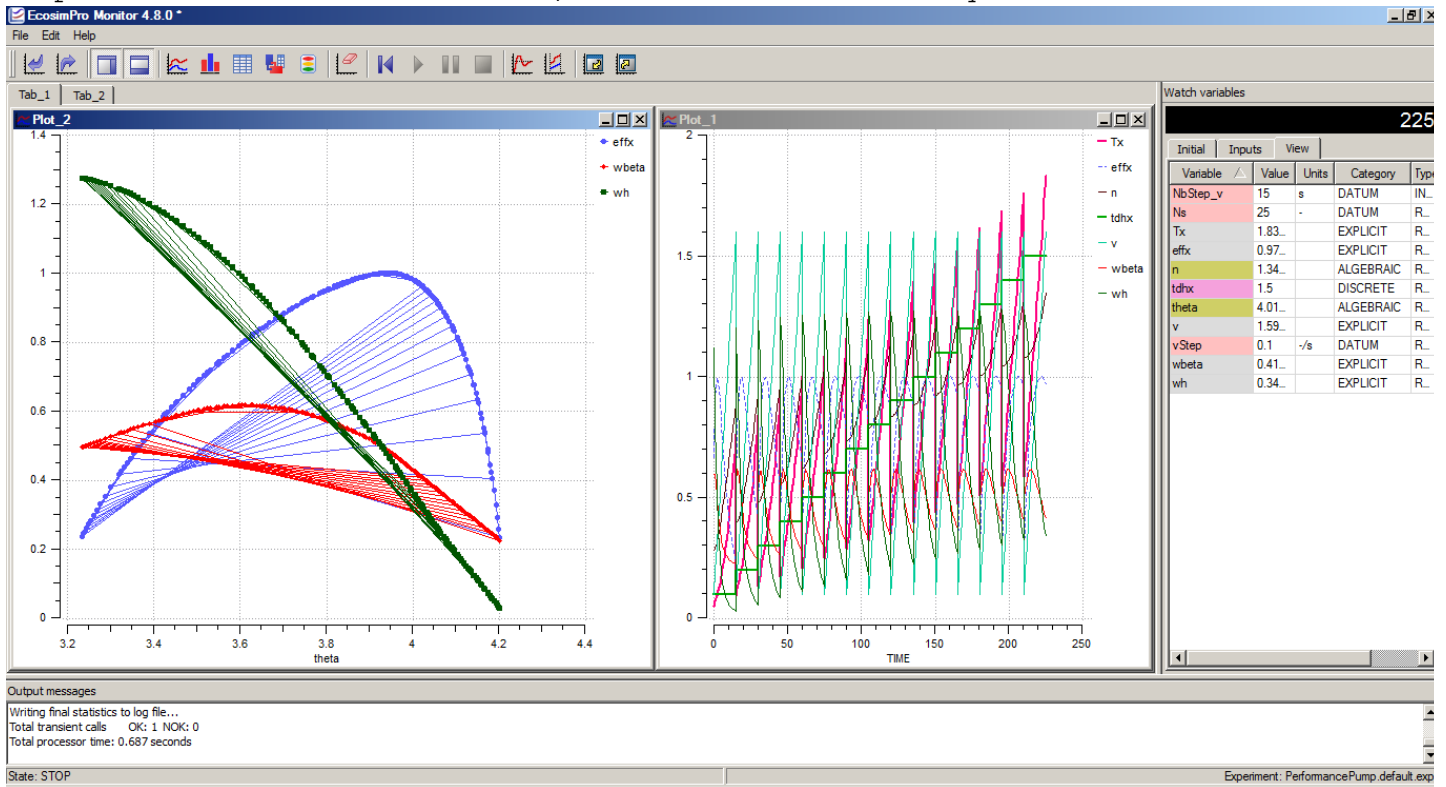
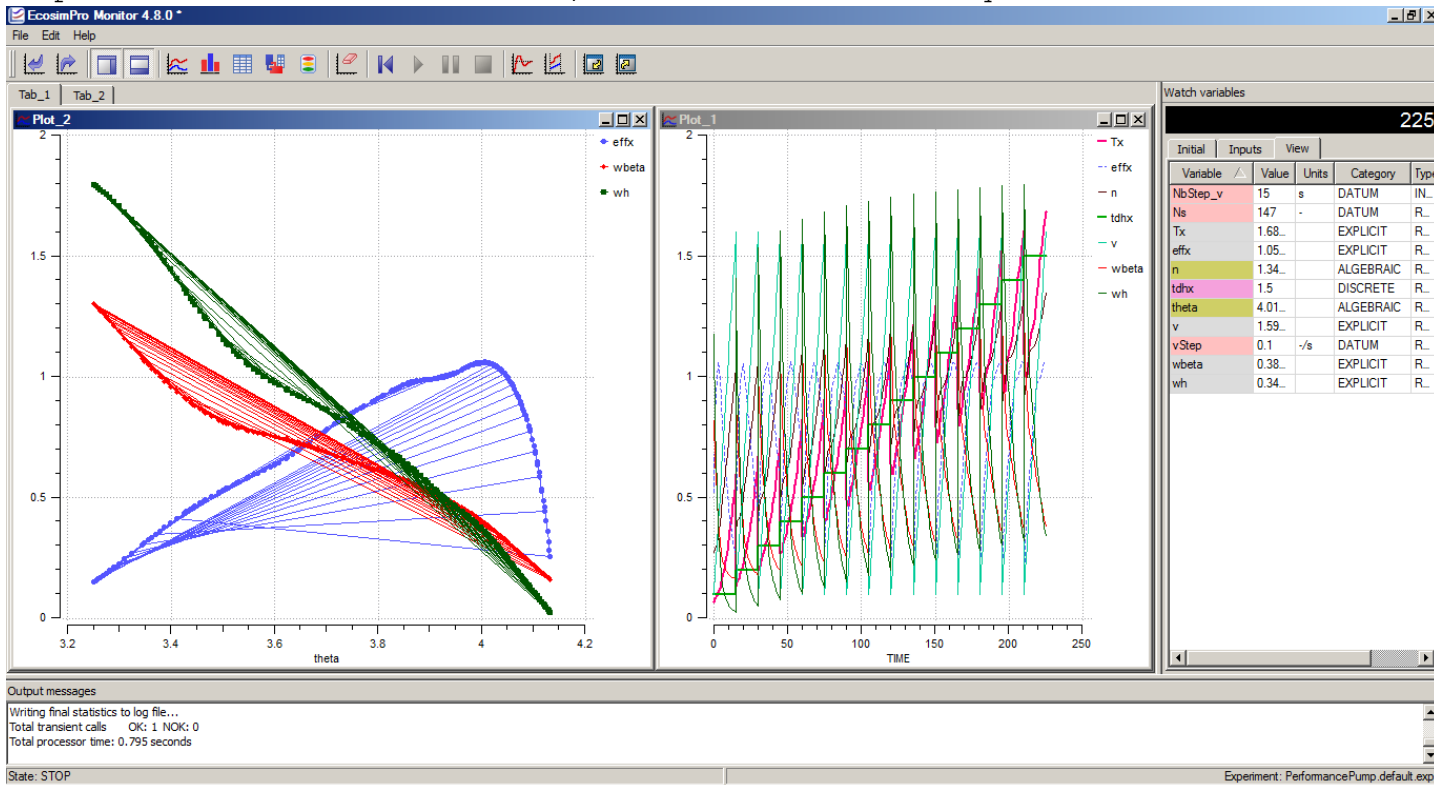


FLUID TRANSIENTS IN SYSTEMS, Pumps

Outputs of the tables for $Ns=25$ wh, wbeta and effx efficiency normalised

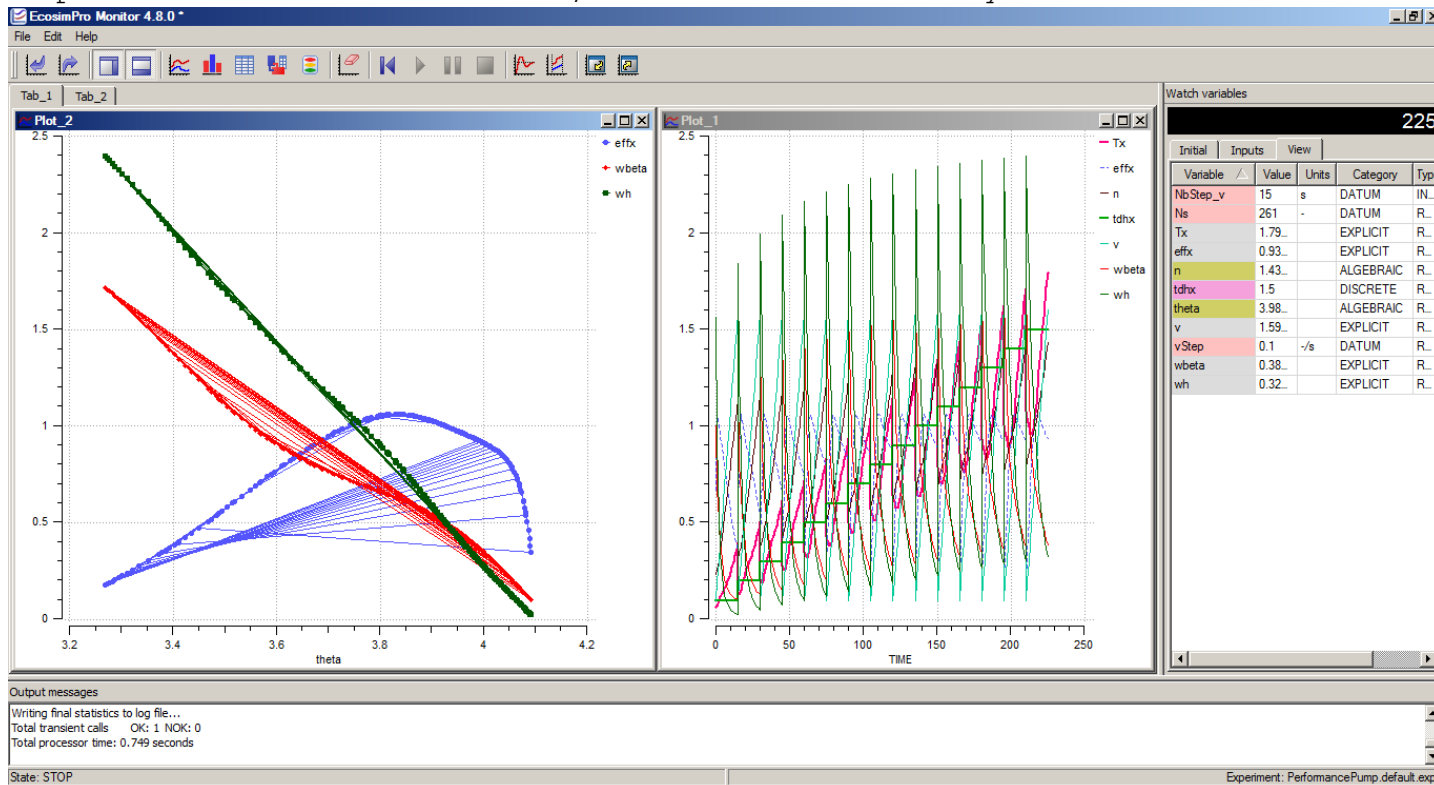


Outputs of the tables for $Ns=147$ wh, wbeta and effx efficiency normalised

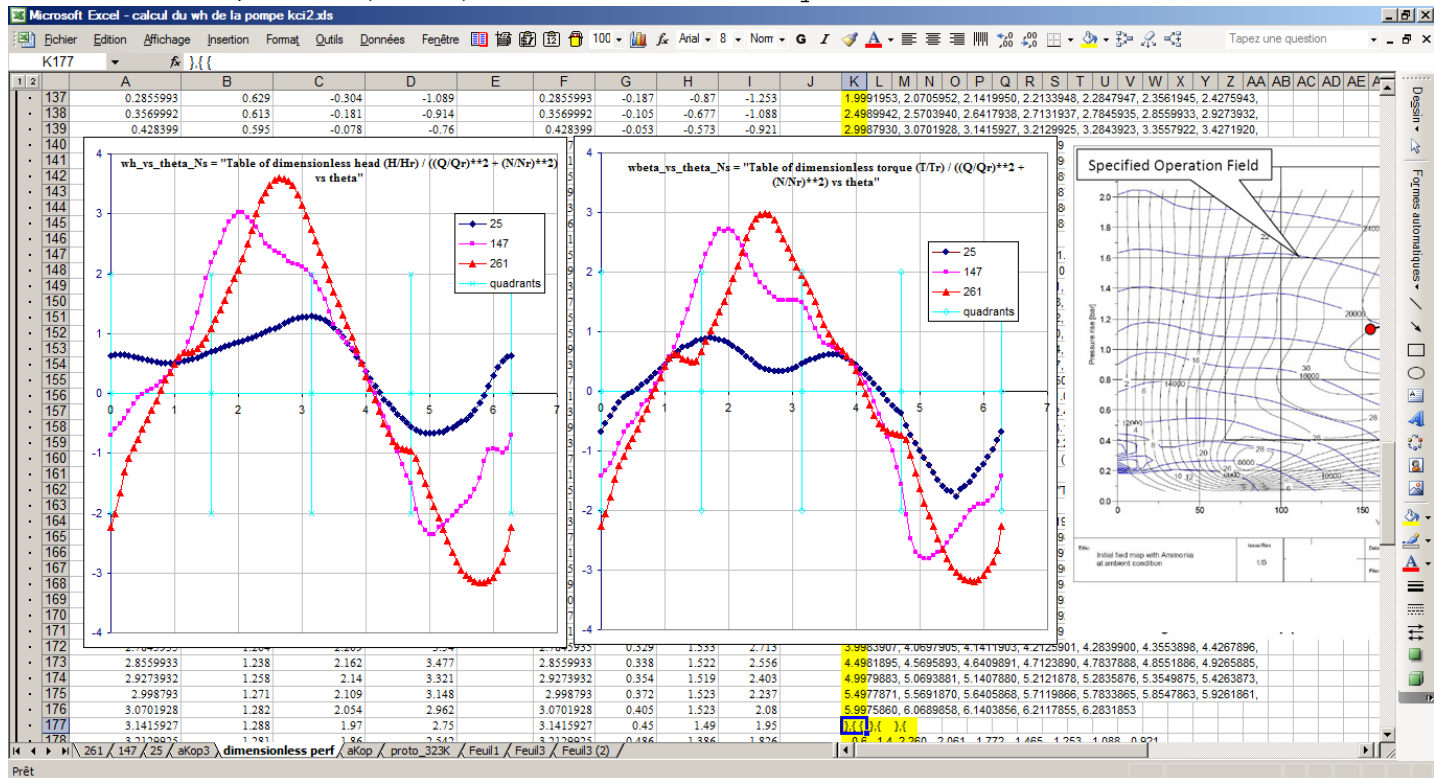




Outputs of the tables for Ns=261 wh, wbeta and effx efficiency normalised



Full curves wh, wbeta (theta) for 3 Ns for each quadrants



USE MATH

FUNCTION NO_TYPE FunctionTablePumps(INTEGER Type,REAL phi, OUT REAL psi, OUT REAL cplus, REAL Ns,REAL theta, OUT REAL wh, OUT REAL wbeta)

Listings 5.15, June 07, Copyright 1992-2007 Support@Kopoo.com, Release 26 juil 2011 21:32:24

```
--torque = cplus*rho*sh_in.omega**2
--dP=rho*sh_in.omega**2*psi
---- for Type=2 Calculation of wh, wbeta with v reduced flow; n reduced speed
--torque = torque_r * (n**2+v**2) * wbeta
```

```
-- tdx = tdx_r * (n**2+v**2) * wh
--call with FunctionTablePumps( Type, phi, psi, cplus, Ns, theta, wh, wbeta)
```

DECLS

```
TABLE_1D psi_vs_phi = {{-0.0082,-0.000114651,-5.70322E-05,-3.76948E-05,-2.79266E-05,-2.19851E-05,-1.79555E-05,-1.50172E-05,-1.27594E-05,-1.09539E-05,-9.4633E-06,-8.2E-06,-7.10534E-06,-6.13844E-06,
```



```

-5.26982E-06,-4.47754E-06,-3.74481E-06,-3.05844E-06,-2.40774E-
06,-1.7838E-06,-1.17898E-06,
-5.86476E-07,3.80564E-13,5.86476E-07,1.17898E-06,1.7838E-
06,2.40774E-06,3.05844E-06,3.74482E-06,
4.47754E-06,5.26982E-06,6.13844E-06,7.10534E-06,8.2E-
06,9.46331E-06,1.09539E-05,1.27594E-05,1.50172E-05,
1.79555E-05,2.1985E-05,2.79266E-05,3.76948E-05,5.70322E-
05,0.000114651,0.0082},
{2352.802353,0.482346756,0.126407686,0.05884385,0.034611154,0.0
23163111,0.01688478,0.013012038,
0.010515494,0.00880452,0.007603255,0.0067728,0.006113551,0.005
623638,0.005236627,0.004961053,
0.00478711,0.00466308,0.004572103,0.004479606,0.004410733,0.00
4381097,0.0043792,0.004377679,
0.00437256,0.004362097,0.004328356,0.0042874,0.00423649,0.0041
5775,0.004045174,0.003888799,
0.003672893,0.0034,0.002917621,0.002272134,0.001454023,0.00016
2836,-0.002009624,-0.004677166,
-0.010923074,-0.025734776,-0.071009889,-0.330026661,-
1890.40189)} "Dp coeff.: Psi+ = f(Phi+ (m^3)) (m^2)"
TABLE_1D Cplus_vs_phi = {{-0.0082,-0.000114651,-5.70322E-05,-
3.76948E-05,-2.79266E-05,-2.19851E-05,
-1.79555E-05,-1.50172E-05,-1.27594E-05,-1.09539E-05,-9.4633E-
06,-8.2E-06,-7.10534E-06,-6.13844E-06,
-5.26982E-06,-4.47754E-06,-3.74481E-06,-3.05844E-06,-2.40774E-
06,-1.7838E-06,-1.17898E-06,
-5.86476E-07,3.80564E-13,5.86476E-07,1.17898E-06,1.7838E-
06,2.40774E-06,3.05844E-06,3.74482E-06,
4.47754E-06,5.26982E-06,6.13844E-06,7.10534E-06,8.2E-
06,9.46331E-06,1.09539E-05,1.27594E-05,1.50172E-05,
1.79555E-05,2.1985E-05,2.79266E-05,3.76948E-05,5.70322E-
05,0.000114651,0.0082},
{0.034600035,6.94007E-06,1.74982E-06,7.76381E-07,4.32891E-
07,2.74472E-07,1.8636E-07,1.3201E-07,
9.6205E-08,7.18395E-08,5.43789E-08,4.16E-08,3.17951E-
08,2.54655E-08,2.09126E-08,1.78108E-08,
1.60013E-08,1.49907E-08,1.46856E-08,1.48301E-08,1.51876E-
08,1.62829E-08,0.000000018,1.95394E-08,
2.123E-08,2.31249E-08,2.51568E-08,2.74754E-08,2.97789E-
08,3.20386E-08,3.42514E-08,3.63258E-08,
3.82381E-08,4E-08,4.02945E-08,4.00965E-08,3.94126E-
08,3.72693E-08,2.85102E-08,1.21187E-08,
-2.67092E-08,-1.42528E-07,-4.89791E-07,-2.46793E-06,-
0.014880015)} "Torque coeff.: C+ = f(Phi+ (m^3)) (m^5)"
CONST TABLE_2D wh_vs_theta_Ns = \
{{ 25, 147, 261 },
{ 0.0000000, 0.0713998, 0.1427997, 0.2141995, 0.2855993,
0.3569992, 0.4283990,
0.4997988, 0.5711987, 0.6425985, 0.7139983, 0.7853982,
0.8567980, 0.9281978,
0.9995977, 1.0709975, 1.1423973, 1.2137972, 1.2851970,
1.3565968, 1.4279967,
1.4993965, 1.5707963, 1.6421962, 1.7135960, 1.7849958,
1.8563957, 1.9277955,
1.9991953, 2.0705952, 2.1419950, 2.2133948, 2.2847947,
2.3561945, 2.4275943,
2.4989942, 2.5703940, 2.6417938, 2.7131937, 2.7845935,
2.8559933, 2.9273932,
2.9987930, 3.0701928, 3.1415927, 3.2129925, 3.2843923,
3.3557922, 3.4271920,
3.4985918, 3.5699917, 3.6413915, 3.7127913, 3.7841912,
3.8555910, 3.9269908,
3.9983907, 4.0697905, 4.1411903, 4.2125901, 4.2839900,
4.3553898, 4.4267896,
4.4981895, 4.5695893, 4.6409891, 4.7123890, 4.7837888,
4.8551886, 4.9265885,
4.9979883, 5.0693881, 5.1407880, 5.2121878, 5.2835876,
5.3549875, 5.4263873,
5.4977871, 5.5691870, 5.6405868, 5.7119866, 5.7833865,
5.8547863, 5.9261861,
5.9975860, 6.0689858, 6.1403856, 6.2117855, 6.2831853
},
{{ 0.634, 0.643, 0.646, 0.640, 0.629, 0.613, 0.595,
0.575, 0.552, 0.533, 0.516, 0.505, 0.504, 0.510,
0.512, 0.522, 0.539, 0.559, 0.580, 0.601, 0.630,
0.662, 0.692, 0.722, 0.753, 0.782, 0.808, 0.832,
0.857, 0.879, 0.904, 0.930, 0.959, 0.996, 1.027,
1.060, 1.090, 1.124, 1.165, 1.204, 1.238, 1.258,
1.271, 1.282, 1.288, 1.281, 1.260, 1.225, 1.172,
1.107, 1.031, 0.942, 0.842, 0.733, 0.617, 0.500,
0.368, 0.240, 0.125, 0.011, -0.102, -0.168, -0.255,
-0.342, -0.423, -0.494, -0.556, -0.620, -0.655, -0.670,
-0.670, -0.660, -0.655, -0.640, -0.600, -0.570, -0.520,
-0.470, -0.430, -0.360, -0.275, -0.160, -0.040, 0.130,

```

```

0.295, 0.430, 0.550, 0.620, 0.634
},
{-0.690, -0.599, -0.512, -0.418, -0.304, -0.181, -0.078,
-0.011, 0.032, 0.074, 0.130, 0.190, 0.265, 0.363,
0.461, 0.553, 0.674, 0.848, 1.075, 1.337, 1.629,
1.929, 2.180, 2.334, 2.518, 2.726, 2.863, 2.948,
3.026, 3.015, 2.927, 2.873, 2.771, 2.640, 2.497,
2.441, 2.378, 2.336, 2.288, 2.209, 2.162, 2.140,
2.109, 2.054, 1.970, 1.860, 1.735, 1.571, 1.357,
1.157, 1.016, 0.927, 0.846, 0.744, 0.640, 0.500,
0.374, 0.191, 0.001, -0.190, -0.384, -0.585, -0.786,
-0.972, -1.185, -1.372, -1.500, -1.940, -2.160, -2.290,
-2.350, -2.350, -2.230, -2.200, -2.130, -2.050, -1.970,
-1.895, -1.810, -1.730, -1.600, -1.420, -1.130, -0.950,
-0.930, -0.950, -1.000, -0.920, -0.690
},
{-2.230, -2.000, -1.662, -1.314, -1.089, -0.914, -0.760,
-0.601, -0.440, -0.284, -0.130, 0.055, 0.222, 0.357,
0.493, 0.616, 0.675, 0.680, 0.691, 0.752, 0.825,
0.930, 1.080, 1.236, 1.389, 1.548, 1.727, 1.919,
2.066, 2.252, 2.490, 2.727, 3.002, 3.225, 3.355,
3.475, 3.562, 3.604, 3.582, 3.540, 3.477, 3.321,
3.148, 2.962, 2.750, 2.542, 2.354, 2.149, 1.909,
1.702, 1.506, 1.310, 1.131, 0.947, 0.737, 0.500,
0.279, 0.082, -0.112, -0.300, -0.505, -0.672, -0.797,
-0.872, -0.920, -0.949, -0.960, -1.080, -1.300, -1.500,
-1.700, -1.890, -2.080, -2.270, -2.470, -2.650, -2.810,
-2.950, -3.040, -3.100, -3.150, -3.170, -3.170, -3.130,
-3.070, -2.960, -2.820, -2.590, -2.230
}
} "Table of dimensionless head (H/Hr) / ((Q/Qr)**2 + (N/Nr)**2) vs
theta"
CONST TABLE_2D wbeta_vs_theta_Ns = \
{{ 25, 147, 261 },
{ 0.0000000, 0.0713998, 0.1427997, 0.2141995, 0.2855993,
0.3569992, 0.4283990,
0.4997988, 0.5711987, 0.6425985, 0.7139983, 0.7853982,
0.8567980, 0.9281978,
0.9995977, 1.0709975, 1.1423973, 1.2137972, 1.2851970,
1.3565968, 1.4279967,
1.4993965, 1.5707963, 1.6421962, 1.7135960, 1.7849958,
1.8563957, 1.9277955,
1.9991953, 2.0705952, 2.1419950, 2.2133948, 2.2847947,
2.3561945, 2.4275943,
2.4989942, 2.5703940, 2.6417938, 2.7131937, 2.7845935,
2.8559933, 2.9273932,
2.9987930, 3.0701928, 3.1415927, 3.2129925, 3.2843923,
3.3557922, 3.4271920,
3.4985918, 3.5699917, 3.6413915, 3.7127913, 3.7841912,
3.8555910, 3.9269908,
3.9983907, 4.0697905, 4.1411903, 4.2125901, 4.2839900,
4.3553898, 4.4267896,
4.4981895, 4.5695893, 4.6409891, 4.7123890, 4.7837888,
4.8551886, 4.9265885,
4.9979883, 5.0693881, 5.1407880, 5.2121878, 5.2835876,
5.3549875, 5.4263873,
5.4977871, 5.5691870, 5.6405868, 5.7119866, 5.7833865,
5.8547863, 5.9261861,
5.9975860, 6.0689858, 6.1403856, 6.2117855, 6.2831853
},
{{ -0.684, -0.547, -0.414, -0.292, -0.187, -0.105, -0.053,
-0.012, 0.042, 0.097, 0.156, 0.227, 0.300, 0.371,
0.444, 0.522, 0.596, 0.672, 0.738, 0.763, 0.797,
0.837, 0.865, 0.883, 0.886, 0.877, 0.859, 0.838,
0.804, 0.758, 0.703, 0.645, 0.583, 0.520, 0.454,
0.408, 0.370, 0.343, 0.331, 0.329, 0.338, 0.354,
0.372, 0.405, 0.450, 0.486, 0.520, 0.552, 0.579,
0.603, 0.616, 0.617, 0.606, 0.582, 0.546, 0.500,
0.432, 0.360, 0.288, 0.214, 0.123, 0.037, -0.053,
-0.161, -0.248, -0.314, -0.372, -0.580, -0.740, -0.880,
-1.000, -1.120, -1.250, -1.370, -1.490, -1.590, -1.660,
-1.690, -1.770, -1.650, -1.590, -1.520, -1.420, -1.320,
-1.230, -1.100, -0.980, -0.820, -0.684
},
{-1.420, -1.328, -1.211, -1.056, -0.870, -0.677, -0.573,
-0.518, -0.380, -0.232, -0.160, 0.000, 0.118, 0.308,
0.442, 0.574, 0.739, 0.929, 1.147, 1.370, 1.599,
1.839, 2.080, 2.300, 2.480, 2.630, 2.724, 2.687,
2.715, 2.688, 2.555, 2.434, 2.288, 2.110, 1.948,
1.825, 1.732, 1.644, 1.576, 1.533, 1.522, 1.519,
1.523, 1.523, 1.490, 1.386, 1.223, 1.048, 0.909,
0.814, 0.766, 0.734, 0.678, 0.624, 0.570, 0.500,
0.407, 0.278, 0.146, 0.023, -0.175, -0.379, -0.585,
-0.778, -1.008, -1.277, -1.560, -2.070, -2.480, -2.700,
-2.770, -2.800, -2.800, -2.760, -2.710, -2.640, -2.540,
-2.440, -2.340, -2.240, -2.120, -2.000, -1.940, -1.900,
-1.900, -1.850, -1.750, -1.630, -1.420
}
},

```



```

{
-2.260, -2.061, -1.772, -1.465, -1.253, -1.088, -0.921,
-0.789, -0.632, -0.457, -0.300, -0.075, 0.052, 0.234,
0.425, 0.558, 0.630, 0.621, 0.546, 0.525, 0.488,
0.512, 0.660, 0.850, 1.014, 1.162, 1.334, 1.512,
1.683, 1.886, 2.105, 2.325, 2.580, 2.770, 2.886,
2.959, 2.979, 2.962, 2.877, 2.713, 2.556, 2.403,
2.237, 2.080, 1.950, 1.826, 1.681, 1.503, 1.301,
1.115, 0.960, 0.840, 0.750, 0.677, 0.604, 0.500,
0.352, 0.161, -0.040, -0.225, -0.403, -0.545, -0.610,
-0.662, -0.699, -0.719, -0.730, -0.810, -1.070, -1.360,
-1.640, -1.880, -2.080, -2.270, -2.470, -2.650, -2.810,
-2.950, -3.040, -3.100, -3.150, -3.170, -3.200, -3.160,
-3.090, -2.990, -2.860, -2.660, -2.260
}
}

```

COMPONENT PerformancePump**PORTS****DATA**

```

REAL Ns=25 "Specific velocity 25 to 261
(-)"
INTEGER Type=2 "2-Calculation of wh,
wbeta with v reduced flow; n reduced speed
(-)"
REAL tdxStep=0.1 "increment on tdx for
each loop (-)"
REAL vStep=0.1 "increment on v for 1
second (-/s)"
INTEGER NbStep_v=15 "total number of
vStep for one value of tdx i.e. duration of
one loop in s (s)"

```

DECLS

```

REAL phi, psi, cplus
REAL wh, wbeta
ALG REAL theta=4
REAL v, n, Tx, effx, nn

```

```

DISCR REAL tdx=0
DISCR REAL dvdt =0
DISCR REAL t0=0
INTEGER i=0

```

TOPOLOGY**INIT****DISCRETE**

```

WHEN TIME==i*NbStep_v THEN -- loop
on tdx, discrete condition every whole period (s)

```

```

i=i+1

```

```

tdx=tdx+tdxStep -- increment tdx

```

```

dvdt=vStep --slope = 1 step per second

```

```

t0=TIME --to reset the local time: TIME-t0

```

```

n=0

```

```

theta =4

```

```

PRINT ("loop tdx done")

```

END WHEN**CONTINUOUS**

```

v= (TIME-t0)*dvdt + vStep --loop on v, volume
flow periodic

```

```

} "Table of dimensionless torque (T/Tr) / ((Q/QR)**2 + (N/Nr)**2) vs
theta"

```

BODY**IF Type==1 THEN**

```

-- Calculation of Psi+

```

```

psi = linearInterp1D(psi_vs_phi,phi)

```

```

-- Calculation of C+

```

```

cplus = linearInterp1D(Cplus_vs_phi,phi)

```

```

ELSEIF Type==2 THEN

```

```

wh = splineInterp2D(wh_vs_theta_Ns, Ns, theta)

```

```

wbeta = splineInterp2D(wbeta_vs_theta_Ns, Ns, theta)

```

END IF**END FUNCTION**

```

n**2+v**2= tdx/wh --tdx=(n**2+v**2)*wh tdx
normalised

```

```

n**2+v**2= Tx/wbeta --Tx=(n**2+v**2)*wbeta T
normalised

```

```

nn=abs(n)

```

```

effx=tdx*v/(nn*Tx) --efficiency normalised (to be
multiplied by the efficiency at rated condition)

```

```

theta=PI+atan2(v,nn)

```

```

FunctionTablePumps( Type, phi, psi,
cplus, Ns, theta, wh, wbeta)

```

```

/

```

```

Table of performance in the 2D space Volume flow;

```

```

Pressure rise, i.e. versus v and tdx

```

```

for Type=2 Calculation of wh, wbeta with v reduced flow; n

```

```

reduced speed

```

```

tdhx reduced tdx = tdx/tdh_r ; Tx reduced torque =

```

```

torque/torque_r

```

```

torque = torque_r * (n**2+v**2) * wbeta

```

```

tdh = tdx_r * (n**2+v**2) * wh

```

```

eff=Pressure*volflow/(omega*T) normalised to 1:

```

```

effx=tdhx*v/(n*Tx)

```

```

/

```

END COMPONENT**EXPERIMENT** exp1 ON PerformancePump.default**DECLS****INIT**

```

-- initial values for algebraics

```

```

theta =4

```

BOUNDS

```

-- Set equations for boundaries: boundVar = f(TIME,...)

```

```

phi = 0

```

BODY

```

-- report results in file reportAll.rpt

```

```

REPORT_TABLE("reportAll.rpt", "**")

```

```

Ns=261 --"Specific velocity 25 to 261 (-)" --261 --147 --25

```

```

Type=2 --"2-Calculation of wh, wbeta with v reduced flow; n reduced speed (-)"

```

```

tdxStep=0.1 -- "increment on tdx for each loop (-)"

```

```

vStep=0.1 --"increment on v for 1 second (-/s)"

```

```

NbStep_v=15 --"total number of vStep for one value of tdx i.e. duration of one loop in s (-)"
-- integrate the model 15 seconds and obtain results every 0.1 seconds

```

```

TIME = 0

```

```

TSTOP = 225-1e-6

```

```

CINT = 1

```

```

ABS_ERROR=1E-6

```

```

REL_ERROR=ABS_ERROR

```

```

--REPORT_MODE=IS_STEP --REPORT_MODE=IS_EVENT,IS_CINT,IS_STEP

```

INTEG()**END EXPERIMENT**



FLUID TRANSIENTS IN SYSTEMS
Wylie and Streeter

This disk contains programs referred to in the textbook, FLUID TRANSIENTS IN SYTEMS, by E. B. Wylie and V. L Streeter, Prentice Hall, Inc., 1993.

The codes are presented for the educational benefit of the users of the book. With the exception of GRAF.BAS the programs are written in FORTRAN and are designed to run on an IBM-compatible personal computer using a FORTRAN compiler. It is assumed that the user has a fundamental knowledge and understanding of FORTRAN programming in order to successfully use the codes. It is necessary to generate an executable (.EXE) file to actually run any of the programs included in the disk. This means a compiler is needed to generate an .OBJ code, which then has to be linked with FORTRAN.LIB to produce the .EXE file. All source files are called xxx.FOR, data files are called xxx.DAT, output files are called xxx.OUT, and data for graphing are stored in xxx.GRF.

The objective in producing these programs and in supplying them in this disk is to facilitate understanding of the theory provided in the book. Thus, the codes do not contain error messages, nor are they particularly user friendly. Users are encouraged to modify the programs to suit their needs.

DISCLAIMER

The authors make no warranties as to the contents of the software. Although every effort has been made to ensure that the programs are accurate and reliable, no responsibility is assumed for results generated through the use of these computer codes. That is, no warranty, expressed or implied, is made as to the accuracy of these programs. Neither the authors nor the publishers can be held responsible for any damages suffered as a result of the use of the computer codes.

E. B. Wylie
Department of Civil and Environmental Engineering
The University of Michigan
Ann Arbor, Michigan 48109-2125

June, 1992

.....