Chinese Physics Letters **40**, 124402 (2023) **Express Letter**

Prediction of Thermal Conductance of Complex Networks with Deep Learning

Changliang Zhu(朱昌良)[1], Xiangying Shen(沈翔瀛)[1*], Guimei Zhu(朱桂妹)[2*], and Baowen Li(李保文)[1,2,3,4]

[1]*Department of Materials Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China*
[2]*School of Microelectronics, Southern University of Science and Technology, Shenzhen 518055, China*
[3]*Department of Physics, Southern University of Science and Technology, Shenzhen 518055, China*
[4]*Shenzhen International Quantum Academy, Shenzhen 518017, China*

(Received 19 October 2023; accepted manuscript online 17 November 2023)

Predicting thermal conductance of complex networks poses a formidable challenge in the field of materials science and engineering. This challenge arises due to the intricate interplay between the parameters of network structure and thermal conductance, encompassing connectivity, network topology, network geometry, node inhomogeneity, and others. Our understanding of how these parameters specifically influence heat transfer performance remains limited. Deep learning offers a promising approach for addressing such complex problems. We find that the well-established convolutional neural network models AlexNet can predict the thermal conductance of complex network efficiently. Our approach further optimizes the calculation efficiency by reducing the image recognition in consideration that the thermal transfer is inherently encoded within the Laplacian matrix. Intriguingly, our findings reveal that adopting a simpler convolutional neural network architecture can achieve a comparable prediction accuracy while requiring less computational time. This result facilitates a more efficient solution for predicting the thermal conductance of complex networks and serves as a reference for machine learning algorithm in related domains.

DOI: 10.1088/0256-307X/40/12/124402

Complex networks manifest themselves ubiquitously in the natural and artificial systems, and have been extensively studied across various disciplines,[1–8] such as social networks,[9,10] neuronal networks,[11,12] thin-film transistors, and nanosensors.[13–15] Owing to the fast development of micro and nanotechnology, real nanowire/nanotube networks now can be meticulously fabricated in laboratory.[16,17]

On the other hand, as one of the most basic energy forms, heat plays a fundamental role in nature and in human being's life. However, compared with electric/electronic transfer, we are short of deep and comprehensive understanding of heat transfer, in particular, in complex networks.[18–22] To better understand heat transfer in networks, nodes can be considered as junctions, while the edges signify the connection between these junctions.[23]

The key distinction between complex networks and 1D/2D lattices lies in the distribution of node degrees: lattices have uniformly distributed degrees for each node, whereas random networks exhibit varying degrees among different nodes.[24–28] Additionally, several other factors influence heat transport in complex networks, including clustering coefficients, average shortest distances, assortativity coefficient, and masses of nodes.[29–33]

Given the intricate nature of these networks, accurately predicting thermal conductance remains a great challenge. The complexity of these networks has led most prior studies to consider only a single heat source node and a single sink node within a given random network. However, practical scenarios often involve multiple nodes serving as heat sources and sinks, rendering the prediction and understanding of their thermal properties limited. Fortunately, machine learning offers a promising approach to predict thermal transport within complex networks.

Deep learning (DL), a prominent subset of machine learning, has become a cornerstone due to its unique capability to process vast and intricate datasets, discern complex patterns, and forecast material behaviors.[34,35] Among the representatives of DL, convolutional neural networks (CNNs) have found notable applications in materials science. CNNs exhibit a remarkable ability to predict the properties of novel materials, consequently, expediting the development of new materials for diverse applications such as energy storage, electronics, and transportation. To achieve robust accuracy of prediction and mitigate overfitting, it is common practice to construct deeper CNN architectures. However, it is crucial to acknowledge that more complex CNN models demand larger datasets and greater computational resources. Moreover, the challenge of constructing a large dataset is exacerbated when data acquisition entails significant experimental and computational costs. Therefore, identifying an appropriate CNN model that balances high prediction accuracy while conserving time and resources is of paramount importance.

In a typical CNN framework, input are images that converted into matrices with three channels corresponding to RGB color. Essentially, once a network's topology

*Corresponding authors. Email: shenxy@sustech.edu.cn; zhugm@sustech.edu.cn
© 2023 Chinese Physical Society and IOP Publishing Ltd

is established, its thermal conductance is decisively determined from its encoded Laplacian matrix. This allows us to utilize the Laplacian matrix as input for CNN models, enabling the prediction of the thermal conductance, within random network systems, as shown in Fig. 1. To the best of our knowledge, this pioneering approach has not been investigated before. In addition, we compare two well-established CNN models with different architectures: the complex VGGNet and the simpler AlexNet. Next, these well-trained models were evaluated by predicting the thermal conductance of complex networks within testing dataset that they have never been encountered during their training. Remarkably, both the VGGNet and AlexNet models demonstrate flawless predictive performance when assessed on the testing dataset.

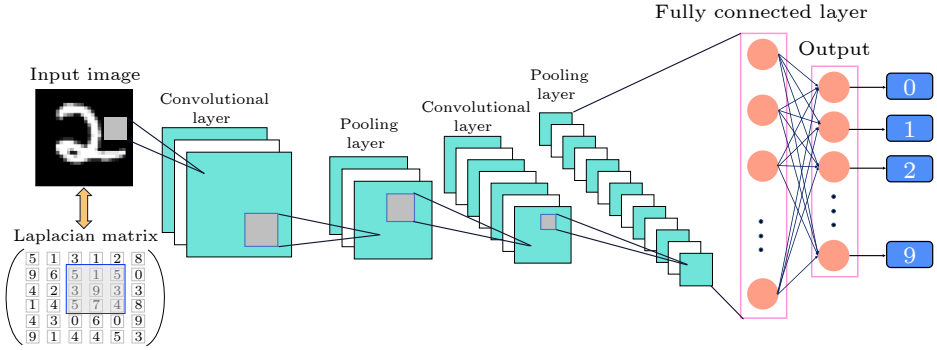

Fig. 1. Architecture of convolutional neural networks. The input data is changed from image to Laplacian matrix that encodes the thermal transport properties of complex networks.

Training the VGGNet model, with its more intricate architecture, required 120 min, whereas the simpler AlexNet architecture demanded only 20 min for training. This revelation underscores the significance of a concise CNN architecture, which can still achieve satisfied accuracy of prediction. Our findings offer valuable insights for physics research pertaining to heat transport within complex network systems.

In general, the heat conduction of complex networks depends on both the contact resistance and the network topology. In our study, we focus on how the temperatures and heat fluxes are distributed on a complex network based on its topological structure and source nodes.[36,37] In this case, the topology of a complex network can be encoded as an adjacency matrix A, with its element $A_{ij} = 1$ if nodes i and j are connected, and 0 otherwise. Define the $A_{ii} = 0$ for avoiding self-connection. Denote $\boldsymbol{M}$ as the temperature field vector of N components. We use this simplified theoretical model to analyze a more complicated situation, considering the number of source and sink nodes are more than 1. Here we consider the case wherein there are m modes with temperature $\boldsymbol{M}_h$ as heat source and n nodes with temperature $\boldsymbol{M}_l$ as heat sink.

The sources nodes are encoded into a vector $\boldsymbol{M}_h = (T_{h1}, T_{h2}, \ldots, T_{hm})$, and the sink nodes are encoded into a vector $\boldsymbol{M}_l = (T_{l1}, T_{l2}, \ldots, T_{ln})$. Thus, the temperature vector M can be expressed as $\boldsymbol{M} = (\boldsymbol{M}_h, \boldsymbol{M}_l, T_{(N-m-n+1)}, \ldots T_N)$ and can be divided into block-1 with $\boldsymbol{M}_1 = (\boldsymbol{M}_h, \boldsymbol{M}_l)$, which is an $(m+n)$-dimensional vector, and block-2 $\boldsymbol{M}_2 = (T_{(N-m-n+1)}, \ldots, T_N)$, which is an $(N-m-n)$-dimensional vector. The discrete Laplace operator on the network L ($N \times N$-dimensional) is an analogue of ∇^2 and can be expressed as $L = D - A$, where D is the diagonal degree matrix. Accordingly, the matrix L can be divided into four submatrices: $(m+n) \times (m+n)$-dimensional L_{11}, $(m+n) \times (N-m-n)$-dimensional L_{12}, $(N-m-n) \times (m+n)$-dimensional L_{21} and $(N-m-n) \times (N-m-n)$-dimensional L_{22}. In the steady state we can obtain $L\boldsymbol{M} = \boldsymbol{J}$, where $\boldsymbol{J}$ is the external flux vector $\boldsymbol{J} = (\boldsymbol{J}_h, \boldsymbol{J}_l, J_{(N-m-n+1)}, \ldots, J_N)$. Note that this is the discrete analogue of $-\kappa\nabla^2\boldsymbol{T}(\boldsymbol{r}) = \nabla \cdot \boldsymbol{j}(\boldsymbol{r})$ with $\boldsymbol{M}$ playing the role of $\kappa\boldsymbol{T}(\boldsymbol{r})$ and $\boldsymbol{J}$ playing the role of $-\nabla \cdot \boldsymbol{j}(\boldsymbol{r})$, where $\boldsymbol{j}(\boldsymbol{r})$ is the heat flux of the vector field. Except for the source and sink nodes, the equilibrium conditions require the zero net heat flux, i.e., $\boldsymbol{J} = 0$ for the nodes $i = N-m-n+1, \ldots, N$. Therefore, we have

$$\begin{cases} L_{11}\boldsymbol{M}_1 + L_{12}\boldsymbol{M}_2 = \boldsymbol{J}, \\ L_{21}\boldsymbol{M}_1 + L_{22}\boldsymbol{M}_2 = 0, \end{cases} \tag{1}$$

where $\boldsymbol{J} = (\boldsymbol{J}_h, \boldsymbol{J}_l)$ represents the heat flux on each source node; $\boldsymbol{J}_h = J_{01}, J_{02}, \ldots, J_{0m}$ represents heat flux on node applied with high temperature, and $\boldsymbol{J}_l = -J_{01}, -J_{02}, \ldots, -J_{0n}$ demonstrates the heat flux on each node with applied low temperature. According to Kichhoff's law, the total amount of heat flow entering the network should be equivalent to the amount of heat flow outgoing the network, $\sum_{p=1}^{m} J_{0p} = -\sum_{q=1}^{n} J_{0q} = J_0$.

The second equation can be expressed as $L_{22}\boldsymbol{M}_2 = -L_{21}\boldsymbol{M}_1$. As $\boldsymbol{M}_1$ and the elements of all submatrices L_{11}, L_{12}, L_{21} and L_{22} are known for a network, $-L_{21}\boldsymbol{M}_1$ is an $(N-m-n)$-dimensional vector and the equation $L_{22}\boldsymbol{M}_2 = -L_{21}\boldsymbol{M}_1$ has a unique nonzero solution $\boldsymbol{M}_2$. Then, substituting $\boldsymbol{M}_2$ into the first equation, we can obtain the $\boldsymbol{J}_0$. Thus, $\boldsymbol{J}_0$ can be analytically solved. Finally, we can calculate the thermal conductance C of the entire complex network as $C = \boldsymbol{J}_0/(T_h - T_l)$. Note that these quantities are dimensionless.

To form the random network, a series of nodes are randomly placed within a square region and then connected utilizing Delaunay triangulation. Note that the

nodes on each boundary are evenly distributed. Subsequently, we designated the nodes on the bottom boundary as the heat source, maintaining an applied temperature of 303.15, while the nodes along the top boundary were identified as the sink, with their temperature set at 293.15. Employing matrix solutions, we were able to determine the temperature distribution for each node within the network, and the resulting distribution is visually represented in Fig. 2. It is noteworthy that variations in their geometrical topologies yield different thermal conductance values, as shown in Figs. 2(a) and 2(b).

To further augment the dataset size, we adopted a strategy of manipulating the random network's topology, thereby inducing variations in thermal conductance. This was accomplished by selectively removing specific bonds within the network. Notably, we observed that an iterative process of bond removal led to a progressive reduction in the thermal conductance of the random network.

As illustrated in Figs. 2(c) and 2(d), the original network exhibited a thermal conductance of 2.05, while the final network, after the removal of certain bonds, registered a substantially lower thermal conductance value of 0.28. Consequently, given an original random network, we were able to generate a substantial number of diverse samples by systematically deleting distinct sets of bonds. An additional advantage of employing bond removal to construct our dataset lies in the capacity to span a broad range of thermal conductance values.

The supervision of this training process relied on the mean squared error (MSE) loss functions, designed to quantify the differences between the predicted thermal conductance values and the corresponding labels: MSE Loss $= \frac{1}{N}\sum_{i=1}^{N}(C_i - \hat{C}_i)^2$, where N represents the number of samples during one training epoch, C_i and $\hat{C}_i$ denote the predicted and label values, respectively. The predictive performance of the optimal trained model can be quantitatively evaluated by the testing dataset with two metrics, i.e., root-mean-square error (RMSE) and determination coefficient (R^2) defined by

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{C}_i - C_i)^2}, \\ R^2 &= 1 - \frac{\sum_{i=1}^{N}(\hat{C}_i - C_i)^2}{\sum_{i=1}^{N}(\bar{C}_i - C_i)^2}, \end{aligned} \tag{2}$$

where $\bar{C}_i$ denotes the average value of testing dataset. Note that the samples within testing dataset are never encountered by CNN models during the training process.

CNN is a widely utilized algorithm in the field of computer vision, primarily designed for image analysis. A CNN comprises multiple hidden layers, which can be broadly categorized into three main types: the convolutional layer (conv), the pooling layer (pooling), and the fully connected layer (FC). The convolutional layer employs a block-based approach to identify specific patterns within the image. Within this layer, a set of filters, represented as $m \times m$ square matrices, slide over the input image, computing the dot product between the filter weights and the corresponding pixel values. These filter weights are optimized through the training process. Typically, the first convolutional layer specializes in detecting low-level features such as edges and colors, while subsequent layers extract increasingly complex, high-level features. This hierarchical progression empowers CNN to develop a holistic understanding of the input dataset. The second crucial component in CNN architecture is the pooling layer, used for dimensionality reduction and mitigating overfitting. Following the convolutional layer, a pooling layer is frequently integrated into the network. This layer offers two kinds of operations: average pooling and max pooling. The third essential layer within a CNN architecture is the fully connected layer. In this layer, each neuron establishes connections with every neuron in the preceding neural networks, fostering comprehensive information exchange.

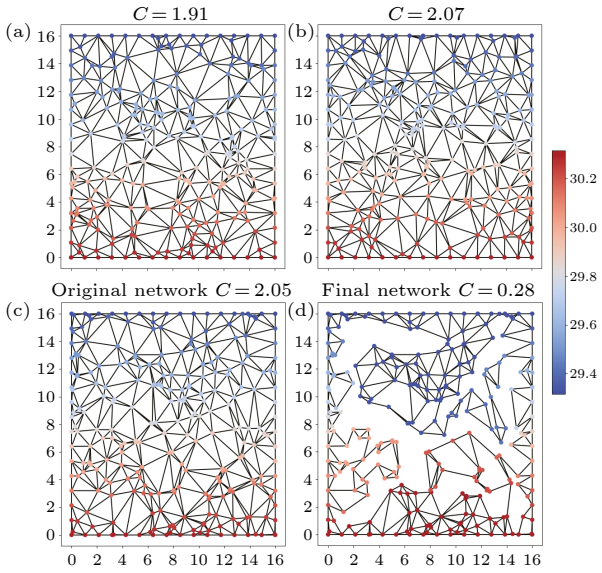

Fig. 2. Dataset construction. The colorbar on the right side of each plot demonstrates the temperature distribution of each node. [(a), (b)] Illustration of different topologies of random networks to generate different thermal conductance. [(c), (d)] Original and final configuration of random networks tailored through bond removing method.

To enhance the predictive accuracy of neural networks, one approach is to increase the network's depth. Nevertheless, an excessive number of convolutional layers can potentially lead to the overfitting problem. To strike a balance between these considerations, the VGGNet architecture was developed, to increase the depth of CNN while maintaining optimal performance. VGG, an abbreviation for visual geometry group, represents a standard deep CNN architecture. The term "deep" in VGGNet alludes to the substantial number of layers, with VGG-16 and VGG-19 models, for instance, comprising 16 and 19 convolutional layers, respectively. This depth enhancement equips VGGNet with the capacity to capture intricate features and patterns, making it a valuable tool for a wide range of computer vision tasks.

We first trained a VGG-16 model to predict the thermal conductance of complex networks, and the schematic

diagram of VGG-16 neural network structure is shown in Fig. 3(a). Throughout the training process in Fig. 3(b), the loss L_2 was reduced to around 10^{-3} after approximately 5 epoches, where it stabilized with minimal oscillation during further optimization. This indicated that the neural network had undergone effective training and was converging towards an optimal state. The entire training process took approximately 120 min to complete. Ultimately, the well-trained neural network serves as a surrogate model, replacing the matrix solving and facilitating the prediction of thermal conductance for a given network configuration.

Subsequently, we leveraged the trained neural network to predict the thermal conductance of samples within the testing dataset. We compared the predicted values $C_{\text{prediction}}$ against the corresponding ground truth label values C_{label}, visualizing the results as a scatter plot, as shown in Fig. 3(c). Remarkably, the scatter points closely align with the reference gray dashed line $C_{\text{prediction}} = C_{\text{label}}$, signifying a high degree of accuracy in the model's predictions. Moreover, the regression estimation value of R^2 is 1.0, which is just the ideal value. This compelling result underscores the exceptional prediction accuracy achieved by the trained VGG model, demonstrating its proficiency in learning the inherent heat transport properties of random networks from the dataset.

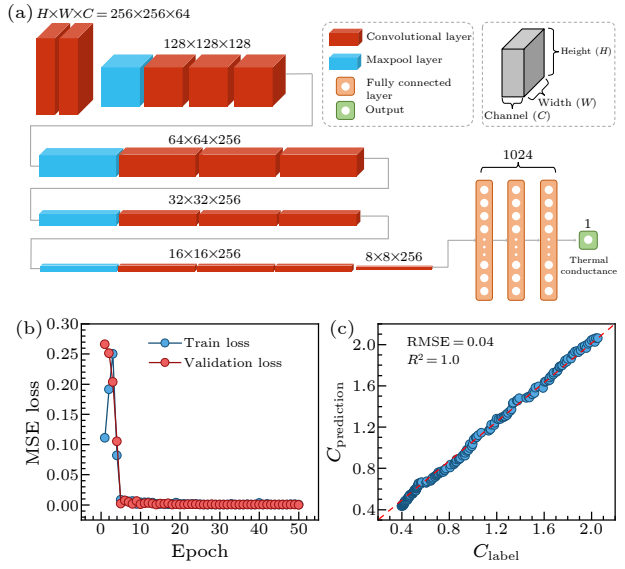

Fig. 3. Training and prediction performance of VGGNet. (a) Architecture of VGGNet. (b) MSE loss decreases as the training iteration proceeding. (c) Predicted thermal conductance $C_{\text{prediction}}$ as a function of label value C_{label}.

However, it is worth noting that VGG-16 is composed of a relatively substantial number of hidden layers, including 13 convolutional layers and 3 fully connected layers. Moreover, each of these fully connected layers consists of 1024 neurons. The inclusion of deeper hidden layers and a multitude of neurons results in a great number of weights and biases that need to be fine-tuned during the training phase. This increased complexity has seriously retarding the training process. Furthermore, if the construction of the training dataset demands substantial computational resources or high experimental costs, it may become challenging to assemble a dataset with an adequate volume of data. Hence, it is prudent to seek a balance between model complexity and prediction accuracy. In this case, AlexNet was selected as our candidate model.

AlexNet made its notable debut in the ImageNet Large Scale Visual Recognition Challenge of 2012, where it demonstrated its prowess in image classification. There are two prevailing versions of architecture of AlexNet with different depths: three and five convolutional layers. For our application, we opted for the version with five convolutional layers, recognizing that deeper neural networks possess the ability to capture more abstract features from the training dataset. Such architecture comprises five convolutional layers and three fully connected layers, as shown in Fig. 4(a). Nevertheless, the five-layer version still has much less layers than VGGNet. Max-pooling operations are applied following the first, second, and fifth convolutional layers. At the end of each layer, a rectified linear unit (ReLu) activation function is applied, except for the final layer. Notably, the nonlinearity of ReLu has demonstrated a significant advantage, enabling the training of deep AlexNet at a much faster pace compared to other activation functions like tanh or sigmoid. This streamlined approach enhances efficiency and accuracy in the prediction of thermal transport properties within complex networks.

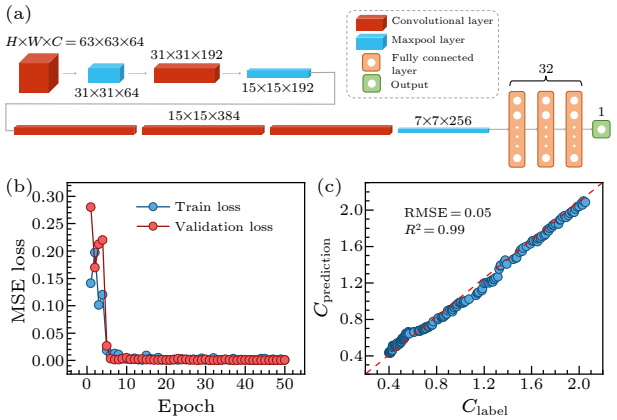

Fig. 4. Training and prediction performance of AlexNet. (a) Architecture of AlexNet. (b) MSE loss decreases as the training iteration proceeding. (c) Predicted thermal conductance $C_{\text{prediction}}$ as a function of label value C_{label}.

To ensure a fair comparison of prediction capabilities between AlexNet and VGGNet, we adopted the same dataset for AlexNet, and we maintained identical training hyperparameters, including the learning rate, batch size, training epochs, and the choice of loss function. Additionally, both models were trained using the same optimizer Adam. During training shown in Fig. 4(b), we observed that the loss value swiftly diminished to approximately 10^{-3} after roughly 5 epoches, thereafter, remaining consistently low. Subsequently, we assessed the performance of the trained AlexNet model on the testing dataset, revealing a regression estimation value of R^2 around 0.99, as shown in Fig. 4(c). The R^2 approached the ideal value of 1, highlighting the remarkable predictive capability of

the model. Particularly noteworthy is the efficiency of AlexNet, as it completed the entire training process in just 20 min, significantly faster than VGGNet, which required six times longer for training. Furthermore, the average computation time to determine the thermal conductance of a single network within the dataset using the Laplacian matrix was 546.54 ms. In contrast, the well-trained VGGNet and AlexNet models significantly reduced the calculation time to 130.04 ms and 76.37 ms, respectively. The traditional method for calculating thermal conductance of a network by using molecular dynamics simulation needs a couple of days for a network of 50×50. These outcomes provide compelling evidence that a simpler neural network structure can achieve equivalent prediction capabilities in significantly less time compared to both the more complex CNN model and traditional methods. This underscores the importance of finding the right balance between model complexity and efficiency in design.

In summary, to mimic the real scenarios of heat conduction in complex networks, we put multiple nodes into heat source and heat sink. We are able to predict the thermal conductance of the complex networks with deep learning. Given that thermal conductance is intricately linked to the network's topology, we diversified the training dataset by generating distinct networks with varying thermal conductance. This has been achieved by selectively removing specific bonds, thereby forming a diverse set of training samples. We compared the prediction performance for thermal conductance of two well-established CNN models, i.e., VGGNet and AlexNet. Remarkably, we discovered that simpler neural network architectures yielded equally satisfying prediction while demanding significantly less training time. This highlights the importance of achieving a balance between CNN model complexity and its prediction efficiency, providing valuable insights for future applications in materials science and engineering.

Acknowledgments. We thank Professor Kezhao Xiong for his supportive information on the computational time cost of molecular dynamics simulation in complex networks. This work was supported by the National Natural Science Foundation of China (Grant Nos. 52250191 and 12205138), and the Shenzhen Science and Technology Innovation Committee (SZSTI) (Grant/Award No. JCYJ20220530113206015).

References

[1] Dorogovtsev S N, Goltsev A V, and Mendes J F 2008 *Rev. Mod. Phys.* **80** 1275
[2] Yan H, Park S H, Finkelstein G, Reif J H, and LaBean T H 2003 *Science* **301** 1882
[3] Liu D G, Park S H, Reif J H, and LaBean T H 2004 *Proc. Natl. Acad. Sci. USA* **101** 717
[4] Yin P, Hariadi R F, Sahu S, Choi H M, Park S H, LaBean T H, and Reif J H 2008 *Science* **321** 824
[5] Wang S, Zeng C, Zhu G, Wang H, and Li B 2023 *Phys. Rev. Res.* **5** 043009
[6] Shen X Y, Zhu G M, and Li B W 2023 *Sci. China Phys. Mech. Astron.* **66** 260501
[7] Shen X Y, Fang C C, Jin Z P, Tong H, Tang S X, Shen H C, Xu N, Lo J H Y, Xu X L, and Xu L 2021 *Nat. Mater.* **20** 1635
[8] Xi Q, Zhong J X, He J X, Xu X F, Nakayama T, Wang Y, Liu J, Zhou J, and Li B W 2020 *Chin. Phys. Lett.* **37** 104401
[9] Newman M E and Park J 2003 *Phys. Rev. E* **68** 036122
[10] Albert R, Jeong H, and Barabási A L 2000 *Nature* **406** 378
[11] Laughlin S B and Sejnowski T J 2003 *Science* **301** 1870
[12] Tian C H, Cao L, Bi H J, Xu K, and Liu Z H 2018 *Nonlinear Dyn.* **93** 1695
[13] Song Y J, Garcia R M, Dorin R M, Wang H, Qiu Y, Coker E N, Steen W A, Miller J E, and Shelnutt J A 2007 *Nano Lett.* **7** 3650
[14] Hu L, Hecht D, and Grüner G 2004 *Nano Lett.* **4** 2513
[15] Hecht D, Hu L, and Grüner G 2006 *Appl. Phys. Lett.* **89** 133112
[16] Rauber M, Alber I, Müller S, Neumann R, Picht O, Roth C, Schökel A, Toimil-Molares M E, and Ensinger W 2011 *Nano Lett.* **11** 2304
[17] van de Groep J, Spinelli P, and Polman A 2012 *Nano Lett.* **12** 3138
[18] Xiong K, Liu Z, Zeng C, and Li B 2020 *Natl. Sci. Rev.* **7** 270
[19] Xiong K Z, Yan Z X, Xie Y, Wang Y X, Zeng C H, and Liu Z H 2022 *Nonlinear Dyn.* **110** 2771
[20] Xiong K Z, Zeng C H, and Liu Z H 2018 *Nonlinear Dyn.* **94** 3067
[21] Xiong K Z, Zeng C H, Liu Z H, and Li B W 2018 *Phys. Rev. E* **98** 022115
[22] Xiong K Z, Zhou J, Tang M, Zeng C H, and Liu Z H 2018 *Phys. Rev. E* **98** 062144
[23] Xiong K Z, Zhou M, Liu W, Zeng C H, and Yan Z X 2023 *Chaos* **33** 083144
[24] Dhar A 2008 *Adv. Phys.* **57** 457
[25] Li N B, Ren J, Wang L, Zhang G, Hänggi P, and Li B W 2012 *Rev. Mod. Phys.* **84** 1045
[26] Cahill D G, Ford W K, Goodson K E, Mahan G D, Majumdar A, Maris H J, Merlin R, and Phillpot S R 2003 *J. Appl. Phys.* **93** 793
[27] Kumar S, Murthy J, and Alam M 2005 *Phys. Rev. Lett.* **95** 066802
[28] Pop E, Mann D, Cao J, Wang Q, Goodson K, and Dai H 2005 *Phys. Rev. Lett.* **95** 155505
[29] Boccaletti S, Latora V, Moreno Y, Chavez M, and Hwang D U 2006 *Phys. Rep.* **424** 175
[30] Albert R and Barabási A L 2002 *Rev. Mod. Phys.* **74** 47
[31] Watts D J and Strogatz S H 1998 *Nature* **393** 440
[32] Oliveira C L, Morais P A, Moreira A A, and Andrade J S 2014 *Phys. Rev. Lett.* **112** 148701
[33] Xiong K Z, Yan Z X, Xie Y, and Liu Z H 2021 *Sci. Rep.* **11** 5501
[34] Zhang Z W, Yang S, Wu Y H, Liu C X, Han Y M, Lee C H, Sun Z, Li G J, and Zhang X 2020 *Chin. Phys. Lett.* **37** 018401
[35] Ouyang Y L, Zhang Z W, Yu C Q, He J, Yan G, and Chen J 2020 *Chin. Phys. Lett.* **37** 126301
[36] Zhang Y C, Blattner M, and Yu Y K 2007 *Phys. Rev. Lett.* **99** 154301
[37] Liu Z H, Wu X, Yang H J, Gupte N, and Li B W 2010 *New J. Phys.* **12** 023016