MOLECULAR DYNAMICS SIMULATION:

In molecular dynamics, integrating Newton’s laws of motion, successive configurations of the system are generated starting from an initial condition. The result is a trajectory that specifies how the positions and velocities of the particles in the system vary with time.

The trajectory is obtained by solving Newton’s Second, \( \dot{P} = \frac{\ddot{F}_i}{m_i} \), where \( m_i \) is the mass of the \( i^{th} \) particle and \( \ddot{F}_i \) is the force acting on the \( i^{th} \) particle.

Finite Difference Method:

Finite difference methods are used to generate molecular dynamics trajectories with continuous potential model. The basic idea is that the integration is discretized by a time interval \( \delta t \). The total force on each particle in the configuration at a time \( t \) is calculated as the vector sum of its interactions terms with all other particles. From the force we can determine the accelerations of the particles, which are then combined with the positions and velocities at a time \( t \) to calculate the positions and velocities at a time \( t + \delta t \). The forces on the particles in their new positions are then determined, leading to positions and velocities at time \( t + 2\delta t \), and so on.

All the algorithms for integrating the equations of motion using finite difference method assume that the positions and dynamics properties (velocities, accelerations, etc.) can be approximated as Taylor series expansions:

\[
\begin{align*}
\vec{r}(t + \delta t) &= \vec{r}(t) + \delta \vec{V}(t) + \frac{1}{2} \delta^2 \vec{a}(t) + \cdots \cdots \cdots \\
\vec{V}(t + \delta t) &= \vec{V}(t) + \delta \vec{a}(t) + \frac{1}{2} \delta^2 \vec{b}(t) + \cdots \cdots \cdots \\
\vec{a}(t + \delta t) &= \vec{a}(t) + \delta \vec{b}(t) + \frac{1}{2} \delta^2 \vec{c}(t) + \cdots \cdots \cdots
\end{align*}
\]

where \( \vec{V} \) is the velocity, \( \vec{a} \) is the acceleration and \( \vec{c} \) is the third derivative, and so on.
Verlet Algorithm:

The Verlet algorithm uses the positions and acceleration at time $t$, and the positions from the previous step, $\ddot{r}(t - \delta t)$, to calculate the new position $\ddot{r}(t + \delta t)$, at time $t + \delta t$. Now,

\[
\ddot{r}(t + \delta t) = \ddot{r}(t) + \delta \ddot{V}(t) + \frac{1}{2} \delta^2 \ddot{a}(t) + \cdots
\]

\[
\ddot{r}(t - \delta t) = \ddot{r}(t) - \delta \ddot{V}(t) + \frac{1}{2} \delta^2 \ddot{a}(t) + \cdots
\]

Adding these two equations gives,

\[
\ddot{r}(t + \delta t) = 2\ddot{r}(t) - \ddot{r}(t - \delta t) + \delta^2 \ddot{a}(t)
\]

In Verlet algorithm, velocities do not appear explicitly. One can get the equations for velocities by simply divide the differences in positions at time $(t + \delta t)$ and $(t - \delta t)$: $\ddot{V}(t) = \frac{\ddot{r}(t + \delta t) - \ddot{r}(t - \delta t)}{2 \delta}$. Replacing $t$ to $(t + \frac{1}{2} \delta)$: $\ddot{V}(t + \frac{1}{2} \delta) = \frac{\ddot{r}(t + \delta t) - \ddot{r}(t)}{\delta}$.

Drawback:

(i) Positions $\ddot{r}(t + \delta t)$ are obtained by adding a small term $\delta^2 \ddot{a}(t)$ to the difference of two large terms $2\ddot{r}(t)$ and $\ddot{r}(t - \delta t)$. This leads to a loss of precision.

(ii) There is no explicit velocity term in the equation.

Leap-Frog Algorithm:

\[
\ddot{r}(t + \delta t) = \ddot{r}(t) + \delta \ddot{V}(t) + \frac{1}{2} \delta^2 \ddot{a}(t)
\]

\[
\ddot{V}(t + \frac{1}{2} \delta) = \ddot{V}(t - \frac{1}{2} \delta) + \delta \ddot{a}(t)
\]

The Leap-Frog method has two advantages over the Verlet algorithm. It explicitly includes the velocity in the equation of motion.

Velocity-Verlet Algorithm:

\[
\ddot{r}(t + \delta t) = \ddot{r}(t) + \delta \ddot{V}(t) + \frac{1}{2} \delta^2 \ddot{a}(t)
\]
In this method the new velocities requires the accelerations at both \( t \) and \( (t + \Delta t) \). So, in the first step the positions at \( (t + \Delta t) \) are calculated by using velocities and accelerations at time \( t \). The velocities at time \( (t + \frac{1}{2} \Delta t) \) are then determined using:

\[
\tilde{V}(t + \frac{1}{2} \Delta t) = \tilde{V}(t) + \frac{1}{2} \Delta \tilde{a}(t)
\]

In the final step velocities at time \( (t + \Delta t) \) are determined, using

\[
\tilde{V}(t + \Delta t) = \tilde{V}(t + \frac{1}{2} \Delta t) + \frac{1}{2} \Delta \tilde{a}(t + \Delta t).
\]

This methodology is used in chapter 4, chapter 5 and chapter 6 to simulate the DNA and protein molecules, as well as their complexes.