]):
33                 Dt[KTS]=PERLEN[i-1]/KSTP1[i-1]
34                 Mtime[KTS]=Mtime[KTS-1]+Dt[KTS]
35                 KTS+=1
```



TEŞEKKÜRLER

info@itunovatto.com.tr



itunovatto

www.itunovatto.com.tr

itunovatto

