Electronic Supplementary Material (ESI) for Physical Chemistry Chemical Physics.

The following is the C++ code employed to generate COOH hydrogen coordinates.

```
#define displacement_bounds 0.5 //this defines the maximum boundary of hydrogen motion

Random_Seed(GetTickCount());

// the following defines the initial hydrogen coordinates

atom_pos H1;
atom_pos H9;

H1.origin.x = H1.current_pos.x = 0.853237;
H1.origin.y = H1.current_pos.y = -0.071849;
H1.origin.z = H1.current_pos.z = -1.770146;
H9.origin.x = H9.current_pos.x = -1.041311;
H9.origin.y = H9.current_pos.y = -0.119732;
H9.origin.z = H9.current_pos.z = -2.065867;

double coord = 0.0;

/*  The code given below generates the random number and then adds/subtracts from the initial
hydrogen positions described above. */

for (int i = 0; i < 2500; i++)
{
        // H1
        coord = Random_Double() * displacement_bounds;
        if (CheckMovePositiveSuccess(H1.current_pos.x, coord, H1.origin.x) == true)
        {
                H1.current_pos.x = H1.current_pos.x + coord;
                sprintf(coord_text1x[i], "%1.6f", H1.current_pos.x);
                TruncateString(coord_text1x[i], 10);
        }
        else if(CheckMoveNegativeSuccess(H1.current_pos.x, coord, H1.origin.x) == true)
        {
                H1.current_pos.x = H1.current_pos.x - coord;
                sprintf(coord_text1x[i], "%1.6f", H1.current_pos.x);
                TruncateString(coord_text1x[i], 10);
        }
        coord = Random_Double() * displacement_bounds;
        if (CheckMovePositiveSuccess(H1.current_pos.y, coord, H1.origin.y) == true)
        {
                H1.current_pos.y = H1.current_pos.y + coord;
                sprintf(coord_text1y[i], "%1.6f", H1.current_pos.y);
                TruncateString(coord_text1y[i], 10);
        }
        else if (CheckMoveNegativeSuccess(H1.current_pos.y, coord, H1.origin.y) == true)
        {
                H1.current_pos.y = H1.current_pos.y - coord;
                sprintf(coord_text1y[i], "%1.6f", H1.current_pos.y);
                TruncateString(coord_text1y[i], 10);
```

```
	}
	coord = Random_Double() * displacement_bounds;
	if (CheckMovePositiveSuccess(H1.current_pos.z, coord, H1.origin.z) == true)
	{
		H1.current_pos.z = H1.current_pos.z + coord;
		sprintf(coord_text1z[i], "%1.6f", H1.current_pos.z);
		TruncateString(coord_text1z[i], 10);
	}
	else if (CheckMoveNegativeSuccess(H1.current_pos.z, coord, H1.origin.z) == true)
	{
		H1.current_pos.z = H1.current_pos.z - coord;
		sprintf(coord_text1z[i], "%1.6f", H1.current_pos.z);
		TruncateString(coord_text1z[i], 10);
	}
}
	for (int i = 0; i < 2500; i++)
{
	// H9
	coord = Random_Double() * displacement_bounds;
	if (CheckMovePositiveSuccess(H9.current_pos.x, coord, H9.origin.x) == true)
	{
		H9.current_pos.x = H9.current_pos.x + coord;
		sprintf(coord_text2x[i], "%1.6f", H9.current_pos.x);
		TruncateString(coord_text2x[i], 10);
	}
	else if (CheckMoveNegativeSuccess(H9.current_pos.x, coord, H9.origin.x) == true)
	{
		H9.current_pos.x = H9.current_pos.x - coord;
		sprintf(coord_text2x[i], "%1.6f", H9.current_pos.x);
		TruncateString(coord_text2x[i], 10);
	}
	coord = Random_Double() * displacement_bounds;
	if (CheckMovePositiveSuccess(H9.current_pos.y, coord, H9.origin.y) == true)
	{
		H9.current_pos.y = H9.current_pos.y + coord;
		sprintf(coord_text2y[i], "%1.6f", H9.current_pos.y);
		TruncateString(coord_text2y[i], 10);
	}
	else if (CheckMoveNegativeSuccess(H9.current_pos.y, coord, H9.origin.y) == true)
	{
		H9.current_pos.y = H9.current_pos.y - coord;
		sprintf(coord_text2y[i], "%1.6f", H9.current_pos.y);
		TruncateString(coord_text2y[i], 10);
	}
	coord = Random_Double() * displacement_bounds;
	if (CheckMovePositiveSuccess(H9.current_pos.z, coord, H9.origin.z) == true)
	{
		H9.current_pos.z = H9.current_pos.z + coord;
		sprintf(coord_text2z[i], "%1.6f", H9.current_pos.z);
		TruncateString(coord_text2z[i], 10);
	}
```

```
        else if (CheckMoveNegativeSuccess(H9.current_pos.z, coord, H9.origin.z) == true)
        {
                H9.current_pos.z = H9.current_pos.z - coord;
                sprintf(coord_text2z[i], "%1.6f", H9.current_pos.z);
                TruncateString(coord_text2z[i], 10);
        }
}
```