SSCHA input file

April 18, 2023

The input file perform the minimization.
To run the SSCHA code with the input file use:

```
>>> sscha -i simple_input.in --save-data simple_input.out
```

The file can have any name (often *.in).
An example of an input file:

```
!
! * * * * * * * * * * * * * *
! *                         *
! *    VC - RELAX EXAMPLE   *
! *                         *
! * * * * * * * * * * * * * *
!
!
! This is the input to perform the sscha minimization, followed
! by the change of the unit cell given by the stress step.
!
! This is not the recommended way to do it (you can do everything automatically)
! But usefull if you want to control manually each submission
!

&relax
type = "vc-relax"
start_pop = 2
max_pop_id = 2
generate_ensemble = .false.
fix_volume = .false.
target_pressure = 0 ! [GPa]
bulk_modulus = 15 ! [GPa]
n_configs = 1000
&end
```

```
&inputscha
n_random = 1000
data_dir = "../ensemble_data_test"
population = 2
fildyn_prefix = "../ensemble_data_test/dyn"
nqirr = 1
supercell_size =  1 1 1
Tg = 0
T = 0
meaningful_factor = 1e-4
gradi_op = "all"
n_random_eff = 500
print_stress = .true.
eq_energy = -144.40680397
lambda_a = 1
lambda_w = 1
root_representation = "normal"
preconditioning = .true.
max_ka= 20
/
```

1 &inputsscha

- "lambda_a"

- "lambda_w"

- "minim_struc"

- "precond_wyck"

- "preconditioning"

- "root_representation"

- "neglect_symmetries"

- "n_random_eff"

- "n_random" The dimension of the ensemble

- "meaningful_factor"

- "eq_energy"

- "fildyn_prefix" [REQUIRED]

- "nqirr" [REQUIRED]

- "data_dir" The position of the ensemble (where the data are stored). Unit of measurements must be in bohr for displacements and $Ry/bohr$ for forces and $Ry/bohr^3$ for stress tensors. Energy is in Ry.

- "load_bin"

- "t" [REQUIRED]

- "tg"

- "supercell_size"

- "max_ka"

- "stress_offset"

- "gradi_op"

- "population" The population id. This is an integer that distinguish different ensembles and allows for use the same data_dir for several minimizations

- "print_stress"

- "use_spglib"

2 &relax

- "type" [REQUIRED]
    - "sscha"
    - "relax"
    - "vc-relax"

- "n_configs" [REQUIRED]

- "max_pop_id"

- "start_pop"

- "ensemble_datadir"

- "generate_ensemble"

- "target_pressure"

- "fix_volume"

- "bulk_modulus"

- "sobol_sampling"

- "sobol_scatter"

3 calculator

- "k_points" [REQUIRED]

- "k_offset"

- "disable_check"

- "program" [REQUIRED]

- "binary"

- "pseudo_"

1. "quantum-espresso"
    - "ecutrho", "ecutwfc", "smearing", "degauss", "occupations", "conv_thr", "tstress", "tprnfor", "verbosity", "disk_io", "input_dft", "use_all_frac"

4 cluster

"template" "SSCHA_CLUSTERS_DIR" "hostname" "pwd" "account" "binary_path" "mpicmd" "reconnect_attempts" "port"
   "shell" "submit_cmd" "queue_directive" "v_nodes" "n_nodes" "use_nodes" "v_cpu" "n_cpu" "use_cpu" "v_time" "n_time" "n_pools" "use_time" "v_memory" "max_ram" "use_memory" "v_partition" "partition_name" "use_partition" "init_script" "max_recalc" "batch_size" "local_workdir" "v_account" "use_account" "sshcmd" "scpcmd" "timeout" "job_numbers" "n_together"
   "workdir"

5 utils

"utils" "save_freq_filename" "save_rho_filename" "mu_lock_start" "mu_lock_end" "mu_free_start" "mu_free_end" "project_dyn" "project_structure"