# EXAMPLES for Tramonto: A Density Functional Theory code for inhomogeneous fluids at equilibrium or steady state

November 2, 2005

## Contents

# 1 Introduction

This document details a suite of test problems that accompanies the Tramonto code. This test suite is found in a directory Tramonto/Examples. Here we briefly discuss each of the test problems, present some input files and density distributions, and give the principle outputs (adsorptions, free energies, and forces, etc.) for each case. This test suite should be used to test any new additions to the code, and examples of new capabilities should be added to the test suite as they become available. Note that under Tramonto/Examples, the input files are found in the directories Ex_inputs, Ex_surfaces, Ex_crfiles, Ex_polyfiles, and Ex_restarts. The density profile outputs for each case are found in the directory Ex_dens_outputs. If a case fails, using the old density output with a restart may be helpful in locating new bugs.

# 2 1D Atomic Fluids at Equilibrium

## 2.1 1 Component Neutral, Hard Sphere system

### 2.1.1 Case 1Dhn1: Two confining surfaces, high resolution

This case solves for a one-component hard sphere fluid confined by two hard walls at high resolution.

### 2.1.2 Case 1Dhn2: 2 confining surfaces, lower resolution

This case repeats case 1 at a lower resolution. $Esize\_x[0] = 0.1$. Principle output for cases 2-11 are given in the table below.

### 2.1.3 Case 1Dhn3: 2 confining surfaces, reflective boundary

This case repeats case 2, but in a domain with a reflective boundary on the right side and a total domain size of $3\sigma$.

### 2.1.4 Case 1Dhn4: 2 confining surfaces, opposite reflective boundary

This case is identical to case 3, but exchanges the left and right boundaries.

### 2.1.5 Case 1Dhn5: 1 surface, bulk boundary condition, Matrix_fill_flag=0

This case is a $10\sigma$ domain with a wall at the left side and a bulk fluid at the right side. It has Matrix_fill_flag=0.

### 2.1.6 Case 1Dhn6: 1 surface, bulk boundary, 4 zones

Same as Case 5 with the domain split into 4 zones with the break points at $2\sigma$, $4\sigma$, and $6\sigma$ from the surface.

### 2.1.7 Case 1Dhn7: 1 surface, 4 zones, Coarser Jacobian

Same as Case 10 with the Jacobian integrals always based on a mesh of $0.2\sigma$.

| Case | niter | Ex. Ads | Total Ads | Surf. Free Energy | Force |
|---|---|---|---|---|---|
| 1Dhn1 | 6 | -0.222308 | 3.527692 | 2.287919 | 5.231941867 |
| 1Dhn2 | 6 | -0.224384 | 3.525616 | 2.393064 | 5.022083086 |
| 1Dhn3 | 5 | -0.224384 | 3.525616 | 2.393064 | 5.022083086 |
| 1Dhn4 | 5 | -0.224384 | 3.525616 | 2.393064 | 5.022083086 |
| 1Dhn5 | 6 | -0.116036 | 7.008964 | 1.198788 | 4.956897154 |
| 1Dhn6 | 6 | -0.115903 | 7.009097 | 1.152352 | 5.165515814 |
| 1Dhn7 | 5 | -0.115903 | 7.009097 | 1.152352 | 5.165517329 |

Table 1: Principle output for 1-dimensional hard sphere/ hard wall cases in the test suite.

## 2.2 1 Component Neutral Lennard-Jones Fluids

### 2.2.1 Case 1Dljn1: LJ Fluid near LJ wall

Dense LJ fluid near a 9-3 LJ wall. (Check the active potential in the dft_potentials.c file if incorrect answers are obtained. Those given here were generated with the cut and shifted 9-3 potential.

### 2.2.2 Case1Dljn2: Test mixing rule

Same as Case 12, but with manual input of wall-fluid parameters.

### 2.2.3 Case 1Dljn3: Test long wall-fluid cutoffs

Same as Case 12, but with wall-fluid cutoff increased from 3 to 20.

### 2.2.4 Case 1Dljn4: Test LJ12-6 integrated walls

Same as Case 14, but does explicit integration of 12-6 potential for external field. Note that the force calculation is not working for this case yet.

| Case | niter | Ex. Ads | Total Ads | Surf. Free Energy | Force |
|---|---|---|---|---|---|
| 1Dljn1 | 6 | -0.178167 | 6.946833 | -2.861495 | 1.625346852 |
| 1Dljn2 | 6 | -0.178167 | 6.946833 | -2.861495 | 1.625346852 |
| 1Dljn3 | 6 | -0.134113 | 6.990887 | -3.578393 | 1.635907510 |
| 1Dljn4 | 6 | -0.137941 | 6.987059 | -3.527798 | 0.000000000 |

Table 2: Principle output for 1-dimensional hard sphere/ hard wall cases in the test suite.

## 2.3 Mixtures

### 2.3.1 Case 1Dmix1: 2 components - neutral hard spheres

Two component neutral hard sphere system with same total density as case all of the above cases. Compare results to case 6 above.

### 2.3.2 Case 1Dmix2: 2 components - neutral Lennard-Jones fluid

Two component neutral Lennard-Jones system with same total density as case all of the above cases. Compare results to case 14 above.

### 2.3.3 Case 1Dmix3: 2 components - Poisson-Boltzmann fluid

Two component Poisson-Boltzmann fluid with surface charge density of 0.3, and total ion density of 0.1 in a 1-1 electrolyte. Note that the free energy computation for this system is not correct at this point.

### 2.3.4 Case 1Dmix4: 2 components - Primitive Model fluid

Two component 1-1 electrolyte of total density 0.1. Finite sized particles of equal size. Two confining surface each with a surface charge density of 0.3. Note that the free energy computation for this system is not correct at this point.

### 2.3.5 Case 1Dmix5: 2 components - Primitive Model fluid - with c(r) corrections

Same as case 1Dmix4, but with corrections to the direct correlation function.

#### 2.3.6 Case 1Dmix6: 3 components - *Civilized* Model fluid

Same as case 1Dmix4, but also treats the solvent as a finite sized particle. The bulk solvent density is set to 0.65.

#### 2.3.7 Case 1Dmix7: 3 components - LJ electrolyte

Same as 1Dmix6, but with LJ terms also turned on. The external field has the coulomb contribution and a 9-3 LJ interaction. The wall-fluid interaction energy is 3. All cutoffs are set to $3\sigma$.

#### 2.3.8 Case 1Dmix8: 3 components - LJ electrolyte # 2

Same as 1Dmix7, but with a coarse mesh.

#### 2.3.9 Case 1Dmix9: 3 components - LJ electrolyte # 3

Same as 1Dmix8, but with a smaller domain and a reflective boundary on the right side.

| Case | niter | Ex. Ads[0] | Ex. Ads[1] | Ex.Ads[2] | Surf. Free Energy | Force |
|---|---|---|---|---|---|---|
| 1Dmix1 | 5 | -0.058018 | -0.058018 | N/A | 1.198788 | 4.956897154 |
| 1Dmix2 | 11 | -0.067056 | -0.067056 | N/A | -3.578393 | 1.635907510 |
| 1Dmix3 | 8 | -0.107513 | 0.492487 | N/A | N/A | N/A |
| 1Dmix4 | 8 | -0.185832 | 0.414168 | N/A | N/A | N/A |
| 1Dmix5 | 8 | -0.160379 | 0.439621 | N/A | N/A | N/A |
| 1Dmix6 | 8 | -0.149902 | 0.450098 | -0.502953 | N/A | N/A |
| 1Dmix7 | 13 | -0.165082 | 0.434918 | -0.607667 | N/A | N/A |
| 1Dmix8 | 8 | -0.164617 | 0.435383 | -0.674387 | N/A | N/A |
| 1Dmix9 | 7 | -0.160178 | 0.439822 | -0.666253 | N/A | N/A |

Table 3: Principle output for 1-dimensional mixture cases in the test suite.

# 3 1D Polymer Fluids at Equilibrium

## 3.1 Case 1Dpoly1: 8-2-8 repulsive polymer at a hard wall

CMS polymers. Total polymer density in the bulk is 0.711. This case does not require a restart from a file. For polymer cases always check the stoichiometry. For the 8-2-8 case, the adsorption of the tail beads should be 16/18 of the total and the adsorption of the head beads should be 2/18 of the total.

## 3.2 Case 1Dpoly2: 8-2-8 attractive polymer at an attractive wall

CMS polymers. In this case, the familiar 9-3 LJ potential is used in the dft_potentials.c file.

## 3.3 Case 1Dpoly3: 8-2-8 polymer in a single site solvent

CMS polymers. This system forms a model lipid bilayer. Use the provided restart file for this case.

## 3.4 Case 1Dpoly4: Charged 8-2-8 polymer in a single site solvent

CMS polymers. The tail (8) segments of the polymer carry a charge of -0.0125 while the head (2) segments of the polymer carry a charge of 0.1. The profile from case 1Dpoly3 is used as the initial guess with the electrostatic field set to 1 everywhere in the domain. This case demonstrates charged polymers and tests the Restart=3 option where the densities, fields, and propagator equations are taken from the files and PDE quantites (electrostatic field, chemical potential fields) are set to something simple.

### 3.5 Case 1Dpoly5: 3-bead Wertheim-Tripathi-Chapman polymer

This case can be compared to the Chapman iSAFT results in the directory *Ex_compare_other*. The discrepancies are due to different treatments of the hard sphere interactions. Note that the adsorption results demonstrate that these functionals do not conserve stoichiometry correctly in their current form.

### 3.6 Case 1Dpoly6: 5-bead Wertheim-Tripathi-Chapman polymer

This case can be compared to the Chapman iSAFT results in the directory *Ex_compare_other*. The discrepancies are due to different treatments of the hard sphere interactions. Note that the adsorption results demonstrate that these functionals do not conserve stoichiometry correctly in their current form. Also note that the reported adsorption units are for segments 0, 1, and 2. To get the component adsorptions to compare with case 1Dpoly7 remember that there are 2 beads of type 0 and 2 beads of type 1 on the chain.

### 3.7 Case 1Dpoly7: 5-bead Wertheim-Tripathi-Chapman polymer

This case is identical to 1Dpoly6 with the exception that beads of the same type are treated as one species in the calculation. This demonstrates that converting between components and segments in the code is all correct.

### 3.8 Case 1Dpoly8: 5-bead Wertheim-Tripathi-Chapman polymer with charges

This case is identical to 1Dpoly6 except very small charges are added to the sites in order to enforce stoichimetry through charge neutrality. Specifically, the two larger end beads have a charge of +0.000003 while the three smaller central beads have a charge of -0.000002.

| Case | niter | Ads[0] | Ads[1] | Ads[2] | Surf. Free Energy |
|---|---|---|---|---|---|
| 1Dpoly1 | 9 | 19.611163 | 2.362834 | N/A | -5.536430 |
| 1Dpoly2 | 10 | 19.335234 | 2.416904 | N/A | -4.355118 |
| 1Dpoly3 | 1 | 3.525591 | 0.440699 | 27.666943 | -26.314286 |
| 1Dpoly4 | 4 | 3.489204 | 0.436150 | 27.700606 | -26.254234 |
| 1Dpoly5 | 13 | 0.457427 | 0.579423 | 0.457427 | -1.198080 |
| 1Dpoly6 | 8 | 0.591319 | 0.575603 | 0.571352 | 0.401681 |
| 1Dpoly7 | 8 | 1.181661 | 1.727377 | | 0.411746 |
| 1Dpoly8 | 9 | 0.587382 | 0.587105 | 0.587942 | 0.402915 |

Table 4: Principle output for 1-dimensional polymer cases in the test suite.

## 4 2D/3D Atomic Fluids Tests

### 4.1 Case 2D3Datomic1: LJ Electrolyte in 2D - y reflections

This case is identical to 1Dmix8 except it is solved in a 2D domain with reflective boundary conditions in y.

### 4.2 Case 2D3Datomic2: LJ Electrolyte in 2D - x reflections

This case is identical to 2D3Datomic1 except the surface normals are rotated $90^o$, and the reflective boundary conditions are in the x-direction.

### 4.3 Case 2D3Datomic3: LJ Electrolyte in 2D - y periodic

This case is identical to 2D3Datomic1, but has periodic boundary conditions in y.

### 4.4 Case 2D3Datomic4: LJ Electrolyte in 2D - x periodic

This case is identical to 2D3Datomic2, but has periodic boundary conditions in x.

### 4.5 Case 2D3Datomic5: LJ Electrolyte in 2D - x continuation

This case is identical to 2D3Datomic1, but has periodic boundary conditions in x.

### 4.6 Case 2D3Datomic6: LJ Electrolyte in 2D - y continuation

This case is identical to 2D3Datomic2, but has periodic boundary conditions in y.

### 4.7 Case 2D3Datomic7: LJ Electrolyte in 3D - y and z reflections

This case is identical to 1Dmix9 except (1) it is solved in a 3D domain with reflective boundary conditions in both y and z, and (2) the size of the domain in x is halved with a reflective boundary on the right side.

### 4.8 Case 2D3Datomic8: LJ Electrolyte in 3D - x and z reflections

This case is identical to 2D3Datomic7 except it is solved in a 3D domain with reflective boundary conditions in both x and z.

### 4.9 Case 2D3Datomic9: LJ Electrolyte in 3D - x and y reflections

This case is identical to 2D3Datomic7 except it is solved in a 3D domain with reflective boundary conditions in both x and y.

### 4.10 Case 2D3Datomic10: LJ Electrolyte in 3D - y and z periodic

This case is identical to 2D3Datomic7 except there are now periodic boundaries in y and z.

### 4.11 Case 2D3Datomic11: LJ Electrolyte in 3D - y and z continue

This case is identical to 2D3Datomic7 except there are now reflective boundaries in y and z.

### 4.12 Case 2D3Datomic12: LJ Electrolyte in 3D - y continue and z periodic

This case is identical to 2D3Datomic7 except there are continuation boundaries in y and periodic boundaries z.

| Case | niter | Ex_Ads[0] | Ex_Ads[1] | Ex_Ads[2] | Surf. Free Energy |
|---|---|---|---|---|---|
| 2D3Datomic1 | 9 | -0.164613 | 0.435387 | -0.674180 | N/A |
| 2D3Datomic2 | 9 | -0.164613 | 0.435387 | -0.674180 | N/A |
| 2D3Datomic3 | 11 | -0.164613 | 0.435387 | -0.674180 | N/A |
| 2D3Datomic4 | 11 | -0.164613 | 0.435387 | -0.674180 | N/A |
| 2D3Datomic5 | 10 | -0.164613 | 0.435387 | -0.674180 | N/A |
| 2D3Datomic6 | 10 | -0.164613 | 0.435387 | -0.674180 | N/A |
| 2D3Datomic7 | 10 | -0.160170 | -0.439830 | -0.666005 | N/A |
| 2D3Datomic8 | 10 | -0.160170 | -0.439830 | -0.666005 | N/A |
| 2D3Datomic9 | 10 | -0.160170 | -0.439830 | -0.666005 | N/A |
| 2D3Datomic10 | 11 | -0.160170 | -0.439830 | -0.666005 | N/A |
| 2D3Datomic11 | 10 | -0.160170 | -0.439830 | -0.666005 | N/A |
| 2D3Datomic12 | 10 | -0.160170 | -0.439830 | -0.666005 | N/A |

Table 5: Principle output for 2 and 3-dimensional problems to test various boundary conditions.

# 5 Continuation Tests

## 5.1 Case 1Dcont1:

0th order continuation varying all densities simultaneously. 1 component Lennard-Jones fluid. Set to compute 30 steps, actually generates 22 solutions. Data for the first, tenth and last are shown below.

## 5.2 Case 1Dcont2:

Arc-length continuation, but otherwise identical to the 1Dcont1 case. Again set up to compute 30 steps. Actually computes 29 solutions. Data for the first, tenth, and last are shown below.

## 5.3 Case 1Dcont3:

Tests spinodal tracking algorithms. Finds the spinodal point in the temperature-density plane. Set up to compute 14 steps. Actually computes 12 solutions. Data for the first, fifth, and last are shown below. Use provided restart files for this case.

## 5.4 Case 1Dcont4:

Tests binodal tracking algorithms. Tracks a binodal in the temperature-density plane. Set up to compute 10 steps. Actually computes 11 solutions. Data for the first and last are shown in the table. Use provided restart files for this case.

| Case | niter | $kT/\epsilon_{ff}$ | $\rho_b$ | Ex_Ads[0] | force | Surf. Free Energy |
|---|---|---|---|---|---|---|
| 1Dcont1 (1) | 9 | 0.769230769 | 0.00000100 | 0.000415 | 0.000001489 | -0.000422 |
| 1Dcont1 (10) | 4 | 0.769230769 | 0.00124611 | 1.314452 | 0.001821120 | -1.504508 |
| 1Dcont1 (22) | 4 | 0.769230769 | 0.00506460 | 2.915346 | -0.106074043 | -3.943196 |
| 1Dcont2 (1) | 9 | 0.769230769 | 0.00000100 | 0.000415 | 0.000001489 | -0.000422 |
| 1Dcont2 (10) | 3 | 0.769230769 | 0.00044223 | 0.848369 | 0.000653671 | -0.344495 |
| 1Dcont2 (29) | 3 | 0.769230769 | 0.00677950 | 4.681015 | -0.319105662 | -0.5916833 |
| 1Dcont3 (1) | 9 | 0.769230769 | 0.00506755 | 2.933112 | -0.115707404 | -3.944801 |
| 1Dcont3 (5) | 3 | 0.8892308 | 0.01242808 | 2.638086 | -0.091864069 | -3.278197 |
| 1Dcont3 (12) | 4 | 1.0729808 | 0.03475432 | 2.456827 | -0.086154772 | -2.753229 |
| 1Dcont4 (1A) | 9 | 0.769230769 | 0.00345787 | 2.021229 | 0.003030797 | -3.036407 |
| 1Dcont4 (1B) | 9 | 0.769230769 | 0.00345787 | 4.391266 | -0.94904113 | -3.036407 |
| 1Dcont4 (11A) | 9 | 1.0692308 | 0.03405399 | 2.373717 | -0.065367901 | -2.756242 |
| 1Dcont4 (11B) | 9 | 1.0692308 | 0.03405399 | 2.640730 | -0.131511943 | -2.756242 |

Table 6: Principle output for test cases centered on continuation algorithms.

# 6 Test dates

**June 22, 2004**.....all test problems working.

One bug found in implementation of the interpolated direct correlation functions for the case of only one correlation function as found in the test suite.

# 7 DEVELOPERS GUIDE

In this section we provide some guidance to a new developer. Specifically we address the necessary steps for the implemetation of new functionals.

## 7.1 modifying input routines

## 7.2 modifying preprocessing routines

### 7.2.1 new thermodynamics

### 7.2.2 new stencils

### 7.2.3 new surfaces

## 7.3 implementing nonlinear equations

## 7.4 implementing new block matrix structures

## 7.5 modifying postprocessing routines

When a new functional is implemented in Tramonto, it is likely that the energy functional used to compute surface free energies will need to be modified. The postprocessing of the field variables is controlled in $dft\_out\_main.c$ where various general variables needed for computing spatial integrals are computed and where the various functions are called. One of the routines called is the free energy constructor routine.

The routine that controls the specific construction of the various contributions to the free energy is $calc\_free\_energy$ and is found in $dft\_out\_energy.c$. This file contains qualifying $if$ statements that build the appropriate free energy modules. It calls the general utility functions $integrateInSpace$ and $integrateInSpace_SumInComp$ for each piece of the functional. When a new functional is added, at least one new section should be added to $dft\_out\_energy.c$.

The specific part of the functional to be built is passed to the integrate utilities via function pointers. These functions all contain the integrands needed for free energy calculations. The necessary integrands are found in physics specific energy routines (for example, $dft\_energy\_att.c$). When a new functional is added to Tramonto, additional integrands should be added to these files if the new functional is similar in nature to existing integrands (e.g. a new kind of attractive functional could be added to $dft\_energy\_att.c$). Or, if a completely new kind of functional has been developed, a new file should be added to the package.

Note that depending on the type of extension, the new functional may be a perturbation to functionals already existing in the code. Or, like the Chandler-McCoy-Singer (CMS) functionals, the new functionals may be completely disconnected from what has gone before. In either case, be sure that the qualifying $if$ statements in $dft\_out\_energy.c$ are modified to be inclusive or exclusive as necessary.