

磁性超表面的单向光传播

路尚润¹⁾

1) (中山大学物理学院, 广州, 中国 510275)

本研究旨在构建双向相位调控的非互易超表面。基于实验室已有材料钇铁石榴石 (YIG) 和永磁体, 利用磁圆二向色性等非互易效应, 实现电磁波的单向传播和相位调控。我们分析了非互易超原子设计中的关键因素, 尤其是相位与超原子参数之间的关系, 并探索了通过表面阻抗控制电磁波传播的机制。研究结果为使用全被动和线性方法实现任意极化的非互易超表面提供了新的思路, 且对进一步开发高效的非互易器件具有指导意义。

关键词: 非互易超表面, 磁性材料, 单向传播

1 引言

超材料由亚波长金属/介电结构 (即超原子) 组成, 可通过与入射电磁场的电或磁分量, 或两者的共振耦合, 表现出自然界中未见的有效电响应 (以介电常数 ϵ 表示) 或磁响应 (以磁导率 μ 表示)^[1]。然而, 三维超材料的制造面临巨大挑战^[2], 相比之下, 二维超材料 (即超表面) 能够通过光刻和纳米印刷等技术更为简便地制造, 这使得它们在光学器件中的应用更加实际可行。

在超表面中, 由于其亚波长厚度, 传统的电磁参数 (如 ϵ 与 μ) 的作用相对削弱, 设计更多依赖于表面阻抗来控制电磁波的传播, 而非体积内的电磁响应^[3]。当前, 超表面的调控方式较为成熟, 包括几何相位^[4]、通过改变纳米结构的尺寸^[5] 以及基于材料特性的调控^[6]。这些方法具有良好的可调节性, 但它们都遵循洛伦兹互易性原则。

近年来, 非互易超表面 (non-reciprocal metasurfaces, NRM) 逐渐受到关注。与传统超表面不同, NRM 能够执行与波传播方向相关的功能。例如, 集成 NRM 的天线罩可以显著减少天线的回波反射。理想情况下, NRM 应能够生成任意相位与幅度的组合, 以便在两个传输方向上按需调制波前。当前已提出多种实现非互易性的方法, 包括利用磁性材料的非对称介电常数或磁导率张量^[7], 通过非对称结构的非线性效应^[8], 或利用整流效应实现单向增益^[9]。其中, 基于有源器件的方法通常需要额外的电源, 设计复杂, 而

非线性材料的实现则受信号强度影响。因此，采用全被动且线性的方式实现任意极化的非互易超表面，具有重要的应用前景。

本文基于钇铁石榴石（YIG）和永磁体材料，探索了磁圆二向色性等非互易效应，实现双向相位调控的非互易超表面。通过构建单向传播的超表面，进一步研究了相位调控与超原子参数的关系，并为构建任意非互易器件提供了指导性建议。

2 装置结构及工作原理

2.1 超表面简介

超表面是一种由亚波长尺度结构单元（通常称为超原子）组成的二维材料，它们可以对电磁波进行精确调控。相比传统三维超材料，超表面由于其低厚度和易制造的特性，在光学、微波和声学等领域得到了广泛的应用。通过设计超原子的几何结构、材料特性或排列方式，超表面能够实现对电磁波的相位、振幅、偏振和波前的调控。根据其功能特性，超表面可分为互易性和非互易性两大类。

同时，Janus 超表面和非互易超表面虽然都涉及方向性的波传播调控，但 Janus 超表面更多依赖于结构的不对称性，而非互易超表面则依赖于打破洛伦兹互易性来实现双向不同的电磁响应。

2.1.1 非互易超表面

非互易超表面是一种打破洛伦兹互易性的超表面，能够在不同方向上执行不同的电磁功能。传统的互易性材料和器件在不同传播方向上具有相同的响应，而非互易超表面则通过引入非对称介电常数或磁导率张量、非线性效应或进行增益，来实现传播方向相关的调控^[10]。例如，基于磁性调控的非互易超表面示意图如图1(a)所示。

2.1.2 Janus 超表面

Janus 超表面是一种具有双向响应的超表面，于 2019 年由我国冯一军教授团队率先提出^[11]，其设计的核心在于两个方向上表现出不同的电磁响应。其团队所设计的超构表面能实现具有方向选择性的双全息成像，当入射电磁波沿前向传播时，该超构表面能够在自由空间中呈现出古罗马 Janus 神（双面神）的右半边像，而沿后向传播时能呈现出 Janus 神的左半边像，因此得名。其示意图如图1(b)所示。

Janus 超表面通过在结构上引入反射对称性破坏，使其在不同方向上可以调控不同的波前、相位或振幅。这种双向不对称功能使得 Janus 超表面在双向全息、光学加密和波前调控等领域有着广泛的应用。

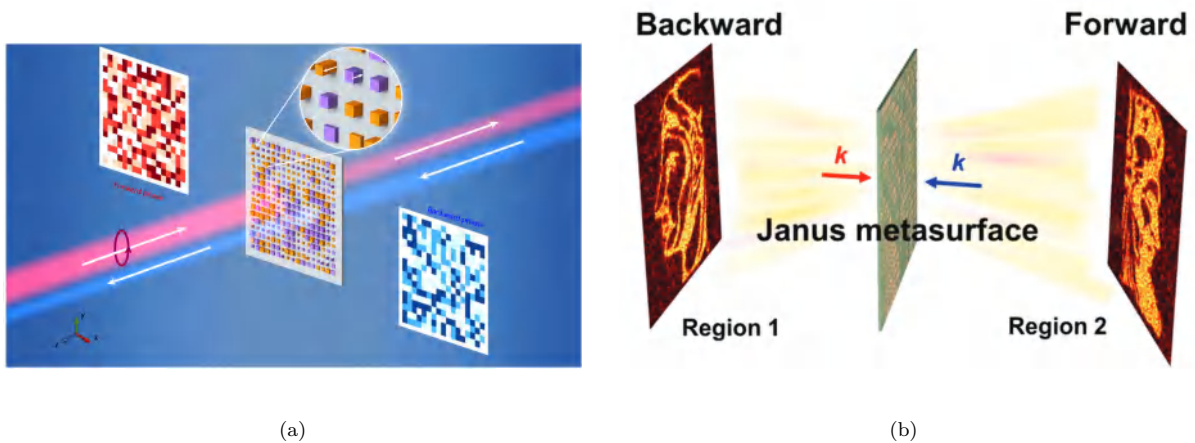


图 1 (a) 基于磁性调控的非互易超表面示意图 (b) Janus 超表面示意图

Fig. 1. Schematic diagram of (a) a NRM based on magnetic control (b) a Janus metasurface

2.2 超原子设计

2.2.1 材料介绍

在本次实验中使用到的材料共有两种，第一种为 YIG，第二种为固定超原子的泡沫板。

钇铁石榴石 (Yttrium Iron Garnet, YIG) 是一种具有独特磁性和光学特性的铁磁材料，化学式为 $\text{Y}_3\text{Fe}_5\text{O}_{11}$ ，晶体结构如图2所示^[12]。

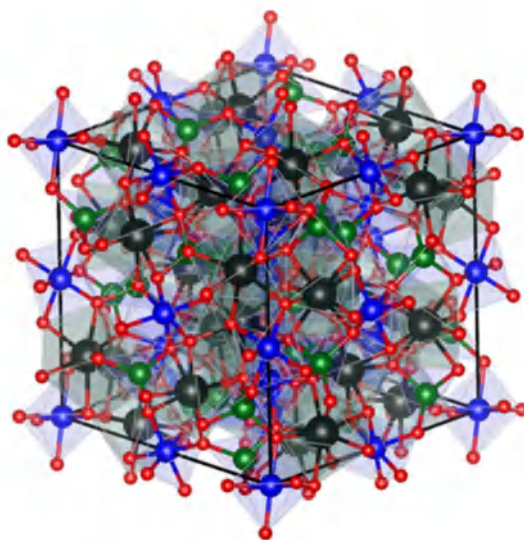


图 2 YIG 晶体结构

Fig. 2. YIG crystal structure

YIG 以其低损耗、高磁导率和优异的磁光效应而著称，广泛应用于微波、光通信和磁光器件领域。在外加磁场的作用下，YIG 展现出显著的磁圆二向色性 (MCD)，可以实现对电磁波的非互易调控，成为非

互易超表面研究中的理想材料。由于其低阻抗和宽频带特性，YIG 在非线性和磁光应用中也表现出优越的性能，尤其适用于高效相位调控和信号隔离等功能。

在 COMSOL 软件中，设置 YIG 的磁导率张量为：

$$\mu = \begin{pmatrix} \mu_1 & i\kappa & 0 \\ -i\kappa & \mu_1 & 0 \\ 0 & 0 & \mu_2 \end{pmatrix} \quad (1)$$

其中 $\kappa = \frac{\omega_m \omega}{(\omega_0 + i\omega\alpha)^2 - \omega^2}$ ， $\mu_1 = 1 + \frac{\omega_m(\omega_0 + i\omega\alpha)}{(\omega_0 + i\omega\alpha)^2 - \omega^2}$ ， $\mu_2 = 1$ ， $\omega_m = \gamma 4\pi M_s$ ， $\omega_0 = \gamma H_0$ ， $\alpha = 0.0088$ 为阻尼系数， ω 为工作频率， $\gamma = 2.8 \text{ MHz/Oe}$ 是旋磁比， $4\pi M_s = 1950 \text{ Oe}$ 是饱和磁矩，以及 $H_0 = 430 \text{ Oe}$ 是实验中在 z 方向上的等效静磁场。

对于泡沫板，其相对介电常数为 1.04，相对磁导率为 1，与空气参数基本相当，使用目的是为了很好地固定 YIG 及永磁体。

除了以上两种材料，实验中还使用了外加永磁体，可以认为其是电磁波的理想导体，电磁波不会穿过，因此其参数对结果并无影响。

2.2.2 结构设计

为了在 YIG 内部产生恒定磁场，我们在 YIG 的两侧表面分别放置永磁体，形成 ABA 型排列结构。如果采用多层结构，则可以按照 ABABA……的方式排列。此外，为了确保超原子按照预定设计精确排列，我们将其插入定制的泡沫板中，因此泡沫板也可以视为超原子结构的一部分。单层 YIG 超原子的结构示意图如图3所示。

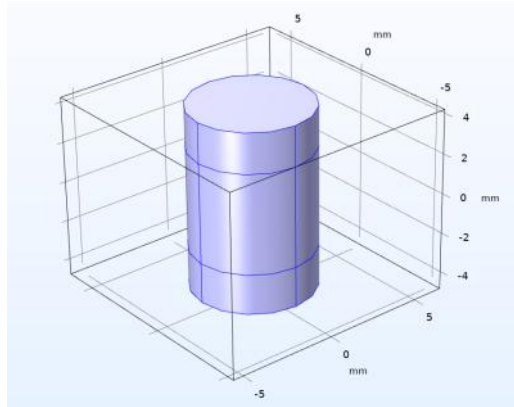


图 3 单层 YIG 超原子结构示意图

Fig. 3. Single-layer YIG superatomic structure schematic diagram

图中上下两层为永磁体，用于产生 YIG 内部的静磁场；永磁体中间即为 YIG 材料，与电磁波发生响应；两者结合插入到预留的泡沫板孔位中，实现超原子的精确定位。

2.3 工作原理

在非互易超表面设计中，磁效应是实现非互易传输的重要机制。磁效应打破了波在传播路径上的对称性，使电磁波在不同传播方向上具有不同的相位或偏振响应。我们主要介绍三种与非互易相位调控密切相关的磁效应：法拉第旋磁效应、磁圆二向色性和磁双折射。

对于线偏振，我们一般不会衡量其相位，反而更多的是衡量电矢量在垂直于波矢面的旋转角度。因此，对于单向传播来讲，使用线偏振并无异议。但对于相位调控来说，为了使相位有意义，此时应当使用圆偏振。

无论对于线偏振光还是圆偏振光，其相关磁效应均不满足洛伦兹互易性，因此可以很好地构建所需的非互易超表面。

2.3.1 法拉第旋磁效应

法拉第旋磁效应是一种典型的非互易效应，通常发生在电磁波通过磁性介质时。当电磁波通过具有外加磁场的介质时，线偏振光的电矢量会围绕传播方向发生旋转，这个旋转角度与磁场的方向和强度成正比，如图4(a)所示。值得注意的是，这种旋转具有单向性：当波在反向传播时，旋转角度不会反转，这正是法拉第旋磁效应的非互易特性。

在超表面设计中，法拉第效应通常用于调控电矢量的旋转角度。例如，在磁性材料（如钇铁石榴石 YIG）中，通过外加磁场产生法拉第旋转，这种效应适用于基于线偏振光的非互易器件。

2.3.2 磁圆二向色性

磁圆二向色性（Magnetic Circular Dichroism, MCD）是指磁性介质对左旋圆偏振光（LCP）和右旋圆偏振光（RCP）具有不同的吸收率，如图4(b)所示。具体来说，LCP 和 RCP 在通过磁性材料时，经历不同的吸收损耗，从而导致它们的透射特性差异。这种现象也依赖于外加磁场的存在，并且具有显著的方向性差异，因而是典型的非互易效应之一。

在非互易超表面中，磁圆二向色性效应可以用于区分两种圆偏振光的透射率，并实现方向性相关的调控。这种效应常用于构建单向传播器件，使得特定方向的电磁波得到增强或抑制。

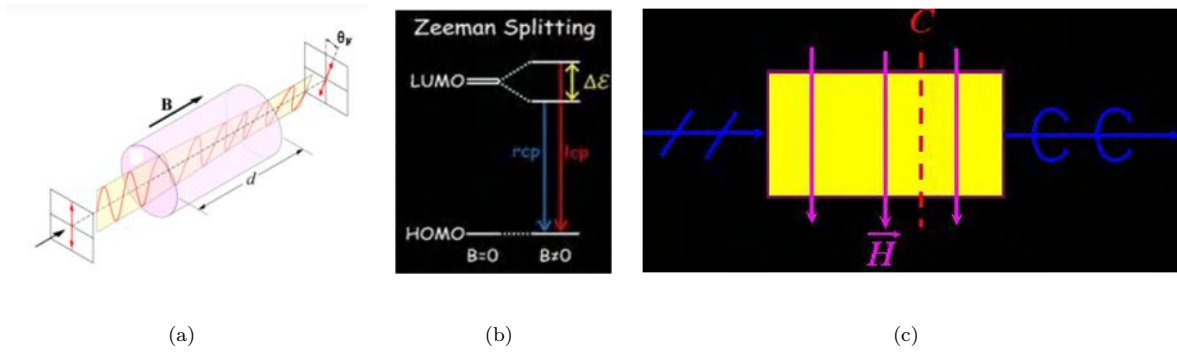


图 4 (a) 法拉第旋磁效应 (b) 磁圆二向色性 (c) 磁双折射

Fig. 4. (a) Faraday's Magnetic Effect (b) Magnetic Circular Dichroism (c) Magnetic Birefringence

2.3.3 磁双折射

磁双折射是指电磁波在磁性介质中传播时，由于磁场的作用，不同偏振态的光会经历不同的折射率，导致相位差的产生。具体而言，左旋和右旋圆偏振光在磁场作用下的折射率不同，从而在传播过程中累积了不同的相位，如图4(c)所示。磁双折射效应不仅影响光的传播速度，还能够调控电磁波的偏振态，尤其在相位调控应用中起到关键作用。

通过在超表面结构中引入磁双折射效应，可以实现圆偏振光的非互易相位调控。结合钇铁石榴石等磁性材料，设计特定的纳米结构，能够精确控制波在正反两个方向上的相位差异，使得非互易超表面能够在双向传播中实现不同的相位调制效果。

3 实验装置

本次实验主要分为两部分：仿真探索和实验验证。仿真所使用的软件为 COMSOL 电磁场求解，COMSOL 版本为 6.0。实验所使用到的仪器为：微波网络分析仪 VNA、电磁远场自动测量系统和透镜天线。

4 实验内容及结果

4.1 单向传输非互易超表面

4.1.1 单层超原子

首先使用单层超原子构造非互易超表面，模型参数为，超原子半径为 3mm，YIG 高度为 5mm，永磁体高度为 2mm，超原子间距 12mm。

首先利用 COMSOL 搭建仿真模型，对正反向传播进行频率扫描，扫描范围为 6GHz-15GHz，使用右旋偏振光，绘制出透射谱如图5(a)和图5(b)所示。

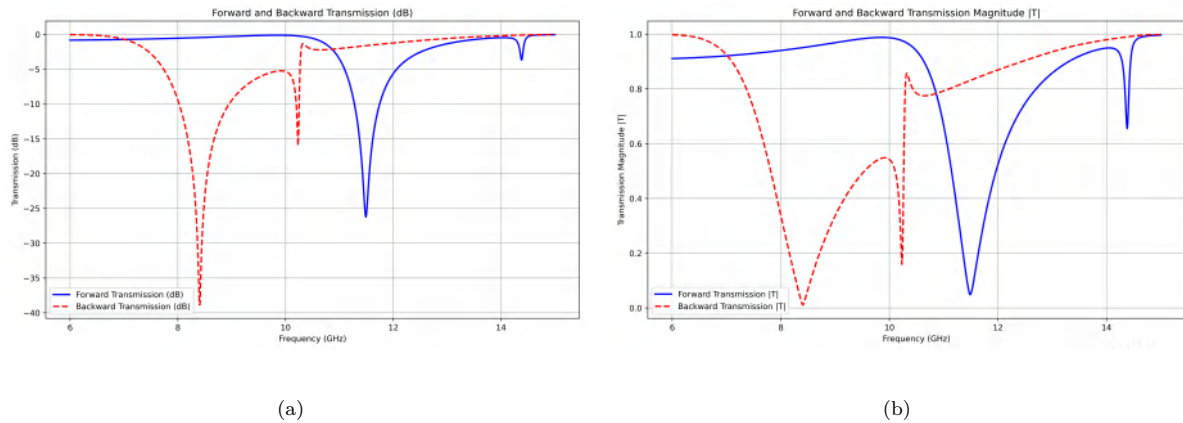


图 5 单层透射谱 (a)dB 单位 (b)% 单位

Fig. 5. Single-layer transmission spectrum (a) in dB units (b) in percentage units

可以看出，正向和反向传播均出现了低谷，并且低谷位置不同，在对应的低谷处即可实现单向传输的非互易超表面。

为衡量单向传输的性能，我们用两个方向的透射率做比值，并称为隔离度，其结果如图6(a)所示，以 dB 为单位。同时我们也绘制了正反向透射率的差值，一定程度上也可以看出很好的非互易性，如图6(b)所示。

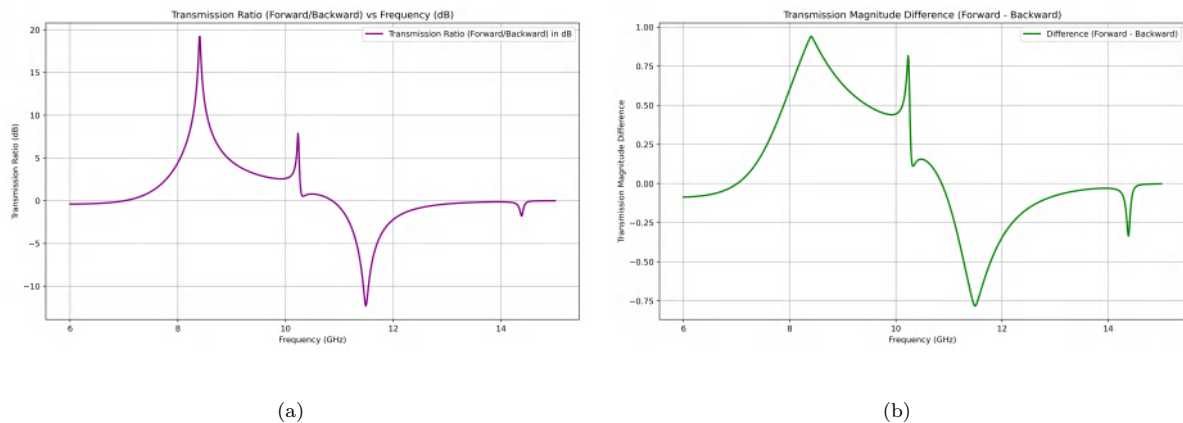


图 6 单层频率响应 (a) 隔离度 (b) 透射率之差

Fig. 6. Single-layer frequency response (a) Isolation (b) Difference in transmission rate

同时，利用代码寻找其全局最大值和最小值，并记录此时的工作频率，实现双向的单向电磁波传输。最终结果如表1所示。

表 1 单层隔离度极值

Table 1. Single-layer Isolation Extreme Value

极值点	频率/GHz	隔离度/dB
极大值	8.40	19.2110
极小值	11.49	-12.3137

可以看出，其隔离度数值均很高，说明此时单向传播效果很好。在极大值处可以实现右旋光的正向单向传输，而在极小值处可以实现右旋光的反向单向传输。同时，正反向的工作频率相差极大，这保证了其工作性能的稳定性，理论上可以很好地实现电磁波的单向传输。

之所以能够实现较好的结果，是因为此时其内部磁场发生了共振响应，将大部分电磁波进行了反射，此时内部磁场的 x 分量如图7(a)和图7(b)所示。

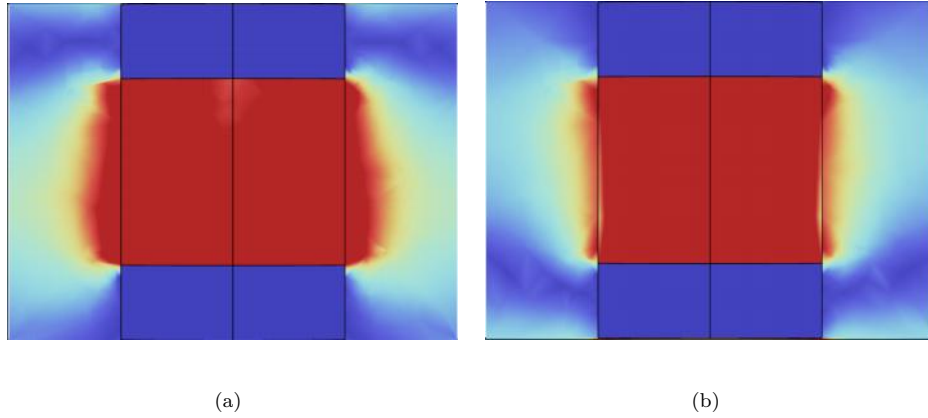


图 7 H_x (a) 反向 8.40GHz(b) 正向 11.94GHz

Fig. 7. H_x (a) Reverse 8.40 GHz (b) Forward 11.94 GHz

根据结果可以看出，在正向和反向透射谱的极小值点处，均发生了磁场共振，因此此时磁效应最强，产生了强烈的非互易结果。

4.1.2 三层超原子

对于单层结果，可以看出其已经呈现出很好的单向传播特性，但美中不足的是其带宽较窄，在实际使用中如果频带发生偏移，透射率会发生较大的改变，单向特性降低。

为解决此问题，我们尝试构建三层超原子，即采用 ABABABA 型排列，期望能看到更强烈的响应区间，其结构如图8所示。

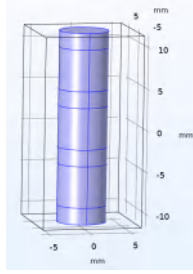


图 8 三层 YIG 超原子结构示意图

Fig. 8. Schematic diagram of a three-layer YIG superatomic structure

与单层超原子进行相同的求解，首先绘制出透射谱如图9(a)和图9(b)所示。

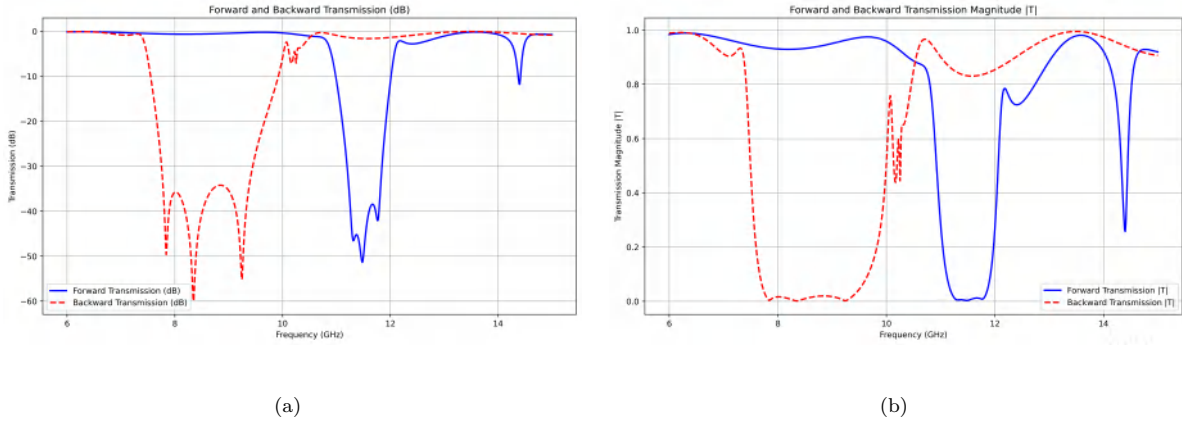


图 9 三层透射谱 (a)dB 单位 (b)% 单位

Fig. 9. Triple-layer transmission spectrum (a) in dB units (b) in percentage units

可以看出，正向和反向传播均出现了低谷，并且低谷位置不同，并且对比单层带宽较大，有三个分别对应的局部极小值，可能是不同程度共振产生的。隔离度果如图10(a)所示，透射率的差值如图10(b)所示。

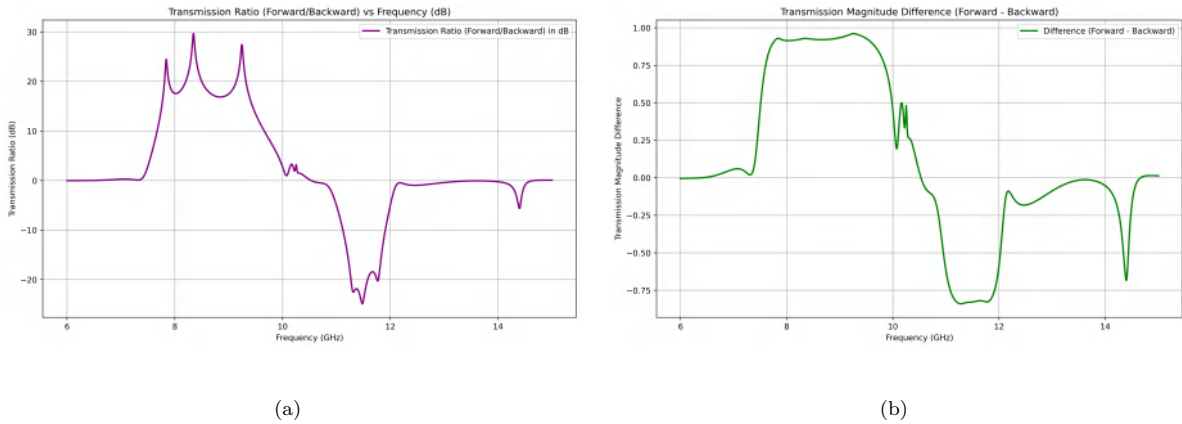


图 10 三层频率响应 (a) 隔离度 (b) 透射率之差

Fig. 10. Triple-layer frequency response (a) Isolation (b) Difference in transmission rate

同时，寻找其全局最大值和最小值，最终结果如表2所示。

表 2 三层隔离度极值

Table 2. Triple-layer isolation extremum

极值点	频率/GHz	隔离度/dB
极大值	8.35	29.6623
极小值	11.48	-24.8946

可以看出，其隔离度数值很高。同时，其隔离度均比单层高出 5dB 左右，说明三层的效果更加优异。

4.2 相位调控

单向传输的非互易超表面只是最基础的非互易器件之一，想要构造更复杂的 NRM 一般都基于相位调控，利用不同尺寸的超原子，实现超表面上点对点的精确相位调控，可以构造非互易光束偏折、非互易超透镜以及非互易全息成像等更加复杂以及应用广泛的非互易超表面。

4.2.1 频率响应

相位调控的自变量应为超原子尺寸，因此工作频率的确定十分重要。我们倾向于正向和反向传播时有较大的透射率，并有不同的透射相位。

采用 150mm 的仿真实空间，首先得到单层的相位频率响应，以及正向和反向的相位差值，分别如图11(a)和11(b)所示。

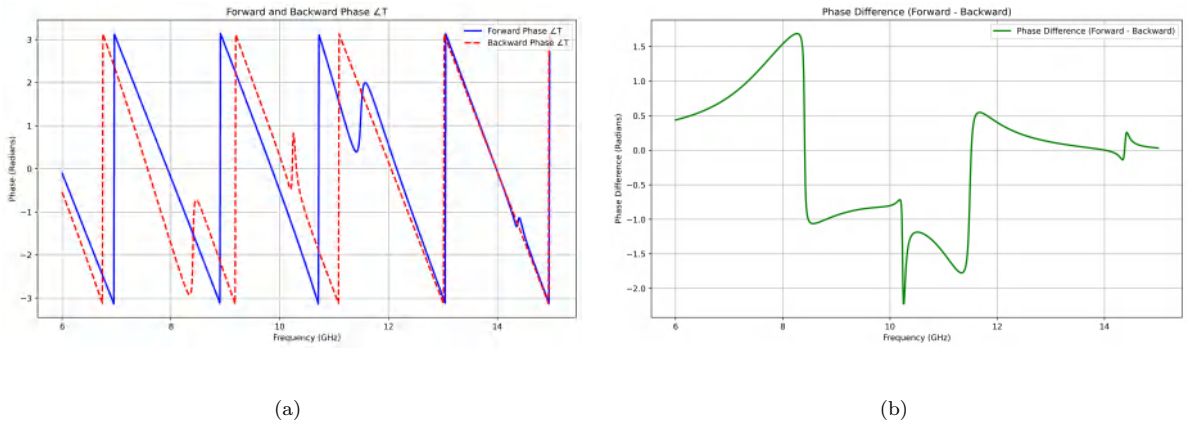


图 11 单层频率响应 (a) 相位 (b) 相位差

Fig. 11. Single-layer frequency response (a) Phase (b) Phase difference

观察结果表明，超表面在磁场作用下的正反向传播相位具有显著的非对称性。特别是在磁场共振点，即透射率较低的频段，正反向传播的相位差值较大，说明此处的非互易相位响应较强。然而，进一步分析发现，相位差的分布范围较为有限，无法完全覆盖 $(-\pi, \pi)$ 的完整区间。

根据先前的研究结果，三层结构的超原子具有比单层结构更强的响应能力。因此，我们拟对三层超原子展开频率响应的详细研究，以期获得更广的相位差范围，以支持实现更具设计灵活性的非互易超表面。

同样在仿真空间长度为 150mm 情况下，得到三层的相位频率响应，以及正反向相位差，分别如图12(a)和12(b)所示。

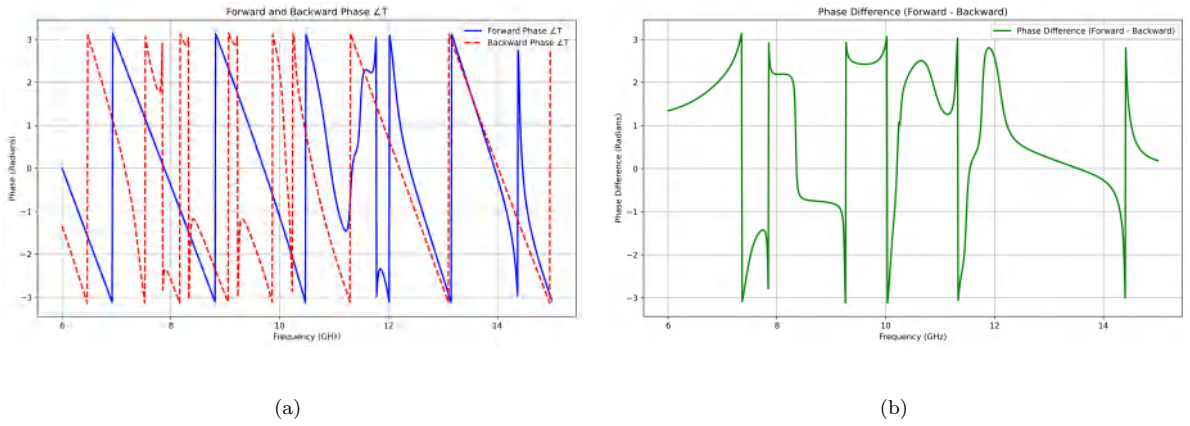


图 12 三层频率响应 (a) 相位 (b) 相位差

Fig. 12. Triple-layer frequency response (a) Phase (b) Phase difference

三层结构的相位响应表现出极高的动态性，正反向传播的相位与相位差在频率变化过程中剧烈波动，能够完整覆盖 $(-\pi, \pi)$ 的相位区间。这种显著的相位响应拓宽了设计的可能性，使三层结构在实现非互易性方面具有更大的潜力。

结合单层和三层结构在频率响应及透射谱方面的表现，我们注意到在 10.5GHz 频段正反向透射率均较高，且此频率位于正反向共振响应的位置之间。这意味着在参数调控下，理论上可以实现丰富的相位差响应。相较而言，三层结构的相位响应在频率上更加剧烈、覆盖范围也更宽，因而为进一步优化设计，拟采用三层结构作为研究重点。

在此基础上，我们将深入研究超原子的几何尺寸如何调控其相位响应特性。通过精细调控尺寸参数，有望获得对非互易性相位调控的更多自由度，为构造更复杂的非互易超表面奠定基础。

4.2.2 参数调控

在固定频率 10.5GHz 下, 研究三层超原子的 YIG 高以及半径对传播频率的影响。以实验参数 $r=3\text{mm}$, $h=5\text{mm}$ 为基准, 上下浮动 2mm 进行参数扫描, 并绘制正反向透射谱、相位谱以及两者的差值, 结果如图13所示。

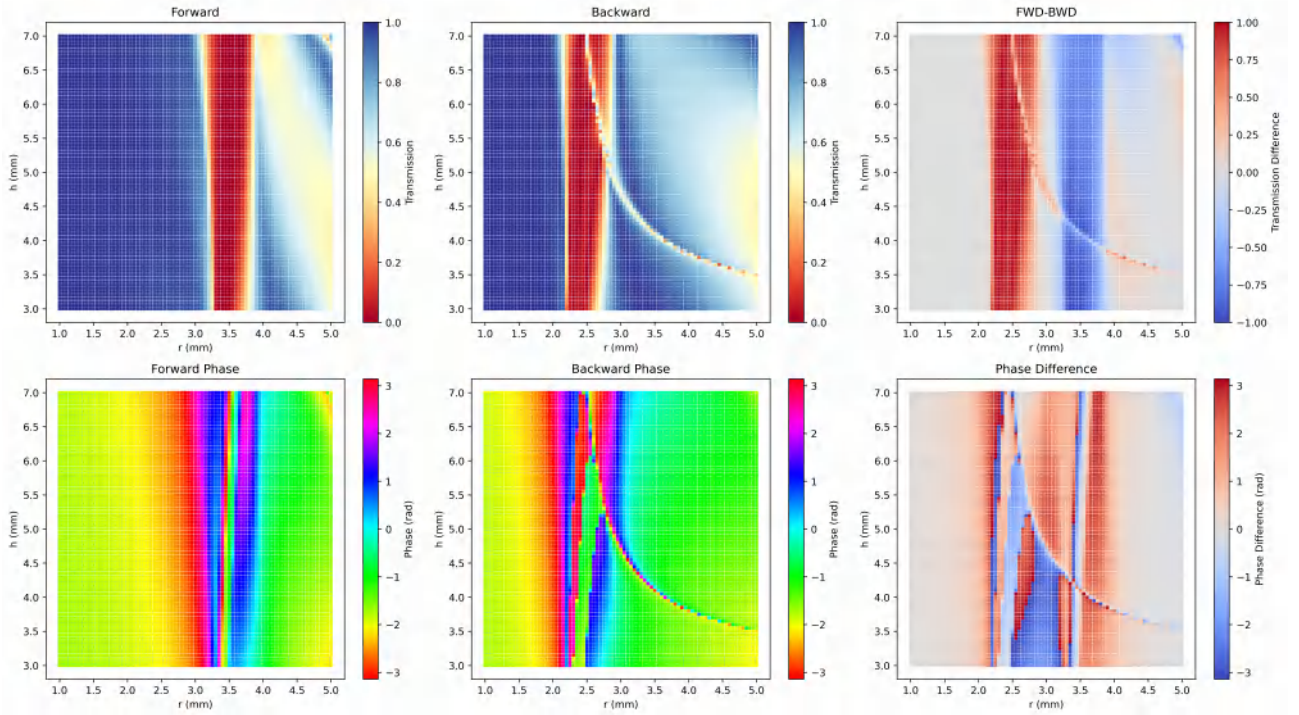


图 13 三层超原子参数响应结果

Fig. 13. Results of the response of the three-layer superatomic parameters

透射谱和相位谱的响应非常显著, 且相位响应通常出现在透射率较低的区域。这一现象的原因在于, 此时磁场发生共振, 磁效应变得更加明显。通过对比结果, 我们观察到反向透射谱的响应区间半径较小。同时, 相较于正向谱线, 其多了一个较细的响应区间, 我们将其称为“局部响应”, 而正向谱线则仅在图像的右上角有初步的显现。

然而, 由于局部响应区间的带宽较窄, 我们无法确定其是否适合用于超表面的构建。因为在这一狭窄范围内, 微小的参数变化可能会导致结果的显著改变, 这使得设计的稳定性和可预测性受到挑战。

针对正反向传播相位的差值, 虽然我们获得了丰富的响应结果, 但这种信息并不直观, 难以直观地感知相位的分布情况。因此, 我们绘制了每个正反向相位所对应的单元个数, 如图14(a)所示。此外, 我们还制作了关于相位差的统计直方图, 如图14(b)所示, 这有助于更清晰地展示相位差分布的特征及其在超表面设计中的应用潜力。通过这些可视化结果, 我们希望能更深入地理解相位的分布情况, 从而为超表面的

优化设计提供有价值的参考。

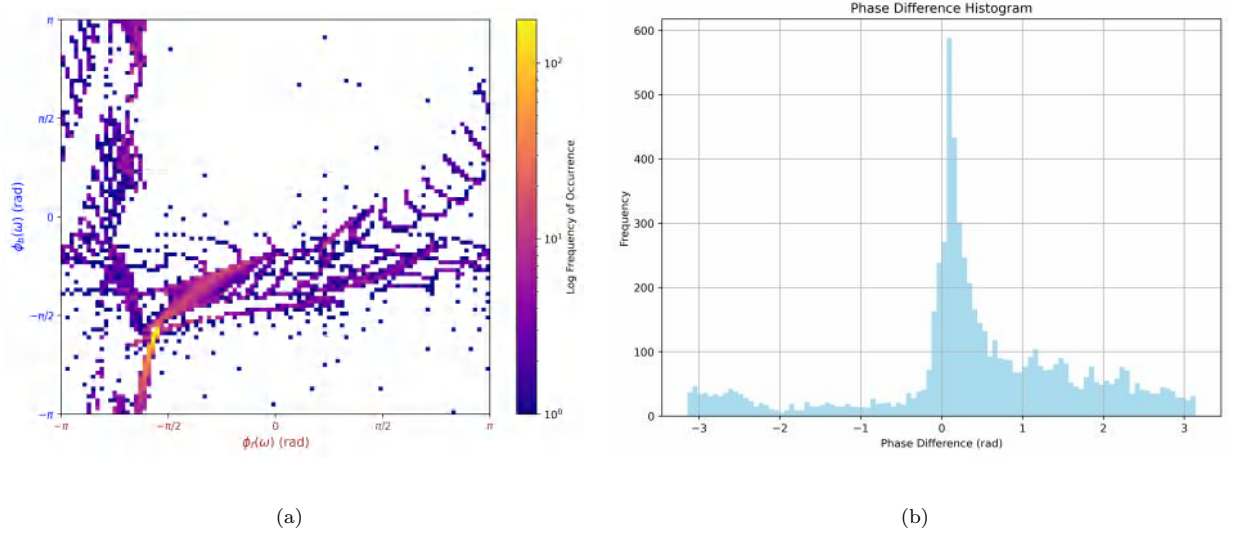


图 14 (a) 单元相位分布 (b) 相位差直方图

Fig. 14. (a) Unit phase distribution (b) Phase difference histogram

首先，对于二维的相位分布图，大部分数据点集中在相位差接近于零的区域，也有一些区域显示出明显的非零相位差，这是由于超表面的非互易特性导致的。

第二张图是相位差直方图，显示了相位差的分布情况，其中相位差为零的区域频率最高，表明大多数情况下前向和反向传输相位是相同的。然而，直方图也显示出在相位差不为零的区域有一定的频率，这进一步证实了超表面具有非互易特性。

同时，根据结果可知，相位的分布十分广泛，相位差也足以覆盖 $(-\pi, \pi)$ 整个区间，据此通过参数调控，可以实现丰富的非互易超表面。

4.3 光束偏折非互易超表面

4.3.1 包含局部响应

首先，我们假设不区分局部响应，并认为其能够实现有效的调控。为实现光束偏折的非互易超表面，我们希望该超表面能够实现单向透射与单向光束偏折。为达到这一目的，相位调控应满足以下要求：一个方向的透射相位基本保持一致，而另一个方向的透射相位呈现 $(0, \pi/2, \pi, 3\pi/2)$ 的梯度变化，从而导致相位延迟和光束偏折。

基于这一思路，我们编写了相应的代码（见附录 A1），以确保某一方向的透射相位基本持平，并在另

一个方向上实现所需的相位变化。然而，结果中不可避免会出现误差，因此如何评估超表面单元的最优性是我们需要解决的问题。

根据文献 [13]，我们采用 FOM (figure of merit) 评估方法，其定义如下：

$$T_{\text{eff},f} = T_f \frac{\sin(|\phi_f - \phi_{\text{target},f}|)}{|\phi_f - \phi_{\text{target},f}|} \quad (2)$$

$$T_{\text{eff},b} = T_b \frac{\sin(|\phi_b - \phi_{\text{target},b}|)}{|\phi_b - \phi_{\text{target},b}|} \quad (3)$$

$$\text{FOM} = T_{\text{eff},f} \cdot T_{\text{eff},b} \quad (4)$$

其中, T_f 和 T_b 分别代表正向和反向透射率, ϕ_f 和 ϕ_b 分别是正向和反向透射的相位, 而 $\phi_{\text{target},f}$ 和 $\phi_{\text{target},b}$ 则是目标的正向和反向透射相位。通过计算参数空间中所有单元的 FOM 值, 具有最大 FOM 值的超原子被视为最优单元。当涉及多个单元时, 其整体评估通过单个单元 FOM 值的乘积进行。

通过搜索, 获得了正向相位持平的结果, 如表3所示。

表 3 正向相位持平最优单元

Table 3. Forward phase holds the optimal unit

半径 (r)	高度 (h)	正向透射率	反向透射率	正向相位	反向相位
3.00 mm	4.70 mm	0.89	0.53	3.08 rad	-2.42 rad
2.90 mm	7.00 mm	0.93	0.68	2.97 rad	2.09 rad
2.95 mm	5.80 mm	0.92	0.88	2.84 rad	0.61 rad
3.10 mm	3.00 mm	0.90	0.97	3.11 rad	-0.55 rad

同时, 我们也搜索了反向相位持平的最优解, 如表4所示。

表 4 反向相位持平最优单元

Table 4. Reverse phase holding the optimal unit

半径 (r)	高度 (h)	正向透射率	反向透射率	正向相位	反向相位
4.55 mm	7.00 mm	0.79	0.87	-0.99 rad	-0.92 rad
3.90 mm	6.45 mm	0.63	0.70	0.80 rad	-0.64 rad
3.20 mm	3.35 mm	0.70	0.99	2.13 rad	-0.64 rad
2.60 mm	6.15 mm	0.96	0.71	-2.52 rad	-0.81 rad

为直观地展示原子所处的位置, 我们将搜索结果绘制于参数扫描结果中, 正向相位持平结果如图15(a)所示, 反向相位持平结果则如图15(b)所示。

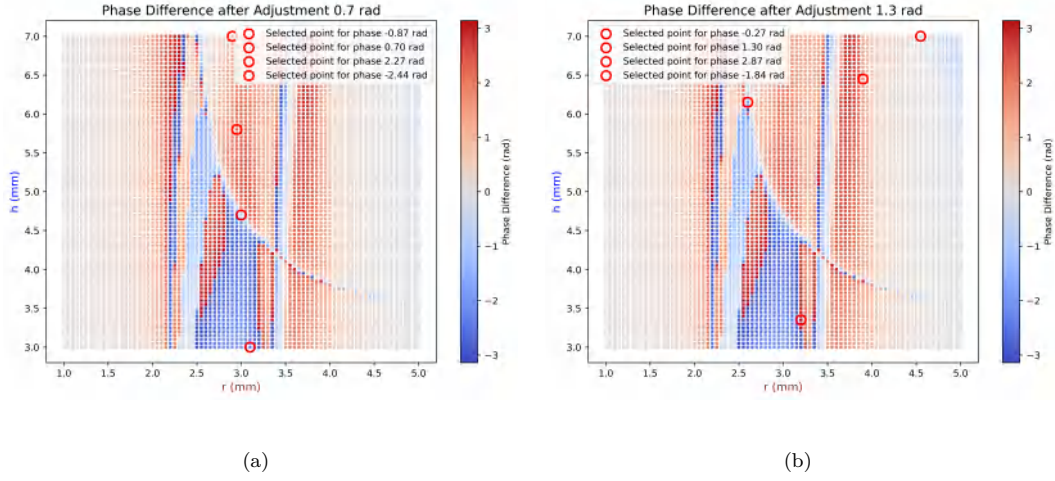


图 15 最优单元位置 (a) 正向 (b) 反向

Fig. 15. Optimal unit position (a) Forward (b) Reverse

可以看出，此时局部响应上分布有若干单元，我们对最优解均进行了仿真计算，但并没有显示出十分优异的相位偏折结果，可能是由于局部响应无法迁移到非对称情况下导致的。

4.3.2 剔除局部响应

通过采点，我们拟合出局局部响应曲线，并将数据剔除，重新绘制参数响应结果如图16所示。

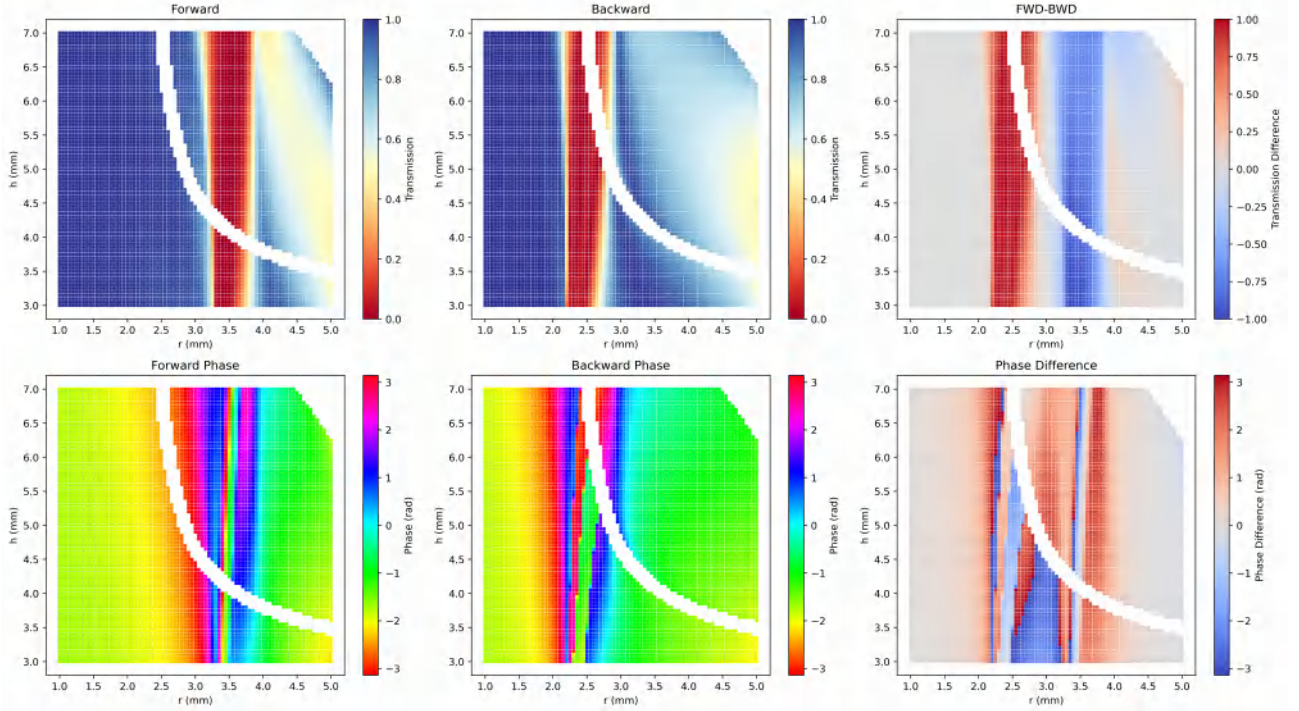


图 16 三层超原子参数响应结果 (剔除局部响应)

Fig. 16. Results of the three-layer superatomic parameter response (excluding local response)

可以看出，此时局部响应已经全部剔除。同时，相位差也仍然有着较为丰富的相应，仍然有构造基于

相位调控的非互易超表面的可能。我们对这些剔除后的数据应用相位识别代码，同样为构造光束偏折的非互易超表面，利用 FOM 协议筛选出最优结果。正向相位持平 and 反向相位持平的结果分别如表5和表6。

表 5 剔除局部响应正向相位持平最优单元

Table 5. Optimal unit phase holding in the reverse direction after excluding local response

半径 (r)	高度 (h)	正向透射率	反向透射率	正向相位	反向相位
2.45 mm	3.95 mm	0.98	0.04	-1.95 rad	-0.53 rad
1.55 mm	6.65 mm	1.00	1.00	-1.98 rad	-2.64 rad
2.10 mm	5.60 mm	0.99	0.93	-2.16 rad	1.93 rad
2.20 mm	4.50 mm	0.99	0.41	-2.19 rad	0.70 rad

表 6 剔除局部响应反向相位持平最优单元

Table 6. Optimal unit phase holding in the reverse direction after excluding local response

半径 (r)	高度 (h)	正向透射率	反向透射率	正向相位	反向相位
2.55 mm	5.00 mm	0.98	0.04	-2.29 rad	-0.91 rad
4.40 mm	7.00 mm	0.65	0.83	-0.65 rad	-0.80 rad
3.85 mm	6.40 mm	0.41	0.70	1.58 rad	-0.63 rad
3.10 mm	3.20 mm	0.89	0.98	2.52 rad	-0.74 rad

将搜索结果绘制于参数扫描结果中,正向相位持平结果如图17(a)所示,反向相位持平结果则如图17(b)所示。

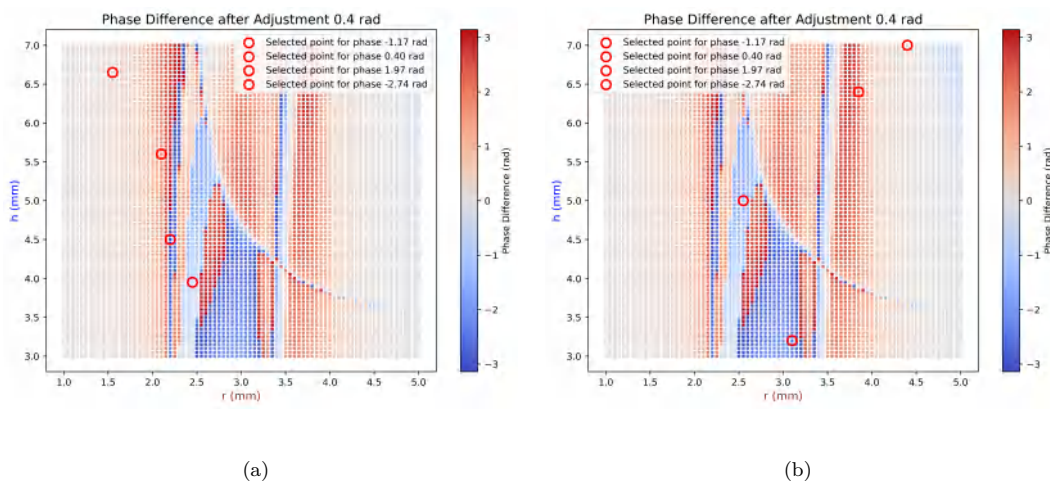


图 17 剔除局部响应的最优单元位置 (a) 正向 (b) 反向

Fig. 17. Optimal unit position excluding local response (a) Forward (b) Reverse

可以看出，此时结果中已经不再包含局部响应的结果。对比未剔除的结果，可以很明显地看出，透射率从基本均能保持 50% 以上，变为存在了一个极小值。但我们并不知道透射率对偏折结果的影响程度。我们利用这些单元对光束偏折进行了仿真，结果仍然没有出现很好的偏折效果。这将在实验展望中进行阐述。

4.4 实验验证

由于实验成本较高，我们仅对单向传播部分的三层超原子结果尝试进行实验验证。超原子结构如图18(a)所示，样品如图18(b)所示。

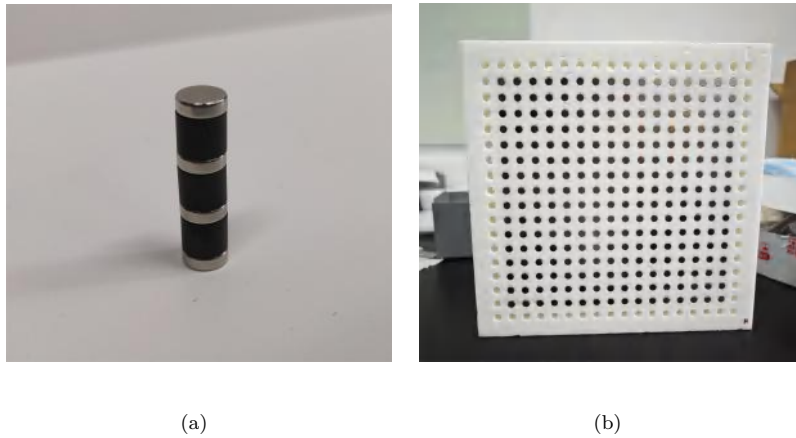


图 18 (a) 超原子 (b) 实验样品图

Fig. 18. (a) Superatom (b) Experimental Sample Diagram

实验装置图如图19(a)所示，测量示意图如图19(b)所示。



图 19 (a) 实验装置 (b) 测量示意图

Fig. 19. (a) Experimental Setup (b) Schematic Diagram of Measurement

实验中，设置微波天线之间的剧烈为 37cm，所用超表面尺寸为 16×16 ，天线的有效区间为 8.2GHz-12.4GHz，但我们测量的区间较大，超出区间的数据仅供参考。

4.4.1 线偏振结果

在实验中，无法直接进行圆偏振的测量，因此需要测量线偏振再进行换算，得到圆偏振的结果。

具体实验中，测量区间为 6GHz-16GHz，共测量了 8 组结果，分别对应 ss、sp、ps、pp 波的出射和入射，以及前向和后向。绘制线偏振的透射谱如图20所示。

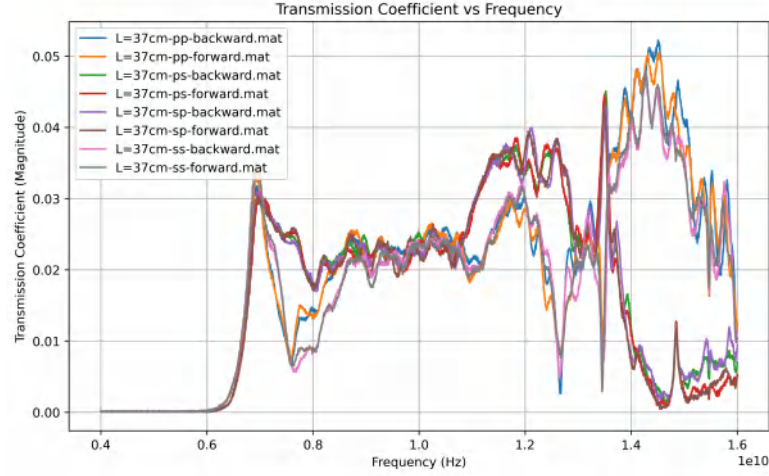


图 20 线偏振透射谱

Fig. 20. Linearly Polarized Transmission Spectrum

可以看出，对于线偏振，其透射谱并没有出现前向和后向的差别，这是因为线性响应本身的透射率结果并非是非互易的，其由于旋磁效应产生的偏转角度是一致的。

4.4.2 圆偏振结果

利用公式将线偏振转换为圆偏振，公式为

$$\begin{aligned}
 T_{rr} &= (T_{xx} + T_{yy}) + i(T_{xy} - T_{yx}), \\
 T_{rl} &= (T_{xx} - T_{yy}) - i(T_{xy} + T_{yx}), \\
 T_{lr} &= (T_{xx} - T_{yy}) + i(T_{xy} + T_{yx}), \\
 T_{ll} &= (T_{xx} + T_{yy}) - i(T_{xy} - T_{yx}).
 \end{aligned} \tag{5}$$

由于样本具有对称性，可以选择 x 为 p 方向，y 为 s 方向，则可以得到前向及后向的八种结果。绘制出右旋光的前后向透射结果，如图21(a)所示，同时绘制其相位差如图21(b)所示。

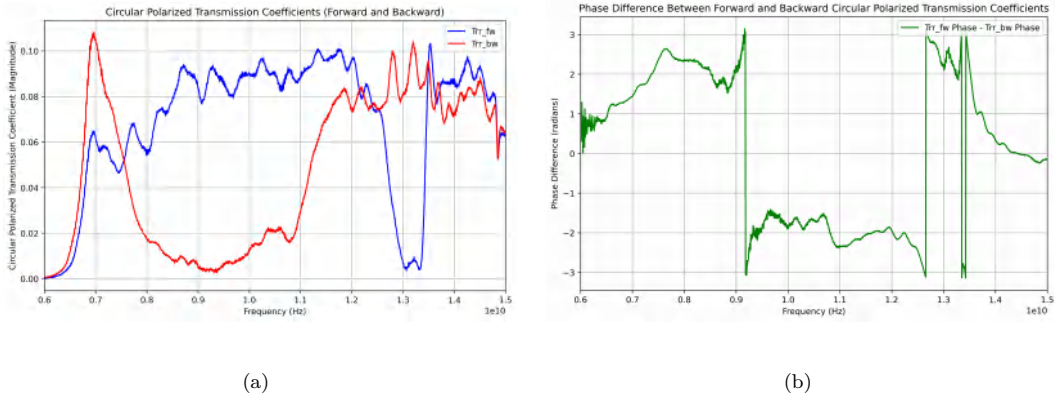


图 21 (a) 圆偏振透射谱 (b) 前后向相位差

Fig. 21. (a) Circular Polarization Transmission Spectrum (b) Forward-Backward Phase Difference

通过对比仿真结果与实验数据，我们能够观察到一些关键的趋势和差异。在透射谱方面，仿真与实验的曲线趋势大体上是一致的，这表明我们的仿真模型在一定程度上能够准确地预测实验结果。然而，我们也注意到在高频区域出现了频移现象，这可能是由于实验条件与仿真假设之间的差异所导致的。这种频移为我们提供了进一步优化仿真模型和实验设置的重要线索。

在相位调控方面，实验结果显示的调控幅度虽然相对较小，但调控范围依然基本覆盖了从 $-\pi$ 到 π 的区间。这意味着尽管存在一定的限制，我们的实验设置仍然具备实现复杂相位超表面构造的潜力。这一发现对于设计和制造具有特定光学特性的超表面具有重要意义，因为它表明我们的方法在理论上是可行的，并且有潜力被进一步发展和完善。

尽管实验中存在可以改进的地方，以及不可避免的误差，但目前的实验结果已经初步验证了超表面的非互易性质。这一点不仅证实了我们的理论预测，而且为未来的研究和应用奠定了坚实的基础。通过细致分析误差来源并针对性地进行改进，我们有望实现更加精确和高效的相位调控，从而推动非互易超表面技术的发展。

5 讨论部分

5.1 误差分析

在本次实验中，误差主要来源于以下几个方面：

1. 仿真过程中，外磁场由永磁体产生，永磁体的尺寸会影响磁场强度。由于缺乏磁场与尺寸变化关系

的函数，仿真中将磁场大小设定为固定值，这可能导致仿真结果与实验结果存在偏差。

2. 仿真扫描阶段，由于时间限制，各尺寸参数之间未能实现连续变化，导致扫描结果不连续，难以捕捉到所有精确的参数点。

3. 将相位响应结果应用于非均匀超表面时，无法确保其有效性，这可能是导致光束偏折非互易超表面实验结果与预期存在差异的原因。

4. 实验中，天线发射的电磁波并非完全均匀，存在一定的系统误差。

5. 由于样品数量有限，无法进行大规模样品实验，这可能会对实验结果产生影响。

5.2 实验展望

1. 通过对比单层与三层响应结果，我们发现三层结构的响应确实更为显著。然而，这也引发了我们对参数敏感性增加可能导致超表面构造难度提升的担忧。因此，我们可以考虑对单层响应进行参数扫描，尽管其响应较弱，但稳定性较高，通过扩展扫描范围，我们可以获得更广泛的响应区间。

2. 在相位识别阶段，我们尚未考虑透射率的一致性。理论上，如果超原子保持统一的透射率，其发射的电磁波将更加均匀，从而提高形成偏折光束的概率。未来，我们可以考虑筛选电磁波的透射率，以确保其一致性。

3. 当前的参数扫描范围较为有限，我们可以通过扩大扫描范围来获得更丰富的相位响应，进而实现更精细的调控。

4. 目前的算法功能较为单一，主要目标是构造光束偏折非互易超表面。实际上，我们可以扩展算法的功能，探索更多相位调控的可能性。

5. 实验还有许多提升空间，例如扩大有效频率范围和增加样品面积等，这些改进都可能显著提高实验效果，并为构建基于相位调控的非互易超表面打下坚实的基础。

6 结论

在本次实验中，我们对基于磁性调控的非互易超表面进行了研究，研究十分系统，成功构造了单向传播非互易超表面，并对更复杂的相位调控非互易超表面进行了初步探究。

首先我们通过构建单层超原子非互易超表面，并设置超原子半径为 3mm、YIG 高度 5mm、永磁体高度 2mm、超原子间距 12mm，在 6GHz 至 15GHz 频率范围内进行了正反向透射的频率扫描。COMSOL

仿真结果显示，在正反向传播中透射率出现明显低谷，且频率不同，说明该设计实现了单向传输的非互易特性。进一步计算得出隔离度极值分别为 8.40GHz 处的 19.21 dB 和 11.49GHz 处的-12.31 dB，表明在这些频率下具有较强的单向传输性能。透射谱低谷处的磁场分布图显示，磁场共振使电磁波被反射，产生了显著的非互易效应。

在单层超原子的基础上，构建了 ABABABA 型排列的三层超原子，以增加带宽和提升单向传播性能。三层结构在频率响应上显示出更宽的带宽，且在正反向传播中出现了三个局部极小值，表明多层结构增强了非互易性。在隔离度方面，三层超原子的极大值达到 8.35GHz 处的 29.66 dB，极小值为 11.48GHz 处的-24.89 dB，比单层提高了约 5 dB，表明该结构的单向传输性能显著优于单层。

单向传输的非互易超表面是最基本的非互易器件，基于相位调控的更复杂非互易器件可实现如非互易光束偏折、非互易超透镜等应用。研究发现，超原子的尺寸决定了相位调控能力，因此工作频率的选择尤为重要。对单层超原子在 150mm 仿真空间下的频率响应分析显示，正反向传播的相位差在磁场共振处较大，但无法覆盖整个 $(-\pi, \pi)$ 区间。相比之下，三层超原子展示出更强的相位频率响应，可覆盖 $(-\pi, \pi)$ 区间，且随频率变化剧烈。综合分析表明，在 10.5GHz 频率下，正反向传播的透射率较高且位于共振响应之间，适合进一步研究尺寸对相位响应的调控，从而实现复杂的非互易超表面设计。

接着，在固定频率 10.5GHz 下，研究了三层超原子中 YIG 的高度和半径对透射率和相位的影响。基于参数 $r = 3\text{mm}$ 和 $h = 5\text{mm}$ ，在上下浮动 2mm 范围内进行扫描，结果显示，透射谱和相位谱的显著响应多集中在低透射率处，这是由于磁场共振导致磁效应增强。反向传播的响应区间比正向传播更窄，且包含一个更细的局部响应区间，表明反向传播中的相位变化更敏感。

正反向相位差分布图显示了丰富的相位差分布，绝大部分数据点集中在接近零相位差的区域，但仍有一些明显的非零相位差点，证实了非互易特性。同时，相位差直方图显示相位差为零的频率最高，但非零相位差的分布也表明了超表面的非互易性。通过此相位覆盖 $(-\pi, \pi)$ 区间的结果，参数调控有望实现更多复杂的非互易超表面设计。

最后，为实现光束偏折的非互易超表面，我们设计了一种结构，在该结构中，正向透射相位保持恒定，而反向透射相位沿着梯度变化。为评估结构性能，我们基于文献 [13] 的评估方法引入 FOM 参数，通过正反向透射率及相位目标值的匹配程度，筛选出最优单元，并记录其参数（表3和4）。

为剔除局部响应带来的干扰，我们拟合并剔除了局部响应单元，并在剔除后重新筛选最优单元（表5和6）。绘图如图16所示。结果表明，透射率有所降低，但仍有构造非互易偏折超表面的可能，让人遗憾的是，最

终仿真结果显示偏折效果不明显。

为验证仿真的准确性，我们做了单向传输非互易超表面的实验验证，对比发现透射谱实验结果与仿真结果的曲线趋势基本一致，只是出现了一些频移，同时相位也具有良好的调控结果，对构造非互易超表面提供了实验可行性验证。

参考文献

- [1] Hou-Tong, Chen A, Taylor J N. A review of metasurfaces: physics and applications[J]. Reports on Progress in Physics, 2016, 79(7): 076401.
- [2] Soukoulis C M, Wegener M. Past achievements and future challenges in the development of three-dimensional photonic metamaterials[J]. Nature Photonics, 2011, 5: 523-530.
- [3] Yu N, Capasso F. Flat optics with designer metasurfaces[J]. Nature Materials, 2014, 13(2): 139-150.
- [4] Luo W, Xiao S, He Q, Sun S, Zhou L. Photonic Spin Hall Effect with Nearly 100% Efficiency. Advanced Optical Materials. 2015, 3(5):1102-1108.
- [5] Sun S, He Q, Xiao S, et al. Gradient-index meta-surfaces as a bridge linking propagating waves and surface waves. Nature Mater. 2012, 11(4):426-431.
- [6] Jia Z, Shu F, Gao Y, Cheng F, Peng R, Fan R, Liu Y, Wang M. Dynamically Switching the Polarization State of Light Based on the Phase Transition of Vanadium Dioxide. Phys. Rev. Applied. 2018, 9(3):034009.
- [7] Pan H, Chen MK, Tsai DP, Wang S. Nonreciprocal Pancharatnam-Berry metasurface for unidirectional wavefront manipulations[J]. Optics Express, 2024, 32(25): 25632.
- [8] Jin B, Argyropoulos C. Self-induced passive nonreciprocal transmission by nonlinear bifacial dielectric metasurfaces. Phys. Rev. Applied. 2020, 13:054056.
- [9] Ma Q, Chen L, Jing H B, Hong Q R, Cui H Y, Liu Y, Li L L, Cui T J. Controllable and programmable nonreciprocity based on detachable digital coding metasurface [J]. Advanced Optical Materials, 2019, 7: 1901285.
- [10] Yang W, Qin J, Long J, Yan W, Yang Y, Li C, Li E, Hu J, Deng L, Du Q, Bi L. A self-biased non-reciprocal magnetic metasurface for bidirectional phase modulation [J]. Nature Electronics, 2023, 6: 234-243.
- [11] K. Chen, G. W. Ding, G. W. Hu, Z. W. Jin, J. M. Zhao, Y. J. Feng, T. Jiang, A. Alu, and C. W. Qiu, Directional Janus Metasurface, Adv. Mater. 32, 1906352 (2020).
- [12] Bhoi B, Kim S-K. Photon-magnon coupling: Historical perspective, status, and future directions. Solid State Physics. 2019.

- [13] Shalaginov, M. Y. et al. Reconfigurable all-dielectric metalens with diffraction-limited performance. Nat. Commun. 12, 1225 (2021).

附录 A1

频率响应绘图:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import glob
4 import os
5
6 # 文件夹路径
7 folder_path = r'C:\Users\14716\Desktop\Magnetic PC Metasurface\COMSOL\YIG'
8
9 # 获取所有文件名, 例如 h=2.txt, h=3.txt 等
10 file_list = glob.glob(os.path.join(folder_path, 'our_canshu_single.txt'))
11
12 # 初始化存储数据的列表
13 f_values = []
14 abs_S21_fwd = []
15 abs_S21_bwd = []
16 arg_S21_fwd = []
17 arg_S21_bwd = []
18
19 # 遍历所有文件并读取数据
20 for file_name in file_list:
21     try:
22         # 使用 genfromtxt 跳过注释行, 并处理非数值字符
23         data = np.genfromtxt(file_name, comments='#', skip_header=1) # 处理以 '#' 开头的注释行
24
25         if data.size == 0:
26             print(f"File {file_name} is empty or has invalid data.")
27             continue
28
29         f = data[:, 0] # 第一列是频率
```

```

30     abs_S21_fwd_data = data[:, 3] # 绝对值 S21 正向 (第四列)
31     abs_S21_bwd_data = data[:, 4] # 绝对值 S21 反向 (第五列)
32     arg_S21_fwd_data = data[:, 1] # 相位 S21 正向 (第二列)
33     arg_S21_bwd_data = data[:, 2] # 相位 S21 反向 (第三列)
34
35     # 将数据添加到列表
36     f_values.append(f)
37     abs_S21_fwd.append(abs_S21_fwd_data)
38     abs_S21_bwd.append(abs_S21_bwd_data)
39     arg_S21_fwd.append(arg_S21_fwd_data)
40     arg_S21_bwd.append(arg_S21_bwd_data)
41
42 except Exception as e:
43     print(f"Error reading {file_name}: {e}")
44
45 # 将数据转换为 numpy 数组
46 f_values = np.concatenate(f_values)
47 abs_S21_fwd = np.concatenate(abs_S21_fwd)
48 abs_S21_bwd = np.concatenate(abs_S21_bwd)
49 arg_S21_fwd = np.concatenate(arg_S21_fwd)
50 arg_S21_bwd = np.concatenate(arg_S21_bwd)
51
52 # 绘制透射率的正向和反向传输的大小
53 plt.figure(figsize=(10, 6), dpi=400)
54 plt.plot(f_values, abs_S21_fwd, label="Forward Transmission |T|", color='blue', lw=2)
55 plt.plot(f_values, abs_S21_bwd, label="Backward Transmission |T|", color='red', lw=2, linestyle='--')
56 plt.xlabel("Frequency (Hz)")
57 plt.ylabel("Transmission Magnitude |T|")
58 plt.title("Forward and Backward Transmission Magnitude |T|")
59 plt.legend()
60 plt.grid(True)

```

```

61 plt.tight_layout()
62
63 # 计算并绘制透射率差异（正向 - 反向）
64 T_difference = abs_S21_fwd - abs_S21_bwd
65
66 plt.figure(figsize=(10, 6), dpi=400)
67 plt.plot(f_values, T_difference, label="Difference (Forward - Backward)", color='green', lw=2)
68 plt.xlabel("Frequency (Hz)")
69 plt.ylabel("Transmission Magnitude Difference")
70 plt.title("Transmission Magnitude Difference (Forward - Backward)")
71 plt.legend()
72 plt.grid(True)
73 plt.tight_layout()
74
75 # 绘制正向和反向的相位
76 plt.figure(figsize=(10, 6), dpi=400)
77 plt.plot(f_values, arg_S21_fwd, label="Forward Phase T", color='blue', lw=2)
78 plt.plot(f_values, arg_S21_bwd, label="Backward Phase T", color='red', lw=2, linestyle='--')
79 plt.xlabel("Frequency (Hz)")
80 plt.ylabel("Phase (Radians)")
81 plt.title("Forward and Backward Phase T")
82 plt.legend()
83 plt.grid(True)
84 plt.tight_layout()
85
86 # 计算并绘制相位差异（正向 - 反向）
87 phase_difference = np.angle(np.exp(1j * (arg_S21_fwd - arg_S21_bwd)))
88
89 plt.figure(figsize=(10, 6), dpi=400)
90 plt.plot(f_values, phase_difference, label="Phase Difference (Forward - Backward)", color='green',
          lw=2)

```

```

91 plt.xlabel("Frequency (Hz)")
92 plt.ylabel("Phase Difference (Radians)")
93 plt.title("Phase Difference (Forward - Backward)")
94 plt.legend()
95 plt.grid(True)
96 plt.tight_layout()
97
98 # 展示所有图表
99 plt.show()
100
101 # 转换为 dB
102 abs_S21_fwd_dB = 20 * np.log10(abs_S21_fwd)
103 abs_S21_bwd_dB = 20 * np.log10(abs_S21_bwd)
104
105 # 绘制透射谱 (dB)
106 plt.figure(figsize=(10, 6), dpi=400)
107 plt.plot(f_values, abs_S21_fwd_dB, label="Forward Transmission (dB)", color='blue', lw=2)
108 plt.plot(f_values, abs_S21_bwd_dB, label="Backward Transmission (dB)", color='red', lw=2,
109         linestyle='--')
110 plt.xlabel("Frequency (Hz)")
111 plt.ylabel("Transmission (dB)")
112 plt.title("Forward and Backward Transmission (dB)")
113 plt.legend()
114 plt.grid(True)
115 plt.tight_layout()
116
117 # 展示图表
118 plt.show()

```

扫参绘图及相位识别：

```

1 import numpy as np

```

```

2  import matplotlib.pyplot as plt
3  import glob
4  import os
5  import math
6  from matplotlib.colors import LogNorm
7  from scipy.optimize import minimize
8
9
10  '''-----初始化-----'''
11  # 文件夹路径
12  folder_path = r'C:\Users\14716\Desktop\Magnetic PC Metasurface\COMSOL\YIG\10.5'
13
14  # 获取所有文件名, 例如 h=2.txt, h=3.txt 等
15  file_list = glob.glob(os.path.join(folder_path, '*.txt'))
16
17  # 初始化存储数据的列表
18  r_values = []
19  h_values = []
20  abs_S21_fwd = []
21  abs_S21_bwd = []
22  arg_S21_fwd = []
23  arg_S21_bwd = []
24
25  '''-----文件读取-----'''
26  # 初始化存储数据的列表
27  all_data = [] # 用于存储所有单元及其相关信息
28  file_num = 0
29  # 遍历所有文件并读取数据
30  for file_name in file_list:
31      try:
32          # 使用 genfromtxt 跳过注释行, 并处理非数值字符

```

```

33     data = np.genfromtxt(file_name, comments='%', skip_header=1) # 处理以 '%' 开头的注释行
34
35     if data.size == 0:
36         print(f"File {file_name} is empty or has invalid data.")
37         continue
38
39     r = data[:, 0] # 第一列是 r 值
40     h = data[:, 1] # 第二列是 h 值
41     abs_S21_fwd_data = data[:, 2] # 绝对值 S21 正向 (第三列)
42     abs_S21_bwd_data = data[:, 3] # 绝对值 S21 反向 (第四列)
43     arg_S21_fwd_data = data[:, 4] # 相位 S21 正向 (第五列)
44     arg_S21_bwd_data = data[:, 5] # 相位 S21 反向 (第六列)
45
46     # 将数据添加到列表
47     r_values.append(r)
48     h_values.append(h)
49     abs_S21_fwd.append(abs_S21_fwd_data)
50     abs_S21_bwd.append(abs_S21_bwd_data)
51     arg_S21_fwd.append(arg_S21_fwd_data)
52     arg_S21_bwd.append(arg_S21_bwd_data)
53
54     # 为每个数据点添加索引信息, 并将其合并为一行
55     num_points = len(r)
56     indices = np.arange(num_points*file_num, num_points*(file_num + 1), 1).reshape(-1, 1) #
57     # 创建索引列
58     combined_data = np.hstack((indices, r.reshape(-1, 1), h.reshape(-1, 1),
59                                abs_S21_fwd_data.reshape(-1, 1),
60                                abs_S21_bwd_data.reshape(-1, 1),
61                                arg_S21_fwd_data.reshape(-1, 1),
62                                arg_S21_bwd_data.reshape(-1, 1)))

```

```

63     # 将数据添加到列表中
64     all_data.append(combined_data)
65
66     file_num += 1
67
68     except Exception as e:
69         print(f"Error reading {file_name}: {e}")
70
71     # 将数据转换为 numpy 数组
72     r_values = np.concatenate(r_values)
73     h_values = np.concatenate(h_values)
74     abs_S21_fwd = np.concatenate(abs_S21_fwd)
75     abs_S21_bwd = np.concatenate(abs_S21_bwd)
76     arg_S21_fwd = np.concatenate(arg_S21_fwd)
77     arg_S21_bwd = np.concatenate(arg_S21_bwd)
78
79     # 将数据转换为 numpy 数组
80     all_data = np.vstack(all_data) # 将所有数据拼接成一个大的数组
81
82     '''-----最小化距离并筛选-----'''
83     # 定义数据列索引
84     INDEX_COL = 0
85     R_COL = 1
86     H_COL = 2
87
88     # 初始化存储需要保留的行
89     keep_rows = []
90
91     # 定义曲线函数
92     def curve(x):
93         return 162085.51453 * np.exp(-x / 0.22109) + 11.33267 * np.exp(-x / 1.36609) + 3.19609

```

```

94
95 def distance(x, point):
96     y_curve = curve(x)
97     return math.sqrt((x - point[0])*(x - point[0]) + (y_curve - point[1])*(y_curve - point[1]))
98
99 # 遍历所有单元，进行优化
100 for row in all_data:
101     r_val = row[R_COL]
102     h_val = row[H_COL]
103
104     # 使用优化技术找到最小化距离的 x 值
105     result = minimize(lambda x: distance(x, (r_val, h_val)), r_val, bounds=[(2.5, 5.0)])
106
107     # 计算点到曲线的最小距离
108     min_x = result.x[0]
109     min_distance = distance(min_x, (r_val, h_val))
110
111     # 如果最小距离 > 0.1，将当前行数据保留
112     if min_distance > 0.1 and r_val * r_val / 25 + h_val * h_val / 49 < 1.8 :
113         keep_rows.append(row)
114
115 # 将保留的行转换为 numpy 数组
116 filtered_data = np.array(keep_rows)
117
118 # 从筛选后的数据中提取各列信息
119 filtered_indices = filtered_data[:, INDEX_COL]
120 r_filtered = filtered_data[:, R_COL]
121 h_filtered = filtered_data[:, H_COL]
122 abs_S21_fwd_filtered = filtered_data[:, 3]
123 abs_S21_bwd_filtered = filtered_data[:, 4]
124 arg_S21_fwd_filtered = filtered_data[:, 5]

```

```

125 arg_S21_bwd_filtered = filtered_data[:, 6]
126
127 print(f"Remaining matching units: {len(filtered_indices)}")
128
129
130 '''-----计算差异并绘图-----'''
131 # 计算正向和反向的差异
132 abs_S21_diff = abs_S21_fwd - abs_S21_bwd
133
134 # 使用 np.angle 归一化相位差到  $[-\pi, \pi]$  之间
135 arg_S21_diff = np.angle(np.exp(1j * (arg_S21_fwd - arg_S21_bwd)))
136
137 # 创建图形
138 fig, axs = plt.subplots(2, 3, figsize=(18, 10), dpi=400)
139
140 # 设置点的大小和形状
141 point_size = 5 # 调整点的大小
142 marker_style = 's' # 将点的形状设置为方形
143
144 # 绘制传输系数图（正向、反向、差异）
145 im1 = axs[0, 0].scatter(r_values, h_values, c=abs_S21_fwd, cmap='RdYlBu', vmin=0, vmax=1,
146                        s=point_size, marker=marker_style)
147 axs[0, 0].set_title('Forward')
148 axs[0, 0].set_xlabel('r (mm)')
149 axs[0, 0].set_ylabel('h (mm)')
150 plt.colorbar(im1, ax=axs[0, 0], label='Transmission')
151
152 im2 = axs[0, 1].scatter(r_values, h_values, c=abs_S21_bwd, cmap='RdYlBu', vmin=0, vmax=1,
153                        s=point_size, marker=marker_style)
154 axs[0, 1].set_title('Backward')
155 axs[0, 1].set_xlabel('r (mm)')

```

```

154  axs[0, 1].set_ylabel('h (mm)')
155  plt.colorbar(im2, ax=axs[0, 1], label='Transmission')
156
157  im3 = axs[0, 2].scatter(r_values, h_values, c=abs_S21_diff, cmap='coolwarm', vmin=-1, vmax=1,
                           s=point_size, marker=marker_style)
158
159  axs[0, 2].set_title('FWD-BWD')
160  axs[0, 2].set_xlabel('r (mm)')
161  axs[0, 2].set_ylabel('h (mm)')
162  plt.colorbar(im3, ax=axs[0, 2], label='Transmission Difference')
163  # 插值点
164  x = np.array([2.5, 2.65, 2.85, 3.0, 3.4, 3.65, 3.85, 4.25, 4.75, 4.95])
165  y = np.array([7.0, 5.85, 5.0, 4.65, 4.2, 4.0, 3.85, 3.7, 3.55, 3.5])
166  im3 = axs[0, 2].scatter(x, y, facecolors='none', edgecolors='red', s=100, linewidths=2,
                           label='Selected point')
167
168  x_line = np.arange(2.5, 5.0, 0.01)
169  y_line = 162085.51453 * np.exp(-x_line / 0.22109) + 11.33267 * np.exp(-x_line / 1.36609) + 3.19609
170  im3 = axs[0, 2].plot(x_line, y_line)
171
172  # 绘制相位图 (正向、反向、差异)
173  im4 = axs[1, 0].scatter(r_values, h_values, c=arg_S21_fwd, cmap='hsv', vmin=-np.pi, vmax=np.pi,
                           s=point_size, marker=marker_style)
174  axs[1, 0].set_title('Forward Phase')
175  axs[1, 0].set_xlabel('r (mm)')
176  axs[1, 0].set_ylabel('h (mm)')
177  plt.colorbar(im4, ax=axs[1, 0], label='Phase (rad)')
178
179  im5 = axs[1, 1].scatter(r_values, h_values, c=arg_S21_bwd, cmap='hsv', vmin=-np.pi, vmax=np.pi,
                           s=point_size, marker=marker_style)
180  axs[1, 1].set_title('Backward Phase')

```

```

181  axs[1, 1].set_xlabel('r (mm)')
182  axs[1, 1].set_ylabel('h (mm)')
183  plt.colorbar(im5, ax=axs[1, 1], label='Phase (rad)')
184
185  im6 = axs[1, 2].scatter(r_values, h_values, c=arg_S21_diff, cmap='coolwarm', s=point_size,
                           marker=marker_style)
186  axs[1, 2].set_title('Phase Difference')
187  axs[1, 2].set_xlabel('r (mm)')
188  axs[1, 2].set_ylabel('h (mm)')
189  plt.colorbar(im6, ax=axs[1, 2], label='Phase Difference (rad)')
190
191  plt.tight_layout()
192  plt.show()
193
194  '''-----绘制提出数据后的结果-----'''
195  # 计算正向和反向的差异
196  abs_S21_diff_filtered = abs_S21_fwd_filtered - abs_S21_bwd_filtered
197
198  # 使用 np.angle 归一化相位差到  $[-\pi, \pi]$  之间
199  arg_S21_diff_filtered = np.angle(np.exp(1j * (arg_S21_fwd_filtered - arg_S21_bwd_filtered)))
200
201  # 创建图形
202  fig, axs = plt.subplots(2, 3, figsize=(18, 10), dpi=400)
203
204  # 设置点的大小和形状
205  point_size = 5 # 调整点的大小
206  marker_style = 's' # 将点的形状设置为方形
207
208  # 绘制传输系数图（正向、反向、差异）
209  im1 = axs[0, 0].scatter(r_filtered, h_filtered, c=abs_S21_fwd_filtered, cmap='RdYlBu', vmin=0,
                           vmax=1, s=point_size, marker=marker_style)

```

```

210  axs[0, 0].set_title('Forward')
211  axs[0, 0].set_xlabel('r (mm)')
212  axs[0, 0].set_ylabel('h (mm)')
213  plt.colorbar(im1, ax=axs[0, 0], label='Transmission')
214
215  im2 = axs[0, 1].scatter(r_filtered, h_filtered, c=abs_S21_bwd_filtered, cmap='RdYlBu', vmin=0,
        vmax=1, s=point_size, marker=marker_style)
216  axs[0, 1].set_title('Backward')
217  axs[0, 1].set_xlabel('r (mm)')
218  axs[0, 1].set_ylabel('h (mm)')
219  plt.colorbar(im2, ax=axs[0, 1], label='Transmission')
220
221  im3 = axs[0, 2].scatter(r_filtered, h_filtered, c=abs_S21_diff_filtered, cmap='coolwarm', vmin=-1,
        vmax=1, s=point_size, marker=marker_style)
222  axs[0, 2].set_title('FWD-BWD')
223  axs[0, 2].set_xlabel('r (mm)')
224  axs[0, 2].set_ylabel('h (mm)')
225  plt.colorbar(im3, ax=axs[0, 2], label='Transmission Difference')
226
227  # 绘制相位图 (正向、反向、差异)
228  im4 = axs[1, 0].scatter(r_filtered, h_filtered, c=arg_S21_fwd_filtered, cmap='hsv', vmin=-np.pi,
        vmax=np.pi, s=point_size, marker=marker_style)
229  axs[1, 0].set_title('Forward Phase')
230  axs[1, 0].set_xlabel('r (mm)')
231  axs[1, 0].set_ylabel('h (mm)')
232  plt.colorbar(im4, ax=axs[1, 0], label='Phase (rad)')
233
234  im5 = axs[1, 1].scatter(r_filtered, h_filtered, c=arg_S21_bwd_filtered, cmap='hsv', vmin=-np.pi,
        vmax=np.pi, s=point_size, marker=marker_style)
235  axs[1, 1].set_title('Backward Phase')
236  axs[1, 1].set_xlabel('r (mm)')

```

```

237  axs[1, 1].set_ylabel('h (mm)')
238  plt.colorbar(im5, ax=axs[1, 1], label='Phase (rad)')
239
240  im6 = axs[1, 2].scatter(r_filtered, h_filtered, c=arg_S21_diff_filtered, cmap='coolwarm',
                           s=point_size, marker=marker_style)
241  axs[1, 2].set_title('Phase Difference')
242  axs[1, 2].set_xlabel('r (mm)')
243  axs[1, 2].set_ylabel('h (mm)')
244  plt.colorbar(im6, ax=axs[1, 2], label='Phase Difference (rad)')
245
246  plt.tight_layout()
247  plt.show()
248
249
250  '''-----相位分布图-----'''
251  # 定义二维直方图的bin数
252  n_bins = 100
253
254  # 计算二维直方图：向前相位和向后相位的联合分布
255  hist, xedges, yedges = np.histogram2d(arg_S21_fwd, arg_S21_bwd, bins=n_bins, range=[[-np.pi, np.pi],
                                             [-np.pi, np.pi]])
256
257  # 创建网格，绘制热力图，并使用 LogNorm 来显示对数颜色条
258  plt.figure(figsize=(8, 6), dpi=400)
259  plt.imshow(hist.T, origin='lower', extent=[-np.pi, np.pi, -np.pi, np.pi], cmap='plasma',
              norm=LogNorm(), aspect='auto')
260
261  # 设置坐标轴
262  plt.xlabel(r'$\phi_f(\omega)$ (rad)', fontsize=12, color='brown')
263  plt.ylabel(r'$\phi_b(\omega)$ (rad)', fontsize=12, color='blue')
264

```

```

265 # 设置刻度
266 plt.xticks([-np.pi, -np.pi/2, 0, np.pi/2, np.pi], [r'$-\pi$', r'$-\pi/2$', '0', r'$\pi/2$',
    r'$\pi$'], color='brown')
267 plt.yticks([-np.pi, -np.pi/2, 0, np.pi/2, np.pi], [r'$-\pi$', r'$-\pi/2$', '0', r'$\pi/2$',
    r'$\pi$'], color='blue')
268
269 # 添加颜色条, 使用 LogNorm 控制
270 plt.colorbar(label='Log Frequency of Occurrence')
271
272 # 显示图像
273 plt.tight_layout()
274 plt.show()
275
276 '''-----相位直方图-----'''
277 # 绘制相位差直方图
278 plt.figure(figsize=(8, 6), dpi=400)
279 n_bins = 100 # 设置直方图的柱子数量
280 plt.hist(arg_S21_diff, bins=n_bins, range=(-np.pi, np.pi), color='skyblue', alpha=0.7)
281
282 # 设置图表标题和标签
283 plt.title('Phase Difference Histogram')
284 plt.xlabel('Phase Difference (rad)')
285 plt.ylabel('Frequency')
286
287 # 添加网格线便于观察
288 plt.grid(True)
289
290 # 显示图像
291 plt.tight_layout()
292 plt.show()
293

```

```

294  '''-----相位识别-----'''
295  # 用户选择是否使用相位识别
296  use_phase_recognition = 'Y'#input("是否使用相位识别? 输入 'Y' 继续, 输入 'N' 跳出: ").strip().upper()
297
298  if use_phase_recognition == 'Y':
299      # 用户输入多个相位差, 分隔为浮点数列表
300      phase_diffs = input("请输入多个目标相位差值 (用逗号分隔, 单位: 弧度): ").split(',')
301      phase_diffs = [float(phase) for phase in phase_diffs]
302      tolerance = 0.3#float(input("请输入允许的误差范围 (单位: 弧度): "))
303
304      # 定义函数调整相位到 [-, ] 范围内
305      def adjust_phase(phase):
306          while phase > np.pi:
307              phase -= 2 * np.pi # 超出范围向下调整
308          while phase < -np.pi:
309              phase += 2 * np.pi # 超出范围向上调整
310          return phase
311
312      # 初始化最优结果的字典, 包含最优相位调整量
313      best_global_result = {'total_max_value': -1, 'results': {}, 'best_adjustment': None}
314
315      # 搜索直到找到符合条件的单元或达到最大相位
316      for adjustment in np.arange(0, 2 * np.pi / len(phase_diffs), 0.1):
317          # 调整相位差
318          adjusted_phase_diffs = [adjust_phase(phase + adjustment) for phase in phase_diffs]
319          matching_units = [] # 存储符合条件的单元索引
320
321          # 查找符合每个目标相位差的单元
322          for target_phase_diff in adjusted_phase_diffs:
323              matching_indices = np.where((arg_S21_diff >= target_phase_diff - tolerance) &
324                                          (arg_S21_diff <= target_phase_diff + tolerance))[0]

```

```

325     matching_units.extend(matching_indices.tolist()) # 添加符合条件的单元索引
326
327     matching_units = np.unique(matching_units) # 去重
328     indices_to_remove = [] # 用于存储需要删除的索引
329
330     for idx in matching_units:
331         # 使用优化技术找到最小化距离的x值
332         result = minimize(lambda x: distance(x, (r_values[idx], h_values[idx])), r_values[idx],
333                             bounds=[(2.5, 5.0)])
334
335         # 计算点到曲线的最小距离
336
337         min_x = result.x[0]
338         min_distance = distance(min_x, (r_values[idx], h_values[idx]))
339
340         # 如果最小距离小于0.1, 将其索引记录下来
341
342         if min_distance < 0.1 or r_values[idx] * r_values[idx] / 25 + h_values[idx] *
343             h_values[idx] / 49 > 1.8:
344             indices_to_remove.append(idx)
345
346     # 删除记录的单元
347
348     matching_units = [unit for unit in matching_units if unit not in indices_to_remove]
349
350     matching_units = np.array(matching_units)
351
352     #用正向相位区分
353
354     fwd_phases = arg_S21_fwd[matching_units] # 获取正向相位
355
356     # 计算正向相位的直方图
357
358     hist, bin_edges = np.histogram(fwd_phases, bins=20, density=True)
359
360     # 遍历每个直方图区间
361
362     found = False # 标志是否找到符合条件的区间
363
364     single_best_result = {'total_max_value': -1, 'results': {}} # 存储单次最优结果
365
366     for i in range(len(bin_edges) - 1):

```

```

354     bin_start = bin_edges[i]
355     bin_end = bin_edges[i + 1]
356     in_bin_indices = matching_units[(fwd_phases >= bin_start) & (fwd_phases < bin_end)] #
        找到当前区间内的单元
357
358     # 检查该区间内是否包含所有目标相位差
359     all_phase_diffs_present = all(
360         any((arg_S21_diff[in_bin_indices] >= adjust_phase(phase - tolerance)) &
361             (arg_S21_diff[in_bin_indices] <= adjust_phase(phase + tolerance))
362         ) for phase in adjusted_phase_diffs)
363
364     # 如果当前区间包含所有目标相位差
365     if len(in_bin_indices) > 0 and all_phase_diffs_present:
366         found = True # 设置标志, 表示找到符合条件的区间
367         total_max_value = 1 # 初始化当前区间的FOM值
368         current_results = {} # 存储当前区间的结果
369
370         # 在该区间内寻找每个目标相位差的最优解
371         for phase in adjusted_phase_diffs:
372             max_forward = None
373             max_backward = None
374             best_fom = -1 # 初始化FOM值
375             forward_phase = None
376             backward_phase = None
377             best_unit_params = None
378
379         # 遍历当前区间内的单元
380         for idx in in_bin_indices:
381             if (arg_S21_diff[idx] >= adjust_phase(phase - tolerance)) and \
382                 (arg_S21_diff[idx] <= adjust_phase(phase + tolerance)):
383                 forward_transmission = abs_S21_fwd[idx] # 正向透射率

```

```

384         backward_transmission = abs_S21_bwd[idx] # 反向透射率
385
386         forward_phase = arg_S21_fwd[idx] # 正向相位
387
388         backward_phase = arg_S21_bwd[idx] # 反向相位
389
390     # 计算 FOM 值
391
392     phase_f = adjust_phase(forward_phase - phase)
393
394     phase_b = adjust_phase(backward_phase - phase)
395
396     Teff_f = forward_transmission * np.sin(abs(phase_f)) / abs(phase_f)
397
398     Teff_b = backward_transmission * np.sin(abs(phase_b)) / abs(phase_b)
399
400     fom_value = Teff_f * Teff_b
401
402     # 更新最优FOM值和相关参数
403
404     if fom_value > best_fom:
405
406         max_forward = forward_transmission
407
408         max_backward = backward_transmission
409
410         max_value = forward_transmission + backward_transmission
411
412         forward_phase = arg_S21_fwd[idx] # 对应的正向相位
413
414         backward_phase = arg_S21_bwd[idx] # 对应的反向相位
415
416         best_unit_params = (r_values[idx], h_values[idx]) # 单元参数
417
418         best_fom = fom_value
419
420     total_max_value *= best_fom # 累积当前区间所有相位差的FOM值
421
422     current_results[phase] = {
423
424         'max_forward': max_forward,
425
426         'max_backward': max_backward,
427
428         'max_value': max_value,
429
430         'forward_phase': forward_phase,
431
432         'backward_phase': backward_phase,
433
434         'unit_params': best_unit_params,
435
436         'FOM': best_fom
437     }
438 }

```

```

415
416     # 判断当前区间是否为单次最优结果
417     if total_max_value > single_best_result['total_max_value']:
418         single_best_result['total_max_value'] = total_max_value
419         single_best_result['results'] = current_results
420
421 # 输出当前相位调整后的单次最优结果
422 if found:
423     print(f"调整相位 {adjustment:.1f} rad: 找到符合条件的区间!")
424     # 在输出单次最优结果后, 绘制参数(r, h)图, 并圈出符合条件的点
425     plt.figure(figsize=(8, 6), dpi=400)
426     plt.scatter(r_values, h_values, c=arg_S21_diff, cmap='coolwarm', s=point_size,
427                 marker=marker_style)
428
429     # 设置坐标轴
430     plt.xlabel('r (mm)', fontsize=12, color='brown')
431     plt.ylabel('h (mm)', fontsize=12, color='blue')
432
433     # 添加颜色条
434     plt.colorbar(label='Phase Difference (rad)')
435
436     # 添加标题
437     plt.title(f'Phase Difference after Adjustment {adjustment:.1f} rad', fontsize=14)
438
439     print(f"此相位调整的FOM乘积为: {single_best_result['total_max_value']:.8f}")
440
441     # 遍历每个目标相位差结果并绘制红圈标记
442     for phase, result in single_best_result['results'].items():
443         unit_r, unit_h = result['unit_params']
444         phase_diff_single = adjust_phase(result['forward_phase']-result['backward_phase'])
445         print(f"目标相位差 = {phase:.2f} rad, "

```

```

445         f"实际相位差 = {phase_diff_single:.2f} rad, "
446         f"正向透射率 = {result['max_forward']:.2f}, "
447         f"反向透射率 = {result['max_backward']:.2f}, "
448         f"对应正向相位 = {result['forward_phase']:.2f} rad, "
449         f"对应反向相位 = {result['backward_phase']:.2f} rad, "
450         f"单元参数 (r, h) = ({unit_r:.2f} mm, {unit_h:.2f} mm), "
451         f"FOM最优值 = {result['FOM']:.6f}")
452
453     # 标记筛选出的点（单元参数）并为每个目标相位差添加标签
454     selected_r = unit_r
455     selected_h = unit_h
456     plt.scatter(selected_r, selected_h, facecolors='none', edgecolors='red', s=100,
457                linewidths=2,
458                label=f'Selected point for phase {phase:.2f} rad')
459
460     # 显示图例
461     plt.legend()
462     # 紧凑布局
463     plt.tight_layout()
464     # 显示图像
465     plt.show()
466     print(" ")
467
468     # 如果当前单次FOM乘积大于全局最优FOM乘积，则更新全局最优
469     if single_best_result['total_max_value'] > best_global_result['total_max_value']:
470         best_global_result = single_best_result.copy()
471         best_global_result['best_adjustment'] = adjustment # 记录最优相位调整量
472     else:
473         print(f"调整相位 {adjustment:.1f} rad: 未找到符合条件的区间，继续调整。\\n")
474
475     # 输出最终全局最优结果

```

```

475     if best_global_result['total_max_value'] > -1:
476         print(f"全局最优相位调整量为 {best_global_result['best_adjustment']:.2f} rad")
477         print(f"全局最优FOM乘积为: {best_global_result['total_max_value']:.8f}")
478         for phase, result in best_global_result['results'].items():
479             unit_r, unit_h = result['unit_params']
480             phase_diff = adjust_phase(result['forward_phase']-result['backward_phase'])
481             print(f"目标相位差 = {phase:.2f} rad, "
482                   f"实际相位差 = {phase_diff:.2f} rad, "
483                   f"正向透射率 = {result['max_forward']:.2f}, "
484                   f"反向透射率 = {result['max_backward']:.2f}, "
485                   f"对应正向相位 = {result['forward_phase']:.2f} rad, "
486                   f"对应反向相位 = {result['backward_phase']:.2f} rad, "
487                   f"单元参数 (r, h) = ({unit_r:.2f} mm, {unit_h:.2f} mm), "
488                   f"FOM最优值 = {result['FOM']:.6f}")
489     else:
490         print("未找到符合条件的全局最优结果。")
491
492 else:
493     print("已选择跳出相位识别部分，不进行筛选。")

```

实验换算及绘图代码：

```

1  import numpy as np
2  import h5py
3  import glob
4  import os
5  import matplotlib.pyplot as plt
6
7  # 定义函数调整相位到 [- , ] 范围内
8  def adjust_phase(phase):
9      while phase > np.pi:
10         phase -= 2 * np.pi # 超出范围向下调整

```

```

11     while phase < -np.pi:
12         phase += 2 * np.pi # 超出范围向上调整
13     return phase
14
15 # 向量化adjust_phase函数
16 adjust_phase_vec = np.vectorize(adjust_phase)
17
18 # 文件夹路径
19 folder_path = r'C:\Users\14716\Desktop\Magnetic PC Metasurface\正方晶格单向传输超表面\M0_Meta'
20
21 # 获取文件夹中所有的.mat文件
22 file_names = glob.glob(os.path.join(folder_path, '*.mat'))
23
24 # 存储频率和透射系数的字典
25 data_dict = {}
26
27 for file_name in file_names:
28     # 读取.mat文件
29     with h5py.File(file_name, 'r') as mat_data:
30         # 假设频率和透射系数的变量名为 'Frequency_Hz' 和 'Spectrum'
31         frequency = np.array(mat_data['Frequency_Hz']).flatten() # 转换为一维数组
32         transmission_coefficient = np.array(mat_data['Spectrum'])
33
34         # 提取实部和虚部，计算模
35         if transmission_coefficient.dtype.names: # 检查是否是结构数组
36             real_part = transmission_coefficient['real']
37             imag_part = transmission_coefficient['imag']
38             transmission_coefficient = np.sqrt(real_part**2 + imag_part**2) # 计算模
39
40         # 将结果存储到字典中
41         data_dict[file_name] = {

```

```

42         'frequency': frequency,
43         'transmission_coefficient': transmission_coefficient,
44         'real_part': real_part,
45         'imag_part': imag_part,
46         f'T_{os.path.basename(file_name)}': real_part + 1j * imag_part
47     }
48
49 # 提取特定文件的数据
50 Tpp_fw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-pp-forward.mat']['T_L=37cm-pp-forward.mat']
51 Tps_fw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-ps-forward.mat']['T_L=37cm-ps-forward.mat']
52 Tsp_fw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-sp-forward.mat']['T_L=37cm-sp-forward.mat']
53 Tss_fw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-ss-forward.mat']['T_L=37cm-ss-forward.mat']
54
55 Tpp_bw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-pp-backward.mat']['T_L=37cm-pp-backward.mat']
56 Tps_bw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-ps-backward.mat']['T_L=37cm-ps-backward.mat']
57 Tsp_bw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-sp-backward.mat']['T_L=37cm-sp-backward.mat']
58 Tss_bw = data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
        Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-ss-backward.mat']['T_L=37cm-ss-backward.mat']
59
60 # 计算圆偏振透射系数
61 Trr_fw = (Tpp_fw + Tss_fw) + 1j * (Tps_fw - Tsp_fw)
62 Trl_fw = (Tpp_fw - Tss_fw) - 1j * (Tps_fw + Tsp_fw)
63 Tlr_fw = (Tpp_fw - Tss_fw) + 1j * (Tps_fw + Tsp_fw)
64 Tll_fw = (Tpp_fw + Tss_fw) - 1j * (Tps_fw - Tsp_fw)

```

```

65
66 Trr_bw = (Tpp_bw + Tss_bw) + 1j * (Tps_bw - Tsp_bw)
67 Trl_bw = (Tpp_bw - Tss_bw) - 1j * (Tps_bw + Tsp_bw)
68 Tlr_bw = (Tpp_bw - Tss_bw) + 1j * (Tps_bw + Tsp_bw)
69 Tll_bw = (Tpp_bw + Tss_bw) - 1j * (Tps_bw - Tsp_bw)
70
71 #计算模和相位
72 circular_magnitudes_fw = {
73     'Trr_fw': np.abs(Trr_fw),
74     'Trl_fw': np.abs(Trl_fw),
75     'Tlr_fw': np.abs(Tlr_fw),
76     'Tll_fw': np.abs(Tll_fw),
77 }
78
79 circular_phases_fw = {
80     'Trr_fw': np.angle(Trr_fw),
81     'Trl_fw': np.angle(Trl_fw),
82     'Tlr_fw': np.angle(Tlr_fw),
83     'Tll_fw': np.angle(Tll_fw),
84 }
85
86 circular_magnitudes_bw = {
87     'Trr_bw': np.abs(Trr_bw),
88     'Trl_bw': np.abs(Trl_bw),
89     'Tlr_bw': np.abs(Tlr_bw),
90     'Tll_bw': np.abs(Tll_bw),
91 }
92
93 circular_phases_bw = {
94     'Trr_bw': np.angle(Trr_bw),
95     'Trl_bw': np.angle(Trl_bw),

```

```

96  'Tlr_bw': np.angle(Tlr_bw),
97  'Tll_bw': np.angle(Tll_bw),
98  }
99
100 #将结果存储到字典中
101 data_dict['C:\\Users\\14716\\Desktop\\Magnetic PC
      Metasurface\\正方晶格单向传输超表面\\M0_Meta\\L=37cm-pp-forward.mat'].update({
102  'circular_magnitudes_fw': circular_magnitudes_fw,
103  'circular_phases_fw': circular_phases_fw,
104  'circular_magnitudes_bw': circular_magnitudes_bw,
105  'circular_phases_bw': circular_phases_bw,
106  })
107
108 #绘制圆偏振透射谱
109 plt.figure(figsize=(10, 6), dpi=400)
110 for file_name, data in data_dict.items():
111     if 'circular_magnitudes_fw' in data and 'circular_magnitudes_bw' in data:
112         # 过滤频率，只保留6GHz到15GHz的部分
113         mask = (data['frequency'] >= 6e9) & (data['frequency'] <= 15e9)
114         plt.plot(data['frequency'][mask], data['circular_magnitudes_fw']['Trr_fw'][mask],
115                 label="Trr_fw", color="blue")
116         plt.plot(data['frequency'][mask], data['circular_magnitudes_bw']['Trr_bw'][mask],
117                 label="Trr_bw", color="red")
118
119 plt.xlabel('Frequency (Hz)')
120 plt.ylabel('Circular Polarized Transmission Coefficient (Magnitude)')
121 plt.title('Circular Polarized Transmission Coefficients (Forward and Backward)')
122 plt.legend(loc='upper right') # 将图例放在右上角
123 plt.grid(True)
124 plt.xlim(6e9, 15e9) # 设置x轴范围为6GHz到15GHz
125 plt.show()

```

```

124 #绘制相位差
125 plt.figure(figsize=(10, 6), dpi=400)
126 for file_name, data in data_dict.items():
127     if 'circular_phases_fw' in data and 'circular_phases_bw' in data:
128         # 过滤频率, 只保留6GHz到15GHz的部分
129         mask = (data['frequency'] >= 6e9) & (data['frequency'] <= 15e9)
130         plt.plot(data['frequency'][mask], adjust_phase_vec(data['circular_phases_fw']['Trr_fw'][mask]
131             - data['circular_phases_bw']['Trr_bw'][mask]), label="Trr_fw Phase - Trr_bw Phase",
132             color="green")
131 plt.xlabel('Frequency (Hz)')
132 plt.ylabel('Phase Difference (radians)')
133 plt.title('Phase Difference Between Forward and Backward Circular Polarized Transmission
134           Coefficients')
134 plt.legend(loc='upper right') # 将图例放在右上角
135 plt.grid(True)
136 plt.xlim(6e9, 15e9) # 设置x轴范围为6GHz到15GHz
137 plt.show()

```

Unidirectional Light Propagation in Magnetic Metasurfaces

Shangrun Lu¹⁾

1) (Sun Yat-sen University, School of Physics, Guangzhou 510275, China)

Abstract

This study aims to construct a bidirectional phase-tunable non-reciprocal metasurface. Based on the existing materials yttrium iron garnet (YIG) and permanent magnets in the laboratory, we utilize non-reciprocal effects such as magnetic circular dichroism to achieve unidirectional propagation and phase control of electromagnetic waves. We analyzed the key factors in the design of non-reciprocal meta-atoms, especially the relationship between phase and meta-atom parameters, and explored the mechanism of controlling electromagnetic wave propagation through surface impedance. The results of this study provide new insights for achieving non-reciprocal metasurfaces with arbitrary polarization using entirely passive and linear methods, and have guiding significance for the further development of efficient non-reciprocal devices.

Keywords: Non-reciprocal metasurfaces, magnetic materials, unidirectional propagation