

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ ДЛЯ РЕШЕНИЯ ПРОБЛЕМЫ ЗНАКА В ФИЗИЧЕСКИХ МОДЕЛЯХ

М. В. Улыбышев^{1, *}, *В. И. Дорожнинский*², *О. В. Павловский*^{1, 2, **}

¹ Институт теоретической и экспериментальной физики им. А. И. Алиханова
Национального исследовательского центра «Курчатовский институт», Москва

² Московский государственный университет им. М. В. Ломоносова, Москва

В работе изучалась возможность ослабления проблемы знака, возникающей при исследовании фермионных систем с конечным химическим потенциалом, с помощью алгоритмов нейронных сетей. Решение проблемы знака крайне важно как для многих актуальных задач физики конденсированного состояния вещества, так и для физики кварк-глюонной плазмы высокой плотности — нового состояния вещества, свойства которого будут изучаться на строящихся ускорителях FAIR и NICA. Обученные нейронные сети в предложенном подходе приближенно воспроизводят «наперстки Лefшца» — множества, на которых мнимая часть действия является константой. Показано, что обученная сеть существенно ускоряет построение множества в комплексном пространстве, по которому производится интегрирование, по сравнению с использованием обычного алгоритма градиентного потока. Также показано, что флуктуации мнимой части действия на приближенном множестве, определенном нейронной сетью, все еще гораздо меньше, чем в обычном методе перевешивания.

We study the possibility to weaken the sign problem, which appears in the investigation of fermionic systems with finite chemical potential, using the neural networks. Solution of the sign problem is extremely important both for numerous problems in condensed matter physics as well as for the physics of quark–gluon plasma: a new state of matter to be studied using the new accelerators FAIR and NICA, which are both under construction now. In our approach, the trained neural network approximately reproduces Lefschetz thimbles: manifolds in complex space, where the imaginary part of the action is constant. We show that the trained neural network can substantially speed up the construction of the integration manifold in complex space, in comparison with usually employed gradient flow algorithm. We also show that the fluctuations of the imaginary part of the action on the approximation manifold defined by the neural network are still substantially smaller than in usual reweighting method.

PACS: 05.50.+q; 75.10.Hk; 02.70.-c; 02.70.Uu

*E-mail: ulybyshev@goa.bog.msu.ru

**E-mail: pavlovsky@physics.msu.ru

ВВЕДЕНИЕ

Вычисление методом Монте-Карло фейнмановского интеграла по путям — это уникальный способ для непертурбативного изучения физики сильно связанных квантовых систем большого числа частиц без каких-либо физических предположений и упрощений. Этот метод обычно применяется к евклидовым интегралам по путям, поскольку после выполнения поворота Вика из пространства Минковского в евклидово пространство действие системы становится действительным, и, таким образом, подынтегральная функция больше не является сильно осциллирующей. Последнее означает, что для вычисления интегралов по путям в евклидовом пространстве могут применяться статистические методы, такие как метод Монте-Карло. Однако существует много систем, представляющих теоретический и экспериментальный интерес, в которых даже евклидово действие является комплексным. Как правило, такие задачи возникают при интегрировании систем фермионов. В таких случаях мы не можем применить метод существенной выборки без значительных модификаций: трудно, а часто даже невозможно выбрать соответствующее положительно определенное распределение вероятностей. Возникающая проблема называется проблемой знака.

Проблема знака является одной из основных проблем в современной вычислительной физике. В частности, она препятствует применению высокопроизводительного метода квантового Монте-Карло (КМК) для многочисленных моделей, очень интересных с физической точки зрения.

Наиболее ярким примером в физике конденсированных сред является модель Хаббарда на бипартитной решетке с ненулевым химическим потенциалом, которая имеет особое значение в силу того, что она, возможно, может быть применена для описания физики высокотемпературных сверхпроводников. Ряд других моделей для квантовых систем (например модель Хаббарда на треугольной решетке) также подвержены проблеме знака. Еще одним примером из физики конденсированного состояния является унитарный ферми-газ. Эта система может быть получена экспериментально в лаборатории [1–3].

В физике высоких энергий проблема знака также играет существенную роль. Наиболее ярким примером в области физики высоких энергий является решеточная квантовая хромодинамика (КХД), опять же с конечным химическим потенциалом. Из-за проблемы знака становится достаточно сложным исследование непертурбативных явлений при конечной барионной плотности. Эта проблема ограничивает наши знания о фазовой диаграмме КХД, которая связана с другими важными проблемами, начиная от физики ранней Вселенной и заканчивая нейтронными звездами и космологией. Теоретические исследования фазовой картины КХД привлекают большое внимание, различные аспекты этих явлений очень активно изучаются также экспериментально на ЛНС и некоторых других установках, существующих и строящихся в настоя-

щее время, таких как RHIC, FAIR и NICA. Кроме того, хорошо известно, что калибровочные теории с добавлением θ -члена имеют комплексное евклидово действие.

Конечно, известны различные приближенные методы для работы со всеми вышеупомянутыми системами, но все же применение квантового Монте-Карло (КМК) было бы выгодно, так как этот метод не привлекает никаких дополнительных физических предположений, кроме тех, которые были сделаны при построении модели. Таким образом, как только проблема знака будет решена (или сделана достаточно слабой), роль КМК резко возрастет, так как этот метод может дать ответ с заданной числовой точностью для данного квантового гамильтониана.

На самом деле модели с комплексным евклидовым действием встречаются настолько часто, что можно утверждать, что системы с проблемой знака являются не исключением, а, скорее, составляют большинство. К сожалению, Тройер и Визе [4] показали, что проблема знака является NP-полной задачей по крайней мере в одном конкретном случае. Следовательно, для проблемы знака вряд ли будет найдено общее решение.

Сложность проблемы может быть легко проиллюстрирована, если мы попробуем справиться с проблемой знака напрямую, «грубой силой». Такой метод существует и известен как метод перевзвешивания. Он основан на соотношении

$$\begin{aligned} \langle \mathcal{O} \rangle &= \frac{1}{\mathcal{Z}} \int \mathcal{D}\Phi \mathcal{O}[\Phi] e^{-S[\Phi]} = \frac{\int \mathcal{D}\Phi \mathcal{O}[\Phi] e^{-S[\Phi]}}{\int \mathcal{D}\Phi e^{-S[\Phi]}} = \\ &= \frac{1}{\mathcal{Z}_{\text{pq}}} \int \mathcal{D}\Phi \mathcal{O}[\Phi] \frac{e^{-S[\Phi]}}{e^{-S_R[\Phi]}} e^{-S_R[\Phi]} = \frac{\langle \mathcal{O} e^{-iS_I} \rangle_{S_R}}{\langle e^{-iS_I} \rangle_{S_R}}, \quad (1) \\ &= \frac{1}{\mathcal{Z}_{\text{pq}}} \int \mathcal{D}\Phi \frac{e^{-S[\Phi]}}{e^{-S_R[\Phi]}} e^{-S_R[\Phi]} \end{aligned}$$

где $S = S_R + iS_I$, а отношение $e^{-S[\Phi]}/e^{-S_R[\Phi]}$ является фактором перевзвешивания. $\mathcal{Z}_{\text{pq}} = \int \mathcal{D}\Phi e^{-S_R[\Phi]}$ — статистическая сумма без учета комплексной фазы. Угловые скобки обозначают среднее по ансамблю с мерой $\mathcal{D}\Phi e^{-S_R}$. Несмотря на то, что это просто эквивалентные преобразования, практическое применение метода перевзвешивания ведет к экспоненциально растущим затратам вычислительных ресурсов.

Основная проблема — это пересечение ансамбля конфигураций, в котором используется только реальная часть действия, с изначальным целевым ансамблем конфигураций, который включает в себя полное действие. Величину $\langle e^{-S_I} \rangle_{S_R}$ можно рассматривать как отношение двух статистических сумм $\langle \mathcal{Z} / \mathcal{Z}_{\text{pq}} \rangle = e^{-V\beta\Delta f}$, где $\mathcal{Z}_{\text{pq}} = \int \mathcal{D}\Phi e^{-S_R[\Phi]}$ — статистическая сумма с исключенной фазой. Здесь мы также ввели объем системы V , обратную температуру β и Δf , которая является разницей плотности свободной энергии

между двумя ансамблями. Связь между V , β и средним знаком показывает, что сложность проблемы знака экспоненциально зависит от объема и обратной температуры. Таким образом, как правило, с помощью простого метода перевзвешивания можно добиться лишь незначительного прогресса.

Для решения проблемы знака применяются различные аналитические и численные подходы. Например, многие интересные результаты были получены методом разложения действия по химическому потенциалу [5, 6]. Этот метод показал свою высокую эффективность, но, к сожалению, ограничен в применении только областью малых значений химического потенциала. Также весьма перспективным во многих задачах выглядит метод канонического производящего функционала [7]. Этот метод требует существенных вычислительных ресурсов, и его применимость пока весьма ограничена.

В последние годы был достигнут некоторый прогресс в преодолении проблемы знака благодаря идее, заключающейся в аналитическом продолжении полей в комплексную плоскость. В результате чего для некоторых систем было продемонстрировано, что проблема знака, весьма серьезная в исходной формулировке, облегчается или иногда даже устраняется. Эта идея, которую легко продемонстрировать в простых одномерных интегралах, нашла применение в нескольких нетривиальных физических системах. В настоящее время используется два подхода с комплексными полями — комплексная ланжевеновская динамика [8–15] и метод «наперстков Лефшеца» [16–26, 28–33].

В последние годы метод «наперстков Лефшеца» является наиболее привлекательным, поскольку он опирается на строгие математические теоремы и имеет строгое математическое обоснование. В традиционных подходах КМК мы работаем с континуальным интегралом по действительным непрерывным полям, Φ . Если рассмотреть продолжение континуального интеграла в область комплексных полей, $\Phi \in \mathbb{C}^N$, то по теореме Коши для интегрирования можно выбрать любой подходящий контур в таком комплексном пространстве. Если некоторые специальные точки (например полюсы) подынтегральной функции не пересекаются во время сдвига контура интегрирования, то интеграл остается тем же самым. Как правило, подынтегральное выражение в (1) не имеет полюсов, поэтому переход к комплексному полю является безопасной процедурой. Контур с особенно интересными свойствами может быть построен с помощью теории Морса (или теории Пикара–Лефшеца). Он известен как разложение Лефшеца для интеграла по путям. Рассмотрим идею подхода на примере конкретной физической модели.

1. ФИЗИЧЕСКАЯ МОДЕЛЬ

Алгоритмы квантового Монте-Карло обычно имеют дело со статистической суммой системы

$$\mathcal{Z} = \text{Tr} e^{-\beta \hat{H}} \quad (2)$$

и соответствующими термодинамическими средними для наблюдаемых

$$\langle \hat{O} \rangle = \frac{1}{Z} \text{Tr} (\hat{O} e^{-\beta \hat{H}}). \quad (3)$$

Здесь \hat{H} и \hat{O} — гамильтониан и наблюдаемая соответственно и β — обратная температура. Рассмотрим гамильтониан фермионной системы \hat{H} , состоящий из двух частей:

$$\hat{H} = \hat{H}_{(2)} + \hat{H}_{(4)} = \sum_{x,y,\sigma,\sigma'} t_{xy\sigma\sigma'} \hat{c}_{x\sigma}^\dagger \hat{c}_{y\sigma'} + \sum_{x,y,\sigma,\sigma'} U_{xy\sigma\sigma'} \hat{n}_{x\sigma} \hat{n}_{y\sigma'}, \quad (4)$$

где индексы x и y обозначают узлы решетки, $\sigma, \sigma' = \uparrow, \downarrow$ соответствуют спиновым индексам, а $\hat{n}_{x\sigma} = \hat{c}_{x\sigma}^\dagger \hat{c}_{x\sigma}$ — оператор числа частиц на узле. Первая часть содержит только билинейные фермионные слагаемые, которые включают в себя кинетическую энергию, а также химический потенциал. Вторая часть содержит четырехфермионные члены, описывающие электрон-электронное взаимодействие. В (4) показан наиболее общий вид этого взаимодействия. Такую модель можно называть «расширенной моделью Хаббарда», так как сама модель Хаббарда включает только локальное взаимодействие в узлах $\sum_x U \hat{n}_{x\uparrow} \hat{n}_{x\downarrow}$.

Представление статистической суммы в виде интеграла по путям (2) начинается с разложения Троттера:

$$\text{Tr} e^{-\beta \hat{H}} \approx \text{Tr} \left(e^{-\delta \hat{H}_{(2)}} e^{-\delta \hat{H}_{(4)}} e^{-\delta \hat{H}_{(2)}} e^{-\delta \hat{H}_{(4)}} \dots \right). \quad (5)$$

После разложения мы получаем произведение $2N_t$ экспонент, которые составляют евклидово временное измерение решетки. δ — это шаг в евклидовом времени: $N_t \delta = \beta$. Чтобы конвертировать след в интегралы по путям, вводятся грассмановы когерентные состояния $|\xi\rangle$ и грассмановы переменные ξ для каждого оператора рождения и уничтожения. Дальнейшие подробности построения представления в виде интеграла по путям можно найти в [34–36], где это было сделано для модели Хаббарда–Кулона на гексагональной решетке.

Мы хотели бы выделить один этап в этом выводе, который важен для нашего исследования проблемы знака. А именно, преобразование четырехфермионной части гамильтониана $\hat{H}_{(4)}$ в билинейную. Этот шаг важен, поскольку для билинейных членов в показателе степени мы имеем простой набор соотношений, который позволяет преобразовать многомерный интеграл по грассмановым переменным в форму, удобную для алгоритма Монте-Карло.

Рассмотрим вариант, основанный на обычном гауссовом преобразовании Хаббарда–Стратоновича:

$$\begin{aligned} \exp\left(-\frac{\delta}{2} \sum_{x,y} U_{x,y} \hat{n}_x \hat{n}_y\right) &\cong \\ &\cong \int D\phi_x \exp\left(-\frac{1}{2\delta} \sum_{x,y} \phi_x U_{xy}^{-1} \phi_y\right) \exp\left(i \sum_x \phi_x \hat{n}_x\right), \quad (6) \end{aligned}$$

$$\begin{aligned} \exp\left(\frac{\delta}{2} \sum_{x,y} U_{x,y} \hat{n}_x \hat{n}_y\right) &\cong \\ &\cong \int D\phi_x \exp\left(-\frac{1}{2\delta} \sum_{x,y} \phi_x U_{xy}^{-1} \phi_y\right) \exp\left(\sum_x \phi_x \hat{n}_x\right). \quad (7) \end{aligned}$$

Это преобразование может приводить к действительным (7) и мнимым (6) экспонентам. Важное преимущество этого преобразования в том, что оно также работает для нелокальных взаимодействий, поэтому нам не нужно вводить новое дополнительное поле для каждой пары взаимодействующих электронов и количество хаббардовских полей равно количеству узлов решетки. Это свойство используется, например, для модели Хаббарда–Кулона [35, 37–40].

Теперь перейдем к описанию проблемы знака в данной задаче. Отметим, что в особых случаях, когда существуют некоторые дополнительные симметрии (например симметрия обращения времен [41]), расширенная модель Хаббарда поддается моделированию с помощью квантового Монте-Карло. В частности, это возможно в случае бипартитной решетки. Таким образом, мы собираемся сосредоточиться на следующем гамильтониане, действующем на бипартитной решетке, причем только с хаббардовским взаимодействием:

$$\hat{H} = -\kappa \sum_{\langle x,y \rangle, \sigma} \hat{c}_{x\sigma}^\dagger \hat{c}_{y\sigma} + U \sum_x \hat{n}_{x\uparrow} \hat{n}_{x\downarrow} - \left(\frac{U}{2} - \mu\right) \sum_x (\hat{n}_{x\uparrow} + \hat{n}_{x\downarrow} - 1). \quad (8)$$

Кинетическая энергия включает в себя исключительно туннелирование электронов между ближайшими соседями. Химический потенциал μ определяет сдвиг от половинного заполнения, которое соответствует $\mu = 0,0$. В идеальной ситуации (при отсутствии проблемы знака), алгоритмы квантового Монте-Карло требуют как минимум неотрицательного веса для хаббардовских полей. Бипартитная решетка предоставляет нам такую возможность при половинном заполнении, после хорошо известного трюка, который преобразует электроны со спином вверх и вниз ($\hat{c}_{x,\uparrow}$ и $\hat{c}_{x,\downarrow}$) в электроны и дырки (\hat{a}_x и \hat{b}_x):

$$\begin{cases} \hat{c}_{x,\uparrow}, \hat{c}_{x,\uparrow}^\dagger \rightarrow \hat{a}_x, \hat{a}_x^\dagger, \\ \hat{c}_{x,\downarrow}, \hat{c}_{x,\downarrow}^\dagger \rightarrow \pm \hat{b}_x, \pm \hat{b}_x^\dagger. \end{cases} \quad (9)$$

Знак во второй строке чередуется в зависимости от подрешетки. Гамильтониан (8) приобретает следующий вид после перехода к новым переменным:

$$\hat{H} = -\kappa \sum_{\langle x,y \rangle} (\hat{a}_x^\dagger \hat{a}_y + \hat{b}_x^\dagger \hat{b}_y) + \frac{U}{2} \sum_x (\hat{n}_{x,\text{el}} - \hat{n}_{x,h})^2 + \mu \sum_x (\hat{n}_{x,\text{el}} - \hat{n}_{x,h}), \quad (10)$$

где $\hat{n}_{x,\text{el}} = \hat{a}_x^\dagger \hat{a}_x$ и $\hat{n}_{x,h} = \hat{b}_x^\dagger \hat{b}_x$ — операторы числа частиц для электронов и дырок соответственно.

В случае непрерывных хаббардовских полей можно написать преобразование Хаббарда–Стратоновича более общим способом, используя как вещественные (7), так и комплексные (6) экспоненты:

$$\frac{U}{2} (\hat{n}_{\text{el}} - \hat{n}_h)^2 = \frac{\alpha U}{2} (\hat{n}_{\text{el}} - \hat{n}_h)^2 - \frac{(1-\alpha)U}{2} (\hat{n}_{\text{el}} + \hat{n}_h)^2 + (1-\alpha)U (\hat{n}_{\text{el}} + \hat{n}_h). \quad (11)$$

Параметр $\alpha \in [0, 1]$ определяет баланс между действительными и комплексными показателями в интеграле. Первый четырехфермионный член можно преобразовать в билинейный, используя (6), а второй — используя (7). Хотя это и не самое общее возможное разложение четырехфермионных членов на билинейные, зато оно наиболее часто используется в алгоритмах квантового Монте-Карло с непрерывными хаббардовскими полями. Это представление было впервые предложено в [42] и использовано в недавней статье [43].

В итоге статистическая сумма может быть записана как следующий интеграл:

$$\begin{aligned} \mathcal{Z}_c &= \int \mathcal{D}\phi_{x,t} \chi_{x,t} \exp^{-S_\alpha} \det M_{\text{el}} \det M_h, \\ S_\alpha(\phi_{x,t}, \chi_{x,t}) &= \sum_{x,t} \frac{\phi_{x,t}^2}{2\alpha\delta U} + \sum_{x,t} \frac{(\chi_{x,t} - (1-\alpha)\delta U)^2}{2(1-\alpha)\delta U}, \end{aligned} \quad (12)$$

где фермионные операторы для непрерывных хаббардовских полей записываются как

$$\begin{aligned} M_{\text{el}} &= I + \prod_{t=1}^{N_t} \left(e^{-\delta(h+\mu)} \text{diag} e^{i\phi_{x,t} + \chi_{x,t}} \right), \\ M_h &= I + \prod_{t=1}^{N_t} \left(e^{-\delta(h-\mu)} \text{diag} e^{-i\phi_{x,t} + \chi_{x,t}} \right). \end{aligned} \quad (13)$$

В дальнейшем мы будем иметь дело с полным действием, которое включает в себя как квадратичную форму S_α , так и логарифмы детерминантов фермионных операторов:

$$S = S_\alpha - \ln(\det M_{\text{el}} \det M_h). \quad (14)$$

Отдельно укажем, что все параметры гамильтониана (10), а также температура будут в дальнейшем указываться в единицах κ . Если все отношения размерных параметров к параметру хоппинга (β/κ , U/κ , μ/κ) фиксированы, то физика уже никак не зависит от значения κ . Параметр хоппинга выступает в этом случае просто как размерная константа, задающая общий энергетический масштаб и используемая для обезразмеривания всех остальных параметров модели.

2. МЕТОД РАЗЛОЖЕНИЯ ПО «НАПЕРСТКАМ»

Чтобы проиллюстрировать основные идеи метода разложения по «наперсткам», начнем с наиболее общего вида интегралов, появляющихся в КМК с непрерывными хаббардовскими полями:

$$\mathcal{Z}(\beta, \mu, \dots) = \int_{\mathbb{R}^N} d^N x e^{-S(\beta, \mu, \dots, x)}. \quad (15)$$

Если мы рассмотрим продолжение этого интеграла в область комплексных переменных $x \in \mathbb{C}^N$, то по теореме Коши можно выбрать любой подходящий контур в комплексном пространстве для интегрирования. Представление с особенно полезными свойствами может быть построено с помощью теории Морса (или теории Пикара–Лефшеца). Оно известно как разложение Лефшеца для интеграла по путям [16, 17]:

$$\mathcal{Z}(\beta, \mu, \dots) = \sum_{\sigma} k_{\sigma}(\beta, \mu, \dots) \mathcal{Z}_{\sigma}(\beta, \mu, \dots), \quad (16)$$

$$\mathcal{Z}_{\sigma}(\beta, \mu, \dots) = \int_{\mathcal{I}_{\sigma}(\beta, \mu, \dots)} d^N x e^{-S(\beta, \mu, \dots, x)}, \quad (17)$$

где σ обозначает все комплексные седловые точки $z_{\sigma}(\beta, \mu, \dots) \in \mathbb{C}$:

$$\left. \frac{\partial S}{\partial x} \right|_{x=z_{\sigma}(\beta, \mu, \dots)} = 0, \quad (18)$$

целочисленные коэффициенты $k_{\sigma}(\beta, \mu, \dots)$ — это так называемые числа пересечений; $\mathcal{I}_{\sigma}(\beta, \mu, \dots)$ — поверхности наискорейшего спуска. В данных выражениях для ясности мы указали зависимость всех важных величин от параметров модели, в дальнейшем мы будем ее опускать. Соотношение (16) справедливо, если седловые точки невырождены и изолированы (для обобщения на случай калибровочной теории см. [16]). Вырожденные седловые точки могут появиться из-за спонтанного нарушения некоторой непрерывной симметрии. В таком случае симметрия должна быть явно нарушена некоторым небольшим членом в гамильтониане, и все результаты должны быть экстраполированы к пределу восстановления этой симметрии.

Для того чтобы построить «наперсток Лефшеца» \mathcal{I}_σ , соответствующий заданной седловой точке z_σ , используем уравнение градиентного потока

$$\frac{dx}{d\tau} = \frac{\partial S}{\partial x} \quad (19)$$

со следующими граничными условиями, соответствующими началу в седловой точке:

$$x \in \mathcal{I}_\sigma : x(\tau) = x, x(\tau \rightarrow -\infty) \rightarrow z_\sigma. \quad (20)$$

Это уравнение определяет эволюцию комплексной переменной x относительно фиктивного времени течения τ , и все такие решения составляют «наперсток».

Аналогично мы определяем еще один важный тип многообразия, так называемый «антинаперсток» \mathcal{K}_σ , который состоит из всех возможных решений уравнений потока (19), которые заканчиваются в данной седловой точке z_σ :

$$x \in \mathcal{K}_\sigma : x(\tau) = x, x(\tau \rightarrow +\infty) \rightarrow z_\sigma. \quad (21)$$

С помощью «антинаперстков» можно вычислить целочисленные коэффициенты k_σ в выражении (16), подсчитав количество пересечений \mathcal{K}_σ с оригинальным контуром интегрирования \mathbb{R}^N :

$$k_\sigma = \langle \mathcal{K}_\sigma, \mathbb{R}^N \rangle. \quad (22)$$

Как «наперстки», так и «антинаперстки» являются N -размерными действительными поверхностями в \mathbb{C}^N . Два основных свойства, которые делают их полезными, следующие. Во-первых, реальная часть действия $\text{Re } S$ монотонно возрастает вдоль «наперстка» и монотонно убывает по «антинаперстку», если мы начинаем с седловой точки. Во-вторых, мнимая часть действия $\text{Im } S$ не меняется вдоль каждого из них. Отсюда следует, что ни «наперстки», ни «антинаперстки» не могут пересекаться друг с другом. Также в общем случае ни одна пара седловых точек не может быть соединена каким-либо «наперстком» и все интегралы в правой части выражения (16) сходятся.

Именно благодаря постоянным комплексным фазам на «наперстках» этот метод стал привлекательным для изучения проблемы знака в КМК. Тем не менее существует еще остаточная проблема знака из-за нетривиального комплексного элемента объема на «наперстке», которая, однако, может быть преодолена. На практике «наперстки» можно построить, используя их касательные пространства вблизи седловых точек. А именно, в каждой седловой точке мы можем вычислить $2N \times 2N$ матрицу вторых производных $\text{Re } S$ по вещественной и мнимой частям комплексной переменной x . Эта матрица содержит ровно N положительных и N отрицательных собственных значений. Соответствующие собственные векторы определяют касательные пространства для «наперстка» и «антинаперстка» и предоставляют нам начальные условия для уравнений потока.

Отметим, что несколько особняком к рассматриваемой проблеме стоит так называемое явление Стокса. Это явление происходит тогда, когда при некоторых значениях параметров существуют две или более седловые точки, соединенные некоторым «наперстком». Это может произойти только тогда, когда мнимая часть действия в этих седловых точках совпадает: $\text{Im } S(z_\sigma) = \text{Im } S(z_{\sigma'})$. Тогда интегралы по «наперсткам» \mathcal{Z}_σ , связанные с этими седловыми точками, испытывают скачки, которые должны быть компенсированы скачком в коэффициентах k_σ для того, чтобы обеспечить справедливость теоремы Коши. Скачки в числах пересечений появляются из-за изменения структуры «антинаперстков». Следовательно, некоторые коэффициенты k_σ могут стать ненулевыми или наоборот обратиться в нуль, а структура суммы (16) может резко измениться, поэтому любой разумный алгоритм КМК, основанный на разложении по «наперсткам», должен правильно их учитывать. Это делает непосредственное применение разложения по «наперсткам» очень непрактичным, однако само существование такого разложения мотивирует разработку алгоритмов, которые будут приближаться к «наперсткам» некоторым автоматическим способом и минимизировать проблему знака.

Проблема знака, порожденная флуктуирующей фазой в (15), заменяется проблемой знака, порожденной различными фазовыми множителями, фигурирующими в сумме по «наперсткам» (16):

$$\mathcal{Z} = \sum_{\sigma} k_{\sigma} e^{-i \text{Im } S(z_{\sigma})} \int_{\mathcal{I}_{\sigma}} d^N x e^{-\text{Re } S(x)}, \quad (23)$$

где мы явно записываем комплексные факторы, связанные с различными седловыми точками. Мы говорим, что «наперсток» релевантен, если он имеет ненулевое число пересечения k_{σ} и таким образом участвует в этой сумме. Количество релевантных «наперстков», их вес и распределение мнимой части действия в соответствующих (релевантных) седловых точках определяют остаточную сложность проблемы знака. Чем меньше число релевантных седловых точек, тем менее серьезна проблема знака в (16). Идеальная ситуация, конечно, когда у нас есть только один релевантный «наперсток» или если в сумме важен только один «наперсток» (16) из-за доминирующего абсолютного значения интеграла по нему.

3. ВЫБОР КОНТУРА ИНТЕГРИРОВАНИЯ ДЛЯ МОДЕЛИ ХАББАРДА

Общая схема деформации контура интегрирования представлена на рис. 1. Последовательность деформаций можно описать следующими образом. Вначале мы производим равномерный сдвиг в комплексную плоскость ($\Phi \rightarrow$

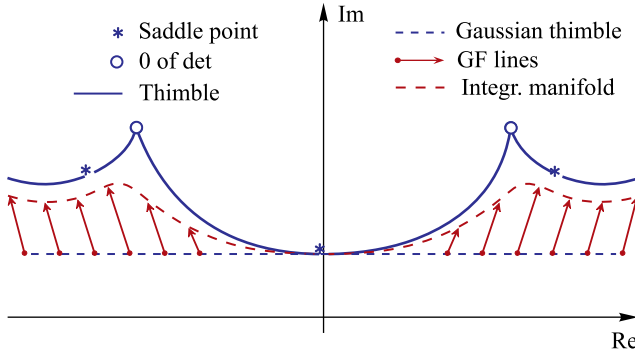


Рис. 1. Схема потока гибридного Монте-Карло

$\Phi + i\Phi_0$). Из-за симметрии

$$S[\text{Re } \Phi + i \text{Im } \Phi] = S[-\text{Re } \Phi + i \text{Im } \Phi]$$

сдвиг соответствует «наперстку», присоединенному к вакуумной седловой точке в гауссовом приближении к действию. Дальнейший сдвиг производится с помощью уравнений градиентного потока (GF). Обозначим $\tilde{\Phi} \in \mathbb{C}^N$ как результат эволюции поля начиная с гауссова «наперстка» $\Phi + i\Phi_0$, $\Phi \in \mathbb{R}$ с временем потока \mathcal{T} . Якобиан трансформаций,

$$J = \frac{\mathcal{D}\tilde{\Phi}}{\mathcal{D}\Phi},$$

появляется на второй стадии. Время потока определяет, как близко мы можем подойти к «наперстку», и таким образом регулирует флуктуации $\text{Im } S$. Так как якобиан является комплексным числом, он тоже может вносить вклад в проблему знака, особенно в случае «вертикально» ориентированных «наперстков». В кратком виде эти преобразования могут быть выражены как

$$\mathcal{Z} = \int_{\mathcal{I}^N} \mathcal{D}^N \tilde{\Phi} e^{-S[\tilde{\Phi}]} = \int_{\mathbb{R}^N} \mathcal{D}\Phi e^{-S[\tilde{\Phi}(\Phi+i\Phi_0)]} \det J|_{\mathbb{R}^N \rightarrow \mathcal{I}^N}. \quad (24)$$

Используем следующую стратегию для генерации полевых конфигураций Φ и $\tilde{\Phi}$ в соответствии со статистической суммой (24):

1) якобиан не учитывается в марковском процессе генерации конфигураций $\tilde{\Phi}$ и будет учтен во время финального перевзвешивания;

2) поля Φ генерируются при помощи гибридного Монте-Карло в соответствии с распределением $\exp \left\{ -\text{Re } S[\tilde{\Phi}(\Phi + i\Phi_0)] \right\}$;

3) поля $\tilde{\Phi}$ вычисляются с помощью градиентного потока. Вторая стадия требует дополнительного пояснения.

В гибридном Монте-Карло используются глобальные обновления полей в марковском случайном процессе. Обновления осуществляются при помощи эффективной гамильтоновой динамики, управляемой гамильтонианом

$$H = \frac{1}{2} \sum_i p_i^2 + \text{Re } S[\tilde{\Phi}(\Phi + i\Phi_0)],$$

где искусственный импульс p_i вводится для каждого хаббардовского поля Φ_i . Для того чтобы решить получающееся в результате уравнение Гамильтона, необходимо посчитать производную $\partial \text{Re } S[\tilde{\Phi}(\Phi + i\Phi_0)]/\partial \Phi_i$. Мы вычисляем эту величину, смещая изначальные поля $\Phi_i \rightarrow \Phi_i + \delta$ и решая уравнения градиентного потока для каждого сдвига. Это позволяет посчитать разницу действий после потока, а также все элементы якобиана. Тем не менее на практике нам нужен $\det J$ только в конце траектории, в то время как траектория обычно состоит из $O(10^2)$ шагов. Таким образом, вычисление якобиана требует относительно мало вычислительных ресурсов. Конечная сложность алгоритма оказывается $C_1 N_S^4 N_\tau^2 + C_2 N_S^3 N_\tau^3$, где первый член соответствует процедуре генерации конфигураций гибридного Монте-Карло, а второй член отвечает за вычисление $\det J$.

Результирующее значение наблюдаемой вычисляется с помощью перевешивания

$$\langle \mathcal{O} \rangle = \frac{\langle \mathcal{O} \exp^{[i \text{Im}(S + \ln \det J) + \text{Re}(\ln \det J)]} \rangle}{\langle \exp^{[i \text{Im}(S + \ln \det J) + \text{Re}(\ln \det J)]} \rangle}, \quad (25)$$

где скобки $\langle \rangle$ обозначают усреднение по конфигурациям, сгенерированным описанным ранее алгоритмом. Учитывая симметрию действия относительно действительных частей полей, мы используем следующую метрику для оценки сложности проблемы знака: $\langle \cos(\text{Im } S) \rangle$ и $\langle \cos(\text{Im } \ln \det J) \rangle$ для конфигурации и якобиана соответственно, и общий знак

$$\Sigma_G = \langle \cos(\text{Im}(S + \ln \det J)) \rangle.$$

Мы также оцениваем силу флуктуаций якобиана, вычисляя дисперсию $\text{Re}(\ln \det J)$.

4. ПРЕДСТАВЛЕНИЕ ДАННЫХ И ПОСТАНОВКА ЗАДАЧИ

Данные, с которыми нам предстоит работать, представляют собой набор полей в $2 + 1$ пространстве, задаваемый комплексным значением в узлах гексагональной решетки. Для удобства представим гексагональную решетку как

набор из двух ромбовидных подрешеток одинакового размера, но смещенных друг относительно друга (рис. 2). В таком случае индекс любого узла можно задать с помощью четырех чисел: номера временного слоя, индекса подрешетки и координат внутри подрешетки.

Как правило, количество временных слоев значительно превосходит пространственное разрешение: в наших экспериментах использовалось 256 временных слоев из двух ромбовидных подрешеток размером вплоть до 6×6 .

На поле наложены периодические граничные условия как по времени, так и по пространственным координатам подрешеток.

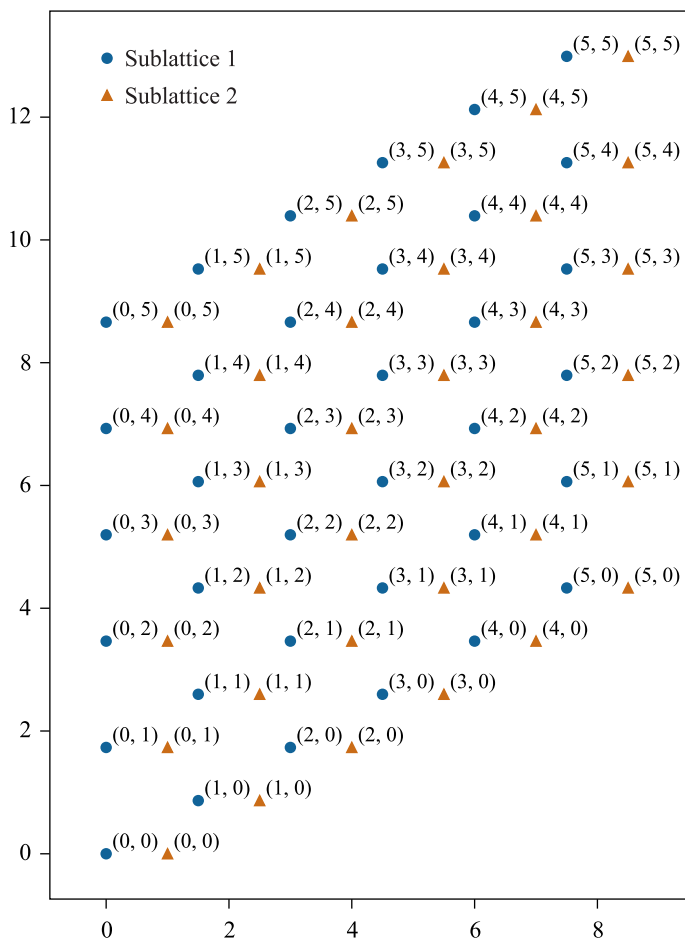


Рис. 2. Гексагональная решетка представлена как комбинация двух ромбовидных (один слой по евклидовому времени)

Данная работа ставит целью построить аппроксимацию изложенного выше алгоритма построения конфигураций, который будет асимптотически быстрее. Говоря конкретно, сложность предложенной аппроксимации оказывается $O(N_S N_\tau \log N_\tau)$.

5. ВЫБОР АРХИТЕКТУРЫ НЕЙРОННОЙ СЕТИ

В данном разделе рассматривается общий вид модели нейронной сети [44, 45] и ее составные части, а также производится сравнение с другими моделями, использовавшимися в подобных задачах.

В качестве алгоритма аппроксимации «наперстка» мы будем использовать нейронную сеть [44, 45]. Представим себе, что у нас есть достаточное число конфигураций, лежащих на некотором множестве интегрирования в комплексном пространстве полей. Эти конфигурации генерируются гибридным Монте-Карло, использующим полный градиентный поток для построения множества интегрирования, в соответствии с алгоритмом, который был описан выше. Эти конфигурации могут быть использованы в качестве основы для некоторой процедуры интерполяции «наперстка», дающей нам необходимые преобразования вида

$$\begin{aligned} (\phi_{x,y}, \chi_{x,t}) &\rightarrow (\tilde{\phi}_{x,y}, \tilde{\chi}_{x,t}), \\ \operatorname{Re} \tilde{\phi}_{x,t} &= \phi_{x,t}, \quad \operatorname{Im} \tilde{\phi}_{x,t} = F_1[\phi, \chi], \\ \operatorname{Re} \tilde{\chi}_{x,t} &= \chi_{x,t}, \quad \operatorname{Im} \tilde{\chi}_{x,t} = F_2[\phi, \chi]. \end{aligned} \quad (26)$$

Таким образом, интегрирование производится снова по действительным полям

$$(\phi_{x,t}, \chi_{x,t}) = \Phi \in \mathbb{R}^N,$$

но необходимое преобразование $\Phi \rightarrow \tilde{\Phi}$ осуществляется функциями F_1 и F_2 , задаваемыми в (26).

Один пример конфигурации полей $\tilde{\Phi}$, полученной с помощью гибридного Монте-Карло с полным градиентным потоком, представлен на рис. 3. Он показывает важное свойство конфигураций, которое влияет на построение преобразований (26). А именно, эти преобразования должны быть нелокальными. Действительно, если бы F_1 и F_2 принимали в качестве аргументов только реальные части полей на одном узле решетки, то такая конфигурация представляла бы собой линию на рис. 3, *a*. Отметим, что рассматриваемое преобразование имеет вид

$$\operatorname{Im} \tilde{\phi}_{x,t} = F_1(\operatorname{Re} \tilde{\phi}_{x,t}), \quad \operatorname{Im} \tilde{\chi}_{x,t} = F_2(\operatorname{Re} \tilde{\chi}_{x,t}).$$

Как видно из рис. 3, вместо линии мы получили размытое облако точек, где, например, одному значению $\operatorname{Re} \tilde{\phi}_{x,t}$ соответствует много значений $\operatorname{Im} \tilde{\phi}_{x,t}$.

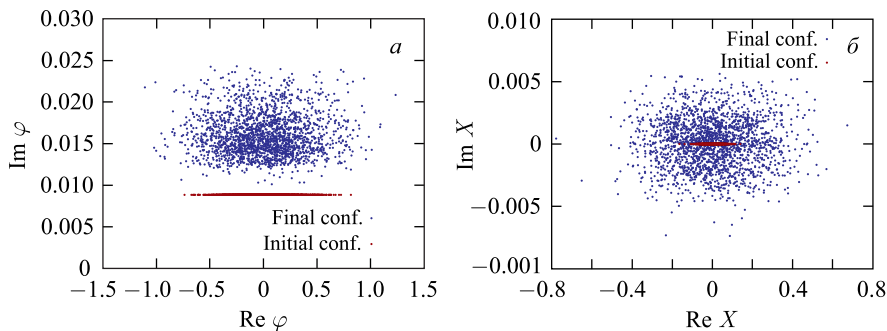


Рис. 3. Пример конфигураций, полученных методом гибридного Монте-Карло с полным градиентным потоком. Показаны вспомогательные поля, свернутые с плотностью заряда (а), и вспомогательные поля, свернутые с плотностью спина (б). В обоих случаях мы показываем начальную конфигурацию, локализованную на гауссовом «наперстке» в окрестности седловой точки тривиального вакуума, и конфигурацию после полного градиентного потока. Вычисления были проведены для гамильтониана (10) с разложением взаимодействия на два члена (11). Параметры вычисления: $\alpha = 0,8$, пространственная гексагональная решетка 2×2 с $N_\tau = 256$ временными слоями, $U = 2,0\kappa$, $\mu = \kappa$, $\beta\kappa = 20,0$ (все параметры указаны в единицах параметра хоппинга). К сожалению, невозможно найти никакого простого соотношения, позволяющего фитировать результат градиентного потока какой-то локальной функцией: $\text{Im } \phi_i = F(\text{Re } \phi_i)$

Таким образом, функции F_1 и F_2 не могут быть локальными в том смысле, что они должны принимать в качестве аргументов всю конфигурацию поля или по крайней мере большую ее часть в окрестности желаемого узла решетки (x, t) . Такая интерполяция может быть осуществлена посредством обучающейся нейронной сети. Обучение мы будем производить по конфигурациям, заранее приготовленным с помощью гибридного Монте-Карло с полным градиентным потоком. Входными данными для сети будут реальные части полей, и нейронная сеть должна будет по этим реальным частям воспроизводить мнимые части полей. Заметим еще раз, что теорема Коши гарантирует нам, что значение интеграла будет одинаковым для любой функции интерполяции (так как любой такой функции соответствует какой-то свой контур интегрирования, а они все дают одно и то же значение интеграла). Таким образом, конечный ответ не зависит от деталей построения нейронной сети, она лишь должна обеспечивать только непрерывные функции для такого преобразования (26).

Важнейшим вопросом в построении нейронной сети и ее обучении является учет симметрий нашей задачи. Очевидно, что исследуемая система обладает трансляционной симметрией по сдвигу вдоль временного и обоим пространственным направлениям. Совершенно необходимо, чтобы архитектура

нейронной сети удовлетворяла этой симметрии. Конечно, на уровне одной конкретной конфигурации эта симметрия может быть нарушена, но интерполяционная функция, которая осуществляется с помощью нейронной сети, должна удовлетворять этой симметрии. Это означает, что если наша исходная конфигурация Φ сдвигается вдоль евклидового времени или пространственных координат, то трансляционная симметрия приводит к тому, что получаемая конфигурация $\tilde{\Phi}$ также сдвигается в том же направлении.

Рассмотрим два возможных варианта архитектуры нейронной сети, удовлетворяющих требованию соблюдения трансляционной инвариантности. Первый вариант такой нейронной сети показан на рис. 4 и имеет вид пирамиды. Такая нейронная сеть получает полную полевую конфигурацию на входе и выдает только одну мнимую часть поля $F_1[\phi, \chi]$ или $F_2[\phi, \chi]$ на выходе, см. (26). Такая архитектура нейронной сети была предложена в одной из предыдущих работ [27] по тематике «наперстков». Теперь, если нам понадобится вычислить мнимую часть поля на другом узле решетки, то мы просто должны сдвинуть исходные поля на входе нейронной сети вдоль пространственных направлений или вдоль евклидового времени в соответствии с трансляционной симметрией и с учетом периодических граничных условий.

На каждом уровне сверточной нейронной сети отдельные связи между нейронами могут иметь свои конкретные веса, отличные друг от друга. Число связей на разных уровнях сети также может быть разным: от связей всех нейронов со всеми до связей только между соседними нейронами. В этом отношении данный метод обеспечивает максимальную гибкость, хотя число параметров в этой сети может быть очень большим, особенно в случае связи всех нейронов со всеми.

Другой возможной архитектурой сверточной нейронной сети, удовлетворяющей требованию трансляционной инвариантности, является предложенная недавно так называемая U-сеть [58]. Число полей на входе и на выходе в этой сети одинаковое, а трансляционная инвариантность обеспечивается единообразным распределением весовых функций на связях между нейро-

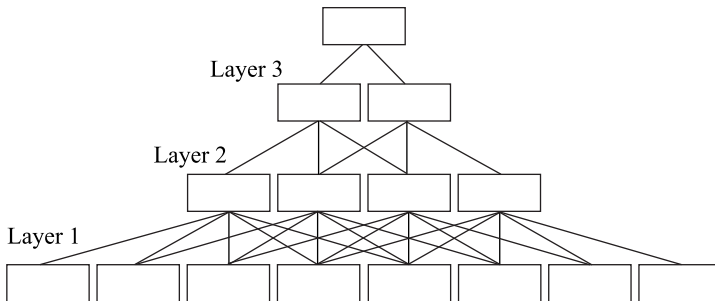


Рис. 4. Архитектура нейронной сети типа «пирамида»

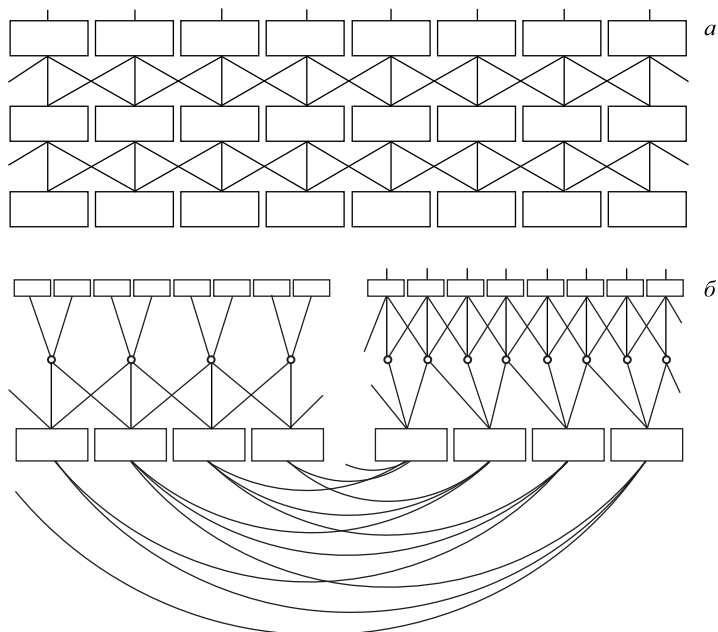


Рис. 5. Архитектура нейронной сети U-типа («U-net»). Связи показаны для случая пространственных направлений (*a*) и для случая направления вдоль евклидового времени (*b*)

нами на каждом уровне. Однако имеется еще одна существенная особенность в генерируемых нами конфигурациях, важная для выбора этого типа архитектуры, — очень большое различие в линейных размерах конфигураций вдоль пространственных направлений (~ 10) и вдоль евклидового времени (~ 100). По этой причине мы выберем две различные стратегии в организации связей между нейронами для случая пространственных направлений и временного направления (см. рис. 5, *a*).

Для пространственных направлений мы можем не уменьшать размерность данных вообще, так как даже небольшого числа слоев достаточно для того, чтобы рецептивное поле нейронов на последнем слое было сравнимо с размерами системы. Причем это верно, даже если учитывать в свертках связи только между соседними нейронами. К сожалению, этот подход не будет работать для направления вдоль евклидового времени. В этом случае нам нужно добиться более быстрого роста размера рецептивного поля нейронов с числом слоев в сети. Учитывая это обстоятельство, мы будем использовать для этого случая архитектуру U-сети с усреднением данных вдоль «нисходящей» ветви нейронной сети (левая часть схемы на рис. 5, *b*, которая начинается от входного слоя и продолжается до нижней части схемы). Далее, для того чтобы

восстановить исходный размер данных, мы будем использовать интерполяцию на «восходящей» ветви нейронной сети (правая часть на схеме, изображенной на рис. 5, б). Основное преимущество предлагаемого подхода состоит в существенном сокращении числа параметров сети, так как в данном случае веса нейронных связей распределены равномерно в каждом конкретном слое. Таким образом, в результате мы можем использовать меньше данных для обучения сети.

Более детально структура нейронной сети, использованной в проекте, описана ниже.

6. АРХИТЕКТУРА U-СЕТИ И ЗАДАЧА ОПТИМИЗАЦИИ

Наша задача формулируется следующим образом: необходимо построить нелинейное отображение M действительной части конфигурации R в мнимую I . У нас $R \in \mathbb{R}^N$ и $I \in \mathbb{R}^N$, где N — это полное число узлов в решетке:

$$R \xrightarrow{M} I. \quad (27)$$

Перейдем от задачи поиска функции к задаче оптимизации. Для этого будем рассматривать отображение M не как произвольную функцию, а как нейронную сеть M_θ с параметрами θ . В таком случае наша задача становится проще, ведь мы перешли от задачи поиска функции к задаче поиска оптимальных параметров, но, зафиксировав функциональный вид (архитектуру) сети, мы теряем возможность представить любую функцию одним только выбором параметров θ . Отметим, однако, что такая возможность все еще присутствует в момент выбора архитектуры [46]. Это означает, что мы должны вначале выбрать наиболее подходящую архитектуру сети, а затем подобрать параметры таким образом, чтобы отображение (27) выполнялось наиболее точно.

Как уже было сказано выше, сеть с фиксированной архитектурой не может представить любую зависимость. Пусть I_{atxy} — предсказанная мнимая часть поля, а \tilde{I}_{atxy} — истинная мнимая часть. Так как в силу фиксированного функционального вида отображения нельзя сделать эти части равными, скажем, что

$$\tilde{I}_{atxy} = I_{atxy} + \epsilon, \quad (28)$$

где ϵ — это добавка в виде «шума», которую наша сеть не может аппроксимировать. Далее будем предполагать, что это нормальный шум с нулевым средним и дисперсией σ^2 . Наша же задача состоит в том, чтобы подобрать такие параметры θ , при которых $I = M_\theta(R)$ оказалась как можно ближе к \tilde{I} .

Чтобы найти оптимальные параметры θ , воспользуемся принципом максимального правдоподобия: найдем такие параметры θ , что

$$P(\tilde{I}|I) = P(\tilde{I}|R, \theta) = \prod_{t,a,x,y} \frac{1}{\sqrt{2\pi\sigma^2}} \exp - \frac{(\tilde{I}_{atxy} - I_{atxy})^2}{2\sigma^2} \rightarrow \max_{\theta}. \quad (29)$$

Обозначая оптимальные параметры θ как $\hat{\theta}$ и принимая $\sigma^2 = \text{const}$, перейдем к логарифму правдоподобия:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{n,t,a,x,y} \left(\tilde{I}_{atxy}^{(n)} - I_{atxy}^{(n)} \right)^2, \quad (30)$$

где индексом n обозначается номер конфигурации.

Схематическое изображение сети в более детальном виде представлено на рис. 6. Для упрощения экспериментов был использован PyTorch [47], автоматизирующий дифференцирование вычислительного графа.

Наша задача во многом схожа с image-to-image задачами [48, 49], с той лишь разницей, что в нашем случае входом и выходом являются не изображения, состоящие из пикселей, а поля на гексагональной решетке.

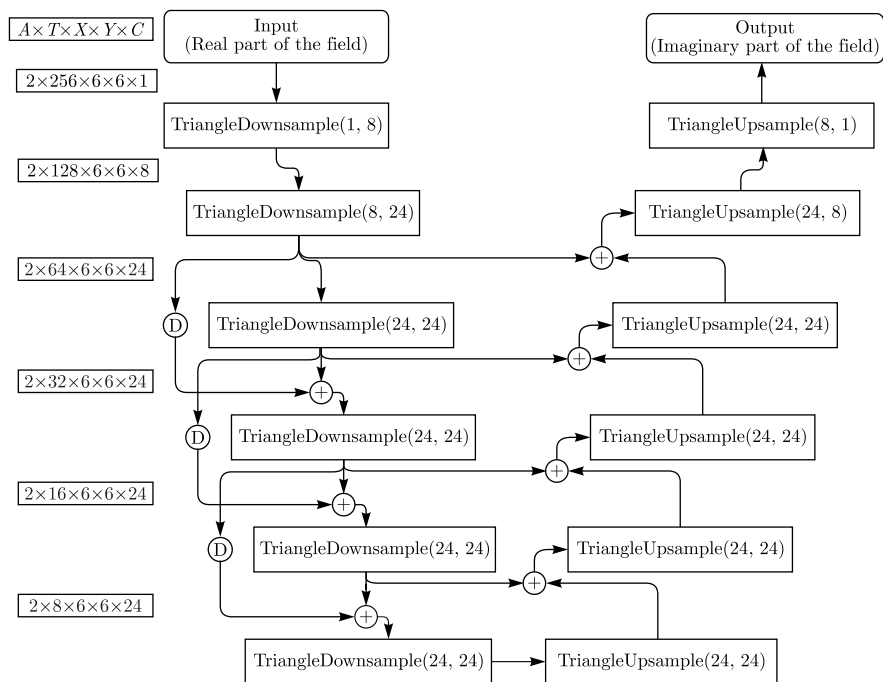


Рис. 6. Передача данных внутри нейронной сети. Блок TriangleDownsample (C_{in}, C_{out}) состоит из последовательного применения треугольной свертки, нелинейности ELU и уменьшения временной размерности путем усреднения. Блок TriangleUpsample (C_{in}, C_{out}) состоит из последовательного применения треугольной свертки, нелинейности ELU (функция активации — экспоненциальная линейная функция) [54] и увеличения временной размерности путем усреднения. Значком Ⓧ обозначается уменьшение временной размерности путем усреднения, а ⊕ — поэлементное сложение тензоров данных

В работе с изображениями хорошо себя зарекомендовали сверточные сети [50–52]. В нашем случае применение сверточных сетей осложняется тем, что данные лежат не в вершинах квадратной решетки, как в случае с изображениями, а образуют гексагональную решетку.

Можно заметить, что у любого элемента такой решетки есть ровно три равноудаленных ближайших соседа (см. рис. 2). Таким образом, ядро свертки может состоять из четырех элементов: центрального и трех ближайших соседей. Например, чтобы вычислить элемент слоя $l + 1$ первой подрешетки с индексом $(3, 2)$, нам понадобятся следующие элементы со слоя l : $(3, 2)$ из первой подрешетки и его ближайшие соседи из второй подрешетки: $(3, 2)$, $(2, 2)$, $(2, 3)$.

В общем случае ближайшие соседи для элемента (i, j) первой подрешетки это (i, j) , $(i - 1, j)$, $(i - 1, j + 1)$, а для элемента (i, j) второй — (i, j) , $(i + 1, j)$, $(i + 1, j - 1)$. При этом, конечно, нужно учитывать периодичность граничных условий.

В нашем случае есть три направления, по которым производится свертка. Среди них два пространственных, работа с которыми была описана выше, и одно временное. Простейший вариант учета временного направления — просто добавить в ядро свертки предыдущий и следующий временные слои.

Конечная формула для свертки выглядит следующим образом. Пусть C_{in} — число каналов на входе в свертку (размерность поля в вершине), $W_{c_i, t, d, c_o}^{\{1,2\}}$ — матрица весов треугольной свертки для каждой подрешетки, $b_{c_o}^{\{1,2\}}$ — вектор смещений свертки для каждой подрешетки, c_o — индекс выходного канала, c_i — индекс входного канала, t — временной индекс, d — пространственный индекс, $L_{a, t, y, x, c}^{\{in, out\}}$, a — индекс подрешетки, x, y — пространственные индексы. Тогда свертки осуществляются в соответствии с выражением

$$L_{1, t, y, x, c_o}^{out} = \sum_{\tau=-1}^1 \sum_{c_i=0}^{C_{in}} \left(L_{1, t+\tau, y, x, c_i}^{in} W_{c_i, t+\tau, 0, c_o}^1 + L_{2, t+\tau, y, x, c_i}^{in} W_{c_i, t+\tau, 1, c_o}^1 + \right. \\ \left. + L_{2, t+\tau, y-1, x, c_i}^{in} W_{c_i, t+\tau, 2, c_o}^1 + L_{2, t+\tau, y-1, x+1, c_i}^{in} W_{c_i, t+\tau, 3, c_o}^1 \right) + b_{c_o}^1, \quad (31)$$

$$L_{2, t, y, x, c_o}^{out} = \sum_{\tau=-1}^1 \sum_{c_i=0}^{C_{in}} \left(L_{2, t+\tau, y, x, c_i}^{in} W_{c_i, t+\tau, 0, c_o}^2 + L_{1, t+\tau, y, x, c_i}^{in} W_{c_i, t+\tau, 1, c_o}^2 + \right. \\ \left. + L_{1, t+\tau, y+1, x, c_i}^{in} W_{c_i, t+\tau, 2, c_o}^2 + L_{1, t+\tau, y+1, x-1, c_i}^{in} W_{c_i, t+\tau, 3, c_o}^2 \right) + b_{c_o}^2, \quad (32)$$

где все операции с временными и пространственными индексами берутся по соответствующему модулю для удовлетворения граничных условий.

Для расчета результата свертки на второй подрешетке мы не будем использовать то же самое ядро, что и для первой подрешетки. Это вызвано тем, что треугольник, образуемый ближайшими соседями элемента первой подрешетки, трансляционно не переводится в ближайших соседей элемента второй подрешетки. Это приводит к тому, что для задания свертки нам потребуется $2 \times 3 \times 4 \times C_{in} \times C_{out}$ значений, где C_{in} и C_{out} — число каналов входного и выходного слоев соответственно.

7. ТРАНСПОРТ ДАННЫХ ЧЕРЕЗ НЕЙРОННУЮ СЕТЬ

Данные проходят через нейронную сеть в соответствии со стрелками на рис. 6. Предложенная архитектура является модифицированной версией U-net [53]. Как и в U-net, горизонтальные связи служат для того, чтобы не терять информацию при понижении временной размерности: понижая временную размерность, мы увеличиваем рецептивную область каждого последующего слоя, но можем терять детали во временном направлении.

Слева записаны размерности представления конфигурации при прохождении через сеть. Первый индекс означает количество подрешеток, их всегда две, второй — количество временных слоев, третий и четвертый — пространственную размерность каждой из решеток, а последний индекс — размерность поля в узле решетки. Так как на вход сети подаются только действительные части поля, то для входных данных размерность поля в узле равна единице.

Например, на выходе из первого слоя сети конфигурация представляется объектом размерности $2 \times 128 \times 6 \times 6 \times 8$, т. е. двумя решетками, в каждой из которых 128 временных слоев с пространственной размерностью 6×6 и теперь в каждом узле решетки поле представлено вектором из восьми компонент.

Можно заметить, что основными блоками нашей сети являются

$$\text{TriangleDownsample}(C_{in}, C_{out})$$

и

$$\text{TriangleUpsample}(C_{in}, C_{out}).$$

В них происходит применение треугольной свертки, нелинейности ELU (функция активации — экспоненциальная линейная функция) [54] с последующим изменением временного разрешения. Обратит внимание на то, как информация, проходя через сеть, может пойти по короткому пути, не доходя до самых нижних слоев. В соответствии с [53] это позволяет улучшить точность восстановления информации после понижения разрешения. Вместо конкатенации мы используем сложение, как предложено в [55].

Кроме того, мы используем подход, схожий с [56], чтобы позволить информации пропускать определенные слои в ветке понижения размерности. Чтобы это стало возможным с точки зрения размерности, мы применяем

только операцию понижения размерности к такому «обходному» пути. Это позволяет добиться большей устойчивости обучения относительно выбора начальных параметров.

8. ОБУЧЕНИЕ МОДЕЛИ

Для обучения нейронной сети использовался специально подготовленный набор конфигураций. Затравочные конфигурации для этого набора были созданы с помощью гибридного Монте-Карло с полным градиентным потоком. Распределение затравочных конфигураций имеет одну особенность, связанную с большой размерностью данных. Рассмотрим гауссово распределение в пространствах большой размерности (рис. 7). Видно, что элемент объема сдвигает максимум в итоговой плотности вероятности для дистанции от начала координат, где находится седловая точка. Сдвиг тем больше, чем больше размерность данных. В итоге все конфигурации, которые мы можем получить с помощью Монте-Карло, лежат достаточно далеко от начала координат (седловой точки) и не могут дать нейронной сети необходимую информацию о поведении мнимой части полей в окрестности седла.

Для гладкой интерполяции «наперстка», начиная от седловой точки и дальше на всех важных дистанциях, необходимо обучать нейронные сети на модифицированном наборе конфигураций, отвечающем разным участкам «наперстка». Для этого генерация обучающих конфигураций производилась в два этапа (рис. 8). Сначала выбиралась конфигурация вдали от седловой точки (конфигурация 1 на рис. 8). Дальше с помощью градиентного потока вдоль гауссова «наперстка» генерировалась серия конфигураций, приближающихся к седловой точке (конфигурации 2, 3, 4 и т. д. на рис. 8). Так как

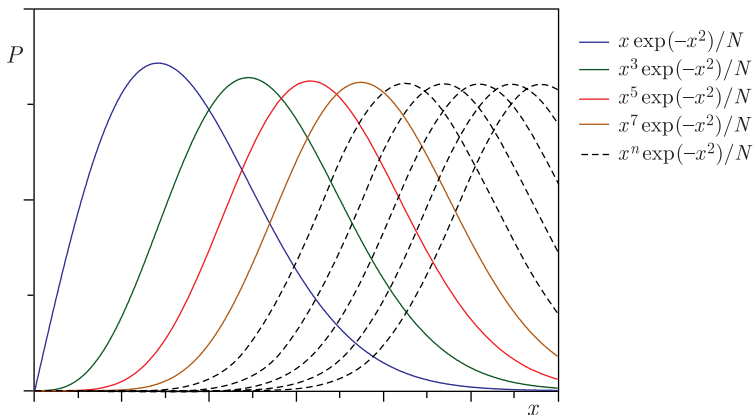


Рис. 7. Радиальная часть распределения плотности вероятности конфигураций ($P(x)$) вдоль «наперстка» начиная от седловой точки ($x = 0$)

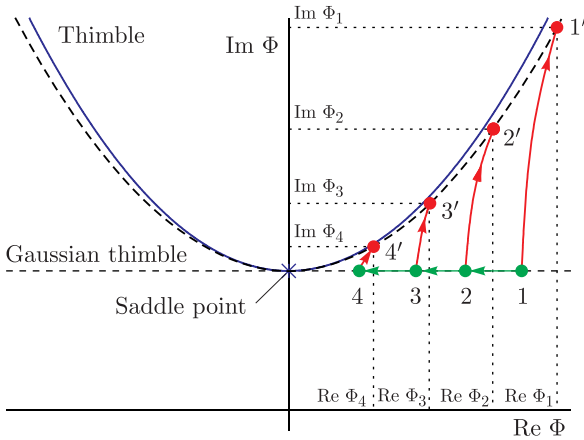


Рис. 8. Алгоритм генерации конфигураций для обучения нейронной сети

градиенты при приближении к седловой точке обращаются в нуль, такой градиентный поток не может пройти мимо седловой точки, а упирается в нее. Таким образом, мы получаем серию конфигураций, приближающихся к седловой точке. На втором этапе из каждой такой конфигурации запускается полный градиентный поток по направлению к «наперстку». Поток также не может пересечь «наперсток», а только приближается к нему. Используя этот подход, мы получаем из каждой конфигурации на гауссовом «наперстке» одну конфигурацию из окрестности искомого «наперстка» (конфигурации $1'$, $2'$, $3'$, $4'$ и т. д. на рис. 8), при этом полученные конфигурации также приближаются к седловой точке, только вдоль искомого «наперстка». Они и были использованы для обучения нейронной сети.

Для обучения использовались конфигурации, представленные 256 временными слоями с пространственной размерностью 2×2 . Всего было использовано 200 затравочных конфигураций, сгенерированных с параметром $\alpha = 1$ (таким образом, только поле ϕ участвует в расчете).

Обучение модели производилось путем минимизации функции потерь при помощи градиентного спуска. В качестве функции потерь было выбрано евклидово расстояние между предсказанным и истинным значениями:

$$L = \mathbb{E}_{\omega \sim \Omega} \|N_{\theta}(\omega^{(\text{re})}) - \omega^{(\text{im})}\|^2, \quad (33)$$

где Ω — распределение всех возможных конфигураций; $\omega^{(\text{re})}$ и $\omega^{(\text{im})}$ — действительная и мнимая части поля соответственно, а $N_{\theta}(\dots)$ — наша нейронная сеть с параметрами θ , принимающая на вход действительную часть поля и возвращающая оценку мнимой части. В случае конечного объема обучающей

выборки уравнение (34) переходит в

$$L = \frac{1}{n} \sum_{i=1}^n \|N_{\theta}(\omega_i^{(re)}) - \omega_i^{(im)}\|^2, \tag{34}$$

где n — число конфигураций в обучающей выборке.

Необходимо найти такие параметры нейронной сети $\hat{\theta}$, чтобы $\hat{\theta} = \underset{\theta}{\operatorname{argmin}}(L)$. Для нахождения $\hat{\theta}$ был использован алгоритм Adam [57] с параметрами $\alpha = 0,001$, $\beta_1 = 0,9$ и $\beta_2 = 0,999$.

9. РЕЗУЛЬТАТЫ

В результате была построена нейронная сеть, асимптотическая сложность применения которой $O(N_S N_{\tau} \log N_{\tau})$. Сложность каждого слоя $O(N_S N_{\tau})$,

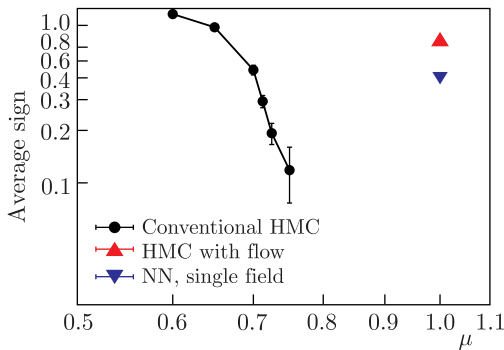


Рис. 9. Зависимость проблемы знака от химического потенциала

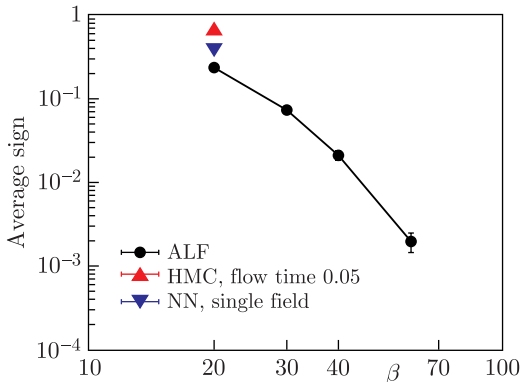


Рис. 10. Зависимость проблемы знака от температуры

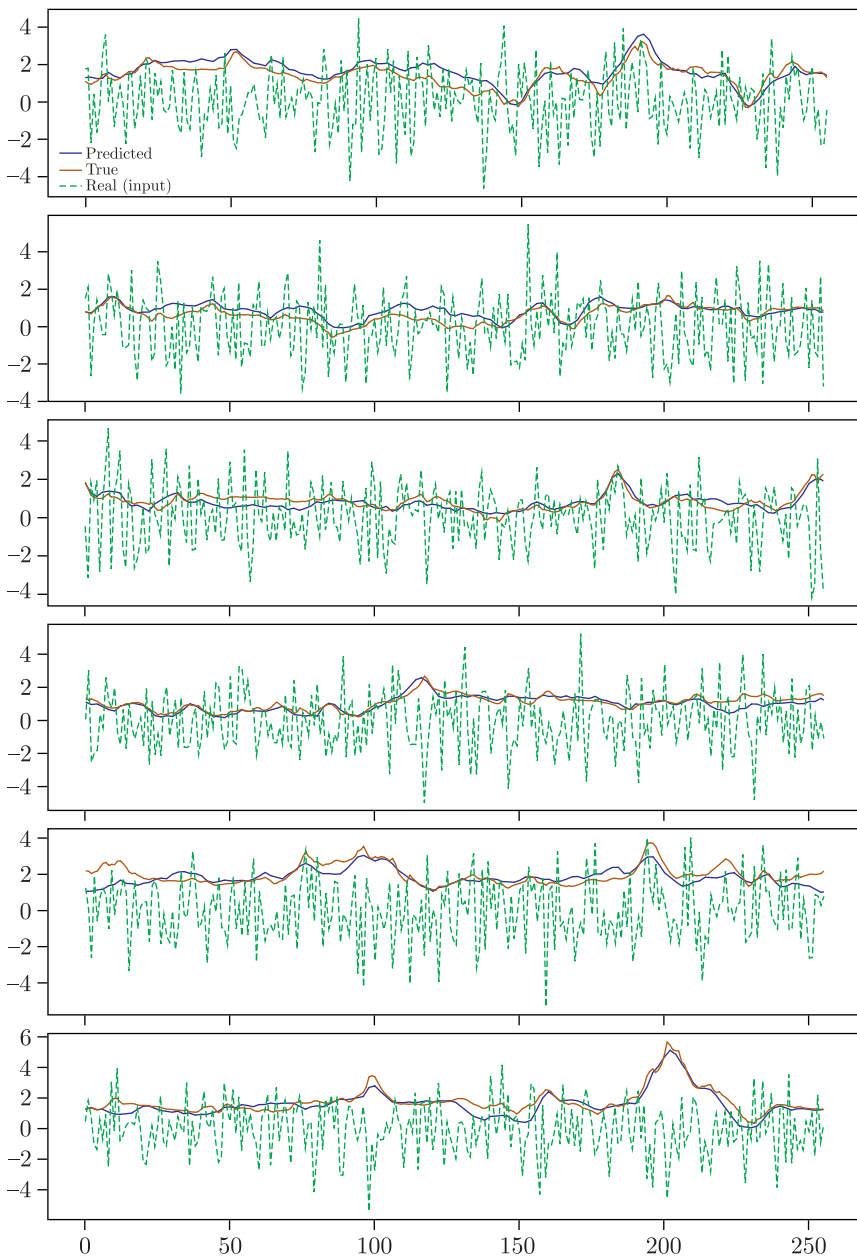


Рис. 11. Пример результата работы сети

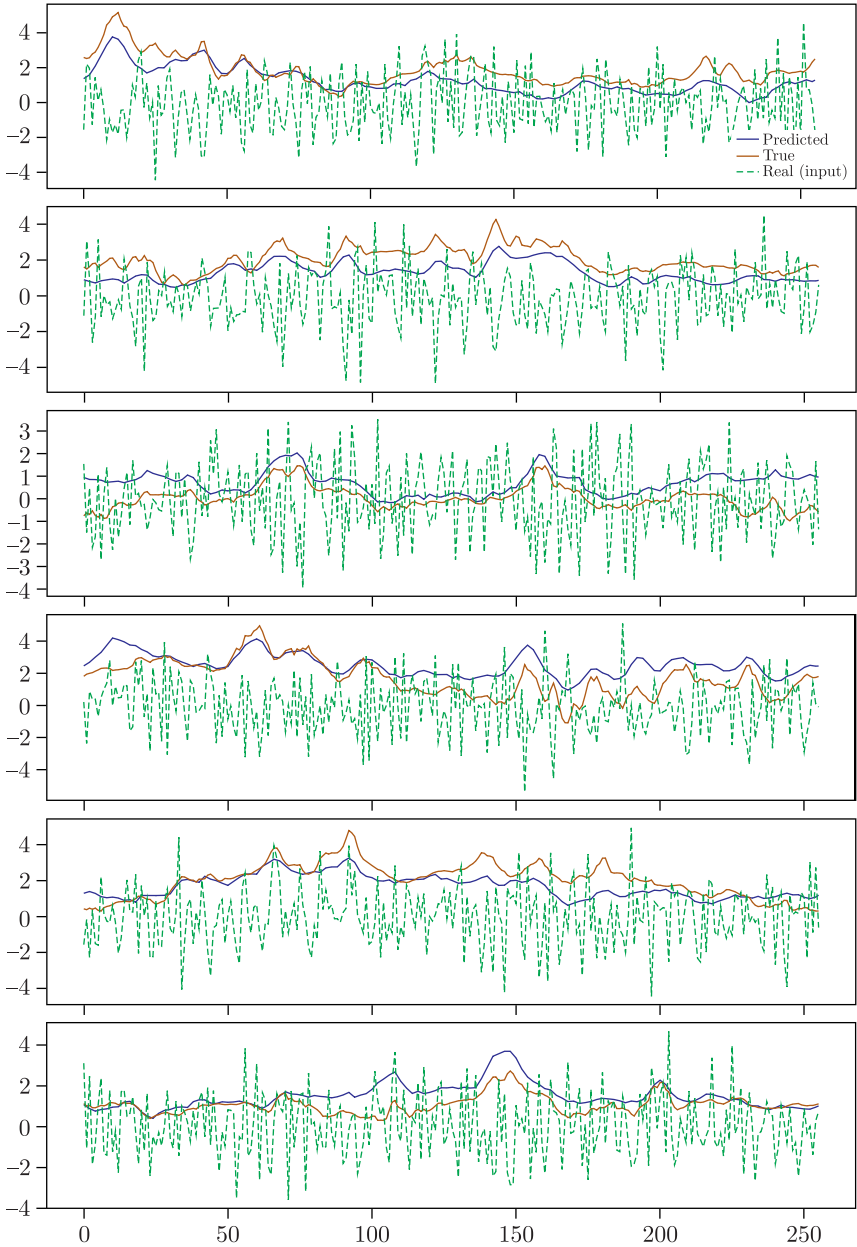


Рис. 12. Пример смещения

а число слоев растет как $O(\log N_\tau)$. Это позволяет сэмплировать траектории намного быстрее, чем при помощи точного алгоритма, работающего за $O(N_S^4 N_\tau^2)$, что позволяет снизить дисперсию монте-карло-оценки.

На рис. 9 и 10 приведено сравнение нашей аппроксимации и точного алгоритма, использующего градиентный поток и обычное перевзвешивание. Видно, что средний знак для предложенной аппроксимации выше, чем средний знак при перевзвешивании, что подтверждает жизнеспособность предлагаемого подхода.

На рис. 11 показаны результаты работы сети. Видно, что нейронная сеть способна уловить зависимость между действительной и мнимой частями поля, несмотря на сложный характер действительной части.

На рис. 12 приведены примеры, в которых сеть сильно ошибается. Как можно заметить, в подавляющем большинстве случаев оказывается, что часть предсказанной сетью траектории смещена относительно истинной.

10. ВЫВОДЫ И ДАЛЬНЕЙШИЕ ПЛАНЫ

Из полученных результатов можно сделать вывод о том, что предложенный алгоритм аппроксимации справляется с поставленной задачей: асимптотическая сложность аппроксимации оказывается значительно меньше сложности «честного» алгоритма, при этом сохраняется удовлетворительная точность. Таким образом, сверточные нейронные сети показывают свою эффективность в задачах на решетке.

Отметим также, что в рамках предлагаемого подхода существует еще большое число открытых проблем, которые ждут своего решения. В первую очередь метод протестирован пока только для случая одного вспомогательного поля, заданного на каждом узле решетки ($\alpha = 1$). В этом случае структура нейронной сети гораздо проще, чем в случае формализма с двумя вспомогательными полями ($\alpha \neq 1$). Однако последний формализм гораздо менее чувствителен к проблеме эргодичности. Эта проблема возникает, когда гибридное Монте-Карло не может перейти от одного релевантного «наперстка» к другому, поэтому проходится не все конфигурационное пространство полей. Как было показано в [59, 60], $\alpha = 1$ соответствует случаю большого количества релевантных «наперстков», и проблема эргодичности стоит достаточно остро в данном конкретном случае. Поэтому предложенный подход с нейронными сетями было бы интересно обобщить на случай формализма с двумя вспомогательными полями.

Другой важный момент — это исследование применимости предлагаемого метода на решетках разного размера. В идеале мы должны были бы обучать нашу нейронную сеть на маленьких решетках, а затем применять ее предсказания для больших решеток, так как корреляции между вспомогательными полями, полученными с помощью уравнений градиентного потока, быстро

спадают с расстоянием в пространстве и евклидовом времени. Однако применимость такого подхода к реальным задачам до сих пор остается открытым вопросом и требует подтверждения. Например, мы могли бы попытаться использовать предсказания нейронной сети, полученные на решетке $2 \times 2 \times 256$, для получения конфигураций на решетке $4 \times 2 \times 256$ или $4 \times 4 \times 256$. Отметим, что архитектура нейронной сети типа «пирамида» (см. рис. 4), возможно, окажется более подходящей для решения поставленной задачи (обучение сети на маленьких решетках и использование обученной сети для больших). Все эти идеи пока что не воплощены в нашем компьютерном коде.

Интересным развитием предложенного подхода может стать использование более гладкой функции активации. В нашем подходе использовалась функция ELU, которая при многих достоинствах обладает и недостатком — разрывностью первой производной в нуле. Возможно, это оказывает влияние на процесс обучения.

Еще одним открытым вопросом остается учет ротационной и подрешеточной симметрий. Учет этих симметрий может привести к уменьшению числа параметров в нейронной сети и облегчению процесса обучения. Однако архитектура сети также должна значительно усложниться. Поэтому необходимость учета дополнительных симметрий до сих пор не ясна.

Благодарности. Работа выполнена при финансовой поддержке Российского научного фонда (грант РНФ 16-12-10059-П).

СПИСОК ЛИТЕРАТУРЫ

1. *Inguscio M., Ketterlen W., Salomon C.* Ultra-Cold Fermi Gases // Proc. of Intern. School of Physics “Enrico Fermi 164th Course”. 2007. V. 164. P. 1–904.
2. *Scheffler T., Teaney D.* Nearly Perfect Fluidity: From Cold Atomic Gases to Hot Quark Gluon Plasmas // Rep. Prog. Phys. 2009. V. 72, No. 12. P. 126001; arXiv:0904.3107 [hep-ph].
3. *Rammelmuller L., Loheac A. C., Drut J. E., Braun J.* Finite Temperature Equation of State of Polarized Fermions at Unitarity // Phys. Rev. Lett. 2018. V. 121, No. 17. P. 173001; arXiv:1807.04664 [cond-mat].
4. *Troyer M., Wiese U.-J.* Computational Complexity and Fundamental Limitations to Fermionic Quantum Monte Carlo Simulations // Phys. Rev. Lett. 2005. V. 94, No. 17. P. 170201; arXiv:0408370 [cond-mat].
5. *Karsch F., Schaefer B.-J., Wagner M.* Towards Finite Density QCD with Taylor Expansions // Phys. Lett. B. 2011. V. 698, No. 3. P. 256–264; arXiv:1009.5211 [hep-ph].
6. *Bazavov A., Ding H.-T., Hegde P., Kaczmarek O., Karsch F., Laermann E., Maezawa Y., Mukherjee S., Ohno H., Petreczky P., Sandmeyer H., Steinbrecher P., Schmidt C., Sharma S., Seldner W., Wagner M.* QCD Equation of State to $\mathcal{O}(\mu_B^6)$ from Lattice QCD // Phys. Rev. D. 2017. V. 95, No. 5. P. 054504; arXiv:1701.04325 [hep-lat].

7. *Bornyakov V. G., Boyda D. L., Goy V. A., Molochkov A. V., Nakamura A., Nikolaev A. A., Zakharov V. I.* New Approach to Canonical Partition Functions Computation in $N_f = 2$ Lattice QCD at Finite Baryon Density // *Phys. Rev. D.* 2017. V. 95, No. 9. P. 094506; arXiv:1611.04229 [hep-lat].
8. *Aarts G.* Complex Langevin Dynamics and Other Approaches at Finite Chemical Potential // *PoS Lattice2012, Cairns, Australia, June 24–29, 2012.* P.017; arXiv:1302.3028 [hep-lat].
9. *Aarts G., Bongiovanni L., Seiler E., Sexty D., Stamatescu I.-O.* Controlling Complex Langevin Dynamics at Finite Density // *Eur. Phys. J. A.* 2013. V. 49. P. 89; arXiv:1303.6425 [hep-lat].
10. *Sexty D.* Simulating Full QCD at Nonzero Density Using the Complex Langevin Equation // *Phys. Lett. B.* 2014. V. 729. P. 108-111; arXiv:1307.7748 [hep-lat].
11. *Nagata K., Nishimura J., Shimasaki S.* Gauge Cooling for the Singular-Drift Problem in the Complex Langevin Method — A Test in Random Matrix Theory for Finite Density QCD // *JHEP.* 2016. V. 07. P. 073; arXiv:1604.07717 [hep-lat].
12. *Aarts G., Seiler E., Sexty D., Stamatescu I.-O.* Complex Langevin Dynamics and Zeros of the Fermion Determinant // *JHEP.* 2017. V. 05 P. 044; arXiv:1701.02322 [hep-lat].
13. *Bloch J., Glesaaen J., Verbaarschot J. J. M., Zafeiropoulos S.* Complex Langevin Simulation of a Random Matrix Model at Nonzero Chemical Potential // *JHEP.* 2018. V. 03. P. 15; arXiv:1712.07514 [hep-lat].
14. *Anagnostopoulos K. N., Azuma T., Ito Y., Nishimura J., Kovalkov Papadoudis S.* Complex Langevin Analysis of the Spontaneous Symmetry Breaking in Dimensionally Reduced Super Yang–Mills Models. // <https://doi.org/10.1140/epja/i2013-13089-4> *JHEP* 2018. V. 02 P. 151; arXiv:1712.07562 [hep-lat].
15. *Scherzer M., Seiler E., Sexty D., Stamatescu I.-O.* Complex Langevin and Boundary Terms // *Phys. Rev. D.* 2019. V. 99, No. 1. P. 014512; arXiv:1808.05187 [hep-lat].
16. *Witten E.* A New Look at the Path Integral of Quantum Mechanics. 2010. arXiv:1009.6032 [hep-th].
17. *Witten E.* Analytic Continuation of Chern–Simons Theory // *AMS/IP Stud. Adv. Math.* 2011. V. 50. P. 347–446; arXiv:1001.2933 [hep-th].
18. *Cristoforetti M., Di Renzo F., Scorzato L.* New Approach to the Sign Problem in Quantum Field Theories: High Density QCD on a Lefschetz Thimble // *Phys. Rev. D.* 2012. V. 86. P. 074506; arXiv:1205.3996 [hep-lat].
19. *Cristoforetti M., Di Renzo F., Mukherjee A., Scorzato L.* Quantum Field Theories on the Lefschetz Thimble // *PoS Lattice2013 2014, Mainz, Germany, July 29 – August 3, 2013.* P. 197; arXiv:1312.1052 [hep-lat].
20. *Cristoforetti M., Di Renzo F., Mukherjee A., Scorzato L.* Monte Carlo Simulations on the Lefschetz Thimble: Taming the Sign Problem // *Phys. Rev. D.* 2013. V. 88. P. 051501; arXiv:1303.7204 [hep-lat].
21. *Fujii H., Honda D. H., Kato M., Kikukawa Y., Komatsu S., Sano T.* Hybrid Monte Carlo on Lefschetz Thimbles — A Study of the Residual Sign Problem // *JHEP.* 2013. V. 10. P. 147; arXiv:1309.4371 [hep-lat].
22. *Fujii H., Kamata S., Kikukawa Y.* Lefschetz Thimble Structure in One-Dimensional Lattice Thirring Model at Finite Density // *JHEP.* 2015. V. 11. P. 078; arXiv:1509.08176 [hep-lat].

23. *Tanizaki Y., Hidaka Y., Hayata T.* Lefschetz Thimble Analysis of the Sign Problem in One-Site Fermion Model // *New J. Phys.* 2016. V. 18, No. 3. P. 033002; arXiv:1509.07146 [hep-th].
24. *Kanazawa T., Tanizaki Y.* Structure of Lefschetz Thimbles in Simple Fermionic Systems // *JHEP.* 2015. V. 03. P. 044; arXiv:1412.2802 [hep-lat].
25. *Alexandru A., Basar G., Bedaque P.* Monte Carlo Algorithm for Simulating Fermions on Lefschetz Thimbles // *Phys. Rev. D.* 2016. V. 93, No. 1. P. 014504; arXiv:1510.03258 [hep-lat].
26. *Alexandru A., Basar G., Bedaque P. F., Ridgway G. W., Warrington N. C.* Sign Problem and Monte Carlo Calculations beyond Lefschetz Thimbles // *JHEP.* 2016. V. 05. P. 053; arXiv:1512.08764 [hep-lat].
27. *Alexandru A., Bedaque P. F., Lamm H., Lawrence S.* Deep Learning beyond Lefschetz Thimbles // *Phys. Rev. D.* 2017. V. 96, No. 9. P. 094505; arXiv:1709.01971 [hep-lat].
28. *Di Renzo F., Eruzzi G.* Thimble Regularization at Work: From Toy Models to Chiral Random Matrix Theories // *Phys. Rev. D.* 2015. V. 92, No. 8. P. 08; arXiv:1507.03858 [hep-lat].
29. *Alexandru A., Basar G., Bedaque P. F., Ridgway G. W., Warrington N. C.* Monte Carlo Calculations of the Finite Density Thirring Model // *Phys. Rev. D.* 2017. V. 95, No. 1. P. 014502; arXiv:1609.01730 [hep-lat].
30. *Alexandru A., Basar G., Bedaque P. F., Warrington N. C.* Spin Polarized Nonrelativistic Fermions in $1+1$ Dimensions // *Phys. Rev. D.* 2018. V. 98, No. 5. P. 054514; arXiv:1805.00125 [hep-lat].
31. *Alexandru A., Basar G., Bedaque P. F., Lamm H., Lawrence S.* Finite Density QED $_{1+1}$ near Lefschetz Thimbles // *Phys. Rev. D.* 2018. V. 98, No. 3. P. 034506; arXiv:1807.02027 [hep-lat].
32. *Alexandru A., Basar G., Bedaque P. F., Lamm H., Lawrence S., Warrington N. C.* Fermions at Finite Density in $2+1$ Dimensions with Sign-Optimized Manifolds // *Phys. Rev. Lett.* 2018. V. 121, No. 19. P. 191602; arXiv:1808.09799 [hep-lat].
33. *Bluecher S., Pawłowski J. M., Scherzer M., Schlosser M., Stamatescu I.-O., Syrkowski S., Ziegler F. P. G.* Reweighting Lefschetz Thimbles // *SciPost Phys.* 2018. V. 5, No. 5. P. 044; arXiv:1803.08418 [hep-lat].
34. *Buividovich P. V., Polikarpov M. I.* Monte Carlo Study of the Electron Transport Properties of Monolayer Graphene within the Tight-Binding Model // *Phys. Rev. B.* 2012. V. 86, No. 5. P. 245117; arXiv:1206.0619 [cond-mat.str-el].
35. *Ulybyshev M. V., Buividovich P. V., Katsnelson M. I., Polikarpov M. I.* Monte Carlo Study of the Semimetal-Insulator Phase Transition in Monolayer Graphene with a Realistic Interelectron Interaction Potential // *Phys. Rev. Lett.* 2013. V. 111, No. 5. P. 056801; arXiv:1304.3660 [cond-mat.str-el].
36. *Smith D., von Smekal L.* Monte Carlo Simulation of the Tight-Binding Model of Graphene with Partially Screened Coulomb Interactions // *Phys. Rev. B.* 2014. V. 89, No. 5. P. 195429; arXiv:1403.3620 [hep-lat].
37. *Buividovich P. V., Ulybyshev M. V.* Applications of Lattice QCD Techniques for Condensed Matter Systems // *Intern. J. Mod. Phys. A.* 2016. V. 31, No. 22. P. 1643008; arXiv:1602.08431 [hep-lat].
38. *Hohenadler M., Parisen Toldin F., Herbut I. F., Assaad F. F.* Phase Diagram of the Kane-Mele-Coulomb Model // *Phys. Rev. B.* 2014. V. 90, No. 8. P. 085146; arXiv:1407.2708 [cond-mat.str-el].

39. *Ulybyshev M., Winterowd C., Zafeiropoulos S.* Collective Charge Excitations and the Metal-Insulator Transition in the Square Lattice Hubbard–Coulomb Model // *Phys. Rev. B.* 2017. V.96, No. 20. P. 205115; arXiv:1707.04212 [cond-mat.str-el]
40. *Korner M., Smith D., Buividovich P., Ulybyshev M., von Smekal L.* Hybrid Monte Carlo Study of Monolayer Graphene with Partially Screened Coulomb Interactions at Finite Spin Density // *Phys. Rev. B.* 2017. V.96, No.19. P.195408; arXiv:1704.03757 [cond-mat.str-el]
41. *Wu C., Zhang S.-C.* Sufficient Condition for Absence of the Sign Problem in the Fermionic Quantum Monte Carlo Algorithm // *Phys. Rev. B.* 2005. V. 71, No. 15. P. 155115. e-Print: cond-mat/0407272
42. *White S.R., Sugar R.L., Scalletar R.T.* Algorithm for the Simulation of Many-Electron Systems at Low Temperatures // *Phys. Rev. B.* 1988. V.38, No.16. P. 11665–11668.
43. *Beyl S., Goth F., Assaad F.* Revisiting the Hybrid Quantum Monte Carlo Method for Hubbard and Electron–Phonon Model // *Phys. Rev. B* 2018. V. 97, No. 8. P.085144; arXiv:1708.03661 [cond-mat.str-el]
44. *Goodfellow I., Bengio Y., Courville A.* Deep Learning. MIT Press, 2016.
45. *Nielsen M.A.* Neural Networks and Deep Learning. V. 25. San Francisco, CA, USA: Determination Press, 2015.
46. *Cybenko G.* Approximation by Superpositions of a Sigmoidal Function // *Math. of Control, Signals, and Systems.* 1989. V.2, No.4. P.303–314.
47. *Paszke A., Gross S., Chintala S., Chanan G., Yang E., DeVito Z., Lin Z., Desmaison A., Antiga L., Lerer A.* Automatic Differentiation in Pytorch // *Proc. of the 31st Conf. on Neural Information Processing Systems, NIPS 2017.* Long Beach, CA, USA, 2017. P. 1–4.
48. *Noh H., Hong S., Han B.* Learning Deconvolution Network for Semantic Segmentation // *Proc. of IEEE Intern. Conf. on Computer Vision, Santiago, Chile, 2015.* P. 1520–1528.
49. *Long J., Shelhamer E., Darrell T.* Fully Convolutional Networks for Semantic Segmentation // *Ibid.* P. 3431–3440.
50. *LeCun Y., Bottou L., Bengio Y., Hanner P.* Gradient-Based Learning Applied to Document Recognition // *Ibid.* P. 2278–2324.
51. *Krizhevsky A., Sutskever I., Hinton G.E.* Imagenet Classification with Deep Convolutional Neural Networks // *Advances in Neural Information Processing Systems, Curran Associates, Inc. / Ed. by F.Pereira, C.J.C.Burges, L.Bottou, and K.Q.Weinberger.* 2012. P. 1097–1105.
52. *Dumoulin V., Visin F.* A Guide to Convolution Arithmetic for Deep Learning. arXiv:1603.07285. 2016.
53. *Ronneberger O., Fischer P., Brox T.* U-Net: Convolutional Networks for Biomedical Image Segmentation // *Proc. Intern. Conf. on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, Oct. 5–9, 2015.* P. 234–241.
54. *Clevert D.-A., Unterthiner T., Hochreiter S.* Fast and Accurate Deep Network Learning by Exponential Linear Units (Elus). arXiv:1511.0728. 2015.
55. *He K., Zhang X., Ren S., Sun J.* Deep Residual Learning for Image Recognition // *Proc. of IEEE Intern. Conf. on Computer Vision, Las Vegas, NV, USA, 2016.* P. 770–778.

56. *Kumar Srivastava R., Greff K., Schmidhuber J.* Highway Networks. arXiv:1505.00387. 2015.
57. *Kingma D. P., Ba J.* Adam: A Method for Stochastic Optimization. arXiv:1412.6980. 2014.
58. *Ronneberger O., Fischer P., Brox T.* U-Net: Convolutional Networks for Biomedical Image Segmentation, Medical Image Computing and Computer-Assisted Intervention // Proc. of Intern. Conf. on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, Oct. 5–9, 2015. 2016. P. 234–241.
59. *Ulybyshev M., Winterowd C., Zafeiropoulos S.* Lefschetz Thimbles Decomposition for the Hubbard Model on the Hexagonal Lattice. arXiv:1906.07678. 2019.
60. *Ulybyshev M., Winterowd C., Zafeiropoulos S.* Taming the Sign Problem of the Finite Density Hubbard Model via Lefschetz Thimbles. arXiv:1906.02726. 2019.