Fortran 90 source codes for Monte Carlo simulations of one- and two-dimensional random walks on naked and nucleosomal DNA

Martin Kampmann

```
PROGRAM eff1no

! records number of different positions visited as a function of steps in
! a one-dimensional random walk on naked DNA.
!
! Run = Run number (1-10)
! Nsteps = number of random walk steps (0-10000)
! x = x coordinate of protein position on DNA
!
! Random = random number used to determine direction of next step
! Distribution = array containing the frequencies
!

IMPLICIT NONE
INTEGER :: Run, Nsteps, x, j
REAL :: Random, Mean, Sd
INTEGER, DIMENSION(20001) :: Distribution, Count
REAL, DIMENSION(10):: Results

OPEN (UNIT  = 12, FILE = "reff1no.txt", STATUS = "NEW")
WRITE (12, *) "# Steps","Mean","SD"

CALL RANDOM_SEED

DO Nsteps = 0, 10000, 100

        Results = 0

        ! simulation
        DO Run = 1, 10
```

```
                x = 0

                Distribution = 0
                DO j = 1, Nsteps

                        CALL RANDOM_NUMBER (Random)
                        IF (Random < 0.5) THEN
                                x = x + 1
                        ELSE
                                x = x - 1
                        END IF

                    Distribution(10001+x) = Distribution(10001+x) + 1

                END DO

                Count = 0
                WHERE (Distribution > 0) Count = 1
                Results (Run) = SUM (Count)

PRINT *, Results(Run)

        END DO

        ! write results to file

        Mean = SUM(Results)/10
        Results = (Results - Mean)
        Results = Results*Results
        Sd = SQRT(SUM(Results)/10)
        WRITE (12, *) Nsteps, ACHAR(9), Mean, ACHAR(9), Sd
        PRINT *, Nsteps, Mean, Sd

END DO

END PROGRAM eff1no
```

```
PROGRAM eff1nu

! records number of different positions visited as a function of steps in
! a one-dimensional random walk. Nucleosome obstacles are 145 bp in length,
! with 55 bp of linker DNA in between.
!
! Run = Run number (1-10)
! Nsteps = number of random walk steps (0-10000)
! x = x coordinate of protein position on DNA
!
! Random = random number used to determine direction of next step
! Distribution = array containing the frequencies
!

IMPLICIT NONE
INTEGER :: Run, Nsteps, x, j
REAL :: Random, Mean, Sd
INTEGER, DIMENSION(20001) :: Distribution, Count
REAL, DIMENSION(10):: Results

OPEN (UNIT  = 12, FILE = "reff1nu.txt", STATUS = "NEW")
WRITE (12, *) "# Steps","Mean","SD"

CALL RANDOM_SEED

DO Nsteps = 0, 10000, 100

        Results = 0

        ! simulation

        DO Run = 1, 10
                x = 0

                Distribution = 0
```

```
                DO j = 1, Nsteps

                        CALL RANDOM_NUMBER (Random)
                        IF ((Random < 0.5) .AND. (MODULO((ABS(x+1)-28),200) > 144)) THEN
                                x = x + 1
                        ELSE IF ((Random >= 0.5) .AND. (MODULO((ABS(x-1)-28),200) > 144)) THEN
                                x = x - 1
                        END IF

                Distribution(10001+x) = Distribution(10001+x) + 1

                END DO

                Count = 0
                WHERE (Distribution > 0) Count = 1
                Results (Run) = SUM (Count)

PRINT *, Results(Run)

        END DO

        ! write results to file

        Mean = SUM(Results)/10
        Results = (Results - Mean)
        Results = Results*Results
        Sd = SQRT(SUM(Results)/10)

        WRITE (12, *) Nsteps, ACHAR(9), Mean, ACHAR(9), Sd
        PRINT *, Nsteps, Mean, Sd

END DO

END PROGRAM eff1nu
```

```
PROGRAM eff2no

! records number of different positions visited as a function of steps in
! a two-dimensional random walk on naked DNA.
!
! Run = Run number (1-10)
! Nsteps = number of random walk steps (0-10000)
! x = x coordinate of protein position on DNA
! y = y coordinate of protein position on DNA
! Random = random number used to determine direction of next step
! Distribution = array containing the frequencies
!

IMPLICIT NONE
INTEGER :: Run, Nsteps, x, y, j
REAL :: Random, Mean, Sd
INTEGER, DIMENSION(20001) :: Distribution, Count
REAL, DIMENSION(10):: Results

OPEN (UNIT  = 12, FILE = "reff2no.txt", STATUS = "NEW")
WRITE (12, *) "# Steps","Mean","SD"

CALL RANDOM_SEED

DO Nsteps = 0, 10000, 100

        Results = 0

        ! simulation

        DO Run = 1, 10
                x = 0
                y = 0
                Distribution = 0
                DO j = 1, Nsteps
```

```
                DO
                    CALL RANDOM_NUMBER (Random)
                    IF (Random < 0.25) THEN
                        x = x + 1
                        y = y + 1
                    ELSE IF (Random >= 0.25 .AND. Random < 0.5) THEN
                        x = x + 1
                        y = y - 1
                    ELSE IF (Random >= 0.5 .AND. Random < 0.75) THEN
                        x = x - 1
                        y = y + 1
                    ELSE
                        x = x - 1
                        y = y - 1
                    END IF
                    x = MODULO(x,10)
                    IF (x == MODULO(y,10)) EXIT
                END DO

                Distribution(10001+y) = Distribution(10001+y) + 1

            END DO

            Count = 0
            WHERE (Distribution > 0) Count = 1
            Results (Run) = SUM (Count)

PRINT *, Results(Run)

        END DO

        ! write results to file

        Mean = SUM(Results)/10
        Results = (Results - Mean)
        Results = Results*Results
```

```
        Sd = SQRT(SUM(Results)/10)

        WRITE (12, *) Nsteps, ACHAR(9), Mean, ACHAR(9), Sd
        PRINT *, Nsteps, Mean, Sd

END DO

END PROGRAM eff2no
```

```
PROGRAM eff2nu

! records number of different positions visited as a function of steps in
! a two-dimensional random walk. Nucleosome obstacles are 145 bp in length,
! with 55 bp of linker DNA in between. They occlude 50% of the DNA helix.
!
! Run = Run number (1-10)
! Nsteps = number of random walk steps (0-10000)
! x = x coordinate of protein position on DNA
! y = y coordinate of protein position on DNA
! Random = random number used to determine direction of next step
! Distribution = array containing the frequencies
!

IMPLICIT NONE
INTEGER :: Run, Nsteps, x, y, j
REAL :: Random, Mean, Sd
INTEGER, DIMENSION(20001) :: Distribution, Count
REAL, DIMENSION(10):: Results

OPEN (UNIT  = 12, FILE = "reff2nu.txt", STATUS = "NEW")
WRITE (12, *) "# Steps","Mean","SD"

CALL RANDOM_SEED

DO Nsteps = 0, 10000, 100
```

```
Results = 0

! simulation

DO Run = 1, 10
        x = 0
        y = 0
        Distribution = 0
        DO j = 1, Nsteps

                DO
        CALL RANDOM_NUMBER (Random)
        IF (Random<0.25 .AND. (MODULO((ABS(y+1)-28),200)>144 .OR. MODULO((x+1),10)<6)) THEN
                x = x + 1
                y = y + 1
        ELSE IF (Random>=0.25 .AND. Random<0.5 .AND. (MODULO((ABS(y-1)-28),200)>144 .OR. MODULO((x+1),10)<6)) THEN
                x = x + 1
                y = y - 1
        ELSE IF (Random>=0.5 .AND. Random<0.75 .AND. (MODULO((ABS(y+1)-28),200)>144 .OR. MODULO((x-1),10)<6)) THEN
                x = x - 1
                y = y + 1
        ELSE IF (MODULO((ABS(y-1)-28),200)>144 .OR. MODULO((x-1),10)<6) THEN
                x = x - 1
                y = y - 1
        END IF
                x = MODULO(x,10)
                IF (x == MODULO(y,10)) EXIT
                END DO

                Distribution(10001+y) = Distribution(10001+y) + 1

        END DO

        Count = 0
        WHERE (Distribution > 0) Count = 1
        Results (Run) = SUM (Count)
```

```
        END DO

        ! write results to file

        Mean = SUM(Results)/10
        Results = (Results - Mean)
        Results = Results*Results
        Sd = SQRT(SUM(Results)/10)

        WRITE (12, *) Nsteps, ACHAR(9), Mean, ACHAR(9), Sd
        PRINT *, Nsteps, Mean, Sd

END DO

END PROGRAM eff2nu
```