CoCo-MD: A Simple and Effective Method for the Enhanced Sampling of Conformational Space

Ardita Shkurti[1§], Ioanna Danai Styliari[1°], Vivek Balasubramanian[2], Iain Bethune[3], Conrado Pedebos, Shantenu Jha[3], Charles A. Laughton[1]*

[1]School of Pharmacy and Centre for Biomolecular Sciences, University of Nottingham, University Park, Nottingham NG7 2RD, UK.

[2]Department of Electrical and Computer Engineering, Rutgers University, Piscataway, NJ 08854, USA.

[3]EPCC, The University of Edinburgh, James Clerk Maxwell Building, Peter Guthrie Tait Road, Edinburgh, UK.

Supplementary Material

1. Methods

a) Example MD input file for Gromacs simulations:

```
; RUN CONTROL PARAMETERS
integrator               = md
; Start time and timestep in ps
tinit                    = 0
dt                       = 0.002
nsteps                   = 25000
; For exact run continuation or redoing part of a run
init-step                = 0
; Part index is updated automatically on checkpointing (keeps files
separate)
simulation-part          = 1
; mode for center of mass motion removal
comm-mode                = Linear
; number of steps for center of mass motion removal
nstcomm                  = 100
; group(s) for center of mass motion removal
comm-grps                =

; LANGEVIN DYNAMICS OPTIONS
```

```
; Friction coefficient (amu/ps) and random seed
bd-fric                  = 0
ld-seed                  = -1

; ENERGY MINIMIZATION OPTIONS
; Force tolerance and initial step-size
emtol                    = 10
emstep                   = 0.01
; Max number of iterations in relax-shells
niter                    = 20
; Step size (ps^2) for minimization of flexible constraints
fcstep                   = 0
; Frequency of steepest descents steps when doing CG
nstcgsteep               = 1000
nbfgscorr                = 10

; OUTPUT CONTROL OPTIONS
; Output frequency for coords (x), velocities (v) and forces (f)
nstxout                  = 0
nstvout                  = 0
nstfout                  = 0
; Output frequency for energies to log file and energy file
nstlog                   = 1000
nstcalcenergy            = 100
nstenergy                = 1000
; Output frequency and precision for .xtc file
nstxout-compressed       = 500
compressed-x-precision   = 1000
; This selects the subset of atoms for the compressed
; trajectory file. You can select multiple groups. By
; default, all atoms will be written.
compressed-x-grps        =
; Selection of energy groups
energygrps               =

; NEIGHBORSEARCHING PARAMETERS
; cut-off scheme (Verlet: particle based cut-offs, group: using charge
groups)
cutoff-scheme            = verlet
; nblist update frequency
nstlist                  = 10
; ns algorithm (simple or grid)
ns_type                  = grid
; Periodic boundary conditions: xyz, no, xy
pbc                      = xyz
periodic-molecules       = no
; Allowed energy error due to the Verlet buffer in kJ/mol/ps per atom,
; a value of -1 means: use rlist
verlet-buffer-tolerance  = 0.005
; nblist cut-off
rlist                    = 1
; long-range cut-off for switched potentials

; OPTIONS FOR ELECTROSTATICS AND VDW
; Method for doing electrostatics
coulombtype              = PME
coulomb-modifier         = Potential-shift-Verlet
rcoulomb-switch          = 0
rcoulomb                 = 1.0
```

```
; Relative dielectric constant for the medium and the reaction field
epsilon-r               = 1
epsilon-rf              = 0
; Method for doing Van der Waals
vdw-type                = Cut-off
vdw-modifier            = Potential-shift-Verlet
; cut-off lengths
rvdw-switch             = 0
rvdw                    = 1.0
; Apply long range dispersion corrections for Energy and Pressure
DispCorr                = No
; Extension of the potential lookup tables beyond the cut-off
table-extension         = 1
; Separate tables between energy group pairs
energygrp-table         =
; Spacing for the PME/PPPM FFT grid
fourierspacing          = 0.12
; FFT grid size, when a value is 0 fourierspacing will be used
fourier-nx              = 0
fourier-ny              = 0
fourier-nz              = 0
; EWALD/PME/PPPM parameters
pme-order               = 4
ewald-rtol              = 1e-05
ewald-rtol-lj           = 0.001
lj-pme-comb-rule        = Geometric
ewald-geometry          = 3d
epsilon-surface         = 0

; IMPLICIT SOLVENT ALGORITHM
implicit-solvent        = No

; OPTIONS FOR WEAK COUPLING ALGORITHMS
; Temperature coupling
tcoupl                  = Berendsen
nsttcouple              = -1
nh-chain-length         = 10
print-nose-hoover-chain-variables = no
; Groups to couple separately
tc-grps                 = Protein SOL NA
; Time constant (ps) and reference temperature (K)
tau-t                   = 0.2 0.2 0.2
ref-t                   = 300 300 300
; pressure coupling
pcoupl                  = Berendsen
pcoupltype              = Isotropic
nstpcouple              = -1
; Time constant (ps), compressibility (1/bar) and reference P (bar)
tau-p                   = 1
compressibility         = 4.5e-5
ref-p                   = 1.0
; Scaling of reference coordinates, No, All or COM
refcoord-scaling        = COM

; GENERATE VELOCITIES FOR STARTUP RUN
gen-vel                 = no
gen-temp                = 300
gen-seed                = -1
```

```
; OPTIONS FOR BONDS
constraints              = all-bonds
; Type of constraint algorithm
constraint-algorithm     = Lincs
; Do not constrain the start configuration
continuation             = no
; Use successive overrelaxation to reduce the number of shake iterations
Shake-SOR                = no
; Relative tolerance of shake
shake-tol                = 0.0001
; Highest order in the expansion of the constraint coupling matrix
lincs-order              = 4
; Number of iterations in the final step of LINCS. 1 is fine for
; normal simulations, but use 2 to conserve energy in NVE runs.
; For energy minimization with constraints it should be 4 to 8.
lincs-iter               = 1
; Lincs will write a warning to the stderr if in one step a bond
; rotates over more degrees than
lincs-warnangle          = 30
; Convert harmonic bonds to morse potentials
morse                    = no

; ENERGY GROUP EXCLUSIONS
; Pairs of energy groups for which all non-bonded interactions are excluded
energygrp-excl           =
```

b) Example MD input file for AMBER simulations:

```
10ns simulation of alanine pentapeptide
 &cntrl
   imin=0, ntx=1,
   ntpr=1000, ntwr=1000, ntwx=500,
   nstlim=5000000, dt=0.002,
   ntt=3, ig=-1, gamma_ln=5.0,
   ntc=2, ntf=2,
   ntb=2, cut=9.0,
   ntp=1, taup=2.0,
 &end
```

c) Flowchart for the CoCo-MD method.

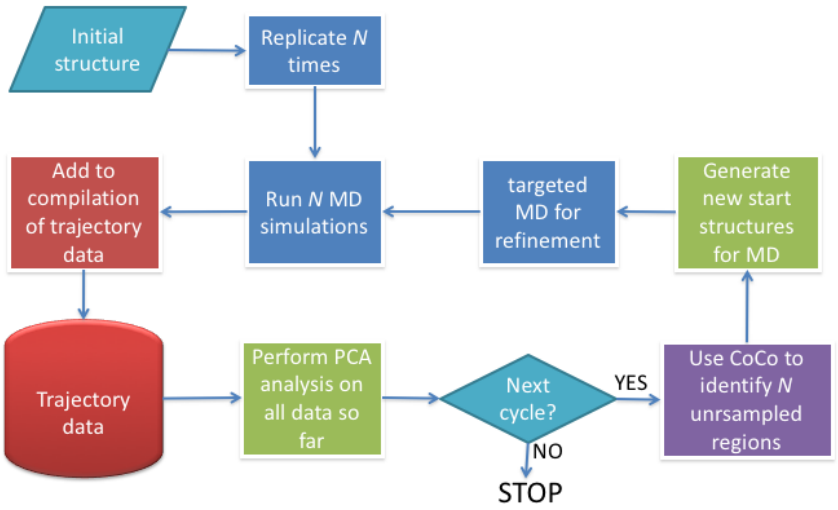


Figure S1: The CoCo-MD iterative simulation/analysis workflow.

d) Development and evaluation of the reweighting/resampling method for CoCo-MD ensembles.

The procedure to correct, approximately, for the biased sampling produced by the CoCo-MD method is as follows:

1. For each member of the ensemble, we obtain the coordinates in a principal component subspace, and the potential energy.
2. By binning coordinate and energy data, we assign each structure to a microstate.
3. A histogram of the potential energy distribution from a short but equilibrated (c 10 ns) conventional MD simulation of the same system is generated. A target number of counts for each bin in the histogram is set: this is what the resampling process will seek to satisfy.
4. A random microstate is selected.
5. A random member of the biased ensemble that is in that microstate is selected.
6. If it has an energy that corresponds to a bin in the potential energy histogram that has a count less than the target value, the member is added to the new ensemble, and the population count in that bin is increased by one.
7. We return to step 4, until there are no bins left that need more samples.

The final result is a new ensemble, formed by sampling with replacement from the original, that has a potential energy distribution that matches a conventional MD simulation of the same system.

To test the approach, we applied it to a double-well two dimensional potential on a 20 x 20 grid (Figure S2a):

$$E(i, j) = \min(E_{0,1} + K_1((i-i_1)^2 + (j-j_1)^2), E_{0,2} + K_2((i-i_2)^2 + (j-j_2)^2))$$

$$E_{0,1} = 0.0,\ E_{0,2} = 0.2,\ K_1 = 0.1,\ K_2 = 0.2,\ i_1 = j_1 = 5,\ i_2 = j_2 = 15$$

Where i and j are grid point indices, and all energies are in kT.

We used a simple Metropolis Monte-Carlo approach to generate a Boltzmann weighted sample on the surface (100000 trials, 22000 samples, acceptance ratio 0.22). From this we generated reference 2D and 1D free energy surfaces (Figure S2b and S2c).

We then repeated the procedure, in the presence of a bias potential that was added to the "true" potential (Figure S2d):

$$E_{biased}(i, j) = E(i, j) + 0.2 * i$$

The apparent 2D and 1D free energy surfaces from this biased sampling is shown in Figure S2e and S2f

Since in the CoCo-MD method we obtain a biased sampling, but have an unbiased potential, we calculated the true (unbiased) energy for each member of the biased sample. For this toy system, the microstate of each sample is defined precisely by the indices (i, j). Therefore to mimic a more realistic scenario, we assumed that only coordinate i was known, plus the potential energy. We then explored how well our reweighting method could recover the true one-dimensional free energy surface (along i), from the biased sample. Without including potential energy in the microstate assignments, all samples are assigned to one of just twenty microstates according to the value of i, and the method over-corrects for the bias: the two minima are of equal energy (Figure S2h). As we add potential energy into the microstate definition the performance of the method improves dramatically. With increasing resolution in the partitioning of the energy (greater number of bins), the number of microstates increases (Figure S2g) and the accuracy with which the 1D free energy surface can be recovered increases. However, the performance with a quite modest discretisation of the energy term is very reasonable (Figure S2h).

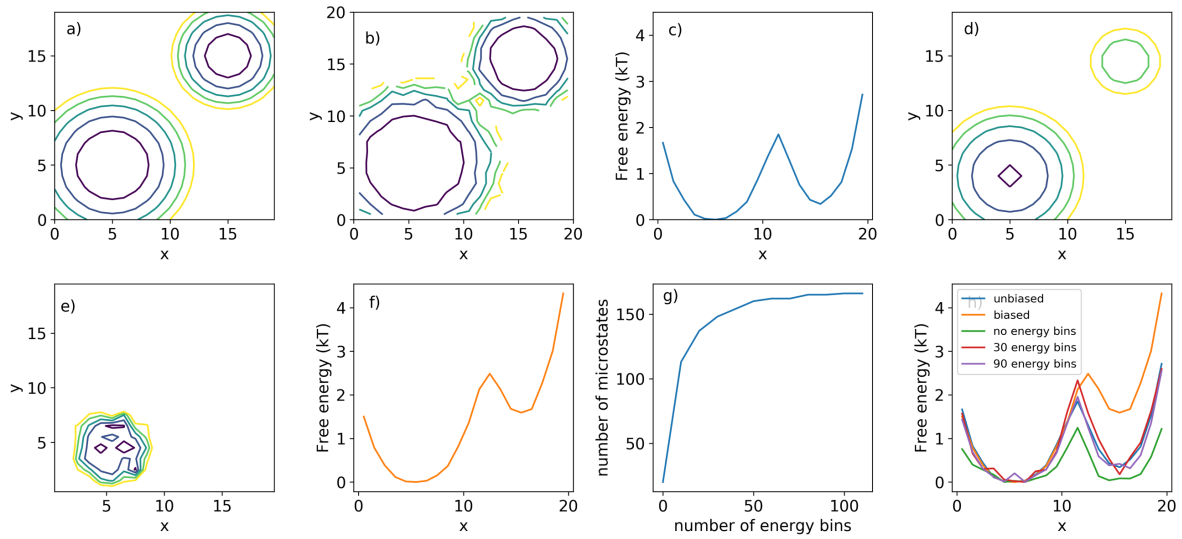


Figure S2: Application of the CoCo-MD ensemble reweighting method to a simple double-well potential. See text for details.

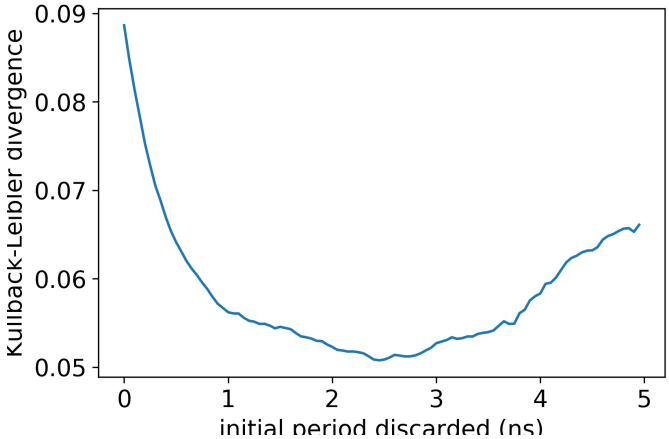


Figure S3. Kullback-Leibler divergence in the histograms generated from the first and second halves of the merged two hundred independent 10ns CMD simulations of alanine pentapeptide, as a function of the amount of each initial trajectory that was trimmed off, as potentially biased by the starting configuration.

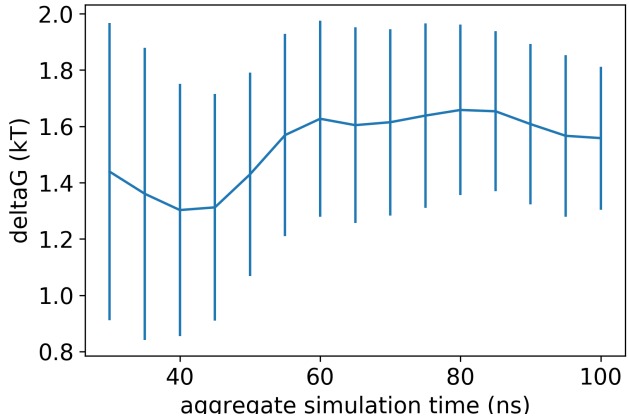
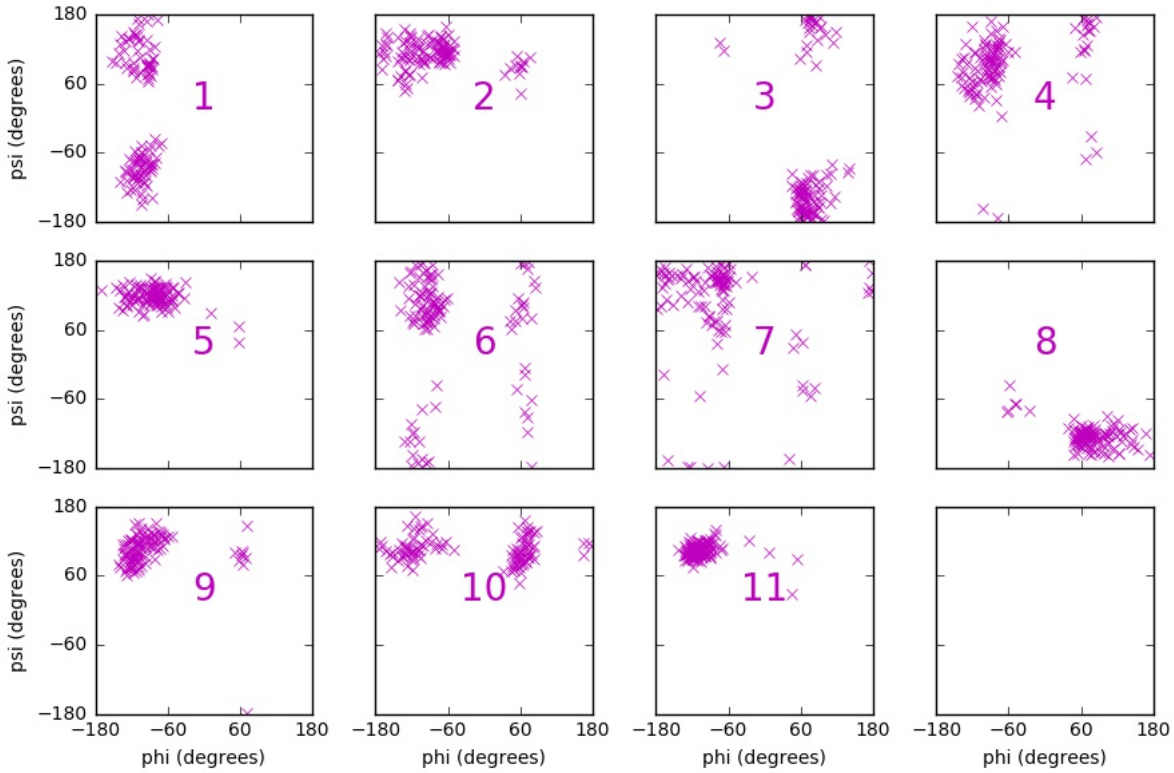
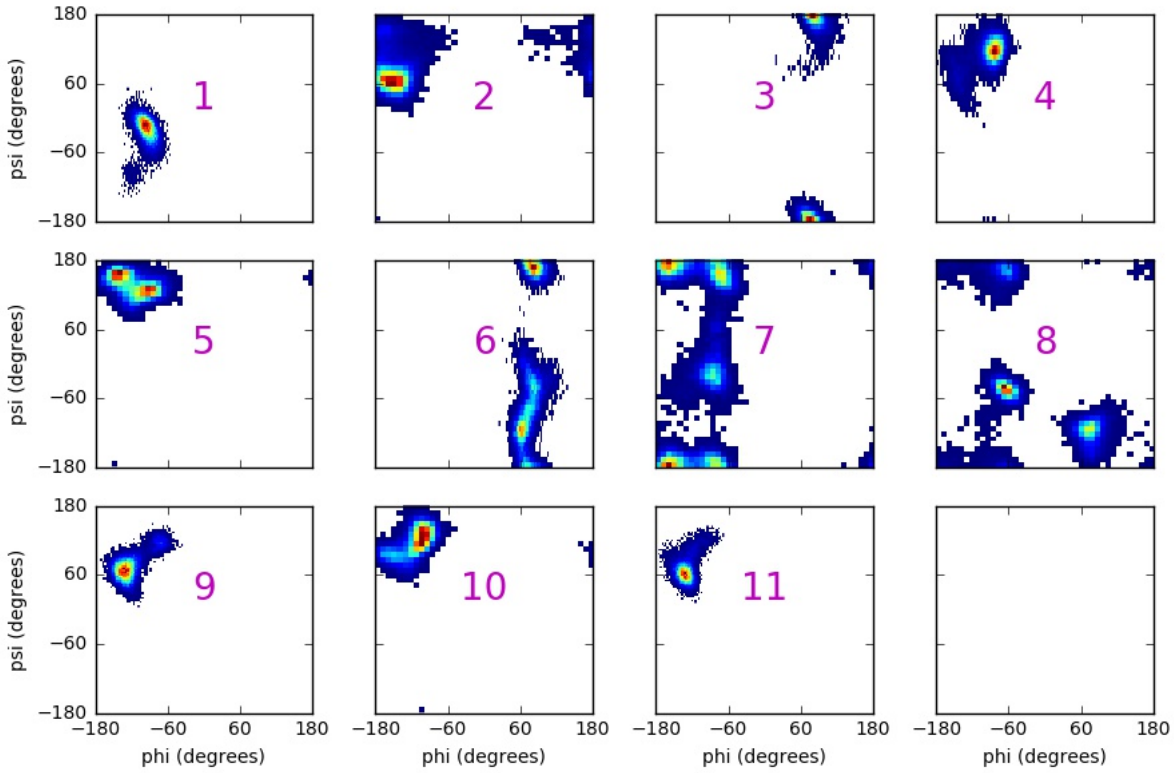


Figure S4: Convergence in the estimated value of the free energy difference between the extended and alpha-helical states of alanine pentapeptide for CMD simulations begun from the alpha-helical state. Each estimation used ten of the one hundred independent 10 ns simulations. Error bars are +/- the standard error (N=10).

2. Analysis of Cyclosporine A conformational sampling.





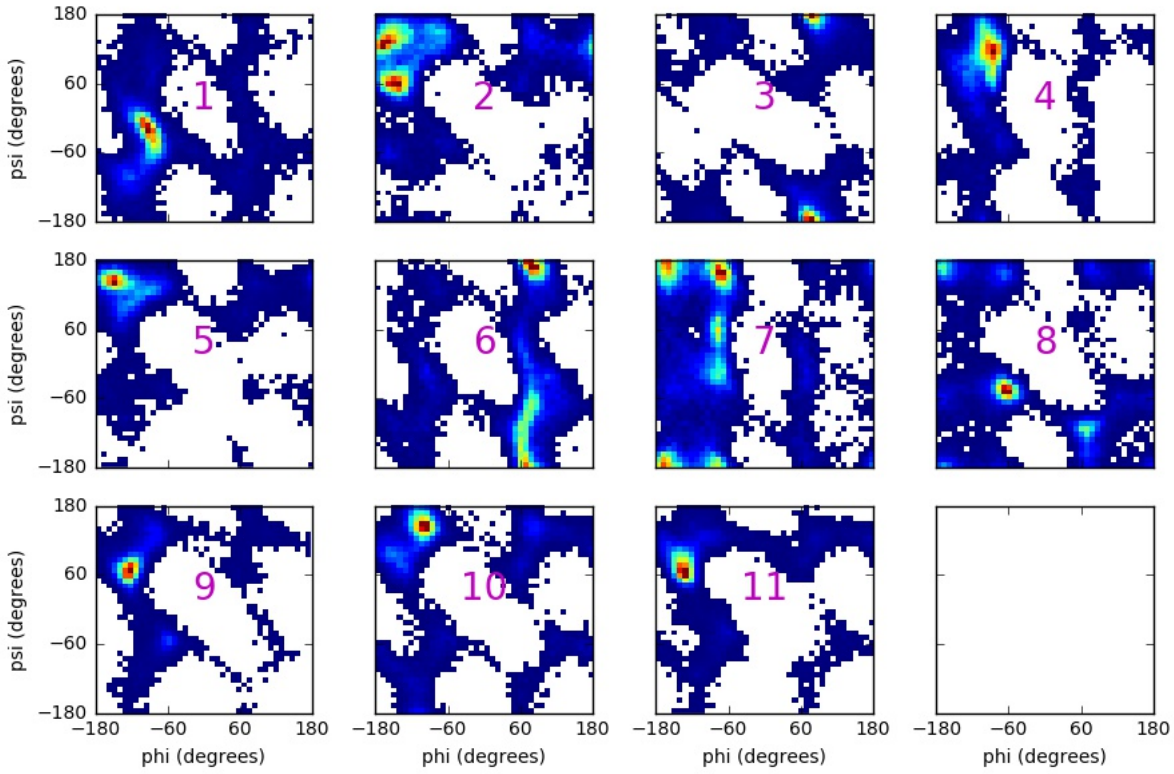


Figure S5: Ramachandran maps for all amino acids in CSA, comparing the distributions found in top panel: the centroids from Witeck et al., centre panel: 10 replicate 2ns conventional MD simulations, bottom panel: CoCo-MD simulations (same aggregate simulation time).

3. Analysis of MBP conformational sampling.

List of crystal and NMR structures (PDB codes) used for the PCA analysis. In the case of NMR ensembles, just the first submitted model was used.

1ANF	1OMP	3RLF
1DMB	2KLF	3RUM
1EZ9	2MV0	3VFJ
1FQA	2N44	4KHZ
1FQB	2N45	4MBP
1FQC	3MBP	5BK1
1FQD	3PUV	5BK2
1JW4	3PUW	5GS2
1JW5	3PUX	5M13
1LLS	3PUY	5M14