Лекция №10. Микроканондық ансамбль (NVE-ансамбль) үшін МД әдісі

1. Раман алгоритмі.

T уақыт мезетінде $\vec{r}_i(t)$, $\vec{\upsilon}_i(t)$ және үдеу $\vec{a}_i(t)$ белгілі болсын. Мына теңдік көмегімен $\vec{r}_i(t+\Delta t)$ шамасын алғашқы жуықтауда бағалаймыз:

$$\vec{r}_i(t+\Delta t) = \vec{r}_i(t) + \vec{\upsilon}_i(t)\Delta t\ . \qquad (6)$$

$\vec{r}_i(t+\Delta t)$ көмегімен және потенциалды пайдалана отырып, $\vec{a}_i(t+\Delta t)$ үдеулер бағалы түрде анықталады. Ал содан кейін $(t+\Delta t)$ уақыт аралығындағы жылдамдықтар да анықталады:

$$\vec{\upsilon}_i(t+\Delta t) = \vec{\upsilon}_i(t) + \frac{1}{2}\left[\vec{a}(t+\Delta t) + \vec{a}_i(t)\right]\Delta t\ .$$

Келесі жуықтауда болса, координаттар мәндері үшін:

$$\vec{r}_i(t+\Delta t) = \vec{r}_i(t) + \frac{1}{2}\left[\vec{\upsilon}_i(t+\Delta t) + \vec{\upsilon}_i(t)\right]\Delta t\ .$$

Осы аталып өткен амалдарды алдын-ала болжамданған және нақтыланған мәндер арасындағы итерация кезінде керекті дәлдікке дейін қол жетпейінше қайталауға болады. Содан кейін есептеулердің барлығын келесі уақыт мезеті $(t+2\Delta t)$ үшін қайталаймыз. Міне осылайша Раман алгоритмі «предиктор-корректор» схемасын пайдаланады да жоғары дәлдікке қол жеткізу үшін қосымша машиналық уақыт керек екені айғақ. Нейтрал бөлшектерден тұратын газдың, сұйықтықтың қасиеттерін модельдеу кезінде Раман алгоритмі өз қуаттылығын көрсетті. Ал осы алгоритмді плазманы модельдеуге пайдаланудың еш керегі жоқ, себебі – алыстан әсер ететін кулон күштерін есептеу үшін компьютерлік уақыттың көпке созылатындығында.

2. Верле алгоритмі.

Бұл ең көп тараған алгоритмдердің бірі, өте дәл схемаға ие, себебі ол үшін итерациялық амалды керек қылмайды. Енді $\vec{r}_i(t+\Delta t)$ функциясын Δt дәрежелері бойынша Тэйлор қатарына жіктейік:

$$\vec{r}_i\left(t+\Delta t\right)=\vec{r}_i\left(t\right)+\vec{\upsilon}_i\Delta t+\frac{1}{2}\vec{a}_i\left(t\right)\left(\Delta t\right)^2+\frac{1}{6}\dot{\vec{a}}_i\left(t\right)\left(\Delta t\right)^3+\cdots. \tag{7}$$

Егер $\vec{r}_i(t+\Delta t)$ координаттары белгілі болса, онда мына түрде жазуға болады:

$$\vec{\upsilon}_i\left(t\right)=\frac{1}{2}\left[\vec{r}_i\left(t+\Delta t\right)-\vec{r}_i\left(t-\Delta t\right)\right]/\Delta t. \tag{8}$$

(7) теңдеудегі аздықтың Δt^2 мүшелерімен шектелейік, қозғалыс теңдеуін ескере отырып (8) теңдеуді (7) теңдеуіне апарып салсақ, мынаны аламыз:

$$\vec{\upsilon}_i\left(t+\Delta t\right)=2\vec{r}_i\left(t\right)-\vec{r}_i\left(t-\Delta t\right)+\frac{\vec{F}_i(t)}{m_i}\left(\Delta t\right)^2. \tag{9}$$

(8) және (9) теңдеулері Верле алгоритмін құрайды. Айта кетер бір жай, (8) және (9) формасындағы Верле алгоритмі өзімен өзі іске қосылатындардың бірі емес, басқалай айтқанда, іске қосу үшін бөлшектердің орналасуын тек бастапқы уақыт мезетінде ғана емес, сонымен қатар келесі уақыт мезетінде де білу керек. Іс жүзінде бөлшектердің алғашқы екі уақыт мезетіндегі орналасуын жоғары дәлдікпен бір қадамды әдістер көмегімен табуға болады (Эйлер әдісі, Рунге-Кутта және т.б.).

Верле алгоритмінің тағы басқа жылдамдық формасы бар:

$$\begin{gathered}\vec{r}_i\left(t+\Delta t\right)=\vec{r}_i\left(t\right)+\vec{\upsilon}_i(t)\Delta t+\frac{\left(\Delta t\right)^2}{2m_i}\vec{F}_i(t),\\ \vec{\upsilon}_i(t+\Delta t)=\vec{\upsilon}_i(t)+\left(\frac{\Delta t}{2m_i}\right)\left(\vec{F}_i(t+\Delta t)-\vec{F}_i(t)\right)\end{gathered} \tag{10}$$

Осы формадағы алгоритмнің бірқатар артықшылықтары бар: біріншіден, бөлшектердің координаттары мен жылдамдықтары тұрақты бір уақыт қадамында есептелінеді; екіншіден, алғашқы алгоритмнің формасына қарағанда, берілген алгоритм формасы тұрақтылау болып келеді, ал бұл болса, ұзақ уақыт бойы жүйені моделдеу кезінде маңызы өте зор зат.

3. Биман алгоритмі.

Верле схемасына қарағанда Биман алгоритмін пайдалануда жоғары дәлдікке ие. Себебі, жүйенің энергиясын есептеуге Верле схемасын пайдаланғанда энергияның шамалары біресе өсіп, біресе кеміп тұратындығында. $\vec{r}_i(t+\Delta t)$ функциясын жіктегенде (7) теңдеуден $\sim O(\Delta t^3)$ шамасына пропорционал мүшелерді сақтап және

$$\dot{\vec{a}}_i(t)\Delta t \cong \vec{a}_i(t) - \vec{a}_i(t-\Delta t),$$

деп ұйғарсақ, мынаны аламыз:

$$\vec{r}_i(t+\Delta t) = \vec{r}_i(t) + \vec{\upsilon}_i\Delta t + \frac{1}{6}\left[4\vec{a}_i(t) - \vec{a}_i(t-\Delta t)\right](\Delta t)^2 . \tag{11}$$

Жылдамдық үшін алгоритм (11) теңдеуі арқылы және (8) теңдгін дифференциалдау арқылы анықталады, бірақ $2\Delta t$ орнына Δt деп аламыз:

$$\vec{\upsilon}_i\left(t+\frac{1}{2}\Delta t\right) \cong \left[\vec{r}_i(t+\Delta t) - \vec{r}_i(t)\right]/(\Delta t) = \vec{\upsilon}_i(t) + \frac{1}{6}\left[4\vec{a}_i(t) - \vec{a}_i(t-\Delta t)\right](\Delta t).$$

Соңғы теңдеудің аргументтінің барлығына Δt қосып отырып, мынаны аламыз:

$$\vec{\upsilon}_i\left(t+\frac{3}{2}\Delta t\right) \cong \vec{\upsilon}_i(t+\Delta t) + \frac{1}{6}\left[4\vec{a}_i(t+\Delta t) - \vec{a}_i(t)\right]\Delta t .$$

Жуықтау қатынасы көмегімен

$$\vec{a}_i(t+\Delta t)\Delta t \cong \vec{\upsilon}_i\left(t+\frac{3}{2}\Delta t\right) - \vec{\upsilon}_i\left(t+\frac{1}{2}\Delta t\right)$$

және оны (12) теңдеуіне қойып, кейбір амалдарды орындағаннан кейін мынаны аламыз:

$$\vec{\upsilon}_i(t+\Delta t) = \vec{\upsilon}_i(t) + \frac{1}{6}\left[2\vec{a}_i(t+\Delta t) + 5\vec{a}_i(t) - \vec{a}_i(t-\Delta t)\right]\Delta t . \tag{12}$$

(11) және (12) қатынастары Биман алгоритмін құрайды.

NVE-ансамблі үшін МД әдісінің алгоритмі.

1. Бөлшектердің жылдамдықтары мен координаттарының бастапқы мәндерін беру:

2. (6)-ға сәйкес i-ші бөлшекке басқа бөлшектер тарапынан әсер ететін $\vec{F}_i(t)$ күшті есептеу;
3. Интегралдау алгоритміне сәйкес $\vec{r}_i(t+\Delta t)$ уақыт мезетіндегі бөлшектердің қозғалыс координаттарының теңдеулерін есептеу;
4. Интегралдау алгоритміне сәйкес $(t+\Delta t)$: $\vec{\upsilon}_i(t+\Delta t)$ уақыт мезетіндегі бөлшектердің қозғалыс жылдамдықтарының теңдеулерін есептеу;
5. $\vec{r}_i(t)$ және $\vec{\upsilon}_i(t)$ шамаларының қазіргі мәндерін компьютердің сыртқы құрылғыларына жинақтау;
6. 2) және 5) пунктерін қайталап орындау.

Сонда жүйені МД әдісімен модельдеу арқасында жүйенің координаттарын, энергиясын, жылдамдықтарын және тағы басқа параметрлерін кез-келген уақыт мезеті үшін табуға болады. Жалпы айтқанда классикалық механиканың негізгі есебі бірмәнді шешіледі.

Енді Δt интегралдау қадамын таңдауға тоқталайық. Гидродинамикалық шекте, сонымен қатар фазалық кеңістіктің үлкен аймағында моделдеу уақыты $T = n\Delta t$ өте ұзақ болуы мүмкін. Компьютерлік уақыт n-ге пропорционал, сондықтан Δt шамасын неғұрылым үлкен етіп таңдау керек. Алайда Δt шамасы өскен сайын сандық интегралдаудың дәлдігі өте тез кеми түседі және жүйенің толық энергиясының флуктуациялары ұлғаяды. Δt шамасының жеке мәндері үшін дәлдікке предиктор-корректор (Раман) типтес алгоритмдері арқасында жетуге болады. Бірақ ол үшін өте көп компьютерлік уақыт кетер еді. Плазманы модельдеу кезінде қойылатын шарттардың бірі келесі шарттың орындалуы болып табылады:

$$\Delta t << t_0 \tag{13}$$

мұндағы $t_0=a_Б/\upsilon_T$; $a_Б$ – бірінші Бор радиусы; υ_T – жылулық жылдамдық.