学习记录

Bo Zhao (赵博)
(Dated: June 21, 2024)

蓝色是以后会用的/需要注意的，红色是待完成项。

手上压的学习任务有点多，之后先专心过基础知识，之后有多余时间把之前看的几篇文章好好整理一下，之后顺着看一下Wave-Function Tomography of Topological Dimer Chains with Long-Range Couplings，这篇应该和之前看的声子晶体Large winding number比较相似，顺便可以尝试一下用comsol复现。都了结之后做comsol官网声子晶体的案例（包含色散关系），尝试复现Large winding number仿真部分。看一下Ref. [1] (Realization of fractional quantum Hall state with interacting photons)。

2024/06/20 2024/06/21

复现宇宸师兄在群里发的文章(基于紧束缚模型求解螺旋波导阵列)Ref. [2]. Ref. [2]是Nature文章Photonic Floquet topological insulators拓扑理论部分的学习笔记 [3]。Ref. [3]的亮点在于：1. 光波段实现（基于打破时间反演）。2. 新的拓扑绝缘体实现方式。3. 无磁，更易小型化集成化（作者没说，我认为的）。螺旋波导电磁理论：Ref. [4]。拓扑绝缘体的哈密顿量是这样实现的：构建模型中的第三维度z映射时间。变换到旋转框架，波导螺旋周期需足够小，以实现z方向的绝热性（不懂，过完量力的绝热近似再说）。根据紧束缚方法，得到电场的动力学演化（看完紧束缚模型再说）。由动力学演化，用海森堡方程，或者量子郎之万方程反向变回去(文章上说用Peierls substitution，不过我大致估算了一下，海森堡方程就能得到相同结果。）就可以得到Hamiltonian。

该Hamiltonian含周期性变化的第三维度，使用Floquet operator method和Fourier expansion method进行处理后按套路计算即可。能谱图还在画，争取今天画出来。

2024/06/19

感觉无心学习，稍微提前回去休息了。18，19号休息周六日补上。

2024/06/18

17号弄到太晚，18号一直很困就回去休息了.

2024/06/17

重复comsol案例：光子晶体和光子晶体带隙分析。注意光子晶体案例解出的电场是有虚部的，所以最后的线图直接用默认的normE和二维图对不上。comsol默认Ez显示实部解，因此ewfd.normE对应的不是sqrt(ewfd.Ex^2 + ewfd.E^2 + ewfd.Ez^2)，而是sqrt(real(ewfd.Ex)^2 + imag(ewfd.Ex)^2 + real(ewfd.Ey)^2 + imag(ewfd.Ey)^2 + real(ewfd.Ez)^2 + imag(ewfd.Ez)^2)。

2024/06/15 & 2024/06/16

休息。

This week

2024/06/14

看完Griffiths量子力学第四章。看高量课程，补画了个别图片。

2024/06/13

看Griffiths量子力学第四章。听了严以京关于开放系统中量子力学的报告。

2024/06/12

看Griffiths量子力学第四章。

2024/06/11

完成Griffiths量子力学第三章，读部分第四章。

2024/06/10

下午：读Griffiths量子力学§3.1和§3.2并完成重点*习题。

2024/06/08 & 2024/06/09

从8号中午看到9号早上。

总结：完成习题1.3，浏览复现文章Ref. [5]（Large Winding number）Winding number图片(其他图片套路和之前基本一致，就没画)。关于comsol仿真部分复现失败，对仿真掌握的知识太少，学过后可以回来尝试复现一下。

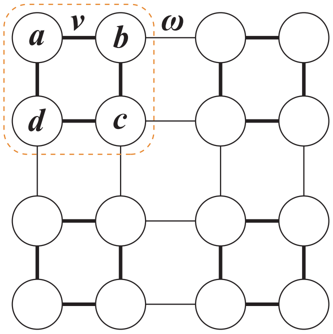


FIG. 1. 二维SSH模型。

习题1.3 *A possible generalization to two dimensions*-Consider a two dimensional generalization of the SSH model. Take parallel copies of the SSH chain and couple them without breaking chiral symmetry. What will happen with the edge states?

答：习题1.3的答案有人在2017年发过prl（该书是2016年出版的）[6]。设二维SSH模型cell中子晶格顺时针标号为abcd，x(y)方向cell标号为i(j),如FIG. 1所示，Hamiltonian为:

$$
\begin{aligned}
\hat{H} =& v\sum_{i,j}^{N_i,N_j}(\hat{a}^\dagger_{i,j}\hat{b}_{i,j}+\hat{b}^\dagger_{i,j}\hat{c}_{i,j}+\hat{c}^\dagger_{i,j}\hat{d}_{i,j}+\hat{d}^\dagger_{i,j}\hat{a}_{i,j})\\
&+w\sum_{i,j}^{N_{i-1},N_j}(\hat{b}^\dagger_{i,j}\hat{a}_{i+1,j}+\hat{c}^\dagger_{i,j}\hat{d}_{i+1,j})\\
&+w\sum_{i,j}^{N_i,N_{j-1}}(\hat{d}^\dagger_{i,j}\hat{a}_{i,j+1}+\hat{c}^\dagger_{i,j}\hat{b}_{i,j+1})+\text{H.c.}.
\end{aligned}
$$

动量空间的Hamiltonian:

$$
\hat{H}(k)=\begin{pmatrix}
0 & v+we^{-ik_x} & 0 & v+we^{-ik_y}\\
v+we^{ik_x} & 0 & v+we^{-ik_y} & 0\\
0 & v+we^{ik_y} & 0 & v+we^{-ik_x}\\
v+we^{ik_y} & 0 & v+we^{ik_x} & 0
\end{pmatrix}.
$$

先画个色散关系，如FIG. 2所示。

按Ref. [7]第一章的思路继续写不出了，因为不知道如何把$\hat{H}$(k)写成实空间的$\hat{H}$来画能谱图以及波函数分布。

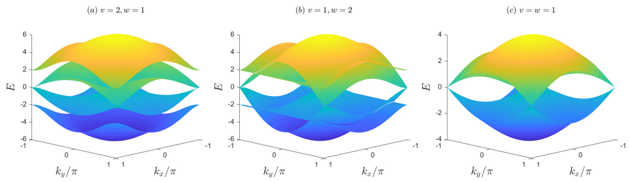


FIG. 2. 2D SSH模型色散关系。

注意到Ref. [6]限制一个方向(y方向)无边界，另一个方向有界，利用该思想可以得到部分实空间的Hamiltonian:

$$
\hat{H}(k)=\begin{pmatrix}
A & B & 0 & \cdots & 0\\
B^\dagger & A & B & \cdots & 0\\
0 & B^\dagger & A & \cdots & 0\\
\vdots & \vdots & \vdots & \ddots & B\\
0 & 0 & 0 & \cdots & A
\end{pmatrix}.
$$

即主对角线是A矩阵，超对角线是B矩阵，次对角线是$B^\dagger$矩阵。其中

$$
A=\begin{pmatrix}
0 & v & 0 & v+we^{-ik_y}\\
v & 0 & v+we^{-ik_y} & 0\\
0 & v+we^{ik_y} & 0 & v\\
v+we^{ik_y} & 0 & v & 0
\end{pmatrix},
$$

$$
B=\begin{pmatrix}
0 & 0 & 0 & 0\\
w & 0 & 0 & 0\\
0 & 0 & 0 & w\\
0 & 0 & 0 & 0
\end{pmatrix}.
$$

可以据此画出能谱图，如FIG. 3所示。原文似乎有些问题，跟据我所绘的图，蓝色能量对应的波函数并非边界态。

关于参考文献Ref. [6]，过完基础知识之后回来仔细看一看，了解物理图像。

下图是Ref. [5]的复现图，相比于我构想的winding number=2的模型，该模型动量空间的哈密顿量e指数符号不同，因此体现在实空间中的耦合项相差甚远。特别是，该模型还考虑的B模式与上一cell中A模式的耦合。但是在代码上几乎相同，此处不给出复现代码（用之前的调调参数，Hamiltonian改改符号即可）。Ref. [5]中Appendix A中动量空间Hamiltonian系数v1对应e指数少了个负号（复现图片会出现差异）。

能谱图如FIG。 5所示。但是看到文中标明4个边界态，同时就尝试在我的模型Winding number=2情况看一下，也出现了4个边界态，如FIG. 6所示。很神奇，后面继续学的时候关注一下相关知识。

2024/06/07

来了办公室但是超级困，周六日再来学吧。

2024/06/06

上午：睡觉（前两天熬的太晚）。

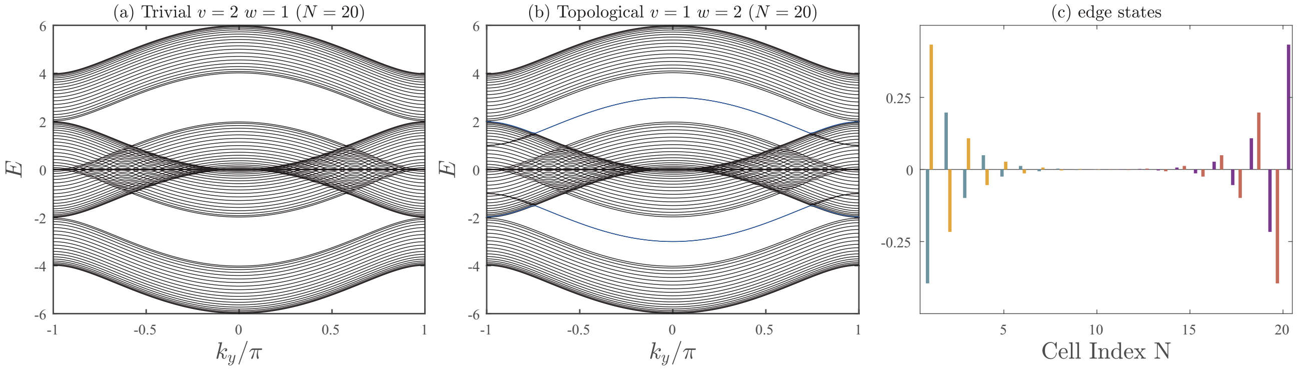


FIG. 3. 2D SSH模型能带(a)(b)以及边界态波函数分布(c)。

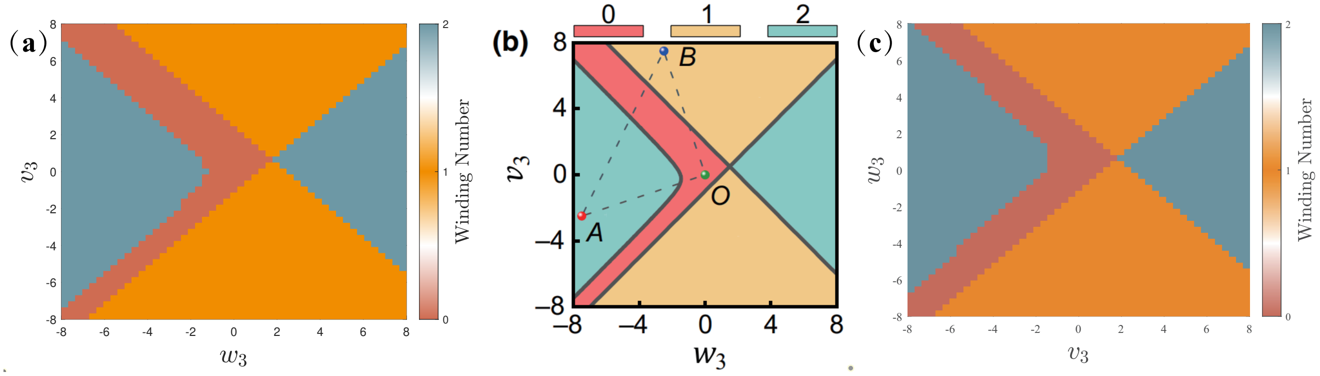


FIG. 4. Winding Number of Ref. [5]，（a）使用修正后Hamitonian。（b）原图。（c）使用原文Hamitonian。

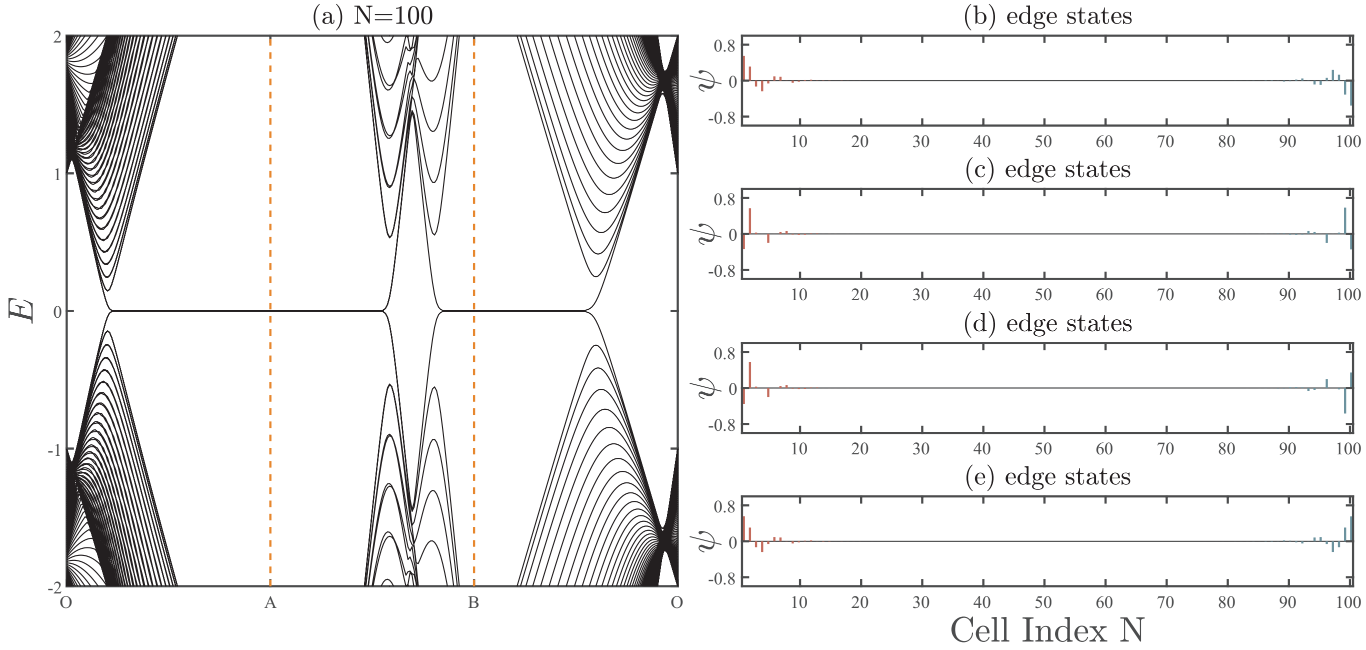


FIG. 5. Winding Number=2存在四个边界态,原文复现（原文N=120,但结果相同），2024/06/14补画

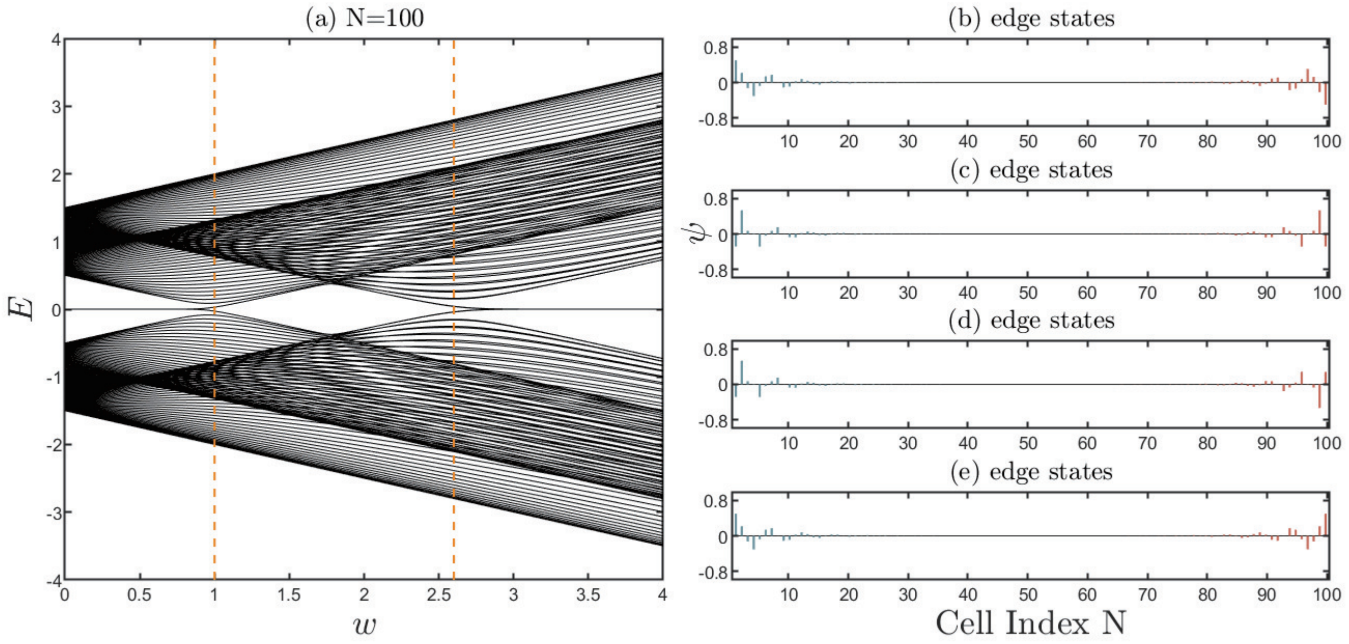


FIG. 6. Winding Number=2存在四个边界态

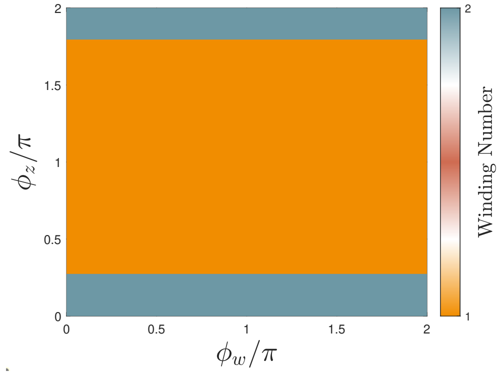


FIG. 7. 相位控制的Winding number。

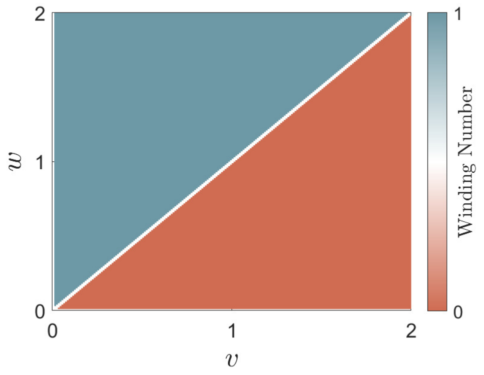


FIG. 8. SSH模型的Winding number。

下午：整理重绘复现图片，完成习题1.2（处理winding number图片变化交界处的方法不一样，所以三张图风格不统一，分别使用的是手动取消网格，光滑处理，光滑处理）。

习题1.2：*Complex-valued hopping amplitudes*- Generalize the SSH model in the following way. Assume that the hopping amplitudes $v = |v|e^{i\psi_v}$ and $w = |w|e^{i\psi_w}$ are complex, and include a third complex-valued hopping amplitude $z = |z|e^{i\psi_z}$ between the states $|m, A\rangle$ and $|m+1, B\rangle$ for every m. Provide a specific example where the tuning of one of the phases changes the bulk winding number.

答：和我设计的结构很像，但是远耦合方向相反且差个系数，并考虑相位。相位控制的Winding number如Fig. 7所示，v=0.5。w=0.5。代码如CODE4所示。

2024/06/04 & 2024/06/05

读完*A Short Course on Topological Insulators*的第一章并复现图片 [7]，如FIG. 8所示,代码位于CODE3。写了一个自动算2 × 2 Hamiltonian（频域）系统Winding number的函数，复现代码中可查。完成习题1.1并绘制图片。

习题1.1：*Higher winding numbers*- The SSH model is one-dimensional in space, and has a two-dimensional internal Hilbert space. Construct a lattice model that has these properties of the SSH model, but which has a bulk winding number of 2. Generalize the construction for an arbitrary integer bulk winding number.

答：如FIG. 9所示。代码如CODE2所示。

$$\hat{H}(k) = \begin{pmatrix} 0 & v + e^{-ik} + 0.5we^{2ik} \\ v + e^{ik} + 0.5we^{-2ik} & 0 \end{pmatrix}.$$

2024/06/03

上午 & 下午：完成Griffiths量子力学第二章重点（*）习题。

2024/06/02

休息 & 读Griffiths量子力学第二章。

2024/06/01

休息。

2024/05/31

摸鱼了。

2024/05/30

上午：做学习记录。
下午：完成Griffiths量子力学第一章重点（*，**）习题。

2024/05/29

上午：做Griffiths量子力学$P_{1.1} - P_{1.7}$习题。
下午：读*A Short Course on Topological Insulators*的§ 1.1-§ 1.3并复现如FIG. 10所示（直积不懂不会算，返回复习基础看过Griffiths第三章再往下阅读复现）[7]。MATLAB代码如CODE1所示（B、C、DE）。

2024/05/28

上午：看高量课程，复习经典力学。
下午：阅读Griffiths量子力学。

2024/05/27

重新制定基础知识学习计划，如FIG. 11所示。图书馆借相关书籍，看高量课程。

2024/05/25 & 2024/05/26

周六日休息。

2024/05/24

收拾房间，初步了解金属自由电子气体模型。

2024/05/23

办理校园卡，登记入住宿舍，收拾房间，选择工位。

[1] C. Wang, *et al.*, Realization of fractional quantum Hall state with interacting photons, Science **384**, 579 (2024), https://www.science.org/doi/pdf/10.1126/science.ado3912.

[2] H. ZHONG, D. MIHALACHE, S. SHEN, and Y. ZHANG, THE BAND STRUCTURE OF HELICAL WAVEGUIDE ARRAYS IN TOPOLOGICAL PHOTONICS: A TUTORIAL, .

[3] M. C. Rechtsman, *et al.*, Photonic Floquet topological insulators, Nature **496**, 196 (2013).

[4] F. Lederer, *et al.*, Discrete solitons in optics, Physics Reports **463**, 1 (2008).

[5] H. Liu, *et al.*, Acoustic Topological Metamaterials of Large Winding Number, Phys. Rev. Appl. **19**, 054028 (2023).

[6] F. Liu and K. Wakabayashi, Novel Topological Phase with a Zero Berry Curvature, Phys. Rev. Lett. **118**, 076803 (2017).

[7] J. K. Asb‘th, L. Oroszl‘ny, and A. P‘lyi, *A Short Course on Topological Insulators* (Springer International Publishing, 2016).

CODE4

```
tic
```

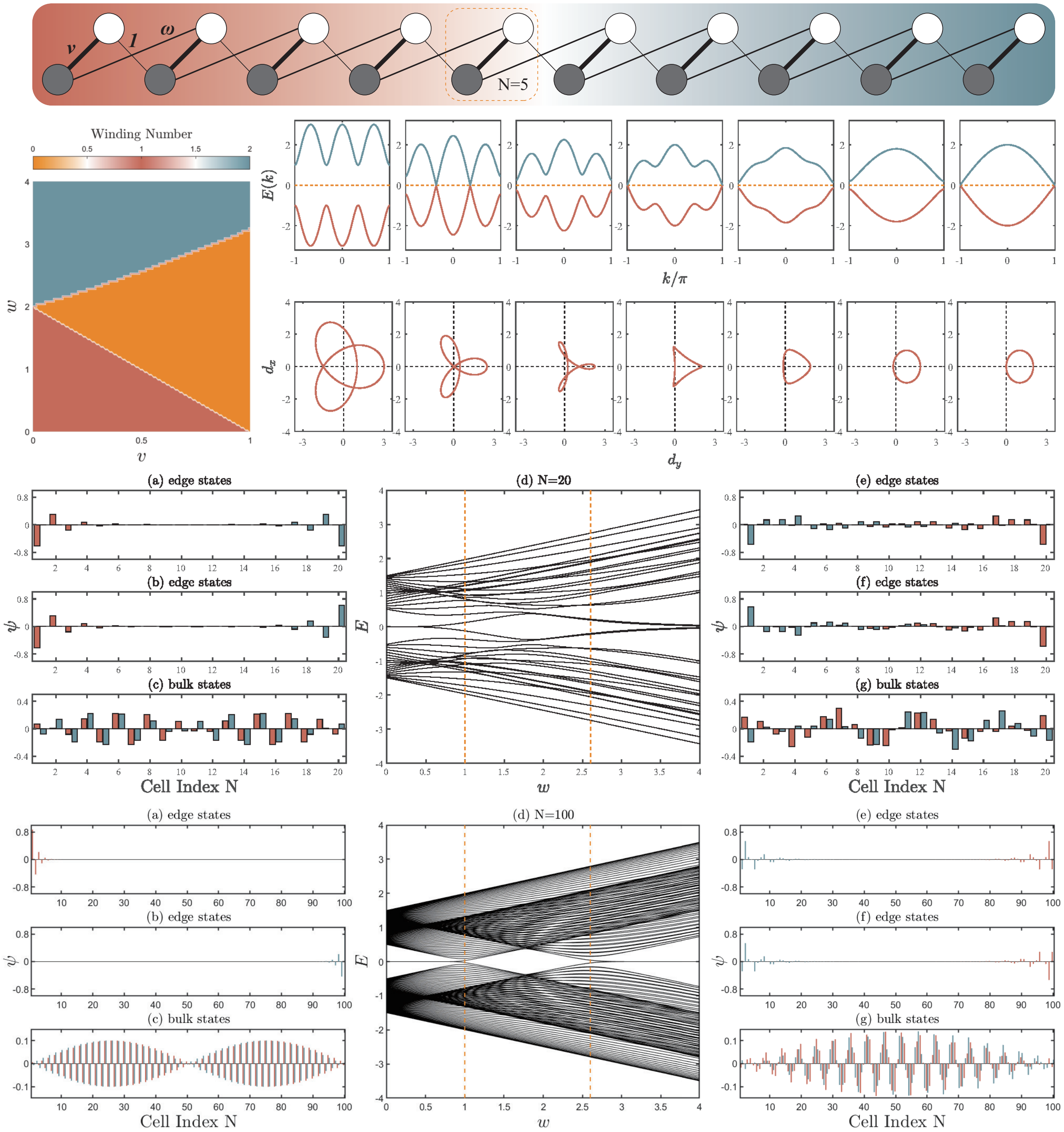


FIG. 9. 设计的Winding number=2的拓扑绝缘体。（Hypocycloid，圆内螺线）

```
clc
clear all

syms k;

rate=30;
parfor_progress(rate);

n=0;
vj=linspace(0,0*pi,rate);
```

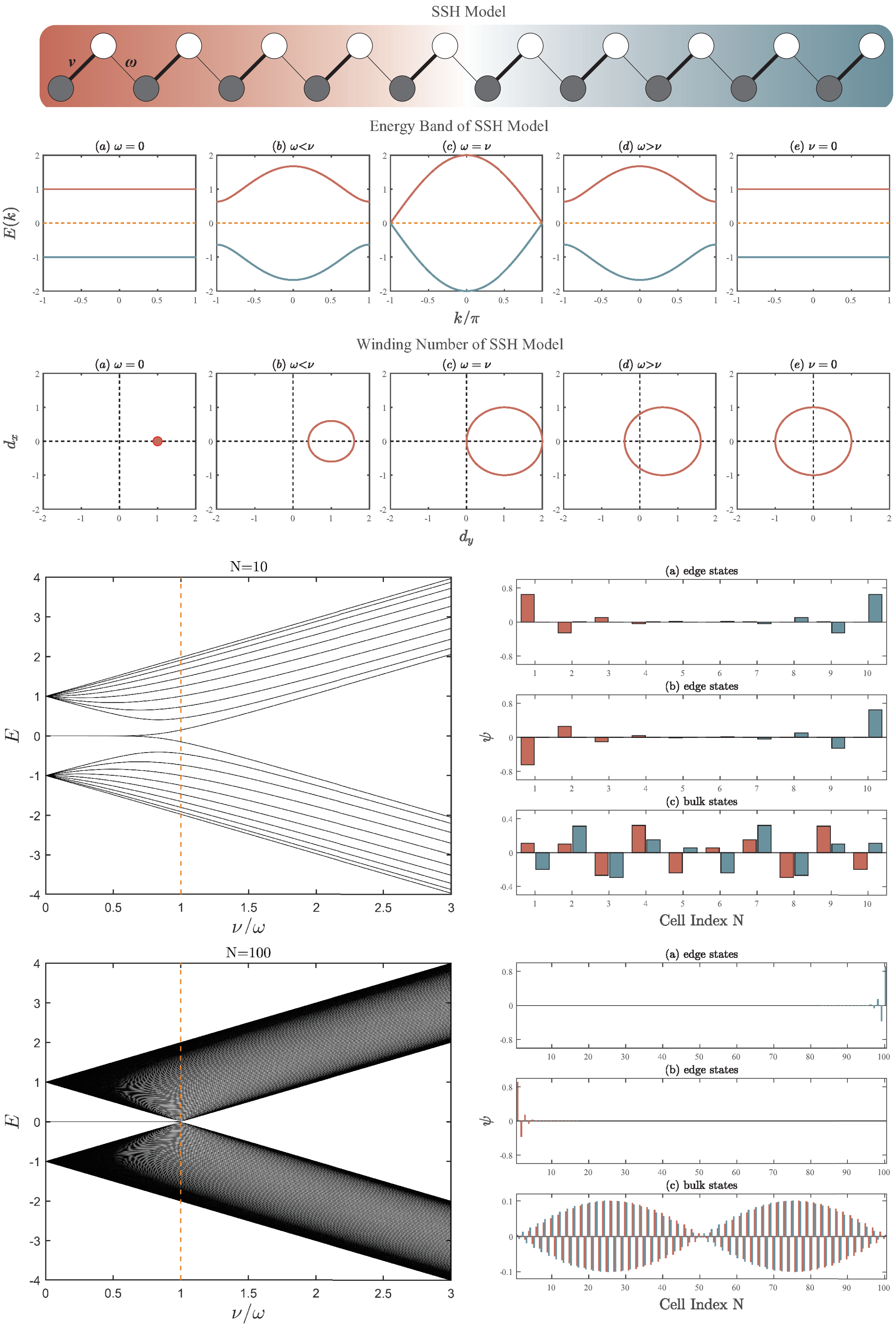


FIG. 10. SSH模型复现图片。自上而下分别为：A. SSH模型图（cell index N=10）。B. SSH模型的能带（bulk部分）。C. SSH模型的Winding number。D. N=10时SSH模型的能谱及波函数。（ν = 0.4, 对应自下而上第10，11，8条能带）E. N=100时SSH模型的能谱及波函数。（ν = 0.4, 对应自下而上第100，101，98条能带）

```
wj=linspace(0,2*pi,rate);
zj=linspace(0,2*pi,rate);
v0=linspace(0.5,0.5,rate);
w0=linspace(1,1,rate);
z0=linspace(1,1,rate);

w=1;

parfor a = 1
    pause(rand); % Replace with real code
    parfor_progress;
            v=v0(a)*exp(1i*vj(a));
    for b = 1 : rate

        w=w0(b)*exp(1i*wj(b));
        z=z0(b)*exp(1i*zj(b));
        H_k = [0 v+w*exp(-1i*k)+z*exp(-2i*k); conj(v)+conj(w)*exp(1i*k)+conj(z)*exp(2i*k) 0];
        [winding_number, d, d_0] = calculateWindingNumber(H_k);
        wn(a,b) = abs(winding_number);
        if abs(wn(a,b)) <= 1e-3 || (abs(wn(a,b))-1) <= 1e-3 || (abs(wn(a,b))-2) <= 1e-3
        else
            wn(a,b) = NaN;
        end
    end
end
parfor_progress(0);
toc
%%
color0=[1,1,1];
color2 = [207, 108, 82] / 255;
color1 = [241,141,0]/255;
color3 = [109, 152, 165] / 255;

Map = [linspace(color1(1), color0(1), 128)', linspace(color1(2), color0(2), 128)', linspace(color1(3), color0(3), 128)';
    linspace(color0(1), color2(1), 128)', linspace(color0(2), color2(2), 128)', linspace(color0(3), color2(3), 128)';
    linspace(color2(1), color0(1), 128)', linspace(color2(2), color0(2), 128)', linspace(color2(3), color0(3), 128)';
    linspace(color0(1), color3(1), 128)', linspace(color0(2), color3(2), 128)', linspace(color0(3), color3(3), 128)'];

%%
pcolor(wj/(pi),zj/(pi),abs(wn)')
xlabel('$\phi_{w}/\pi$','interpreter','latex','FontSize',22)
ylabel('$\phi_{z}/\pi$','interpreter','latex','FontSize',22)
% axis([0 2 0 2])
c = colorbar('Ticks',[1 2],'TickLabels',{'1','2'});;%('ticks',linspace(-3,21,5),'FontSize',11);
c.Label.FontSize=16;
c.Label.Interpreter='Latex';
c.Label.String = '$\mathrm{Winding\ Number}$';
c.Label.HorizontalAlignment = "center";
colormap(Map);
```

```

delete(gcp);


%%
function [winding_number, d, d_0] = calculateWindingNumber(H_k)
syms k real;

sigma_0 = [1 0; 0 1];
sigma_x = [0 1; 1 0];
sigma_y = [0 -1i; 1i 0];
sigma_z = [1 0; 0 -1];

A = [reshape(sigma_0.', [], 1), reshape(sigma_x.', [], 1), reshape(sigma_y.', [], 1), reshape(sigma_z.', [], 1)];
b = reshape(H_k.', [], 1);

coefficients = A \ b;

d_0 = coefficients(1);
d_x = coefficients(2);
d_y = coefficients(3);
d_z = coefficients(4);

d = [d_x; d_y; d_z];
d = simplify(d);

d_dk = diff(d, k);

cross_product = simplify(cross(d, d_dk));
norm_d_squared = simplify(norm(d)^2);
integrand = cross_product(3) / norm_d_squared;

func = matlabFunction(integrand, 'Vars', {k});

winding_number = integral(func, -pi, pi) / (2 * pi);
end

%%
function percent = parfor_progress(N)
%PARFOR_PROGRESS Progress monitor (progress bar) that works with parfor.
%   PARFOR_PROGRESS works by creating a file called parfor_progress.txt in
%   your working directory, and then keeping track of the parfor loop's
%   progress within that file. This workaround is necessary because parfor
%   workers cannot communicate with one another so there is no simple way
%   to know which iterations have finished and which haven't.
%
%   PARFOR_PROGRESS(N) initializes the progress monitor for a set of N
%   upcoming calculations.
%
%   PARFOR_PROGRESS updates the progress inside your parfor loop and
%   displays an updated progress bar.
%
%   PARFOR_PROGRESS(0) deletes parfor_progress.txt and finalizes progress
%   bar.
%
```

```
%   To suppress output from any of these functions, just ask for a return
%   variable from the function calls, like PERCENT = PARFOR_PROGRESS which
%   returns the percentage of completion.
%
%   Example:
%
%      N = 100;
%      parfor_progress(N);
%      parfor i=1:N
%         pause(rand); % Replace with real code
%         parfor_progress;
%      end
%      parfor_progress(0);
%
%   See also PARFOR.

% By Jeremy Scheff - jdscheff@gmail.com - http://www.jeremyscheff.com/

error(nargchk(0, 1, nargin, 'struct'));

if nargin < 1
    N = -1;
end

percent = 0;
w = 50; % Width of progress bar

if N > 0
    f = fopen('parfor_progress.txt', 'w');
    if f<0
        error('Do you have write permissions for %s?', pwd);
    end
    fprintf(f, '%d\n', N); % Save N at the top of progress.txt
    fclose(f);

    if nargout == 0
        disp(['  0%[>', repmat(' ', 1, w), ']']);
    end
elseif N == 0
    delete('parfor_progress.txt');
    percent = 100;

    if nargout == 0
        disp([repmat(char(8), 1, (w+9)), char(10), '100%[', repmat('=', 1, w+1), ']']);
    end
else
    if ~exist('parfor_progress.txt', 'file')
        error('parfor_progress.txt not found. Run PARFOR_PROGRESS(N) before PARFOR_PROGRESS to initialize parfor_progress.txt.');
    end

    f = fopen('parfor_progress.txt', 'a');
    fprintf(f, '1\n');
    fclose(f);

    f = fopen('parfor_progress.txt', 'r');
    progress = fscanf(f, '%d');
    fclose(f);
    percent = (length(progress)-1)/progress(1)*100;

    if nargout == 0
        perc = sprintf('%3.0f%%', percent); % 4 characters wide, percentage
        disp([repmat(char(8), 1, (w+9)), char(10), perc, '[', repmat('=', 1,
            round(percent*w/100)), '>', repmat('-', 1, w - round(percent*w/100)),
            ']']);
    end
end
```

CODE3

```
tic
clc
clear all

syms k;

rate=20;
% parfor_progress(rate);

n=0;
va = linspace(0,2,rate+1);
va(1) = [];
wb = linspace(0,2,rate+1);
wb(1) = [];
wn = NaN(rate, rate);

parpool;
parfor a = 1 : rate
    % pause(rand); % Replace with real code
    % parfor_progress;
    v=va(a);
    for b = 1 : rate
        w=wb(b);
        H_k = [0 v + w*exp(-1i*k); v + w*exp(1i*k) 0];
        [winding_number, d, d_0] = calculateWindingNumber(H_k);
        if v == w
            wn(a,b) = NaN;
        else
            wn(a,b) = winding_number;
        end
    end
end
delete(gcp);
% parfor_progress(0);
toc
%%
color1 = [207, 108, 82] / 255;
color2 = [1, 1, 1];
```

```
color3 = [109, 152, 165] / 255;

Map = [linspace(color1(1), color2(1), 128)', linspace(color1(2), color2(2), 128)', linspace(color1(3), color2(3), 128)';
    linspace(color2(1), color3(1), 128)', linspace(color2(2), color3(2), 128)', linspace(color2(3), color3(3), 128)'];

%%
pcolor(va,wb,wn')
shading interp
set(gca,'XTick',[0,1,2],'FontSize',16)
set(gca,'YTick',[0,1,2],'FontSize',16)
xlabel('$v$','interpreter','latex','FontSize',22)
ylabel('$w$','interpreter','latex','FontSize',22)
axis([0 2 0 2])
c = colorbar('Ticks',[0 1]);;%('ticks',linspace(-3,21,5),'FontSize',11);
c.Label.FontSize=16;
c.Label.Interpreter='Latex';
c.Label.String = '$\mathrm{Winding\ Number}$';
c.Label.HorizontalAlignment = "center";
colormap(Map);
%save("Phase_diagram_winding_number",'-mat')
%%
function [winding_number, d, d_0] = calculateWindingNumber(H_k)
syms k real;

sigma_0 = [1 0; 0 1];
sigma_x = [0 1; 1 0];
sigma_y = [0 -1i; 1i 0];
sigma_z = [1 0; 0 -1];

A = [reshape(sigma_0.', [], 1), reshape(sigma_x.', [], 1), reshape(sigma_y.', [], 1), reshape(sigma_z.', [], 1)];
b = reshape(H_k.', [], 1);

coefficients = A \ b;

d_0 = coefficients(1);
d_x = coefficients(2);
d_y = coefficients(3);
d_z = coefficients(4);

d = [d_x; d_y; d_z];
d = simplify(d);

d_dk = diff(d, k);

cross_product = simplify(cross(d, d_dk));
norm_d_squared = simplify(norm(d)^2);
integrand = cross_product(3) / norm_d_squared;

func = matlabFunction(integrand, 'Vars', {k});

winding_number = integral(func, -pi, pi) / (2 * pi);
end
```

CODE2

```
tic
clc
clear all

syms k;

rate=20;
parfor_progress(rate);

n=0;
va = linspace(0.0001,1,rate);
wb = linspace(0.0001,4,rate);
wn = NaN(rate, rate);

parfor a = 1 : rate
    parfor_progress;
    v=va(a);
    for b = 1 : rate
        w=wb(b);
        H_k = [0 v+1()*exp(-1i*k) + 0.5*w*exp(1i*2*k); v+1*exp(1i*k) + 0.5*w*exp(-1i*2*k) 0];
        [winding_number, d, d_0] = calculateWindingNumber(H_k);
        wn(a,b) = abs(winding_number);
        if abs(wn(a,b)) <= 1e-3 || (abs(wn(a,b))-1) <= 1e-3 || (abs(wn(a,b))-2) <= 1e-3
        else
            wn(a,b) = NaN;
        end
    end
end
parfor_progress(0);
toc
%%
color0=[1,1,1];
color2 = [207, 108, 82] / 255;
color1 = [241,141,0]/255;
color3 = [109, 152, 165] / 255;

Map = [linspace(color1(1), color0(1), 128)', linspace(color1(2), color0(2), 128)', linspace(color1(3), color0(3), 128)';
    linspace(color0(1), color2(1), 128)', linspace(color0(2), color2(2), 128)', linspace(color0(3), color2(3), 128)';
    linspace(color2(1), color0(1), 128)', linspace(color2(2), color0(2), 128)', linspace(color2(3), color0(3), 128)';
    linspace(color0(1), color3(1), 128)', linspace(color0(2), color3(2), 128)', linspace(color0(3), color3(3), 128)'];

%%
pcolor(va,wb,abs(wn)')
shading interp
% set(gca,'XTick',[0,1,2],'FontSize',16)
% set(gca,'YTick',[0,1,2],'FontSize',16)
xlabel('$v$','interpreter','latex','FontSize',22)
```

```
ylabel('$w$','interpreter','latex','FontSize',22)
% axis([0 2 0 2])
c = colorbar;%('ticks',linspace(-3,21,5),'FontSize',11);
c.Label.FontSize=16;
c.Label.Interpreter='Latex';
c.Label.String = '$\mathrm{Winding\~Number}$';
c.Label.HorizontalAlignment = "center";
colormap(Map);
% save('WN_012.mat')
delete(gcp);

%%
function [winding_number, d, d_0] = calculateWindingNumber(H_k)
syms k real;

sigma_0 = [1 0; 0 1];
sigma_x = [0 1; 1 0];
sigma_y = [0 -1i; 1i 0];
sigma_z = [1 0; 0 -1];

A = [reshape(sigma_0.', [], 1), reshape(sigma_x.', [], 1), reshape(sigma_y.', [],
     1), reshape(sigma_z.', [], 1)];
b = reshape(H_k.', [], 1);

coefficients = A \ b;

d_0 = coefficients(1);
d_x = coefficients(2);
d_y = coefficients(3);
d_z = coefficients(4);

d = [d_x; d_y; d_z];
d = simplify(d);

d_dk = diff(d, k);

cross_product = simplify(cross(d, d_dk));
norm_d_squared = simplify(norm(d)^2);
integrand = cross_product(3) / norm_d_squared;

func = matlabFunction(integrand, 'Vars', {k});

winding_number = integral(func, -pi, pi) / (2 * pi);
end

%%
function percent = parfor_progress(N)
%PARFOR_PROGRESS Progress monitor (progress bar) that works with parfor.
%   PARFOR_PROGRESS works by creating a file called parfor_progress.txt in
%   your working directory, and then keeping track of the parfor loop's
%   progress within that file. This workaround is necessary because parfor
%   workers cannot communicate with one another so there is no simple way
%   to know which iterations have finished and which haven't.
%
%   PARFOR_PROGRESS(N) initializes the progress monitor for a set of N
%   upcoming calculations.
```

```
%
%   PARFOR_PROGRESS updates the progress inside your parfor loop and
%   displays an updated progress bar.
%
%   PARFOR_PROGRESS(0) deletes parfor_progress.txt and finalizes progress
%   bar.
%
%   To suppress output from any of these functions, just ask for a return
%   variable from the function calls, like PERCENT = PARFOR_PROGRESS which
%   returns the percentage of completion.
%
%   Example:
%
%      N = 100;
%      parfor_progress(N);
%      parfor i=1:N
%         pause(rand); % Replace with real code
%         parfor_progress;
%      end
%      parfor_progress(0);
%
%   See also PARFOR.

% By Jeremy Scheff - jdscheff@gmail.com - http://www.jeremyscheff.com/

error(nargchk(0, 1, nargin, 'struct'));

if nargin < 1
    N = -1;
end

percent = 0;
w = 50; % Width of progress bar

if N > 0
    f = fopen('parfor_progress.txt', 'w');
    if f<0
        error('Do you have write permissions for %s?', pwd);
    end
    fprintf(f, '%d\n', N); % Save N at the top of progress.txt
    fclose(f);

    if nargout == 0
        disp(['  0%[>', repmat(' ', 1, w), ']']);
    end
elseif N == 0
    delete('parfor_progress.txt');
    percent = 100;

    if nargout == 0
        disp([repmat(char(8), 1, (w+9)), char(10), '100%[', repmat('=', 1, w+1),
            ']']);
    end
else
    if ~exist('parfor_progress.txt', 'file')
```

```
        error('parfor_progress.txt not found. Run PARFOR_PROGRESS(N) before PARFOR_PROGRESS to initialize parfor_progress.txt.');
    end

    f = fopen('parfor_progress.txt', 'a');
    fprintf(f, '1\n');
    fclose(f);

    f = fopen('parfor_progress.txt', 'r');
    progress = fscanf(f, '%d');
    fclose(f);
    percent = (length(progress)-1)/progress(1)*100;

    if nargout == 0
        perc = sprintf('%3.0f%%', percent); % 4 characters wide, percentage
        disp([repmat(char(8), 1, (w+9)), char(10), perc, '[', repmat('=', 1, round(percent*w/100)), '>', repmat(' ', 1, w - round(percent*w/100)), ']']);
    end
end
```

```
clear all
clc
f1=figure(1);
position0=get(f1,'position');
set(f1,'position',position0+[-0.8*position0(3),0,1.6*position0(3),-0.3*position0(4)]);
t=tiledlayout("horizontal","TileSpacing","tight");
titlej={'$(a)\ \omega=0$';'$(b)\ \omega\textless\nu$';'$(c)\ \omega=\nu$';'$(d)\ \omega\textgreater\nu$';'$(e)\ \nu=0$'};

rate_k=1000;

vj=[0,0.3,0.5,0.6,0.7,0.8,1];
wj=[4,2.3,1.5,0.8,0.3,0,0];
ki=linspace(-pi,pi,rate_k);

for j = 1 : 7
    v=vj(j);
    w=wj(j);
    for i = 1 : rate_k
        k=ki(i);
        H_k = [0 v+1*exp(-1i*k) + 0.5*w*exp(1i*2*k); v+1*exp(1i*k) + 0.5*w*exp(-1i*2*k) 0];
        [d, d_0] = WindingNumberFigure(H_k);
        d_x(i)=d(1);
        d_y(i)=d(2);
    end
    nexttile
    plot(linspace(-3.6,3.6,100),linspace(0,0,100),'LineWidth',1,'linestyle','--','Color','k')
    hold on
    plot(linspace(0,0,100),linspace(-4,4,100),'LineWidth',1,'linestyle','--','Color','k')
    hold on
    plot(d_x,d_y,'LineWidth',1.5,'Color',[207,108,82]/255);
    hold on
    set(gca,'LineWidth',1);
    axis([-3.6,3.6,-4,4])
    set(gca,'ytick',[-4 -2 0 2 4],'FontSize',12);
    set(gca,'xtick',[-3 0 3],'FontSize',12);
   % title(titlej{j},'interpreter','latex','FontSize',12);
end
% title(t,'Winding Number of SSH Model','FontSize',18);
xlabel(t,'$d_{y}$','interpreter','latex','FontSize',18);
ylabel(t,'$d_{x}$','interpreter','latex','FontSize',18);

function [d, d_0] = WindingNumberFigure(H_k)
syms k v w

sigma_0 = [1 0; 0 1];
sigma_x = [0 1; 1 0];
sigma_y = [0 -1i; 1i 0];
sigma_z = [1 0; 0 -1];

A = [reshape(sigma_0.', [], 1), reshape(sigma_x.', [], 1), reshape(sigma_y.', [], 1), reshape(sigma_z.', [], 1)];
b = reshape(H_k.', [], 1);

coefficients = A \ b;

d_0 = coefficients(1);
d_x = coefficients(2);
d_y = coefficients(3);
d_z = coefficients(4);

d = [d_x; d_y; d_z];
%d = simplify(d);
end
```

```
clear all
clc

%%
f2=figure(1);
position0=get(f2,'position');
set(f2,'position',position0+[-0.85*position0(3),-0.3*position0(4),1.7*position0(3),0*position0(4)]);
t=tiledlayout(3,3,”TileSpacing”,”tight”);

f2=figure(1);
position0=get(f2,'position');
set(f2,'position',position0+[-0*position0(3),-0*position0(4),-0.2*position0(3),0.34*position0(4)]);

n=100;
%%
rate_v=500;

v=0.5;
```

```
wj=linspace(0,4,rate_v);
for j = 1 : rate_v
    w=wj(j);
    v1=[v,1]'*ones(1,n);
    v1=v1(:)';
    v1(2*n)=[];
    w0=[0.5*w,0]'*ones(1,n);
    w0=w0(:)';
    w0(2*n)=[];
    w0(2*n-1)=[];
    w0(2*n-2)=[];
    w0(2*n-3)=[];
    w0(2*n-4)=[];
    H=diag(v1,1)+diag(v1,-1)+diag(w0,5)+diag(w0,-5);
    [V,D]=eig(H);
    E(j,:)=diag(D);
    %psi(:,:,j)=V;
end
nexttile(2,[3,1]);
for j = 1 : size(E,2)
    plot(wj,E(:,j),"Color",'black');
    hold on;
end
plot(linspace(1,1,100),linspace(-4,4,100),'LineWidth',1,'linestyle','--','Color'
    ,[241,141,0]/255)
hold on;
plot(linspace(2.6,2.6,100),linspace(-4,4,100),'LineWidth',1,'linestyle','--','
    Color',[241,141,0]/255)
hold on;
set(gca,'LineWidth',1);
xlabel('$w$','interpreter','latex','FontSize',16);
ylabel('$E$','interpreter','latex','FontSize',16);
title(sprintf("(d) N=%d", n),'interpreter','latex','FontSize',12);

%ratio=400;

v=0.5;
w=0;
v1=[v,1]'*ones(1,n);
v1=v1(:)';
v1(2*n)=[];
w0=[0.5*w,0]'*ones(1,n);
w0=w0(:)';
w0(2*n)=[];
w0(2*n-1)=[];
w0(2*n-2)=[];
w0(2*n-3)=[];
w0(2*n-4)=[];
H=diag(v1,1)+diag(v1,-1)+diag(w0,5)+diag(w0,-5);
[V,D]=eig(H);
psi=V;

x=linspace(0.75,10.25,20);
nexttile(1);
%bar(x,reshape(psi(:,10,ratio),[2,10]))
b1=bar(reshape(psi(:,n),[2,n])',0.9,'GroupWidth',0.8);
```

```
set(gca,'ytick',[-0.8,0,0.8])
axis([0.5,n+0.5,-1,1])
title({'(a)␣edge␣states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b1(1),'FaceColor',[207,108,82]/255);
set(b1(2),'FaceColor',[109,152,165]/255);
set(gca,'LineWidth',1);

nexttile(4);
%bar(x,reshape(psi(:,11,ratio),[2,10]))
b2=bar(reshape(psi(:,n+1),[2,n])',0.9,'GroupWidth',0.8);
set(gca,'ytick',[-0.8,0,0.8])
axis([0.5,n+0.5,-1,1])
title({'(b)␣edge␣states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b2(1),'FaceColor',[207,108,82]/255);
set(b2(2),'FaceColor',[109,152,165]/255);
set(gca,'LineWidth',1);
ylabel('$\psi$','interpreter','latex','FontSize',16,'VerticalAlignment','middle');

nexttile(7);
%bar(x,reshape(psi(:,8,ratio),[2,10]))
b3=bar(reshape(psi(:,n-2),[2,n])',0.9,'GroupWidth',0.8);
if n >= 100
    set(gca,'ytick',[-0.1,0,0.1])
    axis([0.5,n+0.5,-0.15,0.15])
else
    set(gca,'ytick',[-0.4,0,0.4])
    axis([0.5,n+0.5,-0.5,0.5])
end
title({'(c)␣bulk␣states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b3(1),'FaceColor',[207,108,82]/255);
set(b3(2),'FaceColor',[109,152,165]/255);
set(gca,'LineWidth',1);

xlabel('Cell␣Index␣N','interpreter','latex','FontSize',16);

%%

v=0.5;
w=4;
v1=[v,1]'*ones(1,n);
v1=v1(:)';
v1(2*n)=[];
w0=[0.5*w,0]'*ones(1,n);
w0=w0(:)';
w0(2*n)=[];
w0(2*n-1)=[];
w0(2*n-2)=[];
w0(2*n-3)=[];
w0(2*n-4)=[];
H=diag(v1,1)+diag(v1,-1)+diag(w0,5)+diag(w0,-5);
[V,D]=eig(H);
psi=V;
```

```

x=linspace(0.75,10.25,20);
nexttile(3);
%bar(x,reshape(psi(:,10,ratio),[2,10]))
b1=bar(reshape(psi(:,n),[2,n])',0.9,'GroupWidth',0.8);
set(gca,'ytick',[-0.8,0,0.8])
axis([0.5,n+0.5,-1,1])
title({'(e) edge states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b1(1),'FaceColor',[207,108,82]/255);
set(b1(2),'FaceColor',[109,152,165]/255);
set(gca,'LineWidth',1);

nexttile(6);
%bar(x,reshape(psi(:,11,ratio),[2,10]))
b2=bar(reshape(psi(:,n+1),[2,n])',0.9,'GroupWidth',0.8);
set(gca,'ytick',[-0.8,0,0.8])
axis([0.5,n+0.5,-1,1])
title({'(f) edge states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b2(1),'FaceColor',[207,108,82]/255);
set(b2(2),'FaceColor',[109,152,165]/255);
set(gca,'LineWidth',1);
ylabel('$\psi$','interpreter','latex','FontSize',16,'VerticalAlignment','middle');

nexttile(9);
%bar(x,reshape(psi(:,8,ratio),[2,10]))
b3=bar(reshape(psi(:,n-2),[2,n])',0.9,'GroupWidth',0.8);
if n >= 100
    set(gca,'ytick',[-0.1,0,0.1])
    axis([0.5,n+0.5,-0.15,0.15])
else
    set(gca,'ytick',[-0.4,0,0.4])
    axis([0.5,n+0.5,-0.5,0.5])
end
title({'(g) bulk states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b3(1),'FaceColor',[207,108,82]/255);
set(b3(2),'FaceColor',[109,152,165]/255);
set(gca,'LineWidth',1);

xlabel('Cell Index N','interpreter','latex','FontSize',16);
```

CODE1

```
clear all
clc
f1=figure(1);
position0=get(f1,'position');
set(f1,'position',position0+[-0.75*position0(3),0,1.5*position0(3),-0.27*position0(4)]);
t=tiledlayout("horizontal","TileSpacing","tight");
```

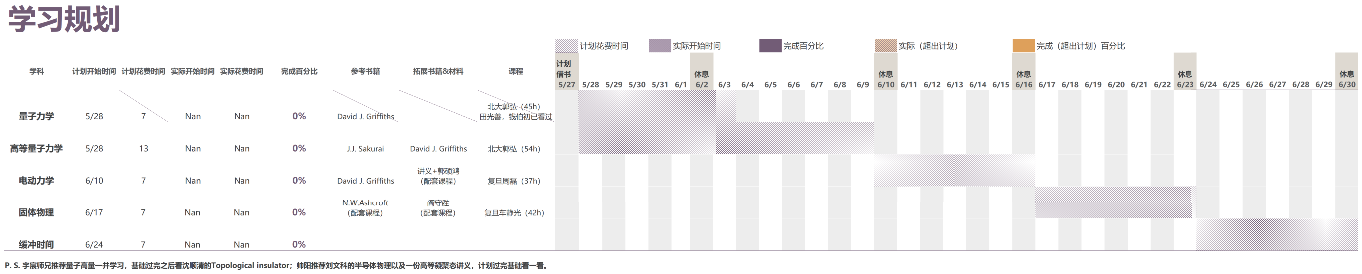
```
titlej={'$(a)\ \omega=0$';'$(b)\ \omega\textless\nu$';'$(c)\ \omega=\nu$';'$(d)\ \omega\textgreater\nu$';'$(e)\ \nu=0$'};

rate_k=1000;

vj=[1,1,1,0.6,0];
wj=[0,0.6,1,1,1];
ki=linspace(-pi,pi,rate_k);

for j = 1 : 5
    v=vj(j);
    w=wj(j);
    for i = 1 : rate_k
        k=ki(i);
        E_1(i)=sqrt(v+w+2*v*w*cos(k));
        E_2(i)=-sqrt(v+w+2*v*w*cos(k));
    end
    nexttile
    plot(ki/pi,linspace(0,0,rate_k),'LineWidth',1,'linestyle','--','Color',[241,141,0]/255)
    hold on
    plot(ki/pi,E_1,'LineWidth',1.5,'Color',[207,108,82]/255);
    hold on
    plot(ki/pi,E_2,'LineWidth',1.5,'Color',[109,152,165]/255);
    hold on
    set(gca,'LineWidth',1);
    axis([-1,1,-2,2])
    title(titlej{j},'interpreter','latex','FontSize',12);
end
title(t,'Energy Band of SSH Model','FontSize',16);
xlabel(t,'$k/\pi$','interpreter','latex','FontSize',16);
ylabel(t,'$E(k)$','interpreter','latex','FontSize',16);
```

```
clear all
clc
f1=figure(1);
position0=get(f1,'position');
set(f1,'position',position0+[-0.75*position0(3),0,1.5*position0(3),-0.27*position0(4)]);
t=tiledlayout("horizontal","TileSpacing","tight");
titlej={'$(a)\ \omega=0$';'$(b)\ \omega\textless\nu$';'$(c)\ \omega=\nu$';'$(d)\ \omega\textgreater\nu$';'$(e)\ \nu=0$'};

rate_k=1000;

vj=[1,1,1,0.6,0];
wj=[0,0.6,1,1,1];
ki=linspace(-pi,pi,rate_k);

for j = 1 : 5
    v=vj(j);
    w=wj(j);
    for i = 1 : rate_k
        k=ki(i);
        d_x(i)=v+w*cos(k);
```

```
            d_y(i)=w*sin(k);
        end
        nexttile
        plot(linspace(-2,2,100),linspace(0,0,100),'LineWidth',1,'linestyle','--','Color','k')
        hold on
        plot(linspace(0,0,100),linspace(-2,2,100),'LineWidth',1,'linestyle','--','Color','k')
        hold on
        if j == 1
            plot(d_x,d_y,'ro','MarkerFaceColor',[207,108,82]/255,'MarkerSize',9);
        else
            plot(d_x,d_y,'LineWidth',1.5,'Color',[207,108,82]/255);
        end
        hold on
        set(gca,'LineWidth',1);
        axis([-2,2,-2,2])
        title(titlej{j},'interpreter','latex','FontSize',12);
    end
    title(t,'Winding␣Number␣of␣SSH␣Model','FontSize',16);
    xlabel(t,'$d_{y}$','interpreter','latex','FontSize',16);
    ylabel(t,'$d_{x}$','interpreter','latex','FontSize',16);
```

```
clear all
clc

rate_v=30000;
n=100;

w=1;
vj=linspace(0,3,rate_v);

for j = 1 : rate_v
    v=vj(j);
    vw=[v,w]'*ones(1,n);
    vw=vw(:)';
    vw(2*n)=[];
    H=diag(vw,1)+diag(vw,-1);
    [V,D]=eig(H);
    E(j,:)=diag(D);
    %psi(:,:,j)=V;
end
for j = 1 : size(E,2)
        plot(vj,E(:,j),"Color",'black');
        hold on;
end
plot(linspace(1,1,100),linspace(-4,4,100),'LineWidth',1,'linestyle','--','Color',[241,141,0]/255)
set(gca,'LineWidth',1);
xlabel('$\nu/\omega$','interpreter','latex','FontSize',16);
ylabel('$E$','interpreter','latex','FontSize',16);
title(sprintf("N=%d", n),'interpreter','latex','FontSize',12);

%%
f2=figure(2);
```

```
position0=get(f2,'position');
set(f2,'position',position0+[-0.4*position0(3),-0.3*position0(4),0.2*position0(3),0.3*position0(4)]);
t=tiledlayout("vertical","TileSpacing","tight");

%ratio=400;

v=0.4;
w=1;
vw=[v,w]'*ones(1,n);
vw=vw(:)';
vw(2*n)=[];
H=diag(vw,1)+diag(vw,-1);
[V,D]=eig(H);
psi=V;

x=linspace(0.75,10.25,20);
nexttile
%bar(x,reshape(psi(:,10,ratio),[2,10]))
b1=bar(reshape(psi(:,n),[2,n])',0.9,'GroupWidth',0.8);
set(gca,'ytick',[-0.8,0,0.8])
axis([0.5,n+0.5,-1,1])
title({'(a) edge states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b1(1),'FaceColor',[207,108,82]/255);
set(b1(2),'FaceColor',[109,152,165]/255);

nexttile
%bar(x,reshape(psi(:,11,ratio),[2,10]))
b2=bar(reshape(psi(:,n+1),[2,n])',0.9,'GroupWidth',0.8);
set(gca,'ytick',[-0.8,0,0.8])
axis([0.5,n+0.5,-1,1])
title({'(b) edge states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b2(1),'FaceColor',[207,108,82]/255);
set(b2(2),'FaceColor',[109,152,165]/255);

nexttile
%bar(x,reshape(psi(:,8,ratio),[2,10]))
b3=bar(reshape(psi(:,n-2),[2,n])',0.9,'GroupWidth',0.8);
if n >= 100
    set(gca,'ytick',[-0.1,0,0.1])
    axis([0.5,n+0.5,-0.12,0.12])
else
    set(gca,'ytick',[-0.4,0,0.4])
axis([0.5,n+0.5,-0.5,0.5])
end
title({'(c) bulk states'},'VerticalAlignment','bottom','interpreter','latex','FontSize',12);
set(b3(1),'FaceColor',[207,108,82]/255);
set(b3(2),'FaceColor',[109,152,165]/255);

ylabel(t,'$\psi$','interpreter','latex','FontSize',16);
xlabel(t,'Cell Index N','interpreter','latex','FontSize',16);
```



学习规划

学科	计划开始时间	计划花费时间	实际开始时间	实际花费时间	完成百分比	参考书籍	拓展书籍&材料	课程
量子力学	5/28	7	Nan	Nan	0%	David J. Griffiths		北大郭弘（45h） 田光善，钱伯初已看过
高等量子力学	5/28	13	Nan	Nan	0%	J.J. Sakurai	David J. Griffiths	北大郭弘（54h）
电动力学	6/10	7	Nan	Nan	0%	David J. Griffiths	讲义+郭硕鸿 （配套课程）	复旦周磊（37h）
固体物理	6/17	7	Nan	Nan	0%	N.W.Ashcroft （配套课程）	阎守胜 （配套课程）	复旦车静光（42h）
缓冲时间	6/24	7	Nan	Nan	0%			

P. S. 宇宸师兄推荐量子高量一并学习，基础过完之后看沈顺清的Topological insulator；帅阳推荐刘文科的半导体物理以及一份高等凝聚态讲义，计划过完基础看一看。

FIG. 11. 基础知识学习计划