



Setting up meshes of interferometers – reversed local light interference method

DAVID A. B. MILLER

Ginzton Laboratory, Spilker Building, Stanford University, 348 Via Pueblo Mall, Stanford, CA 94305-4088, USA

*dabm@ee.stanford.edu

Abstract: Many interesting linear optical networks, such as lattice filters and some interferometer meshes, are difficult to fabricate precisely and cannot be configured progressively even using recent algorithms. Our approach allows a broad category of optical networks to be set up progressively and automatically, including correcting for fabrication imprecision. We null interference locally in the network based on inputs calculated by considering the network operated in reverse. Calibration is only required for the network inputs, not for individual components (though this method can also calibrate those). We illustrate specific cases of lattice filters and rectangular meshes of interferometers, and we expect the approach can be applied broadly to networks in which the light only propagates forward in the network.

© 2017 Optical Society of America

OCIS codes: 130.0130) Integrated optics; (250.5300) Photonic integrated circuits; (270.0270) Quantum optics.

References and links

1. D. A. B. Miller, "Sorting out light," *Science* **347**(6229), 1423–1424 (2015).
2. D. J. Richardson, J. M. Fini, and L. E. Nelson, "Space-division multiplexing in optical fibres," *Nat. Photonics* **7**, 354–362 (2013).
3. P. J. Winzer, "Making spatial multiplexing a reality," *Nat. Photonics* **8**, 345–348 (2014).
4. A. Annoni, E. Guglielmi, M. Carminati, G. Ferrari, M. Sampietro, D. A. B. Miller, A. Melloni, and F. Morichetti, "Unscrambling light – automatically undoing strong mixing between modes," *Light Sci. Appl.* **6**, e17110 (2017).
5. Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljacic, "Deep Learning with Coherent Nanophotonic Circuits," *Nat. Photonics* **11**, 441–446 (2017).
6. J. Carolan, C. Harrold, C. Sparrow, E. Martín-López, N. J. Russell, J. W. Silverstone, P. J. Shadbolt, N. Matsuda, M. Oguma, M. Itoh, G. D. Marshall, M. G. Thompson, J. C. F. Matthews, T. Hashimoto, J. L. O'Brien, and A. Laing, "Universal linear optics," *Science* **349**(6249), 711–716 (2015).
7. J. Mower, N. C. Harris, G. R. Steinbrecher, Y. Lahini, and D. Englund, "High-fidelity quantum state evolution in imperfect photonic integrated circuits," *Phys. Rev. A* **92**, 032322 (2015).
8. N. M. Estakhri, B. E. Edwards, and N. Engheta, "Solving Integral Equations with Optical Metamaterial-Waveguide Networks," in *Conference on Lasers and Electro-Optics, OSA Technical Digest (online)* (Optical Society of America, 2017).
9. N. C. Harris, G. R. Steinbrecher, J. Mower, Y. Lahini, M. Prabhu, D. Bunandar, C. Chen, F. N. C. Wong, T. Baehr-Jones, M. Hochberg, S. Lloyd, and D. Englund, "Quantum transport simulations in a programmable nanophotonic processor," *Nat. Photonics* **11**, 447–452 (2017).
10. C. K. Madsen and J. H. Zhao, *Optical Filter Design and Analysis* (Wiley, 1999).
11. J. M. Wyrwas and M. C. Wu, "Dynamic Range of Frequency Modulated Direct-Detection Analog Fiber Optic Links," *J. Lightwave Technol.* **27**, 5552–5562 (2009).
12. D. Marpaung, C. Roeloffzen, R. Heideman, A. Leinse, S. Sales, and J. Capmany, "Integrated microwave photonics," *Laser Photonics Rev.* **7**, 506–538 (2013).
13. L. Zhuang, C. G. H. Roeloffzen, M. Hoekman, K.-J. Boller, and A. J. Lowery, "Programmable photonic signal processor chip for radiofrequency applications," *Optica* **2**, 854–859 (2015).
14. D. Pérez, I. Gasulla, J. Capmany, and R. A. Soref, "Reconfigurable lattice mesh designs for programmable photonic processors," *Opt. Express* **24**(11), 12093–12106 (2016).
15. T. M. Shay, "Theory of electronically phased coherent beam combination without a reference beam," *Opt. Express* **14**(25), 12188–12195 (2006).
16. D. A. B. Miller, "Self-aligning universal beam coupler," *Opt. Express* **21**(5), 6360–6370 (2013).
17. D. A. B. Miller, "Self-configuring universal linear optical component," *Photon. Res.* **1**, 1–15 (2013).

18. D. A. B. Miller, "Establishing optimal wave communication channels automatically," *J. Lightwave Technol.* **31**, 3987–3994 (2013).
19. D. A. B. Miller, "Reconfigurable add-drop multiplexer for spatial modes," *Opt. Express* **21**(17), 20220–20229 (2013).
20. D. A. B. Miller, "Perfect optics with imperfect components," *Optica* **2**, 747–750 (2015).
21. A. Ribeiro, A. Ruocco, L. Vanacker, and W. Bogaerts, "Demonstration of a 4×4-port universal linear circuit," *Optica* **3**, 1348–1357 (2016).
22. C. M. Wilkes, X. Qiang, J. Wang, R. Santagati, S. Paesani, X. Zhou, D. A. B. Miller, G. D. Marshall, M. G. Thompson, and J. L. O'Brien, "60 dB high-extinction auto-configured Mach-Zehnder interferometer," *Opt. Lett.* **41**(22), 5318–5321 (2016).
23. D. A. B. Miller, L. Zhu, and S. Fan, "Universal modal radiation laws for all thermal emitters," *Proc. Natl. Acad. Sci. U.S.A.* **114**(17), 4336–4341 (2017).
24. M. Reck, A. Zeilinger, H. J. Bernstein, and P. Bertani, "Experimental Realization of Any Discrete Unitary Operator," *Phys. Rev. Lett.* **73**(1), 58–61 (1994).
25. W. R. Clements, P. C. Humphreys, B. J. Metcalf, W. S. Kolthammer, and I. A. Walmsley, "Optimal design for universal multiport interferometers," *Optica* **3**, 1460–1465 (2016).
26. S. Grillanda, F. Morichetti, N. Peserico, P. Ciccarella, A. Annoni, M. Carminati, and A. Melloni, "Non-Invasive Monitoring of Mode-Division Multiplexed Channels on a Silicon Photonic Chip," *J. Lightwave Technol.* **33**, 1197–1201 (2015).

1. Introduction

With increasingly sophisticated technology for making complex integrated optical circuits, such as silicon photonics or other integrated platforms, there are growing opportunities for novel linear optical functions [1]. Meshes of two-beam interferometers, such as Mach-Zehnder interferometers (MZIs), can address a wide range of applications including telecommunications mode-division (de)multiplexing [2–4]; linear optical networks for neural [5], quantum [6,7], and general linear information processing [8,9]; optical filters [10,11]; microwave photonics for analog signal processing [12–14]; laser beam power combining [15,16]; self-aligning beam couplers [16]; and self-configuring optics generally [1,4,16–22]. The concept of such universal linear optical components [17] also allows proofs of novel fundamental laws [23].

Though we can calculate interferometer settings based on some design [17,24,25], calibrating components like beam splitters and phase shifters inside a large mesh can be a complicated task that might have to be repeated as components drift. Algorithms have been proposed [16–18,20] and demonstrated [4,21,22] that can set up certain classes of meshes. Those meshes can be progressively configured after fabrication, including correction of imperfections [20,22], without having to separately calibrate the components inside the mesh, and these procedures can also be used to calibrate mesh components.

These algorithms work in an existing known mesh architecture for unitary transforms [6,16,17,24], in cascaded binary trees [16], and an optimal non-unitary architecture [17]. They operate by successive single-parameter power minimizations or maximization with training inputs. They work most flexibly with embedded monitoring detectors, though external detectors can be used [16,17,20]. Related progressive algorithms allow phase shifter calibrations in meshes with diagonal rows each with external detectors [6,9]. Any linear mapping can be accomplished using meshes of MZIs built with 50:50 beamsplitters [6,17], and a recent extension allows "perfect" meshes to be configured even when fabricated beam splitter ratios are far from 50:50 (up to as bad as 85:15) [20,22]. Such architectures can even continuously adapt to changing problems or component parameter drifts [4,16–19].

Some other mesh architectures, such as lattice filters for wavelength filtering [10,11], a "compacted" unitary transform mesh [25], and possibly mesh architectures for microwave photonics applications [13,14], cannot use these previous algorithms, however.

Here, using a "Reversed Local Light Interference Method" (RELLIM), we show how much broader categories of mesh architectures can be both "perfected" and progressively configured. We expect this method can work for any mesh of connected two-beam interferometers in which the light only flows in one sense – "forward-only" optics – or at least

that, during some setup or calibration, the mesh can be arranged to be “forward-only”, as may be the case for some microwave photonics architectures [13,14].

2. Concept of the RELIM approach

Consider first just one two-beam interferometer, such as an MZI (Fig. 1) (other approaches, such as controllable directional couplers, could also be used). We need to control a relative phase delay ϕ in the “Top” arm relative to the “Left” arm at the input of the interferometer, and, using a controllable relative phase delay θ in the upper branch in an MZI, the split ratio between the “Right” and “Bottom” outputs, as in the phase shifters ϕ and θ in Fig. 1. We also show detectors DR and DB in the “Right” and “Bottom” outputs (though we only need one of these). For the moment, the beamsplitters are presumed to have perfect 50:50 beamsplitting ratios. The detectors sample only some negligible fraction of the power passing through them, sufficient to run “slow” feedback loops during setup. Such “mostly-transparent” detectors can be fabricated in various ways, including simple power taps to external detectors [21] or capacitively coupled detection [4,26].

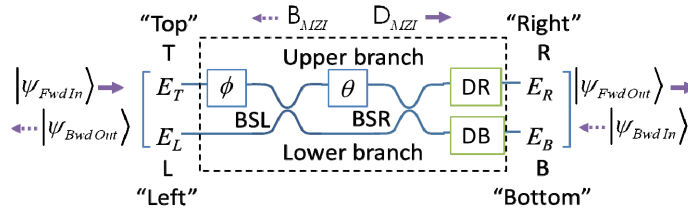


Fig. 1. A waveguide Mach-Zehnder interferometer block (inside dashed line), showing phase shifters ϕ in the “Top” input arm and θ in the upper interferometer branch, beamsplitters BSL and BSR, and “mostly-transparent” detectors DR and DB. The labelling of the ports as “Top”, “Left”, “Right” and “Bottom” corresponds with ports on a simple cube beamsplitter. The “backward” vectors, indicated with dashed arrows, are only calculated beams from a hypothetical specific backwards input.

We presume forward-going wave amplitudes E_T and E_L entering and E_R and E_B exiting (Fig. 1). We describe the MZI with a “device” matrix D_{MZI} acting on a vector $|\psi_{FwdIn}\rangle$ of forward-going input amplitudes to give a vector $|\psi_{FwdOut}\rangle$ of forward-going output amplitudes, using a Dirac notation for convenience,

$$|\psi_{FwdIn}\rangle \equiv \begin{bmatrix} E_T \\ E_L \end{bmatrix}, \quad |\psi_{FwdOut}\rangle \equiv \begin{bmatrix} E_R \\ E_B \end{bmatrix} \quad (1)$$

One possible unitary (and lossless) matrix D_{MZI} would be

$$D_{MZI} = i \exp\left(i \frac{\theta}{2}\right) \begin{bmatrix} \exp(i\phi) \sin(\theta/2) & \cos(\theta/2) \\ \exp(i\phi) \cos(\theta/2) & -\sin(\theta/2) \end{bmatrix} \quad (2)$$

(Other matrices are possible depending on the precise behavior of the beam splitters with regard to phase.) Hence,

$$|\psi_{FwdOut}\rangle \equiv \begin{bmatrix} E_R \\ E_B \end{bmatrix} = D_{MZI} \begin{bmatrix} E_T \\ E_L \end{bmatrix} \equiv D_{MZI} |\psi_{FwdIn}\rangle \quad (3)$$

We want to set up the MZI with values of ϕ and θ to implement the desired “forward-going” device matrix D_{MZI} , based only in minimizing or maximizing power in DR or DB and without separately calibrating components. One key to this is to imagine that we shine power backwards into just one of the “output” ports.

We presume for generality that this matrix could represent a device with N input and output ports, so the vectors have N elements. In matrix D , element d_{mn} gives the (complex) forward amplitude in output waveguide m on the “right” for unit forward amplitude in input waveguide n on the “left”. We presume the optical device is reciprocal. So, if we were instead to shine unit amplitude backwards into “output” waveguide m on the “right”, the amplitude of the backward wave emerging from waveguide n on the “left” would be the (complex) number d_{mn} ; both the magnitude (e.g., the amplitude “attenuation”) and the phase delay are the same in both directions when coupling between specific ports in a reciprocal system. For the matrix B describing this backwards mapping from a vector $|\psi_{BwdIn}\rangle$ of backwards wave amplitudes in the “output” waveguides on the “right” to a vector $|\psi_{BwdOut}\rangle$ of backwards wave amplitudes in the “input” waveguides on the “left”, that is,

$$|\psi_{BwdOut}\rangle = B|\psi_{BwdIn}\rangle \quad (4)$$

then the matrix elements of B are given by $b_{nm} = d_{mn}$. Hence B is the transpose of D , or equivalently the complex conjugate of the Hermitian adjoint of D , that is $B = D^T \equiv (D^\dagger)^*$. Hence

$$|\psi_{BwdOut}\rangle = (D^\dagger)^* |\psi_{BwdIn}\rangle \quad (5)$$

Suppose we shine a forward wave $|\psi_{FwdInPC}\rangle$ in on the left, with amplitudes equal to the complex conjugate (the “phase conjugate”) $|\psi_{BwdOut}\rangle^*$ of this calculated vector $|\psi_{BwdOut}\rangle$. Then the vector of forward wave amplitudes emerging on the right is

$$\begin{aligned} |\psi_{FwdOutPC}\rangle &= D|\psi_{FwdInPC}\rangle = D|\psi_{BwdOut}\rangle^* \\ &= D\left[(D^\dagger)^* |\psi_{BwdIn}\rangle\right]^* = DD^\dagger |\psi_{BwdIn}\rangle^* = |\psi_{BwdIn}\rangle^* \end{aligned} \quad (6)$$

In Eq. (6), since D is unitary, DD^\dagger is just the identity matrix.

Now, imagine shining a beam backwards just into one port on the right – say, the “Right” port. Because we know the design we want, we know the settings ϕ and θ we want, and so we can readily calculate the amplitudes $|\psi_{BwdOut}\rangle$ that would ideally emerge on the “left” from the “input” ports in such a case. So, now we shine actual beams $|\psi_{FwdInPC}\rangle = |\psi_{BwdOut}\rangle^*$ into the input ports. We expect that, at first, our interferometer is not set correctly. But, by successive maximizations of DR power (or minimizations of DB power) [16,17,20], first with respect to ϕ and then with respect to θ , then all the power will emerge from the “Right” port, and we will have set the correct, desired ϕ and θ .

So, from the “reversed” problem, we deduced the input amplitudes for such local light cancellation if the device was set correctly, and then used that set of actual inputs to set up the device by local interference, hence the name RELLIM. This approach requires one well-calibrated set of sources – an “optical setup machine” (OSM) – to drive the correct amplitudes and phases into the inputs on the left. In Appendix A, we discuss such an OSM.

For one two-beam interferometer, we could calibrate it directly, of course. The power of this approach comes with larger networks. We transfer the calibration of the OSM progressively into all the interferometers in the mesh, including those “buried” inside it, either to set some specific mesh configuration, or to calibrate each interferometer. We illustrate this approach for two classes of networks – a linear chain and a rectangular mesh – that otherwise are not easily configured progressively.

Incidentally, if D can be described as a unitary matrix multiplied by a constant, as would be the case for uniform loss on all paths through the device (e.g., from waveguide propagation loss on equal length paths), then the basic method described here will also work. Then DD^\dagger is just the identity matrix multiplied by a constant, and we would have a constant multiplying the right hand side of Eq. (6). The overall amplitude of that vector, however, makes no difference to the conditions for interference cancellation, and hence for the phase settings of the interferometers. Hence, as long as all paths that are to interfere in an interferometer have experienced the same background loss, this technique will still work. Several recent successful experimental demonstrations of operation of large meshes of interferometers (e.g., [4,5,9]) are designed based on the assumption that losses in the waveguides and MZIs do not significantly impair the system. Harris et al. [9] expect, based on the very high “visibility” (or extinction ratio) of their MZIs, that losses are uniform across MZIs and are minimal for each MZI, and that, since waveguide loss per unit length appears to be quite uniform across a wafer, propagation loss on paths of equal length is likely to be very similar. Hence, this assumption of unitarity or unitarity within a factor appears to be a reasonable and useful starting point for designing such systems.

3. Configuring a lattice filter

In a lattice filter [10,11] (Fig. 2), multiple two beam interferometers are cascaded in a simple chain or ladder with a fixed time delay τ in the upper branches between the interferometers. The detailed design of such a filter leads to specific desired split ratios and additional relative phase delays in each of the two-beam interferometer blocks j , labelled with parameters θ_j and ϕ_j , respectively.

Fabricating fixed beamsplitters with the desired split ratios and the additional required phase delays is difficult, and the actual fabricated values may not be known [11]. Now, however, we can configure all the split ratios and phase delays progressively.

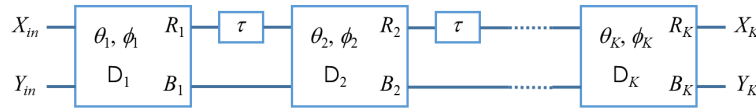


Fig. 2. Lattice filter of K two-beam interferometers, shown as blocks, each with split ratio and relative phase parameters θ_j and ϕ_j , respectively, and with time delays τ between the blocks.

We presume we can send signals, at some chosen wavelength λ , with any chosen relative amplitudes and phases, into the inputs X_{in} and Y_{in} . We also presume we have already designed the filter, which means we know the intended device matrices D_j ; equivalently, we know the desired values of θ_j and ϕ_j . For simplicity, we presume that the delay τ is an integer number of cycles of the frequency corresponding to λ , though ultimately this will not matter. Then the setup of the interferometers proceeds in the following steps.

- 1) Knowing the desired matrix D_1 from our design, we calculate what would be the relative amplitudes and phases at X_{in} and Y_{in} if we were to imagine shining unit amplitude of light at wavelength λ backwards only into the port R_1 . Specifically, we would have, for those backwards amplitudes,

$$\begin{bmatrix} X_{inBwd1} \\ Y_{inBwd1} \end{bmatrix} = (D_1^\dagger)^* \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (7)$$

As discussed above, we shine in forward amplitudes

$$\begin{bmatrix} X_{in1} \\ Y_{in1} \end{bmatrix} = \begin{bmatrix} X_{inBwd1}^* \\ Y_{inBwd1}^* \end{bmatrix} \quad (8)$$

Then we adjust first the parameter ϕ_1 and then parameter θ_1 , maximizing the DR power (or minimizing the DB power) in each case. Then all the power emerges from the “Right” port of Block 1, which now implements the desired operation D_1 . Note that choosing the X and Y inputs as in Eq. (8) simply means that we choose the magnitude of the ratio $|X_{in1} / Y_{in1}| = |X_{inBwd1} / Y_{inBwd1}|$ and, if the calculated phase of X_{inBwd1} was leading the phase of Y_{inBwd1} by some amount ϕ , then the phase of X_{in1} should lag the phase of Y_{in1} by an amount ϕ . Equivalently,

$$\begin{bmatrix} X_{in1} \\ Y_{in1} \end{bmatrix} = D_1^\dagger \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (9)$$

- 2) For Block 2, we imagine shining unit power backwards only into port R_2 . Since we know the matrix D_2 from our design, a correctly configured device, would give backwards amplitudes

$$\begin{bmatrix} X_{inBwd2} \\ Y_{inBwd2} \end{bmatrix} = (D_1^\dagger)^* (D_2^\dagger)^* \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (10)$$

So, now we should shine inputs

$$\begin{bmatrix} X_{in2} \\ Y_{in2} \end{bmatrix} = \begin{bmatrix} X_{inBwd2}^* \\ Y_{inBwd2}^* \end{bmatrix} = D_1^\dagger D_2^\dagger \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (11)$$

on the left. Leaving the Block 1 set to implement D_1 , now we set up Block 2 to implement matrix D_2 ; we first adjust the parameter ϕ_2 to minimize the DB power (or maximize the DR power) inside Block 2 and then similarly adjust parameter θ_2 , setting Block 2 correctly.

- 3) Now we proceed similarly through the remaining Blocks, completing the setup. For the p th Block, we would shine inputs

$$\begin{bmatrix} X_{inp} \\ Y_{inp} \end{bmatrix} = D_1^\dagger D_2^\dagger \cdots D_p^\dagger \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (12)$$

and similarly adjust ϕ_p and θ_p to minimize the DB power or maximize the DR power in Block p . Note we only required one calibrated set of input drivers, and no calibration of the components within the Blocks. (Note it does not actually matter if τ is an integer number of cycles. Any fractional cycle is automatically compensated as if it were incorporated in the phase shifter inside the block.)

We expect similar approaches can be applied to other optical filter designs [10] in which the light only flows in one direction in any given waveguide, thereby allowing a progressive approach.

4. Configuring a two-dimensional mesh

The two-dimensional mesh example of Fig. 3 implements a 5×5 unitary matrix with a rectangular mesh of 2×2 interferometers as in [25]. Now, this system can be configured progressively by setting up the first column of blocks, for example from top to bottom in sequence, and then proceeding to the next column of blocks. In each case, we pretend we are shining a unit backwards beam into a specific port R_{ij} . Again, because we presume we know the design, and hence all the desired elements in the 2×2 matrices D_{ij} , we can calculate the

resulting vector of backwards outputs $|\psi_{BwdOut}\rangle$ that would emerge on the left if the system were set correctly.

Formally, the m th column of blocks can be represented by 5×5 block-diagonal matrices. Then we have an equation like Eq. (12) for setting elements in the p th column of blocks, with the difference that the input and output vectors have 5 elements, not 2. The column vector on the right will have 5 elements, not 2, with a single 1 in the appropriate place for each “Right” output in the corresponding block, as required. This approach lets us calculate the column vector

$$|\psi_{FwdIn}\rangle = |\psi_{BwdOut}\rangle^* \equiv [E_{1T} \ E_{1B} \ E_{2T} \ E_{2B} \ E_{3T}]^T \quad (13)$$

to feed in on the left when setting ϕ and θ in each block by power maximization or minimization in the appropriate detector. Progressing through all the blocks this way sets up the entire mesh. In Appendix B, we show an explicit alignment sequence and a matrix way of envisaging the required calculations

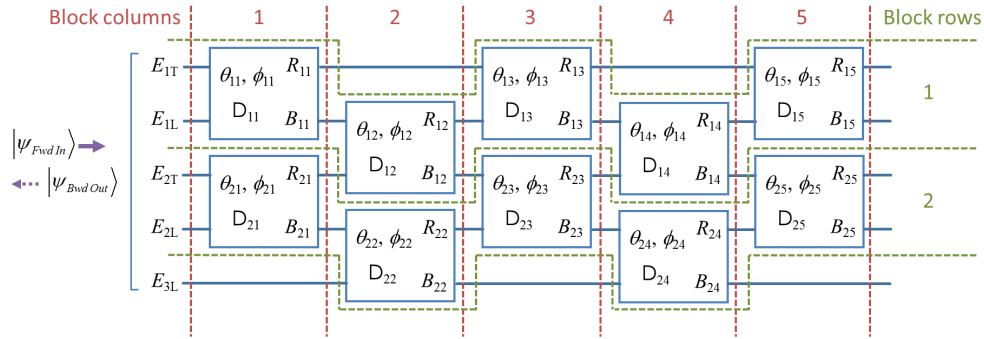


Fig. 3. An example two-dimensional mesh that can implement an arbitrary unitary transform between inputs and outputs according to the scheme of Ref [25], shown here for five inputs and outputs. Individual blocks correspond to the MZI block of Fig. 1, and they are indexed here according to their block “row” i and block “column” j , as in the parameters θ_{ij} and ϕ_{ij} , and the resulting matrix D_{ij} .

5. Discussion

We expect that this kind of progressive approach can be applied quite generally to networks in which the light only flows in a “forward” direction. A very broad category of meshes can, for example, be “embedded” in a rectangular mesh, with just some blocks used actively and the others left in their “bar” or “cross” states, so the proof here of configurability of the rectangular mesh applies automatically to all such “embeddable” meshes.

Using this method does require some additional complexity in the fabricated system, adding some setup network to generate the necessary input vectors (Appendix A). For a mesh representing an $N \times N$ matrix, an additional N MZIs are required; as N becomes large, however, this becomes relatively a negligible increase in complexity. It is arguable that any optical method to set up such a network will require some additional comparable complexity somewhere at the inputs during setup. In one sense, too, the method proposed here is only exactly as complicated as needed to perform its task; only one optical operation is required for each variable that has to be set in the network.

Power sampling detectors are also required for each MZI in this method. Those certainly add some circuit and fabrication complexity for each MZI block, though this has not prevented demonstrations with such sampling [4,21]. A legitimate question is whether such sampling would itself be a significant source of loss in the system, even unbalancing the unitarity (within a factor) of the network. However, since this power sampling is only used

for the setup of the network, and that process need not be particularly fast, very little power need be sampled. For example, we could envisage setting up the network using input optical powers in the range of 100's of μW or higher. Since it is relatively straightforward to detect nanowatt optical powers at low speeds using simple detectors, and even lower powers can be detected routinely, only very small sampling fractions – e.g., 10^{-3} or less – are required. Such numbers can be less than the background loss in the other components of the network. The detection used in [4] requires no additional loss at all since it relies on photoconduction from background absorption that is already taking place in the waveguides.

The fact that light does not flow back to any block that we have already set up is key to allowing a progressive algorithm. That requirement means that Fabry-Perot resonators should be avoided in the network because they generally will have back reflection. Ring resonators in some configurations, such as a ring evanescently coupled to just one waveguide, do not necessarily have significant back reflections, so progressive algorithms may also be possible for such cases.

So far, we presumed the individual two-beam interferometers are perfect, as in MZIs with perfect 50:50 beamsplitters, though such perfect fabrication is difficult. We can, however, “perfect” such interferometers if there is a way to trim the beamsplitter reflectivities [20]. One way to allow trimming substitutes further MZIs for each of the beamsplitters in a given MZI [20]; then, the fabricated fixed beamsplitters inside those MZIs can have actual split ratios anywhere from 15:85 to 85:15. The embedded detectors in the blocks here make it straightforward to add such a progressive “perfecting” operation [20] to the present meshes. See Appendix C.

6. Conclusions

We have shown that, by exploiting a new configuration algorithm, with embedded monitoring detectors, broad categories of linear optical networks can be perfected [20], set up and/or calibrated progressively, based only on calibrated and controlled inputs. Specifically, this method can be applied generally to “forward-only” networks in which the light only flows in one direction. It will work for loss-less networks and those in which the loss is substantially equal on all interfering paths, as is the case for a wide range of MZI networks.

Appendix A – an optical setup machine

As discussed in the main text, to run this approach, we need to be able to construct specific calibrated input vectors of “complex” amplitudes (magnitude and phase) of the beams in the input waveguides to the mesh to be configured. This requires that we can construct and calibrate what we call an optical setup machine (OSM) that must have several key attributes:

- i. we have to be able to calibrate this OSM itself;
- ii. it must be able to create all the necessary relative magnitudes and phases of inputs for the setup of each interferometer in the mesh network – this is the machine’s basic function – and we must be able to calculate how to set it up to generate any vector;
- iii. we need to be able to use it to set up the mesh to be configured
- iv. we need to be able to turn it “off”, allowing the actual inputs in the waveguides to flow through to the network without being disturbed by the OSM.

An architecture of machine that can satisfy all these requirements is shown in Fig. 4. We will describe how we satisfy each of these four attributes, in corresponding subsections (i) to (iv).

We use the example of a machine intended for the 5-input mesh as in Fig 3, though the extension to larger (or smaller) machines for programming meshes with different numbers of inputs is straightforward, just requiring a number of MZIs in the chain that corresponds to the number of inputs to the mesh.

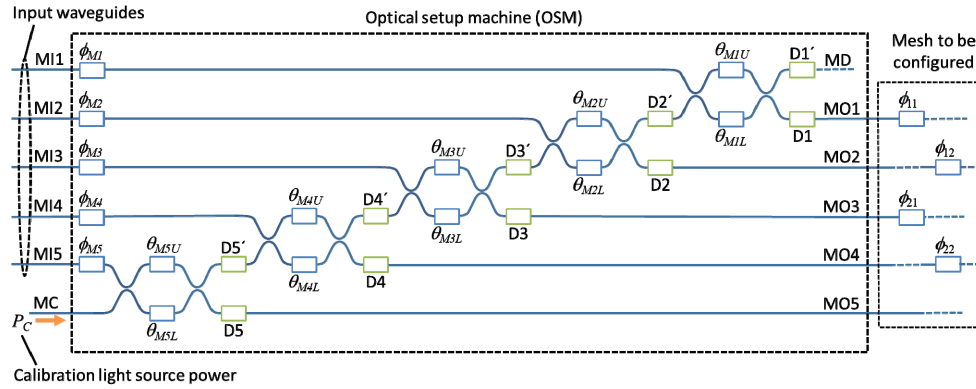


Fig. 4. An OSM for configuring an MZI mesh, using a chain of MZIs (MZ1 to MZ5), for the example of a mesh with 5 inputs. Each MZI has phase shifters on both the upper (θ_{uj}) and lower (θ_{lj}) arms. A set of 5 mostly-transparent detectors is placed either on the “lower” output arms (shown as D1 to D5), or, alternatively, on the “upper” output arms (shown as D1' to D5'). Phase shifters ϕ_{M1} to ϕ_{M5} are provided for each of the input guides. The first phase shifters inside the mesh that is to be configured, here ϕ_{11} to ϕ_{22} , are also shown. The output waveguides, MO1 to MO5, form the input waveguides for the mesh to be configured. During the first parts of the process to calibrate the mesh with an external light source, power P_C is shone into the bottom waveguide MC, but no power is shone into the guides MI1 to MI5. During actual use of the mesh to perform its desired operation, all the MZIs are set to their “cross” state, and the input light of interest is shone into input waveguides MI1 to MI5. The set of phase shifters ϕ_{M1} to ϕ_{M5} are provided for phase equalization or compensation of the inputs, and these are set in a final part of the calibration procedure.

This OSM is basically the first diagonal “row” of a “triangular” MZI network, as in the self-aligning beam coupler [16], though here it is run backwards compared to its use there. Here, at least in the first calibration and mesh setup approach we will discuss, we feed power P_C into only the bottom input guide MC. The sets of phase shifters on both the upper (θ_{uj}) and lower (θ_{lj}) arms of the MZIs give sufficient controls to set the relative amplitudes and phases of the resulting outputs in waveguides MO1 to MO5. Changing the relative phase between the two arms of the MZI over a range of $\pm\pi$ – e.g., by driving the two phase shifters in opposite directions, in a “differential-mode” drive – allows a full range of split ratios, equivalent to changing the output from one arm to the other and back again, and controlling an additional common-mode phase (i.e., a phase that is added or subtracted equally in both arms) in MZj changes the phase in the corresponding output waveguide MOj. (In fact, a range either of 0 to π or of $-\pi$ to 0 would be sufficient; each such allows a full range of magnitudes of split ratio, but with a different sign in each case.)

(i) Calibration of the OSM

We can perform this calibration using power minimization and monitoring in (mostly-transparent) detectors D1 to D5 in the output waveguide paths MO1 to MO5, or, alternatively, using power maximization and monitoring in (mostly-transparent) detectors D1' to D5' positioned in the waveguides linking the MZIs. We only need one or other of these sets of detectors; which detector positions we use depends on whether we care more about minimizing any loss on the signal, which would favor the use of the detectors in the positions D1' to D5' with power maximizations, or if we care more about minimizing loss along the path through the chain MZ1 to MZ5 for simpler power calibrations, in which case we would use detectors in the positions D1 to D5 with power minimizations. We do not need explicitly to calibrate these detectors (for example, we do not need to know the absolute power

corresponding to any given detector output signal), but these detectors should give a signal that is linear in the input power.

In the scheme as shown in Fig. 4, for the phase shifters inside the MZIs a total range of $5\pi/2$ would allow full range and flexibility of function. That allows for a full 2π range for controlling the overall phase of the output from a given MZI using the common-mode phase shifts ϕ_{CMj} and an additional $\pm\pi/2$ on each phase shifter to allow full control of differential phase shifts $\Delta\theta_j$ over $\pm\pi$ to set the split ratio of the MZI. In operation, we should choose a “zero” position in the middle of the range of each phase shifter, a position that allows the phase to be varied by at least $\pm5\pi/4$ around this point. We presume for simplicity that we start with all the phase shifters set approximately to their zero point, though as long as this $\pm5\pi/4$ range is available round this point, it is not critical exactly what the absolute phase delay is at this nominal zero point.

Note that we deliberately have chosen an MZI version for this OSM in which we have two phase shifters within each Mach-Zehnder, one on each arm. This allows us to use each MZI to calibrate itself, at least within an overall phase shift. We are also presuming here that the beamsplitters in the MZIs all have fixed 50:50 split ratios. We could implement the protocols in [20] and [22] to achieve this.

The calibration process for the OSM then proceeds as follows.

1. We “write down” the driving conditions for what we have chosen as the nominal “zero” point for the phase θ_{MSL} in MZ5, so we know how to get back to it and to refer other phase settings relative to it, and we set the phase θ_{MSL} in MZ5 to this point. (For example, we could conveniently choose this in the middle of the phase shifter range.)
2. Shine a power P_C into guide MC.
3. Adjust θ_{MSU} to minimize the power in D5 (or maximize the power in D5'). This sets MZ5 in its “cross” state, with all the power P_C flowing out of the top right port of MZ5. This set of driving conditions now gives us a specific defined “zero” point for the θ_{MSU} phase shifter. (If the phase delays in both arms of an MZI are equal, the MZI will be in its “cross” state.)
4. Calibrate the θ_{MSU} phase shifter (i.e., the phase delay, relative to this defined “zero” point) by observing the power in D5 (or in D5') as we vary the θ_{MSU} phase shifter drive. We presume we know from the physics of the phase shifters what sense of drive corresponds to increasing phase shift, so from the sinusoidal variation in the detected power with phase shift, we can accurately calibrate the phase shifter.
5. Set the θ_{MSU} phase shifter to its now-defined “zero” point.
6. Similarly calibrate the θ_{MSL} phase shifter (i.e., the phase delay, relative to its defined “zero” point) by observing the power in D5 (or in D5') as we vary the θ_{MSL} phase shifter drive.
7. Set the θ_{MSU} and θ_{MSL} phase shifters to their now-defined “zero” points, which will put MZ5 in its “cross” state
8. Now repeat steps (3) to (7), but for MZ4, with its phase shifters and detectors.
9. Repeat step (8) for each of the other MZIs progressively up the chain.

This set of steps will give us calibrations for all the MZIs. We could now set them to give any set of amplitude split ratios and control all the phases in the output guides MO1 to MO5. (We note, incidentally, that we do not know the absolute phase in any of the outputs because the absolute phase delay through any given MZI and in propagation through the various waveguides MO1 to MO5 is not known. So, we might think we do not have enough OSM calibration information to program the mesh; nonetheless, we can program the mesh, including the necessary phases, exploiting one further “trick” to which we will return below in subsection (iii).)

(ii) Calculation of the MZI phase shifter settings for any desired input vector

If this OSM is calibrated, with light shining only into MC, it is relatively straightforward to calculate the necessary settings for all the phase shifters to give any desired output “vector” of amplitudes in waveguides MO1 to MO5. This general mathematics is given in [4], and we will not repeat it here explicitly. Conceptually, in calculating the necessary split-ratios and phase shifts for each MZI, we work progressively from MZ5 back to MZ1. We can summarize this calculation process for the MZI split ratios and phase shifts for any desired output vector as follows.

First, for the required MZI split ratios, for MZ5, according to the vector we want to set up, we know the desired fraction of power to be split off into MO5, thereby establishing the desired split ratio of MZ5. For the split ratio of MZ4, we have to set it to give the desired output power fraction in MO4, establishing its split ratio based on the amount of power remaining after having split power off into MO5 already. We can continue similarly through the remaining MZIs calculating their split ratios based on the desired output power fraction and the amount of power remaining after the preceding splits in earlier MZIs. We end with MZ1, for which we presumably set the split ratio to be 100% – i.e., all of the remaining power will be routed into MO1.

Second, for calculating the desired phase shifts, we proceed in an analogous fashion. We first choose the additional common-mode phase shift we want in MO5, according to the vector we want to set up. (Since usually only relative phase shifts matter in an actual system, we may well just choose this one to be zero.) Then, noting this phase shift in MZ5, we calculate the phase shift we need in MZ4 to give the desired phase difference between MO4 and MO5. We proceed similarly to calculate all the desired phase shifts in the remaining MZIs, giving a set of “common mode” phase shifts ϕ_{CMj} for the MZI setting.

Hence, we have established what split ratios and phase shifts we want for all the MZIs to give the desired output vector of relative amplitudes and phases in the output waveguides MO1 to MO5, thus giving the OSM our first attribute (i) above. Calculating the corresponding necessary differential phases $\Delta\theta_j = \theta_{Uj} - \theta_{Lj}$ for each MZI from the desired split ratios is straightforward (see, e.g., [17]). Hence, the final settings we want for the phase shifters in the MZIs are given by $\theta_{Uj} = \phi_{CMj} + \Delta\theta_j / 2$ and $\theta_{Lj} = \phi_{CMj} - \Delta\theta_j / 2$.

(iii) Setting up the mesh

Now that we have a calibration of the OSM MZIs that allows us to configure any set of amplitude split ratios and control all the phases in the output guides MO1 to MO5, we can start to set up the mesh itself with the desired input functions.

We note that we do not know the absolute phase in any of the outputs MO1 to MO5 because the absolute phase delay through any given MZI is not known (and possibly neither is the phase delay in each of the waveguides from the MZIs to the outputs MO1 to MO5); we can, however, make a mathematical choice that, when all the MZI phase shifters are in their now-defined “zero” state we will call the phase in all the output guides MO1 to MO5 “zero”.

So, with these calibrations and our definition of “zero” phase at the outputs MO1 to MO5, we can proceed to set up the mesh as discussed in the main text, starting with the top “left”

element of the mesh. Note that, in doing so, we first adjust the phase shifter ϕ_{11} inside the mesh to minimize the power in the appropriate detector inside Block 11. That adjustment will itself take care of any unknown fixed relative phase shift between MO1 and MO2, so the fact we did not know that relative phase shift does not matter. Similarly, as we proceed next to set up Block 21, we will adjust phase shifter ϕ_{21} inside the mesh to minimize the power in the appropriate detector inside Block 21, which will take care of any unknown fixed relative phase shift between MO3 and MO4. Then as we move to column 2, Block 12, we will be adjusting phase shifter ϕ_{12} , which will take care of any unknown fixed relative phase shift between MO2 and MO3, and similarly for Block 22, adjusting phase shifter ϕ_{22} will take care of any unknown fixed relative phase shift between MO4 and MO5. In effect, then, the fact that we have phase shifters inside the mesh at one input to each Block means we do not need to know the absolute phase shifts along any of the paths – we automatically compensate any unknown fixed phase differences. (This is the trick referred to above that allows us not to know the actual phase delay through the MZIs.)

(iv) *Turning off the OSM*

We note first that, once we have set up the mesh, we can then set all the MZIs, MZ1 to MZ5, to their “cross” state and set all phase shifts to their “zero” condition. Then, any light from the calibration light source will pass directly “up” the chain, to be “dumped” out the waveguide MD; none of the light from the calibration light source is sent to the mesh. We would not even need to turn off this source, though in practice we might well do so. Since all the MZIs are in their cross state, input light from the input waveguides MI1 to MI5 also passes directly through the MZIs, with no additional added phases. Note that, if we have chosen the detectors D1’ to D5’ positioned between the MZIs, there are now no detectors in the path of the actual inputs to the mesh. Note also that all the input signals pass through nominally equivalent paths on their way from the input OSM input waveguides MI1 to MI5, each encountering exactly one MZI in its cross-state and presumably substantially equal total lengths of waveguides (which is the case for the layout sketched in Fig. 4).

To use this approach with actual inputs, in MI1 to MI5, we finally need to deal with compensating relative phases in these inputs so that those are what we think they are or want them to be.

Input phase compensation with the calibration light source

In practice, we likely would want to add in additional input compensating phase shifters ϕ_{M1} to ϕ_{M5} because the path lengths from the inputs MI j to the corresponding outputs MO j might not all be identical; in practice it would be difficult to fabricate paths that were absolutely identical in optical length to within a small fraction of a wavelength. To perform this compensation, we should first shine the equivalent of a “plane wave” into the inputs MI1 to MI5 – that is, a set of optical inputs that we regard as all having the same phase at these inputs; we want the resulting phases at the outputs of the MZIs (the bottom right ports in each case – the “Bottom” port in Fig. 1 of the main text) to be in phase with the field from the calibration light source at each of these points.

To achieve this phase equalization, we first shine the calibration source light into MC as before, while simultaneously shining in our “plane wave” input into MI1 to MI5. We set MZ5 (a) so that light from both (upper and lower) input ports flows to both output ports (the simplest and most obvious setting is to set MZ5 to its “50:50” split-ratio settings), and (b) to its calibrated “zero” common-mode phase delay setting. Next, we adjust each input phase shifter ϕ_{M5} to maximize the power in the corresponding detector D5 (or minimize the power in D5’); that means that the calibration source power and the power from input MI5 are adding in phase, so the phase delay from the input MI5 is equal to the phase delay of the

calibration power source from MC. Then we can change MZ5 to its “cross” state so the power from MC now flows to MZ4, and repeat the process with MZ4, and so on up the line of MZIs. With this calibration performed for all the input phase shifters ϕ_{Mj} , all the input phases are correctly compensated. We would then “turn off” the OSM as described above.

Input phase compensation without the calibration light source

Note for the above input phase compensation process, the input “plane wave” into MI1 to MI5 has to be phase-coherent with the calibration light source; effectively, all the input light would need to be derived from the same source. To avoid that requirement, we can perform this phase equalization without using the calibration light source at all.

To understand how we do this, we shine the “plane wave” into MI1 to MI5 as if we are phase compensating with the calibration light source as above, though the calibration light source is now turned off. We set the split ratio of MZ5 so that some of the light from MI5 flows out of the upper right port of MZ5 (i.e., the one in which detector D5' would sit); perhaps the simplest and most useful choice here is to set MZ5 in its “bar” state so all the power from MI5 flows out of this port. Since we are only going to be setting relative phases between the various outputs MO1 to MO5, it does not matter what phase we set for ϕ_{M5} , and perhaps most conveniently, we set this to a nominal “zero” phase shift; we could even eliminate this phase shifter altogether, though it may be better to keep it as a “dummy” so all paths from the input to the output waveguides are similar.

The resulting power flowing through MZ5 from MI5 can be used instead of the power that would have flowed from the calibration power source as we set up the subsequent phase shifters. There is, however, a subtlety regarding the phase of this power. When we were setting up the phase shifter ϕ_{M5} using the calibration source power, we were setting that up so that the power was being maximized in the MO5 output port (i.e., the one in which the D5 detector would sit), which means that we were setting constructive interference in that port. That in turn means we would have destructive interference in the other output port (i.e., the one in which detector D5' would sit). That in turn implies that the light flowing from MI5 to the D5' output is actually 180° or π out of phase with the light that would have been flowing from the calibration power source. Hence, if we want to use this light instead of the calibration power source light, we need to phase shift it by 180° or π . So, if we use this method, we should set the common-mode phase shift of MZ5 to 180° or π (or minus this value – in phase shift terms, there is no difference). Thereafter, the phase calibration of the remaining phase shifters ϕ_{M4} to ϕ_{M1} proceeds as if we were using the calibration power source. Once we have completed this phase compensation, we should reset the common-mode phase shift of MZ5 to zero again, and “turn off” the OSM as described above (setting all the MZIs to their cross states and zero common-mode phase delays).

Appendix B - Matrix approach for rectangular mesh calculations

It is relatively straightforward to proceed block-by-block to calculate all the required input vectors to set up the rectangular mesh of Fig. 3 given the known 2×2 matrices D_{ij} as discussed in the main text above, working progressively from the left, block column by block column, starting with block column 1. To write this in a convenient algebraic form, we can write matrices C_j for each entire column. For all the odd-numbered and even-numbered columns in the “ 5×5 ” example of Fig. 3, we have, respectively matrices

$$C_{j(\text{odd})} = \begin{bmatrix} \boxed{D_{1j}} & 0 & 0 & 0 \\ 0 & 0 & \boxed{D_{2j}} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C_{j(\text{even})} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \boxed{D_{1j}} & 0 & 0 \\ 0 & 0 & 0 & \boxed{D_{2j}} \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (14)$$

where the 2×2 blocks D_{ij} can be as in Eq. (2), and we presume we know them based on a design, such as the approach of [25]. Then the configuration process proceeds as follows.

1) Calculate the vector

$$|\psi_{BwdOut}\rangle = C_1 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (15)$$

and train with a physical input vector of amplitudes $|\psi_{FwdIn}\rangle = |\psi_{BwdOut}\rangle^*$, adjusting first ϕ_{11} to minimize the power in the “bottom” detector DB in block “11” and then adjusting θ_{11} to minimize that power again (now essentially to zero), to set the required parameters in block “11”.

2) Calculate the vector

$$|\psi_{BwdOut}\rangle = C_1 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (16)$$

and train with a physical input vector of amplitudes $|\psi_{FwdIn}\rangle = |\psi_{BwdOut}\rangle^*$, adjusting first ϕ_{21} to minimize the power in the “bottom” detector DB in block “21” and then adjusting θ_{21} to minimize that power again (now essentially to zero), to set the required parameters in block “21”.

3) For block “12” in the second column, proceed similarly to step 1 above, but work with the calculated vector

$$|\psi_{BwdOut}\rangle = C_1 C_2 \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (17)$$

and the DB detector in block “12” to set ϕ_{12} and θ_{12} .

- 4) For block “22” in the second column, proceed similarly to step two above, but work with the calculated vector.

$$|\psi_{BwdOut}\rangle = C_1 C_2 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad (18)$$

and the DB detector in block “22” to set ϕ_{22} and θ_{22} .

- 5) For the blocks “13” and “23” in the third column, proceed similarly to steps 1 and 2 above, but with the calculated vectors

$$|\psi_{BwdOut}\rangle = C_1 C_2 C_3 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ and } |\psi_{BwdOut}\rangle = C_1 C_2 C_3 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (19)$$

respectively.

- 6) For the blocks “14” and “24” in the fourth column, proceed similarly to steps 3 and 4 above, but with the calculated vectors

$$|\psi_{BwdOut}\rangle = C_1 C_2 C_3 C_4 \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ and } |\psi_{BwdOut}\rangle = C_1 C_2 C_3 C_4 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad (20)$$

respectively.

- 7) For the blocks “15” and “25” in the fifth column, proceed similarly to steps 1 and 2 above, but with the calculated vectors

$$|\psi_{BwdOut}\rangle = C_1 C_2 C_3 C_4 C_5 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \text{ and } |\psi_{BwdOut}\rangle = C_1 C_2 C_3 C_4 C_5 \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (21)$$

respectively.

We can see how this approach can be generalized for larger such rectangular meshes, with alternating vector calculations between adjacent block columns.

Appendix C - Combining with “perfecting” optics

The presence of embedded detectors DR or DB in the blocks means that the “perfecting” approach that allows automatic adjustment of beamsplitters to 50:50 ratios [20] can be applied straightforwardly here. We can run such a perfecting algorithm on one block after

another before running the actual configuration process that sets the system up for its desired function.

Such perfecting approaches can be based on shining light just into the “top” port of a given interferometer block to run the “perfecting” algorithm. This is straightforward in the first “column” of the lattice filter or the rectangular mesh. For the lattice filter example, for the second interferometer block, we just arrange to leave the first block in its “bar” state (i.e., “top” input connected to “right” output and “left” input connected to “bottom” output), and then we have light in the “top” input port for this second block to run the “perfecting” algorithm on it, and so on.

For the rectangular mesh example, there are several different ways we could perform this sequence of “perfecting” algorithms. Generally, we would perfect a given “column” of interferometer blocks before proceeding to the next. We note that the “perfecting” algorithm can be run for a given block either with light shining into the “top” input or into the “left” input, and it is generally straightforward after perfecting a given interferometer block to set it in either the “bar” or the opposite “cross” state as desired. So, it is similarly straightforward to arrange to shine light only into an appropriate port to run the “perfecting” algorithm in any interferometers in subsequent columns.

Funding

Air Force Office of Scientific Research (FA9550-15-1-0335 and FA9550-17-1-0002).