Dynamical network models of the turbulent cascade

Özgür D. Gürcan

Dynamical Network Models of the Turbulent Cascade

Ö. D. Gürcan

Laboratoire de Physique des Plasmas, CNRS, Ecole Polytechnique, Sorbonne Université, Université Paris-Saclay, Observatoire de Paris, F-91120 Palaiseau, France

Abstract

Cascade models based on dynamical complex networks are proposed as models of the turbulent energy cascade. Taking a simple shell model as the initial regular lattice with only nearest neighbor interactions, small world network models are constructed by adding or replacing some of the existing local interactions by nonlocal ones. The models are then evolved over time, both by solving for the shell velocity variable using an arbitrary network generalization of the shell model evolution and by rewiring the network each time from the original lattice in regular time intervals. This results in a more intermittent time evolution with larger variations of the wavenumber spectrum. It also results in an increase in intermittency, computed from the exponents of structure functions obtained from these models. It is observed that the intermittency increases as the ratio of random nonlocal connections to local nearest-neighbor connections increases. The possibility of extraction of such models from direct numerical simulations are discussed and a detailed example for two dimensional turbulence is given.

1. Introduction

Turbulence is a duality of chaotic disorder and hierarchical organization across a large range of scales in the evolution of a fluid. This aspect of turbulence is shared with many other self-organizing complex systems, that are commonly described using networks, such as the internet [1], the brain [2], the public transport infrastructure [3], and the economy [4] to give a few examples.

The three dimensional incompressible fluid, described by the Navier-Stokes equation, mixed at large scales by an external forcing and dissipated at small scales due to molecular viscosity, provide the canonical example of

the turbulent cascade. The idealized process of self-similar energy transfer from large scales, where the energy is injected, to small scales, where it is dissipated, can be described using various simplified models including shell models [5], differential approximations [6] or closures [7]. While the study of turbulence has a long history [8], and networks are ubiquitous in modern nonlinear science [9], the connection between the two is an emerging field with many open questions [10, 11].

Network theory is as much about spreading or flow of various constituents such as information, ideas, or pathogens within a given (or evolving) network structure, as it is about the topology of the network itself. Flow of packages through the internet [12], people through the public transport system [13], the spreading of financial crises through the global economy [14], or a deadly virus across a network of human contacts [15] are all examples of such phenomena that fall under the umbrella term of percolation in complex networks [16, 17]. The turbulent cascade of energy in a complex network, representing the wavenumber domain, fits right in with the rest of these examples. However there is a key difference in the turbulent problem: the interactions in Fourier space are between three nodes -representing three interacting wavenumbers- instead of two, since they are produced by triadic interactions. In this sense the turbulent cascade takes place on a network with *"three body"* interactions [18, 19].

It is probably easier to imagine turbulence as an evolving network of vortices or semi-coherent structures (that play the role of nodes of the network) that interact with each other mainly by shearing or stretching [10, 20]. However since some aspects of networks, such as scale independence are more apparent for turbulence in Fourier space, it makes sense to develop a formulation of turbulence as a network, there as well.

Dynamically, when a turbulent system is driven at some scale, depending on its dimensionality and conserved quantities its energy cascades forward or backward and gets dissipated at small scales or through boundaries. Such a system can provide an open, out of equilibrium, steady state with a constant flux of energy across its scales. In the ideal, self similar picture, each scale gets its energy from the previous one and transfers it to the next. This view of the turbulence cascade bears some resemblance to what happens in a food web where the energy provided by the sun is transferred upscale through the food chain via predation relations (i.e. big fish eat the little fish) [21, 22]. This results in a power law biomass/size distribution somewhat reminiscent of the power law scaling of the turbulent energy as a function of scale [23].

In fact a more precise analogy exists between the onset of turbulence in some (usually quasi two dimensional) turbulent systems and the simple predator-prey dynamics [24, 25, 26]. At the onset when only few degrees of freedom are excited, turbulence and mean flow act like a coupled predator-prey system, where the turbulence plays the role of the prey and mean flow that of the predator [27, 28]. The analogy could be extended to fully developed turbulence as a complete food web with complex predation relations, or rather as a network of large number of elements whose interactions can transfer and transform various conserved quantities.

In this spirit, we propose a concrete working example in the form of a simple model of the turbulent cascade using dynamical complex networks by generalizing the Gledzer-Ohkitani-Yamada (GOY) model [29] to a percolation model on a complex, small world network, which is a type of network where most nodes can be reached from other nodes by a relatively small number of jumps due to existence of a small number of longe range connections [30]. This allows the exploration of different strategies of random rewiring of a regular lattice in order to form nontrivial small-world networks on which the turbulence is allowed to develop. Two different rewiring strategies based on Watts-Strogatz and Newman-Watts are discussed and dynamical network models are considered where the network is regularly rewired. The energy cascade is described on top of this evolving small world network using a simple shell model-like evolution of the observables $u_n(t) \equiv \sqrt{2\int_{k_n}^{k_{n+1}} E(k)\,dk}$. While this would probably appear quite unnatural to a specialist in turbulence, since the turbulence can be described nicely on a constant regular grid with deterministic equations, its power comes from the additional degree of freedom that the network topology provides, which allows us to represent part of what has been lost in the reduction leading up to the shell model at a very modest cost: the solution of the network model is not slower than the GOY model.

Shell models are closely linked to the concept of spectral reduction [31], which basically amounts to reducing the regular spectral domain to a smaller set of regions in k-space. When the full spectral domain is thus reduced, detailed phase relations between regions are lost. The shell model does have a complex phase that evolves, but this has almost nothing to do with the actual phase of the full system. The direction of energy transfer at a given instant, or its efficiency depends on these phase relations. If the phases are aligned between two regions, the energy can be transferred efficiently,

while if they are out of phase, there may be no energy transfer. While the evolution of phases is deterministic in the full system, it is fairly chaotic and usually irregular. Therefore its effect on a shell model like reduction could be represented plausibly by connections being turned on and off randomly or following a simple algorithm. Note that on top of the connection being turned on, the phase relations of the shell model itself still has to be satisfied for the energy transfer to take place. For a real physical problem, phase relations may be random or regular, for instance as in the case of weak wave turbulence [32]. In such a case, the network topology may be constructed respecting the dispersion relation of the underlying waves, and the network may be used to represent those "enhanced" connections between disparate scales due to resonant interactions [33].

Note that, the novelty of this work is not the models themselves, but the idea that we can interpret the structure of shell models as primitive initial networks that can be modified, rewired etc. to generate more complex set of network models. The turbulent cascade can be seen as a percolation in such a network that may also be evolving in time. We argue that this new way of looking at the turbulent cascade, opens the door to development of a completely new class of models that can pave the way to better conceptual understanding and eventually to the development of well optimized reduced models, especially with the input from numerical simulations.

We have also made some effort to make a connection to direct numerical simulations (DNS) by addressing the issue of network extraction. In the general case of such a problem, assuming that we know the underlying equations, there are two nontrivial steps: the identification of the nodes, and the identification of the connections. Since our model is a simple one, the nodes are already fixed as the logarithmically spaced shells in Fourier space. This allows us to avoid the issue of node identification.

The connections, on the other hand can be computed from shell to shell triadic energy transfer $T_{n\ell m}$, filtered in such a way to give the transfer to shell n from the pair ℓ and m. We can argue that if this transfer, normalized properly is relatively large, then those three nodes are "three body connected".

Note that by using Fourier space shells, which are very large regions of the k-space (especially for large k), we are using a broad brush approach. This leads to loss of details at small scales, and has other unfortunate effects such as modifying the equipartition state. Such loss of information is not well justified unless we are interested in aspects of the turbulent cascade that are somehow not lost in this reduction. Since the goal of this paper is not to

present an accurate model of turbulence, but to establish the largely unexplored connection between the Fourier space evolution in a turbulent system and complex networks, such a reduced model achieves this goal brilliantly, as its simplicity helps curtail the conceptual difficulty of the subject.

The rest of the paper is organized as follows. In section 2 we introduce the basic small world network paradigm for turbulence using a generalization of the GOY model. In sections 2.1 and 2.2 we lay out the strategies for constructing Watts-Strogatz and Newman-Watts models respectively, while in 2.3 we discuss the bipartite network perspective using two sets of nodes corresponding to wavenumbers and triads. In section 3 we provide the numerical results of all these different models compared to the basic GOY model. Section 4 describes the method for extracting the network structure from DNS data, which is then detailed and applied to two dimensional turbulence in section 4.1. We conclude this work and discuss possible future directions in Section 5.

2. Small World Network Shell Models

Consider a shell model of turbulence [5] with arbitrary range interactions for three-dimensional turbulence. Using a set of wave-vectors $k_n = k_0 g^n$, where g is the logarithmic scaling factor (usually $g = 2$), the model can be written as follows:

$$\partial_t u_n = i\alpha_m \Big[a_n^m u_{n+m}^* u_{n+m+1}^* + b_n^m u_{n+1}^* u_{n-m}^* + c_n^m u_{n-1}^* u_{n-1-m}^* \Big] , \tag{1}$$

where the interaction coefficients can be written as $a_n^m = M_{n,n+m,n+m+1}$, $b_n^m = M_{n,n-m,n+1}$ and $c_n^m = M_{n,n-1-m,n-1}$ with:

$$M_{n,\ell,\ell'} = \begin{cases} k_\ell + k_{\ell'} & \text{if } n < \ell \\ -\left((-1)^{n-\ell} k_\ell + k_{\ell'} \right) & \text{if } \ell < n \end{cases} \tag{2}$$

Here m is the range of interaction (e.g. $m = 1$ gives us the usual GOY model), and α_m is the average contribution from the geometric factor, which we take as $\alpha_m = g^{-m}$ (see for example [34]). Note that $\ell < \ell'$ is assumed in (2).

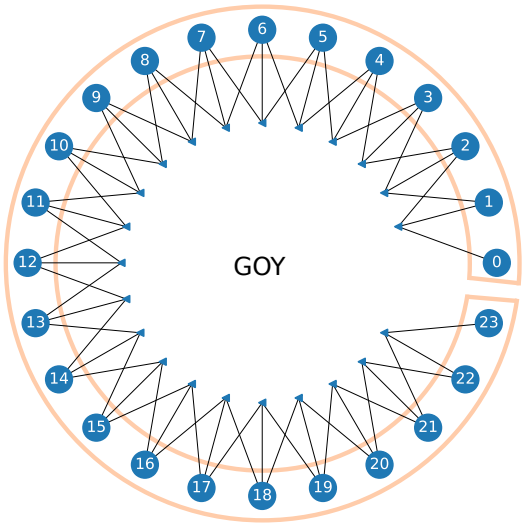

Figure 1: The regular lattice of the GOY model. Apart from the end nodes, all the nodes are connected to three triads and each triad is connected to the three closest nodes. The lattice is shown in a circular representation in order to save space and connect to earlier works on small world networks and synchronization.

This allows conservation of energy

$$E = \sum_n |u_n|^2$$

and helicity

$$H = \sum_n (-1)^n \, k_n^{-1} |u_n|^2 \ ,$$

since

$$M_{n,n+m,n+m+1} + M_{n+m,n,n+m+1} + M_{n+m+1,n,n+m} = 0$$

and

$$\begin{aligned} k_n M_{n,n+m,n+m+1} + (-1)^m \, k_{n+m} M_{n+m,n,n+m+1} \\ + (-1)^{m+1} \, k_{n+m+1} M_{n+m+1,n,n+m} = 0 \ . \end{aligned}$$

The three terms in (1) come from the three triadic interactions that have the same form but shifted with respect to one another. On the other hand a single triadic connection, for example $(n, n+m, n+m+1) = (1, 4, 5)$ introduces terms in the equations for three different wavenumbers: One term in

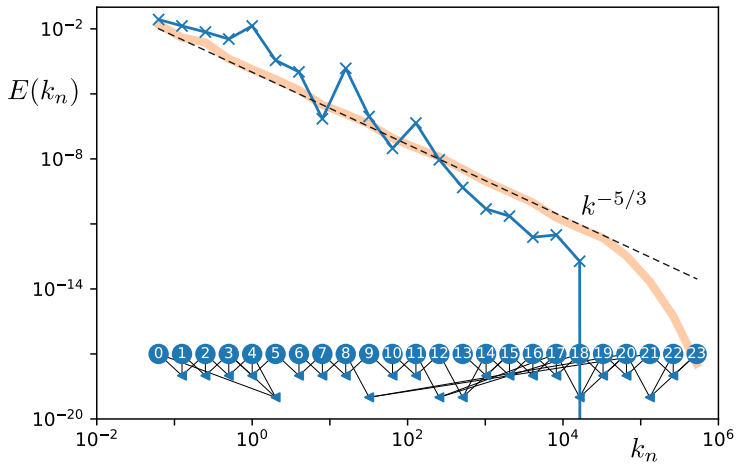

Figure 2: The wave number spectrum for the WS network, generated with $p = 0.4$, $p_f = 0.5$, compared with the GOY model, shown with the thick orange (if in color) line. Nodes 5, 9, 10, 12 and 13 are missing connections, which results in formation of crests or troughs. The fact that energy can go to the dissipative range through non-local connections makes the spectrum fall off rapidly at around 10^4. Here $N = 24$, $k_0 = 2^{-4}$, $\nu = 10^{-8}$ and $f_n = (\delta_{n1}\xi_1 + \delta_{n2}\xi_2)\,10^{-2}$, where ξ_i are random variables with a correlation time of 10^{-2}. The spectra are integrated up to $t = 5 \times 10^3$ and averaged over $t = [3 - 5] \times 10^3$.

the equation for u_1 with the coefficient $a_1^3 = M_{145}$, one term in the equation for u_4 with the coefficient $b_4^3 = M_{415}$ and finally one term in the equation for u_5 with the coefficient $c_5^3 = M_{514}$.

The standard GOY model corresponds to $m = 1$, which represents *"nearest neighbor"* interactions shown in figure 1. If we choose $g = 2$, and absorb a prefactor 6 into the arbitrary constant α, (2) gives $a_n = k_n$, $b_n = -k_{n-1}/2$ and $c_n = -k_{n-2}/2$ of Ref. 29.

2.1. Watts-Strogatz model

There are a total of $N - 2$ triads in the regular lattice of the GOY model with N nodes. In order to create a partially randomized network with non-local interactions, we go over this list of triads and replace the local triad $(n, n + 1, n + 2)$ with a nonlocal one, with either a forward, [i.e. $(n, n + m, n + m + 1)$] or a backward [i.e. $(n, n - m, n - 1)$] coupling with a probability p, where m itself is a random number between 2 and $N-n-2$ for the forward or between 3 and $n-2$ for the backward coupling. We can choose the interaction to be forward with a probability p_f. This basic algorithm is very similar to the one described by Watts and Strogatz in order to build

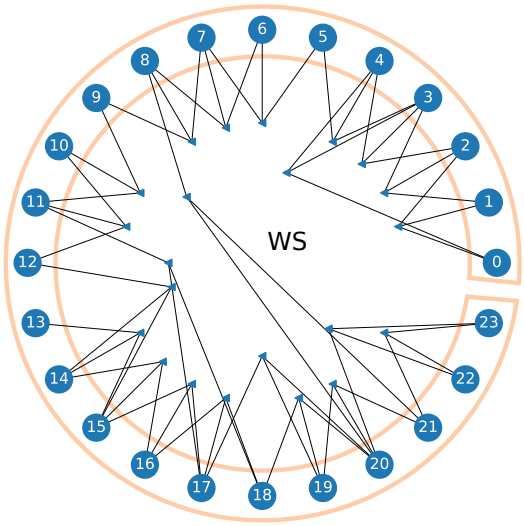

Figure 3: The small world network of the WS model, generated with $p = 0.4$,$p_f = 0.5$ also shown in Figure 2 with less than three connections at nodes 5, 9, 10, 12 and 13.

small world networks [35] (so we call it the Watts-Strogatz model or WS for short), except that the topology of the initial lattice is not really a ring, and the connections are not lines, but triadic interactions.

Having the list of triads thus revised, we can recompute the list of interactions $\mathbf{i}_n = \{\ell, \ell'\}$ and the interaction coefficients (i.e. weights) $M_{n,\ell,\ell'}$ for each node using the triads that connect to it. The idea is to go over the list of triads, and for example when treating the triad $(n, n+m, n+m+1)$, add the three interactions $\mathbf{i}_n = \{n+m, n+m+1\}$, $\mathbf{i}_{n+m} = \{n, n+m+1\}$ and $\mathbf{i}_{n+m+1} = \{n, n+m\}$ to the list of interactions, with the corresponding interaction coefficients $M_{n,n+m,n+m+1}$, $M_{n+m,n,n+m+1}$ and $M_{n+m+1,n,n+m}$ respectively. In the end, one may have nodes with more or less than three connections (the original number of connections of each node in the GOY model), and these connections may be local or nonlocal, but since the contributions from each triad to all of its three nodes are always considered, the conservation laws are automatically respected. The model goes from the regular shell model with local interactions for $p = 0$ to a cascade model with random scale interactions for $p = 1$. In contrast, p_f does not play an important role on network topology, so one could pick all the connections to be forward without loss of generality.

Once the network is constructed, time evolution of the node variables u_n

can be written as:

$$\left(\partial_t + \nu k_n^2\right) u_n = \sum_{\ell,\ell'=\mathbf{i}_n} M_{n,\ell,\ell'} u_\ell^* u_{\ell'}^* \; + f_n, \tag{3}$$

where $\mathbf{i}_n$ is the list of interaction pairs for the nth node, and $M_{n,\ell,\ell'}$ are the interaction coefficients, ν is kinematic viscosity and f_n is (localized and random) forcing.

A static network that is integrated for a certain number of time steps is not a particularly interesting exercise since the result relies on initialization and how the network is wired. The resulting spectrum is a considerably rugged version of the shell model one, as seen in fig. 2, with barriers around nodes that are missing connections. Furthermore each time the network is rewired, the details of how it deviates from the regular shell model would change. Figure 3 shows the particular wiring in more detail that leads to the spectrum shown in figure 2. Notice that some nodes are missing connections, and the energy has difficulty going through those.

A more realistic approach is to rewire the network in regular time intervals (i.e. Δt) as the system evolves. If we run such a model for a long enough time $t \gg \Delta t$ we can obtain good statistics. Various interesting problems related to shell models, such as intermittency etc. can be studied in such a formulation. Note that in order not to completely randomize the network in a few time steps, we apply the WS strategy on the original network and not the modified one at t in order to obtain the network structure at $t + \Delta t$. The results for this dynamical network formulation using the WS strategy can be seen in section 3, along with results for the other strategies.

2.2. Newman-Watts model

Newman and Watts proposed an alternative algorithm for constructing a similar partially randomized network from a regular initial lattice [36]. It translates to shell models as *adding* a non-local triad instead of replacing the local one as in WS, with, m, p and p_f having the same roles as before. We call this, the Newman-Watts strategy or NW for short. The steady-state wavenumber spectrum on a network obtained by this algorithm is shown in figure 4, where the network itself is shown in figure 5. Note that since the algorithm simply adds connections and the interaction coefficients for those nonlocal connections go down as g^{-m}, the result is very similar to GOY.

The primary advantage of the NW is that it keeps the basic structure of the underlying regular local lattice. This allows the basic local transfers to

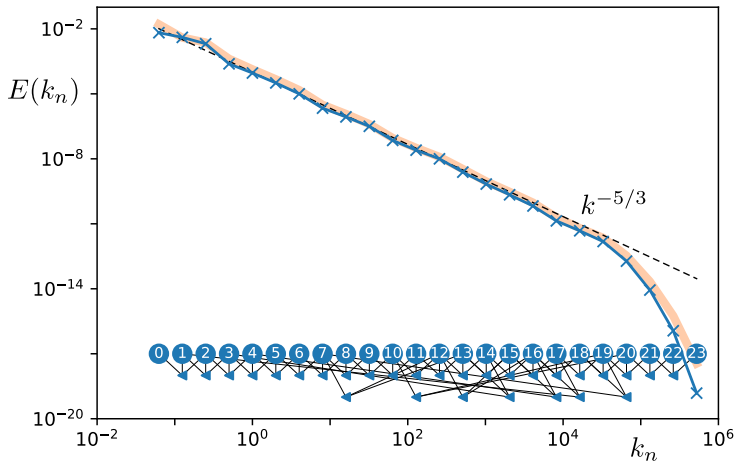

Figure 4: Wave number spectrum for the NW network, generated with $p = 0.4$, $p_f = 1.0$ compared with the GOY model. All the nodes have at least 3 connections therefore no barriers appear. Also the additional connections dissipative range make the spectrum fall off a bit faster. Parameters and run times are the same as in figure 2.

always be present, giving a smoother steady state spectrum. Nonetheless, evolution of the turbulent cascade on a single network instance is not particularly applicable to real turbulence. The more relevant formulation is the dynamically rewired model as in the previous case, which rewires itself from the original regular lattice in regular time intervals as the turbulent cascade proceeds. The results of this dynamically rewired model are shown below in section 3.

2.3. Bipartite Networks of Wavenumbers and Triads for Describing Turbulence

We have introduced the dynamical complex network models using the concept of interactions between nodes and pairs, saying that each node interacts with a list of pairs of other nodes. Those interactions come from triadic interactions, are represented by triads and are thus *"non-pairwise"*.

The graphs of networks in figures 1, 3, and 5 show these triads explicitly, where each node is shown to be connected to a number of triads. This graphical representation is not a coincidence and hints at the underlying nature of the kind of networks that appear in the spectral description of turbulence: Such networks with three body interactions can alternatively be represented as bipartite networks [37], which are networks that exclusively

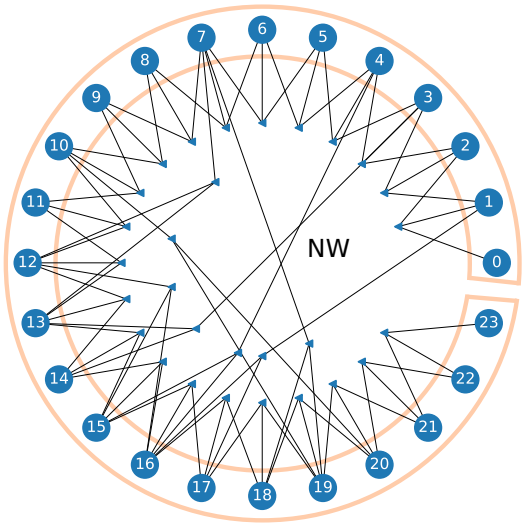

Figure 5: The small world network of the NW model, generated with $p = 0.4$, $p_f = 1.0$, which is the same network as the one in Figure 4.

connect two separate kinds of nodes. In this case, the two separate kinds of nodes are *"wavenumber nodes"* representing some part of the wavenumber domain and *"triad nodes"* representing triadic interactions among those, with the additional constraint that each triad has exactly three connections.

This perspective allows us to transform the three body interaction networks implicit in the turbulent cascade into the simpler and well known class of networks, which are bipartite networks and ask common questions in network topology such as average distance, clustering coefficients, or degree distributions using a more standard formulation, since in this formulation the connection between a wavenumber node and a triad node becomes a simple pairwise connections. In particular, using the bipartite network but focusing on the wavenumber nodes, we can construct a projected simple (or multi or weighted) graph network as discussed in Ref. 37, where the nodes in the projected network are connected only if they are both connected to the same triad.

3. Numerical Results

Here we focus on the results from the dynamical network models discussed above, which are rewired according to either WS or NW in regular intervals of Δt. Unlike the static cases, there is no big difference between the two in

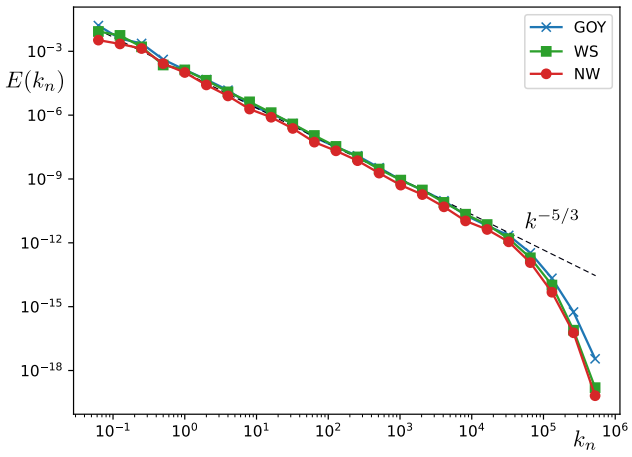

Figure 6: The resulting steady state spectra from the dynamical complex network models WS and NW compared with that of the GOY model, showing that all three models basically capture the $k^{-5/3}$ spectrum that we expect, while NW is very slightly lower in amplitude as opposed to the other two, probably as a result of its extra connections, and therefore higher transfer efficiency. The parameters for these runs are discussed in the text.

terms of their steady state spectra as shown in figure 6, since an evolving network moves its barriers around, and as a result, allows energy transfer, more easily. The results shown in this section uses the parameters $N = 24$, $k_0 = 2^{-4}$, $\nu = 10^{-8}$ and $f_n = (\delta_{n1}\xi_1 + \delta_{n2}\xi_2)\,10^{-2}$, where ξ_i are random variables with a correlation time of 10^{-2}. The spectra are integrated up to $t = 5\times10^3$ using an adaptive fourth order Runge Kutta solver [38], and when steady state results are shown they are usually averaged over $t = [3-5]\times10^3$.

The initial phase of the time evolution for different models can be seen in Figure 7. Here, the nodes with less than three connections act as barriers in the static WS case. This results in a slower buildup and a very noisy final spectrum as shown in figure 2. In contrast, in the static NW case, non-local connections weaken the initial broadening of the energy spectrum around the production region by coupling directly to small scales that are strongly dissipative. This results in a slower buildup as well. However since there are no barriers in NW, the final state is roughly the same as that of GOY. In contrast, since the network evolution time scale $\Delta t = 10^{-2}$ is much faster compared to the time it requires to reach steady state, the evolving network acts as a halo connecting all the nodes to one another, speeding up the redistribution of energy. Changing Δt has a nontrivial impact on the

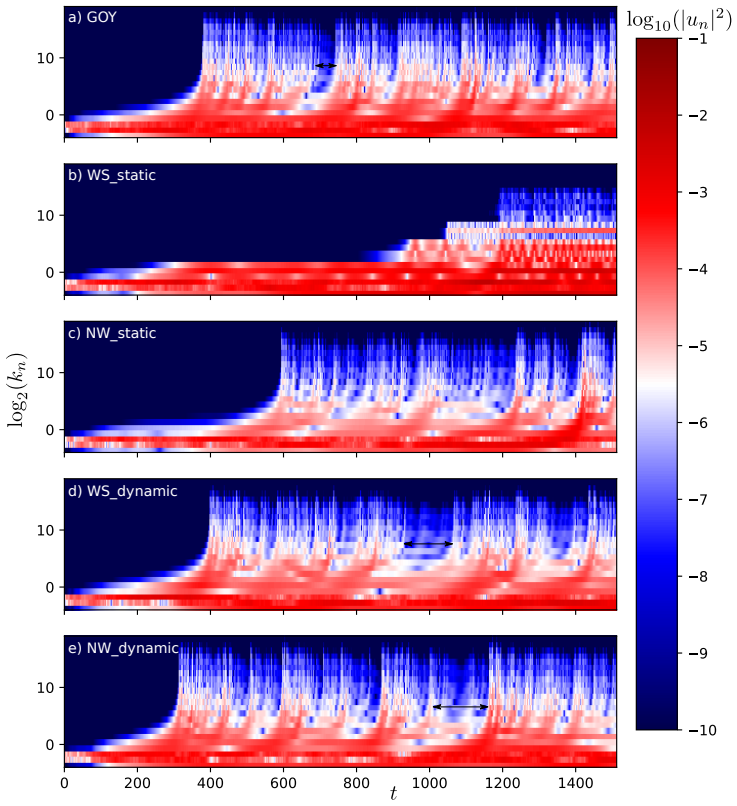

Figure 7: Time evolution of the wave number spectrum, up to $t = 1500$ for a) GOY model, b) static WS network (figure 2), c) static NW network (figure 4), d) dynamic WS network, e) dynamic NW network. Here the x axis is the time, and the y axis is $\log_2(k_n) = n - 2$, where n is the shell index. Barriers that we see in b) are due to nodes with missing connections. It is interesting that while both (b) and (c) are slower to settle to the steady state than (a), both (d) and (e) are faster or same. It also seems that (c), (d), and (e) all have slightly different dynamics from (a) in that they seem to spend more time with energy localized mainly at large scales, which appear as blue gaps around $\log_2(k_n) \approx 10$. We see these gaps for instance between $t = 1000$ and $t = 1200$ in (e) and $t = 900$ and $t = 1050$ in (d). The equivalent gap we see in (a) around $t = 650$ is much narrower in comparison.

dynamics of WS, but not so for NW. Since WS has barriers, how long those stay in one place affects the dynamics. We do not explicitly show plots for different values of Δt here, but this can be seen from the difference between the static (i.e. $\Delta t \to \infty$) vs. the dynamic network versions of the WS shown in figure 7.

Another interesting tool in understanding the dynamics of the turbulent cascade is the structure function, which gives information about the scale by scale distribution of statistical features of the flow field. The shell model equivalent of the ℓth order structure function can be written as $S_\ell^n = \left\langle |u_n|^\ell \right\rangle$ where the average is to be computed over time. Assuming that it has a power law form: $S_\ell^n \propto k_n^{-\xi_\ell}$, one can obtain ξ_ℓ by considering $y_\ell = \log_{10}(S_\ell^n)$, and $x = \log_{10}(k_n)$ and making a linear regression to obtain $y_\ell = a_\ell + b_\ell x$, so that $\xi_\ell \approx -b_\ell$. When we plot this as a function of ℓ as in figure 8, its deviation from the theoretical estimate, $\xi_\ell = \frac{\ell}{3}$ gives us an indication of the intermittency. Somewhat expectedly, the intermittency increases when the ratio of nonlocal to local connections increase.

The GOY model appears to be rather successful in capturing the key features of intermittency [39, 40], at least for a particular choice of its parameters [41], which is thought to be due to instanton dynamics [42]. Therefore including non-local interactions that rewire randomly, increasing intermittency, is not really very useful. However other models [43, 44, 45], more complex than GOY that can address various aspects of turbulence, including anisotropy, may sometimes lack intermittency corrections. It would be interesting to devise similar modifications for these models.

Note finally that in contrast to the case of stochastically perturbed shell models [46] the intermittency increase that we have here due to stochastic evolution of the network structure (i.e. via randomly introduced long-range interactions) without any explicit perturbation of the nonlinear (or other) terms in the model itself, is rather significant.

4. Network Extraction

Since the complex network approach allows us to evolve the connections of a shell model, it would be interesting to develop a method for extracting this evolving network structure from direct numerical simulations. As discussed briefly in the introduction this can be achieved by dividing the k-space of the problem into logarithmically discretized shells, and computing filtered energy

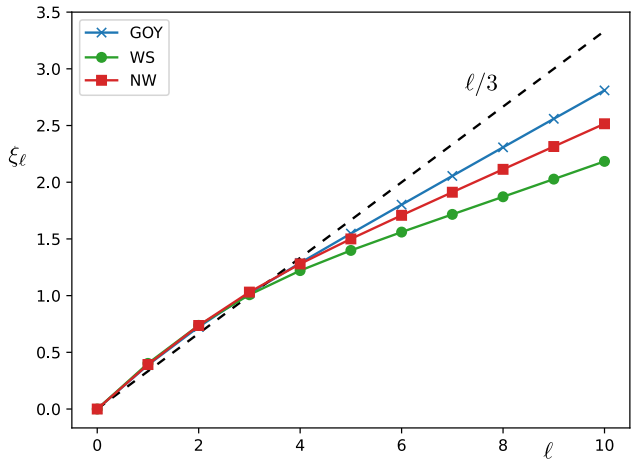

Figure 8: Intermittency in dynamical complex network models. It seems that intermittency increase as the ratio of random non-local connections to local nearest neighbor connections increase. NW increase this ratio by adding non-local interactions, whereas WS increase it further since it also removes local connections as it adds non-local ones.

transfers to see which sets of "triads" interact strongly (i.e. connected) in the DNS.

In order to develop such a method, let's go back to the Navier Stokes equation in Fourier space, which can be written as:

$$\left(\partial_t + \nu k^2\right) \mathbf{u}_k + \mathrm{NL}_k = F_k \ , \tag{4}$$

where

$$\mathrm{NL}_k = \left[\mathcal{P}\left(\mathbf{u}\cdot\nabla\mathbf{u}\right)\right]_k = \mathcal{F}\mathcal{P}\left[\mathcal{F}^{-1}\left(\mathbf{u}_k\right)\cdot\mathcal{F}^{-1}\left(i\mathbf{k}\mathbf{u}_k\right)\right]$$

is the nonlinear term (with $\nabla\cdot\mathbf{u}=0$), $\mathcal{P}\equiv\left(I-\nabla\nabla^{-2}\nabla\right)$ is the projection operator and $\mathcal{F}$ is the Fourier transform, which would be implemented using fast Fourier transforms(FFTs) in a pseudo-spectral formulation. One would normally pad the inverse FFTs with the $2/3$ padding rule and truncate the result after the forward FFT.

We can define a shell filter (i.e. a window) as:

$$W_k^n\left[\mathbf{u}_k\right] = \mathbf{u}_k^n \ ,$$

where

$$W_k^n \equiv \left[\Theta\left(|k|>k_n\right) - \Theta\left(|k|<k_{n+1}\right)\right]$$

in terms of the Heaviside step functions $\Theta\left(x\right)$, which basically sets all the Fourier coefficients outside the nth shell to zero. Using the filter, multiplying

(4) by $\mathbf{u}_k^*$, taking the real part, and summing over the wavenumber k one can write the shell by shell energy budget as:

$$\frac{d}{dt}E_n + \sum_{\ell,m} T^E_{n\ell m} = F_n - D_n \ , \tag{5}$$

where

$$E_n \equiv \frac{1}{2}\sum_k |\mathbf{u}_k^n|^2 = \frac{1}{2}\sum_{k_n<k<k_{n+1}} |\mathbf{u}_k|^2$$

is the energy in the shell n and

$$T^E_{n\ell m} = \mathrm{Re}\sum_k \mathbf{u}_k^{*n}\mathrm{NL}_k^{n\ell m}$$

is the energy transfer between the three (potentially) interacting shells n, ℓ and m. We would compute $T_{n\ell m}$ in practice, by:

$$\mathrm{T}^E_{n\ell m} = \mathrm{Re}\sum_k \mathbf{u}_k^{*n}\mathcal{F}\mathcal{P}\left[\mathcal{F}^{-1}\left(\mathbf{u}_k^\ell\right)\cdot\mathcal{F}^{-1}\left(i\mathbf{k}\mathbf{u}_k^m\right)\right] \tag{6}$$

using band pass filtered velocity fields $\mathbf{u}_k^n$, $\mathbf{u}_k^\ell$ and $\mathbf{u}_k^m$ each one filtered in a different shell.

Note that for a DNS of a resolution of 1024^3, we would have $N = 10$ shells (since $g = 2$, and $g^{10} \to 1024$), and $N^3 = 1000$ transfer terms $\mathrm{T}^E_{n\ell m}$ to compute. Since each combination of $n\ell m$ requires a full pseudo-spectral convolution computation in (6), (and a sum over k) and since the more expensive part of the computation in a pseudo-spectral code is the computation of this nonlinear term, computing $T_{n\ell m}$ in this way would cost up to $N^2 = 100$ times more than taking a single time step (since we have to compute the FFT for each ℓ and m, but the filtering in n can be done afterwards).

Note also that, since energy is conserved, we must have

$$\sum_{n,\ell,m} T^E_{n\ell m} = 0 \ ,$$

which can be used to verify the numerical accuracy or possible double counting errors. Note that in order to achieve this, the first shell should include the origin, and the last shell should extend up to end of the domain so that the partition covers the numerical domain completely. One should of course

treat the origin carefully in order not to cause a division by zero, for example when solving the Poisson equation in 2D, or applying the projection operator in 3D.

Once we obtain the $T_{n\ell m}$ from the DNS, we can compare it to the energy budget of the shell model:

$$\frac{d}{dt}E_n + \sum_{\ell,m} M_{n\ell m} E_n^{1/2} E_m^{1/2} E_\ell^{1/2} = F_n - D_n \ , \tag{7}$$

which can be obtained by multiplying (3) by u_n^* and using $E^{1/2}_{\{n,\ell,m\}} \to u_{\{n,\ell,m\}}$ dropping all the phases, which can be justified based on the fact that here we want to determine if $M_{n\ell m}$ is nonzero (or above a certain threshold) and not compute its exact value. For this, the complex phase of u_n is probably irrelevant. Recall also that the complex phases in a shell model are somewhat arbitrary and there is no obvious way to obtain them from the DNS.

Comparing (5) to (7), we can define the interaction coefficient as

$$M_{n\ell m} = \frac{T^E_{n\ell m}}{E_n^{1/2} E_\ell^{1/2} E_m^{1/2}} \ ,$$

as long as $\{E_n, E_\ell, E_m\} > \epsilon$ (where ϵ is a small number), and zero otherwise. We can also define a threshold value δ and add the connection $\{n, \ell, m\}$ to the list of connections if $|M_{n\ell m}| > \delta$. Finally it is probably important to check the connections for consistency. In other words if we add the connection $n \leftarrow \ell, m$, we should automatically add the other two connections $\ell \leftarrow m, n$ and $m \leftarrow n, \ell$. This can be achieved easily by adding these connections into a list of triads, and then constructing the list of connected pairs for each n using this list of triads.

4.1. Two dimensional turbulence

While the mathematical formulation is *a priori* independent of the number of dimensions, in order to demonstrate the approach, it is simpler to use a two dimensional Navier-Stokes system, as it is easier to run in higher resolutions and simpler to visualize than 3D. However the network model is slightly different in 2D since we can use the stream-function Φ instead of the velocity field, which is linked to it via $\mathbf{u} = \hat{\mathbf{z}} \times \nabla\Phi$ in order to formulate the problem.

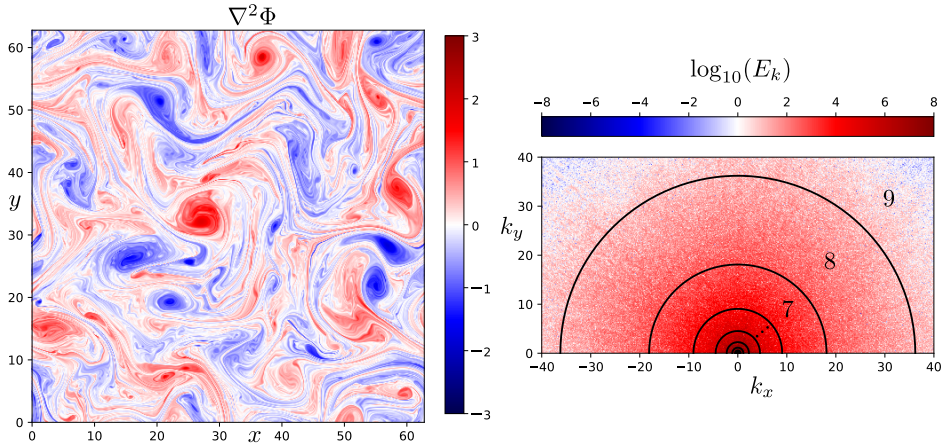

Figure 9: A snapshot (at $t = 1000$) of the vorticity field (on the left) and its energy (on the right) where the k-space shells that are used for bandpass filtering are explicitly shown.

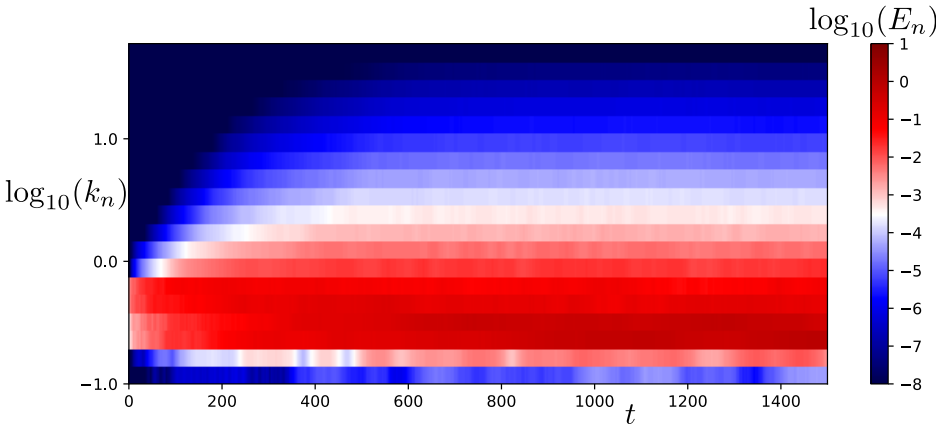

Figure 10: The steady state spectra from the DNS. Since this is 2D Navier-Stokes driven at large scales, the resulting k-spectrum is steeper with $E(k) \propto k^{-3.2}$. Here we plot the spectrum using a logarithmically spaced grid (using $g = 1.4$ to reduce pixellization). Notice that the range of wavenumbers covered by the DNS (with a resolution of 2048×2048) is much smaller than the shell model.

For two dimensional turbulence, which can be formulated in real space using the 2D Navier-Stokes equations:

$$\partial_t \nabla^2 \Phi + \hat{\mathbf{z}} \times \nabla\Phi \cdot \nabla\nabla^2\Phi = \nu\nabla^4\Phi + F$$

the network model can be written as:

$$\left(\partial_t + \nu k_n^2\right)\Phi_n = \sum_{\ell,m=\mathbf{i}_n} M_{n,\ell,m}\Phi_\ell^*\Phi_m^* \ + f_n, \tag{8}$$

with

$$M_{n,\ell,m} = \begin{cases} \frac{k_m^2 - k_\ell^2}{k_n^2} & \ell > n \text{ or } m < n \\ \frac{k_\ell^2 - k_m^2}{k_n^2} & \ell < n < m \end{cases} \tag{9}$$

for $m > \ell$. This means that we can write (using $E_n = k_n^2\Phi_n^2$ for the shell energy in terms of the shell variables Φ_n):

$$\left(\partial_t + \nu k_n^2\right)E_n = \sum_{\ell,m=\mathbf{i}_n} M_{n,\ell,m}\Phi_\ell^*\Phi_m^*\Phi_n^* k_n^2 \ ,$$

where $M_{n\ell m}$ can be computed either using energy

$$\widetilde{M}_{n\ell m}^E = \frac{\mathrm{T}_{n\ell m}^E}{E_n^{1/2}E_\ell^{1/2}E_m^{1/2}}\frac{k_\ell k_m}{k_n} \tag{10}$$

or using enstrophy:

$$\widetilde{M}_{n\ell m}^W = \frac{\mathrm{T}_{n\ell m}^W}{E_n^{1/2}E_\ell^{1/2}E_m^{1/2}}\frac{k_\ell k_m}{k_n^3} \ . \tag{11}$$

While for a shell model, we have $\widetilde{M}_{n\ell m}^E = \widetilde{M}_{n\ell m}^W = M_{n\ell m}$, since the coupling coefficients are given by (9), for the DNS these two may not give exactly the same results (although we find that they are generally close). The two dimensional energy and enstrophy transfers can be computed explicitly in a pseudo-spectral code using:

$$\mathrm{T}_{n\ell m}^E \equiv \mathrm{Re}\sum_k \Phi_k^{*n}\mathcal{F}\left[\mathcal{F}^{-1}\left(\hat{\mathbf{z}} \times i\mathbf{k}\mathbf{\Phi}_k^\ell\right)\cdot\mathcal{F}^{-1}\left(i\mathbf{k}k^2\Phi_k^m\right)\right] \tag{12}$$

$$\mathrm{T}_{n\ell m}^W \equiv \mathrm{Re}\sum_k k^2\Phi_k^{*n}\mathcal{F}\left[\mathcal{F}^{-1}\left(\hat{\mathbf{z}} \times i\mathbf{k}\mathbf{\Phi}_k^\ell\right)\cdot\mathcal{F}^{-1}\left(i\mathbf{k}k^2\Phi_k^m\right)\right] \ , \tag{13}$$

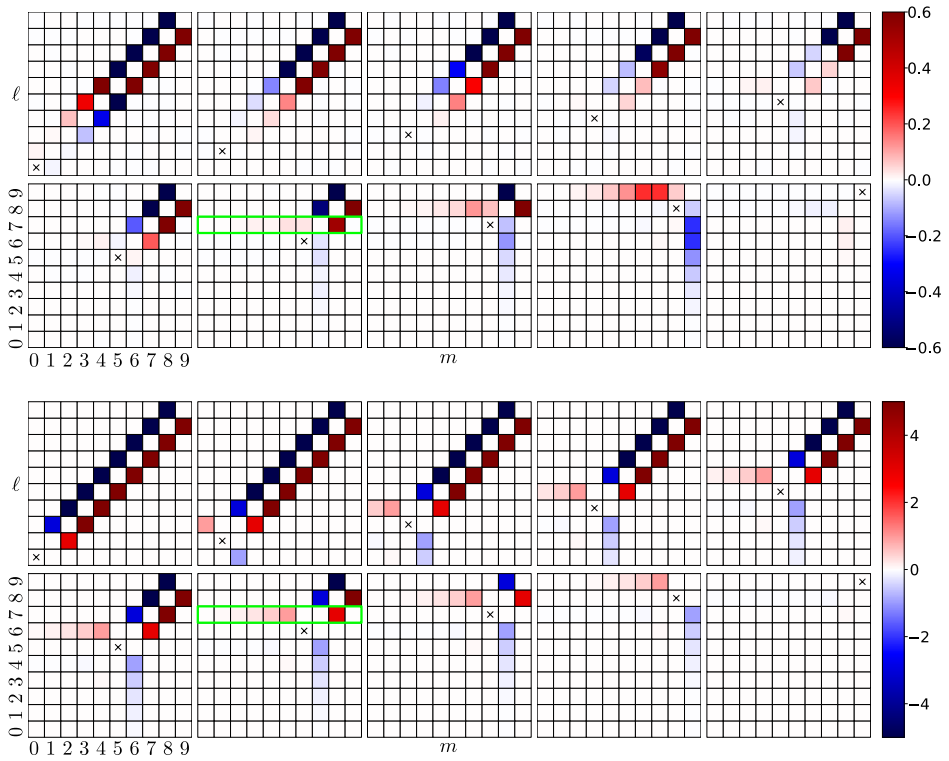

Figure 11: The snapshot of the interaction coefficients as computed from (11) using the DNS data at $t = 1500$ (top figure) for each n (from 0 to 9) from top left to bottom right. For each n, the diagonal matrix element $m = \ell = n$ is shown with a cross. Local interactions are those that are close to the crosses, whereas nonlocal ones are the ones that are far from them. The bottom figure shows the analytical expression that we use in (9) for comparison, with all the possible interactions active. The green boxes in both figures show the $n = 6$, $\ell = 7$ cross-section that we look at in more detail in figure 12.

where $\Phi_k^{*n} = W_k^n [\Phi_k]$ are the *band pass* filtered Fourier transforms of the stream function in the nth shell. Since the expression (13) is not symmetrical with respect to the exchange of ℓ and m, when it is used in (11) in order to compute $M_{n\ell m}$ one has to "symmetrize" the result, and since the exchange of the two wave-vectors k_ℓ and k_m of an interacting triad results in its reflection, this can be done by considering $M_{n\ell m} = \left(\widetilde{M}^W_{n\ell m} - \widetilde{M}^W_{nml}\right)/2$ which is the form to be compared with (9).

Here we used a 2D pseudo-spectral Navier-Stokes solver, developed in python using numpy [47] and scipy's [48] RK45 solver (a 4th order Runge Kutta solver with error estimation and adaptive time stepping) and numba

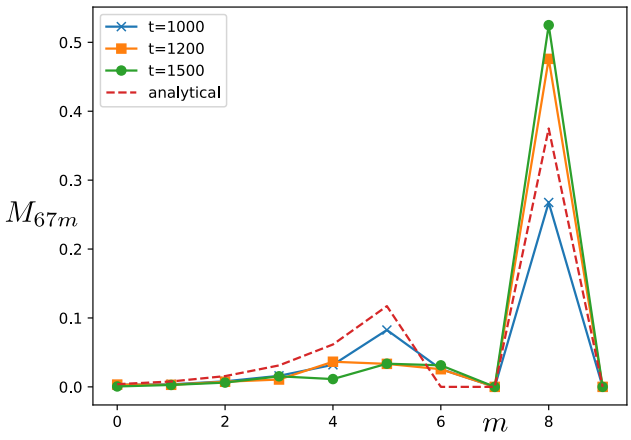

Figure 12: Snapshots of the interaction coefficients for $n = 6$, $\ell = 7$ as computed from (11) using the DNS data at different times compared with the analytical expression. Here we show $M_{67m}/8$ for the analytical expression so that it is roughly of the same with the others. We clearly observe some time evolution of the interaction coefficients, which can be represented as an evolution of the underlying network.

for number crunching [49]. We compute the convolutions using a python wrapper to the fftw [50] with the 2/3 padding rule. The results that are presented here correspond to a padded resolution of 2048×2048 and box size of $Lx, Ly = 2\pi/k_{x0}, 2\pi/k_{y0}$, where $k_{x0} = k_{y0} = 0.1$, with a random forcing between $k = 0.2$ to $k = 0.6$ (with an amplitude $F_A = 10^4$) and a dissipation function of the form:

$$D(k^2) = \nu k^4 + \nu_L k^{-8}$$

with $\nu = \nu_L = 10^{-8}$. This setup is suitable for observing and studying the forward enstrophy cascade which gives a k-spectrum of the form $E(k) \propto k^{-3}$. The simulation has been run up to $t = 1500$, with a saturation roughly around $t = 400$ (see figure 10). Note that the range of wavenumbers in such a run corresponds to about 10 shells with $g = 2$. In order to obtain 24 shells with $g = 2$ as in figure 7, we would need a padded resolution of about $[2^{25}]^2 = 33,554,432^2$ for the DNS. This is clearly out of reach of our computing capabilities, if not impossible all together.

Using the method outlined previously, we can compute the transfer of enstrophy -since it is the enstrophy that cascades here, it seemed more logical to use enstrophy, but using energy gives roughly the same result- of the shell filtered Φ_k variable. The coefficients of interaction computed using (11) can

be seen in figure 11. This is actually the snapshot at $t = 1500$, we also show the cross-section M_{67m} of the interaction coefficient as a function of m at different times in figure (12), which shows that the network of three body interactions represented by the $M_{n\ell m}$ evolves in time. Using a threshold δ, we can also use $M_{n\ell m}/M_{n\ell m}^{\text{analytical}} > \delta$ in order to determine wether or not the particular interaction is active at a given instant.

5. Conclusion

We have introduced the concept of complex network models of turbulent cascade as a generalization of the GOY model to arbitrary networks, where the shells represent nodes and the network consist of three body connections among them. This structure can be identified alternatively as a bipartite network between wavenumber nodes and triad nodes, where each triad is connected to exactly three separate wavenumber nodes. The approach allows us to decouple the setting up of the evolution of the network topology from the evolution of the node variables $u_n(t)$ on the network.

We have discussed two basic strategies of network wiring based on replacing existing local interactions by nonlocal ones (WS) or adding nonlocal interactions (NW) on top of the existing connections. While static results show an exaggerated effect of the network topology on the cascade, especially through the appearance of k-space barriers, it is shown that dynamically evolving network models do not have this issue.

In fact, when the network is dynamically rewired from an original regular lattice with a time step Δt, we find that for $\Delta t \sim \delta t$, where δt is the correlation time of the forcing, we get almost exactly the same k-spectrum but slightly higher intermittency, observable both in terms of temporal dynamics (i.e. appearance of larger gaps in the time evolution) and when it is computed properly using deviation of the scaling of higher-order structure functions from Kolmogorov theory. We find that in particular for the WS case, how fast the network evolves plays an important role in both the dynamics and in the final steady state result. Since WS can have wavenumber nodes with a degree less than 3, it can produce barriers for the energy cascade, and how long those barriers remain in one place is detrimental to the evolution of the spectrum.

We have also developed a method for extracting the network topology from a fully resolved DNS by using the shell-to-shell energy transfer in order to estimate the triple interaction coefficient $M_{n\ell m}$ between three shells

in k-space. While the method can be applied, in principle to any k-space partition and any number of dimensions, in order to demonstrate it, we used a two dimensional Navier-Stokes pseudo-spectral simulation, computing the analytical interaction coefficient for this case and using shell-to-shell transfers in order to estimate the interaction coefficients. We argue that using either $M_{n\ell m} > \delta$ or $M_{n\ell m}/M_{n\ell m}^{\text{analytical}} > \delta$, one can decide whether or not a particular interaction is active at a given instant.

Various obvious extensions, such as the use of preferential attachment strategies that lead to scale-free networks have been left to future studies. We believe that focusing on the formulation and considering a few simple strategies allows us to perform a more detailed study and present a more coherent picture of the connection between turbulence and networks, which constitute the primary novelty of this work. Similarly a proper one-to-one comparison of an extracted network dynamics from a high resolution 3D Navier-Stokes simulation and a complex network is something that should be done in the future as it may clarify which rewiring strategy is the optimum one in terms of its ability to reproduce the dynamics of the interaction topology in real turbulence.

Acknowledgments

I would like to thank Prof. W.-C. Müller and Dr. Ö. Gültekin for fruitful discussions.

References

[1] R. Albert, H. Jeong, A. Barabási, Diameter of the world-wide web, Nature 401 (1999) 130–131. doi:10.1038/43601. 1

[2] E. Bullmore, O. Sporns, Complex brain networks: graph theoretical analysis of structural and functional systems, Nature Reviews Neuroscience 10 (2009) 186–198. doi:10.1038/nrn2575. 1

[3] K. A. Seaton, L. M. Hackett, Stations, trains and small-world networks, Physica A: Statistical Mechanics and its Applications 339 (3) (2004) 635 – 644. doi:10.1016/j.physa.2004.03.019. 1

[4] A. Kirman, The economy as an evolving network, Journal of Evolutionary Economics 7 (1997) 339 – 353. doi:10.1007/s001910050047. 1

[5] L. Biferale, Shell models of energy cascade in turbulence, Ann. Rev. Fluid Mech. 35 (2003) 441–468. doi:10.1146/annurev.fluid.35.101101.161122. 1, 2

[6] C. E. Leith, Diffusion approximation to inertial energy transfer in isotropic turbulence, The Physics of Fluids 10 (7) (1967) 1409–1416. doi:10.1063/1.1762300. 1

[7] M. Lesieur, Turbulence in Fluids, third edition Edition, Kluwer, Dordrecht, 1997. 1

[8] U. Frisch, Turbulence: The Legacy of A. N. Kolmogorov, Cambridge University Press, Cambridge, 1995. 1

[9] A.-L. Barabasi, The network takeover, Nature Physics 8 (2011) 14–16. doi:10.1038/nphys2188. 1

[10] K. Taira, A. G. Nair, S. L. Brunton, Network structure of two-dimensional decaying isotropic turbulence, Journal of Fluid Mechanics 795 (2016) R2. doi:10.1017/jfm.2016.235. 1

[11] Ö. D. Gürcan, Y. Li, P. Morel, Turbulence as a network of fourier modes, Mathematics 8 (4) (2020) 530. doi:10.3390/math8040530.
URL http://dx.doi.org/10.3390/math8040530 1

[12] K. L. Calvert, M. B. Doar, E. W. Zegura, Modeling internet topology, IEEE Communications Magazine 35 (6) (1997) 160–163. doi:10.1109/35.587723. 1

[13] K. Lee, W.-S. Jung, J. S. Park, M. Choi, Statistical analysis of the metropolitan seoul subway system: Network structure and passenger flows, Physica A: Statistical Mechanics and its Applications 387 (24) (2008) 6231 – 6234. doi:10.1016/j.physa.2008.06.035. 1

[14] R. Kali, J. Reyes, Financial contagion on the international trade network, Economic Inquiry 48 (4) (2010) 1072–1101. doi:10.1111/j.1465-7295.2009.00249.x. 1

[15] M. J. Keeling, K. T. Eames, Networks and epidemic models, Journal of The Royal Society Interface 2 (4) (2005) 295–307. doi:10.1098/rsif.2005.0051. 1

[16] S. N. Dorogovtsev, A. V. Goltsev, J. F. F. Mendes, Critical phenomena in complex networks, Rev. Mod. Phys. 80 (2008) 1275–1335. doi:10.1103/RevModPhys.80.1275. 1

[17] D. J. Watts, A simple model of global cascades on random networks, Proceedings of the National Academy of Sciences 99 (9) (2002) 5766–5771. doi:10.1073/pnas.082090499. 1

[18] L. Neuhäuser, A. Mellor, R. Lambiotte, Multibody interactions and nonlinear consensus dynamics on networked systems, Phys. Rev. E 101 (2020) 032310. doi:10.1103/PhysRevE.101.032310.
URL https://link.aps.org/doi/10.1103/PhysRevE.101.032310 1

[19] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, M. Lucas, A. Patania, J.-G. Young, G. Petri, Networks beyond pairwise interactions: Structure and dynamics, Physics Reports 874 (2020) 1 – 92, networks beyond pairwise interactions: Structure and dynamics. doi:10.1016/j.physrep.2020.05.004.
URL https://doi.org/10.1016/j.physrep.2020.05.004 1

[20] M. G. Meena, Network community-based analysis of complex vortical flows:Laminar and turbulent flows, Ph.D. thesis, UCLA (2020).
URL https://escholarship.org/uc/item/8tt3q4r7 1

[21] S. Hall, D. Raffaelli, Food-web patterns: Lessons from a species-rich web, Journal of Animal Ecology 60 (3) (1991) 823–841.
URL http://www.jstor.org/stable/5416 1

[22] N. D. Martinez, Artifacts or attributes? effects of resolution on the little rock lake food web, Ecological Monographs 61 (4) (1991) 367–392. doi:10.2307/2937047.
URL https://doi.org/10.2307/2937047 1

[23] P. A. Marquet, R. A. Quiñones, S. Abades, F. Labra, M. Tognelli, M. Arim, M. Rivadeneira, Scaling and power-laws in ecological systems, Journal of Experimental Biology 208 (9) (2005) 1749–1769. doi:10.1242/jeb.01588.
URL https://jeb.biologists.org/content/208/9/1749 1

[24] P. H. Diamond, Y.-M. Liang, B. A. Carreras, P. W. Terry, Self-regulating shear flow turbulence: A paradigm for the L to H transition, Phys. Rev.

Lett. 72 (16) (1994) 2565–2568. doi:10.1103/PhysRevLett.72.2565.
URL https://doi.org/10.1103/PhysRevLett.72.2565 1

[25] Ö. D. Gürcan, P. H. Diamond, Zonal flows and pattern formation, Journal of Physics A: Mathematical and Theoretical 48 (29) (2015) 293001. doi:10.1088/1751-8113/48/29/293001. 1

[26] N. Goldenfeld, H.-Y. Shih, Turbulence as a problem in non-equilibrium statistical mechanics, Journal of Statistical Physics 167 (2017) 575, 594. doi:10.1007/s10955-016-1682-x.
URL https://doi.org/10.1007/s10955-016-1682-x 1

[27] V. Berionni, Ö. D. Gürcan, Predator prey oscillations in a simple cascade model of drift wave turbulence, Physics of Plasmas 18 (11) (2011) 112301. doi:10.1063/1.3656953. 1

[28] S. Kobayashi, Ö. D. Gürcan, P. H. Diamond, Direct identification of predator-prey dynamics in gyrokinetic simulations, Physics of Plasmas 22 (9) (2015) 090702. doi:10.1063/1.4930127. 1

[29] K. Ohkitani, M. Yamada, Temporal intermittency in the energy cascade process and local lyapunov analysis in fully-developed model turbulence, Progress of Theoretical Physics 81 (2) (1989) 329–341. doi:10.1143/PTP.81.329. 1, 2

[30] M. E. J. Newman, The structure and function of complex networks, SIAM Review 45 (2) (2003) 167–256. doi:10.1137/S003614450342480.
URL https://doi.org/10.1137/S003614450342480 1

[31] J. C. Bowman, B. A. Shadwick, P. J. Morrison, Spectral reduction: A statistical description of turbulence, Phys. Rev. Lett. 83 (1999) 5491–5494. doi:10.1103/PhysRevLett.83.5491. 1

[32] A. C. Newell, B. Rumpf, Wave turbulence, Annual Review of Fluid Mechanics 43 (1) (2011) 59–78. doi:10.1146/annurev-fluid-122109-160807.
URL https://doi.org/10.1146/annurev-fluid-122109-160807 1

[33] E. Kartashova, Nonlinear resonance analysis, Nonlinear Resonance Analysis, by Elena Kartashova, Cambridge, UK: Cambridge University Press, 2010 1 (2010). 1

[34] F. Plunian, R. Stepanov, A non-local shell model of hydrodynamic and magnetohydrodynamic turbulence, New Journal of Physics 9 (8) (2007) 294. doi:10.1088/1367-2630/9/8/294. 2

[35] D. J. Watts, S. H. Strogatz, Collective dynamics of small-world networks, Nature 393 (1998) 440–442. doi:10.1038/30918. 2.1

[36] M. E. J. Newman, D. J. Watts, Scaling and percolation in the small-world network model, Phys. Rev. E 60 (1999) 7332–7342. doi:10.1103/PhysRevE.60.7332. 2.2

[37] D. Vasques Filho, D. R. J. O'Neale, Degree distributions of bipartite networks and their projections, Phys. Rev. E 98 (2018) 022307. doi:10.1103/PhysRevE.98.022307.
URL https://link.aps.org/doi/10.1103/PhysRevE.98.022307 2.3

[38] Ö. D. Gürcan, Dycon : Dynamical complex network models for turbulence (May 2020). doi:10.17605/OSF.IO/FW6XA.
URL osf.io/fw6xa 3

[39] R. Benzi, L. Biferale, G. Parisi, On intermittency in a cascade model for turbulence, Physica D: Nonlinear Phenomena 65 (1) (1993) 163 – 171. doi:10.1016/0167-2789(93)90012-P. 3

[40] D. Pisarenko, L. Biferale, D. Courvoisier, U. Frisch, M. Vergassola, Further results on multifractality in shell models, Physics of Fluids A 5 (10) (1993) 2533–2538. doi:10.1063/1.858766. 3

[41] J. C. Bowman, C. R. Doering, B. Eckhardt, J. Davoudi, M. Roberts, J. Schumacher, Links between dissipation, intermittency, and helicity in the goy model revisited, Physica D: Nonlinear Phenomena 218 (1) (2006) 1–10. doi:10.1016/j.physd.2006.01.028.
URL https://doi.org/10.1016/j.physd.2006.01.028 3

[42] A. A. Mailybaev, Computation of anomalous scaling exponents of turbulence from self-similar instanton dynamics, Phys. Rev. E 86 (2012) 025301(R). doi:10.1103/PhysRevE.86.025301. 3

[43] Ö. D. Gürcan, Nested polyhedra model of turbulence, Phys. Rev. E 95 (2017) 063102. doi:10.1103/PhysRevE.95.063102. 3

[44] Ö. D. Gürcan, Nested polyhedra model of isotropic magnetohydrodynamic turbulence, Phys. Rev. E 97 (6) (2018) 063111. doi:10.1103/PhysRevE.97.063111. 3

[45] Ö. D. Gürcan, S. Xu, P. Morel, Spiral chain models of two-dimensional turbulence, Phys. Rev. E 100 (2019) 043113. doi:10.1103/PhysRevE.100.043113.
URL https://link.aps.org/doi/10.1103/PhysRevE.100.043113 3

[46] L. Biferale, M. Cencini, D. Pierotti, A. Vulpiani, Intermittency in stochastically perturbed turbulent models, Journal of Statistical Physics 88 (1997) 1117–1138. doi:10.1007/BF02732427. 3

[47] C. R. Harris, K. J. Millman, S. J. van der Walt, et al., Array programming with NumPy, Nature 585 (7825) (2020) 357–362. doi:10.1038/s41586-020-2649-2.
URL https://doi.org/10.1038/s41586-020-2649-2 4.1

[48] P. Virtanen, R. Gommers, T. E. Oliphant, et al., SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods 17 (2020) 261–272. doi:10.1038/s41592-019-0686-2. 4.1

[49] S. K. Lam, A. Pitrou, S. Seibert, Numba: A llvm-based python jit compiler, in: Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC, LLVM '15, Association for Computing Machinery, New York, NY, USA, 2015, p. 7. doi:10.1145/2833157.2833162.
URL https://doi.org/10.1145/2833157.2833162 4.1

[50] M. Frigo, S. G. Johnson, The design and implementation of fftw3, Proceedings of the IEEE 93 (2) (2005) 216–231. doi:10.1109/JPROC.2004.840301. 4.1