Mechanical metamaterials: nonlinear beams and excess zero modes

Lubbers, L.A.

4

Topology based counting of excess zero modes

For disordered systems in 2D, the number of zero modes can be determined exactly from the connection topology [47, 48], but for symmetric systems we are not aware of general techniques to do so. Here we develop an approximate counting method for the number of zero modes in diluted symmetric systems consisting of hinging squares. We describe these systems as a collection of clusters coupled by connectors and use their topology to iteratively estimate the number of zero modes. We compare the iterative results of our topology based counting method to exact calculations based on the Hessian matrix, and show that we obtain a tight lower bound on the number of excess zero modes.

A paper based on the work presented in this chapter is in preparation for submission to Phys. Rev. Lett. as:

L.A. Lubbers and M. van Hecke, *Excess floppy modes in metamaterials with symmetries.*

4.1 Introduction

The aim of this chapter is to gain insight into the nature and multiplicity of the excess zero modes that we numerically characterized in chapter 3. First, for the generic case the number of zero modes and states of self stresses can be determined by the pebble game [47, 48], a discrete algorithm that is exact in two dimensions and for generic quads. For symmetric quads however, this algorithm is not suitable. We therefore focus on gaining an approximate, yet accurate, understanding of the number of (excess) zero modes in diluted symmetric systems, which will be topology-based.

We start from the observation that the occurrence of excess modes is driven by densely connected patches, which are rigid in the generic case but, irrespective of size, feature a hinging mode in symmetric systems. Hence, we would like to consider our systems as connected groups of quads, that we will refer to as clusters, and in section 4.2 we give precise definitions that allow us to separate any system in a collection of clusters, connectors and remaining quads. We show examples of this partition, and point out that the remaining quads do not significantly contribute to Δ, as they almost equally contribute to n_s and n_g. We therefore define so-called *pruned* systems, where the remaining quads are removed, and that solely consist of clusters and connectors — we show that Δ for the full system and Δ' for the pruned systems are extremely close in section 4.3. In section 4.4 we consider the three distinct type of connectors that arise between two clusters, and show how they constrain the number of (excess) zero modes. We find that clusters with a sufficient number of connectors should be seen as a single cluster, and develop an iterative cluster merging algorithm. Finally, in section 4.5 we apply this topology-based iterative algorithm to determine n_s' and Δ' for the pruned systems, and show very close correspondence to the numerically obtained Δ'.

4.2 Clusters on the square and dual grid

In this section we introduce the notion of 4-blocks, that, once detected, can be used to unambiguously detect clusters, connectors and remaining quads. A 4-block consist of four quads connected in a loop [Fig. 4.1(a)] and has one internal degree of freedom. A quad can belong to one or

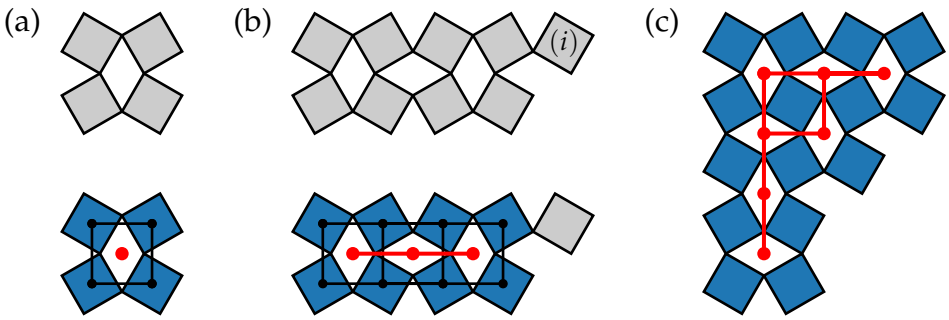

Figure 4.1: Detection of 4-blocks. Systems are initialized with grey quads, which are marked blue if they belong to a 4-block. The square and dual grid associated with the 4-blocks are respectively shown in black and red. (a-b) The top rows show unmarked quads of (a) a single 4-block and (b) a strip with one dangling quad, quad (i). The bottom rows display the associated square and dual grid, and the marked quads. (c) (Square grid not shown) As an additional example we show the dual grid of a large system that is not strip-like.

more 4-blocks, as is for example the case for the strips and loops (linearly connected 4-blocks) described in section 3.1; a quad can also belong to zero 4-blocks, such as quad (i) of the system shown in Fig. 4.1(b).

Formally, the presence of a quad in a diluted system can be expressed as the filling of a node on a square-like grid* (Fig. 4.1). Similarly, the presence of a 4-block can be expressed by filling a node on a dual, square-like grid, whose nodes lie in the center of each 4-block (Fig. 4.1). In a first step, we detect all 4-blocks, and track the quads of all 4-blocks [Figs. (4.1-4.2)]. After detecting all 4-blocks and filling the dual grid accordingly, we note that adjacent nodes on the dual grid correspond to 4-blocks that are connected and are part of the same cluster, and we thus connect these dual nodes [red lines in Figs. (4.1-4.2)]. Subsequently, we assign the same colour to all quads that belong to these connected 4-blocks, and use different colours for 4-blocks that are disconnected on the dual grid (Fig. 4.2). This procedure yields an unambiguous detection of clusters: As illustrated for the systems in Fig. 4.2, we have identified and coloured all connected 4-blocks, and find that each quad belongs to either zero (grey), one (colour) or in some cases, two clusters (bi-colour).

*For the generic quads, there is no regular underlying lattice. Nonetheless, the connectivity and topology of such networks are equivalent to that on a square lattice.

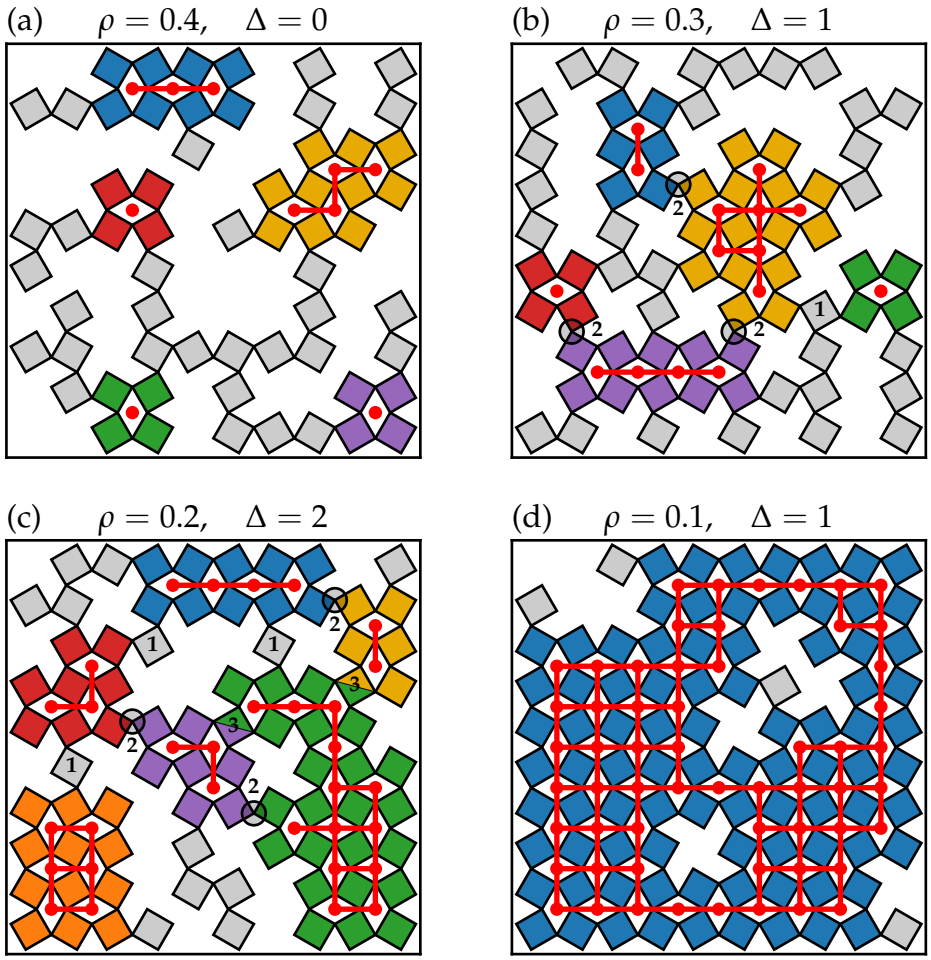

Figure 4.2: Clusters on the dual grid. Distinct clusters are indicated by different colours; the red nodes and lines display the dual grid. (a-d) Four independent randomly diluted systems of varying cutting fraction, shown from high to low ρ. The actual quads of different clusters can connect in three different manners, as indicated by the numbers 1, 2 and 3: Via a grey quad (type 1), direct connection (type 2, marked by small circles) or an shared quad (type 3 and bi-coloured). All grey quads that are not type-1 are remaining quads.

Distinct clusters can be connected in three different manners, as indicated by the numbers in Fig. 4.2, of which a more detailed overview is provided in Fig. 4.8. First, a grey quad may connect to two clusters and we will call this a type-1 connector [Fig. 4.8(a)]. Second, two clusters may directly be connected, leading to a type-2 connector [Fig. 4.8(b)]. Third, two clusters may share a quad, leading to a type-3 connector [Fig. 4.8(c)]. These names are not arbitrary — in what follows we will show that these lead to one, two or three constraints.

Finally, grey quads that are not type-1 connectors are classified as remaining quads. In the following section we will discuss the relevance of the remaining quads and show that most of these are not relevant for Δ. Altogether, in this section we have developed the notion of 4-blocks which allows us to partition any randomly diluted system into clusters, connectors and remaining quads.

4.3 Pruned systems

We now define pruned systems, where most of the remaining quads are removed, motivated by the aim to study the nature and multiplicity of the excess zero modes in the simplest possible setting. We therefore demonstrate that remaining quads contribute almost equally to n_s and n_g, and can thus be pruned without significantly altering Δ. The only remaining quads that easily can be detected and understood to have the potential to change Δ, are self-connectors — isolated quads that are connected to the same cluster [examples are shown in Fig. 4.2(d) and Fig. 4.6(a)]. These never modify n_s, but they may rigidify an otherwise hinging generic cluster, and we have therefore kept these self-connectors in our pruned systems. All other remaining quads are removed. In the following we describe the average number of excess zero modes in the pruned systems by $\langle \Delta' \rangle(\rho, N)$, with ρ the cutting fraction of the full systems, to facilitate comparison with Δ.

In Fig. 4.3 we have visualized the pruned systems that correspond to the full systems shown in Fig. 4.2. For each of these examples we find that the number of excess zero modes before and after pruning is the same ($\Delta = \Delta'$). One can understand this intuitively by noting that the remaining quads form dangling and floppy groups that equally contribute to n_s and n_g. Nonetheless, a quantitative comparison of the ensemble

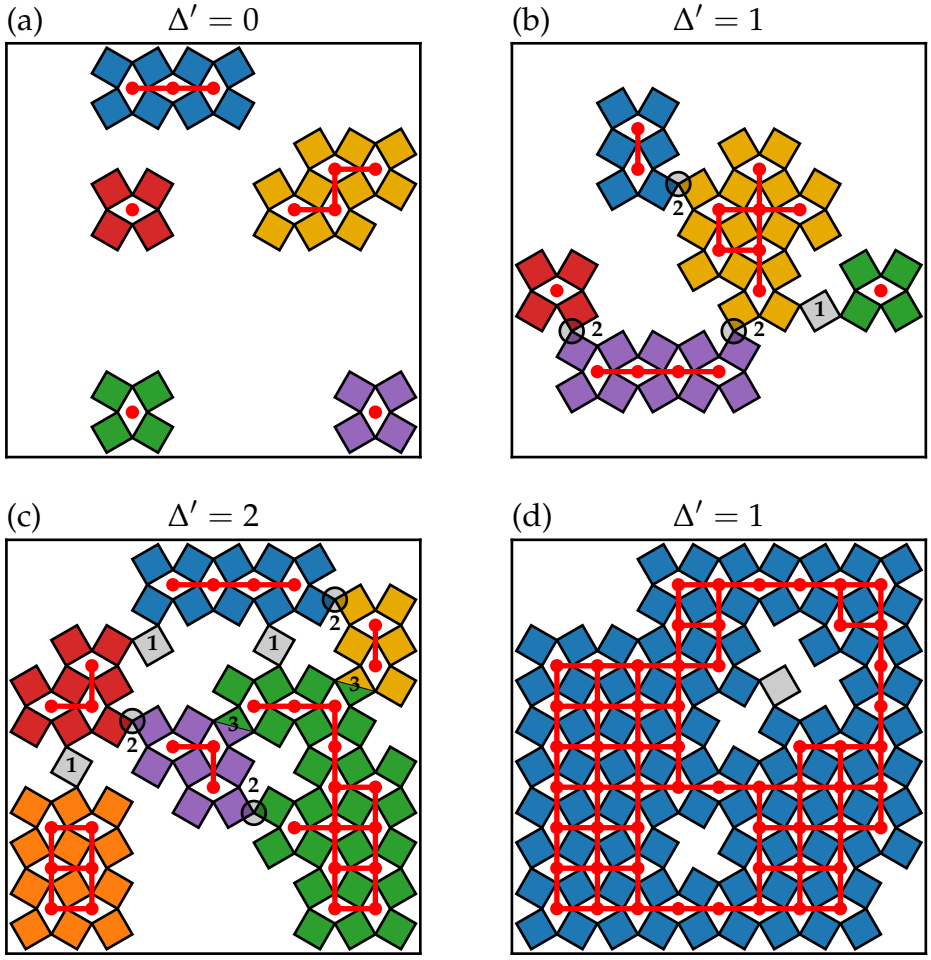

Figure 4.3: Pruned versions of the systems show in Fig. 4.2. Note that $\Delta' = \Delta$ for each of these examples.

averages $\langle\Delta\rangle$ and $\langle\Delta'\rangle$ as function of ρ and N, as obtained from another series of independent simulations, shows slight differences in the number of excess modes of pruned and full systems can occur (Fig. 4.4). As can be observed from panel (a), the curves $\langle\Delta\rangle(\rho, N)$ (solid lines) and $\langle\Delta'\rangle(\rho, N)$ (dashed lines) are essentially indistinguishable, but $[\langle\Delta'\rangle - \langle\Delta\rangle] / \langle\Delta\rangle^{\star}$, as plotted in panel (b), reveals differences that are most pronounced for intermediate cutting fractions. Here, $\langle\Delta\rangle^{\star}$ denotes the peak value of $\langle\Delta\rangle$.

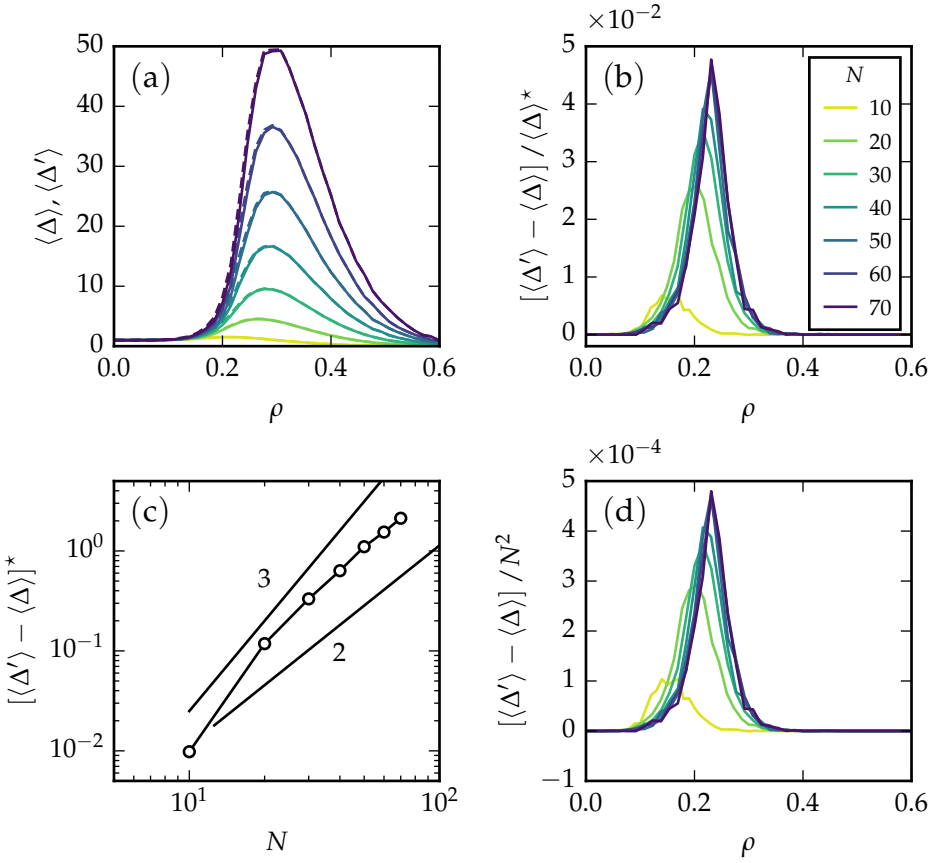

Figure 4.4: Excess zero modes in full and pruned systems, for a range of system sizes. (a) The average number of excess modes as function of ρ for full ($\langle\Delta\rangle$, solid lines) and pruned ($\langle\Delta'\rangle$, dashed lines) systems for a range of system sizes. (b) Deviations as compared to the peak value of the full systems, $\langle\Delta\rangle^\star$, show that relative deviations are maximally 5% for the system sizes considered. (c) The peak value of $\langle\Delta'\rangle - \langle\Delta'\rangle$ as function of N. (d) Deviations rescaled by N^2.

As can be seen, these relative differences never exceed 5% for the system sizes considered. In panel (c) we show the peak value of the deviations, $[\langle\Delta'\rangle - \langle\Delta\rangle]^\star$, as function of N. This shows that the deviations grow rapidly for small N, but for larger N, our data is consistent with a crossover to N^2 scaling — to accurately measure the asymptotic scaling exponent much larger system sizes are required. For completeness, in

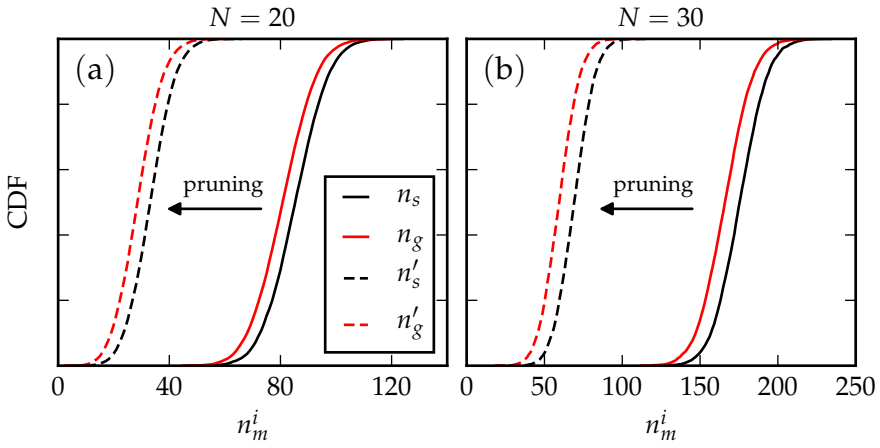

Figure 4.5: CDFs for the number of internal zero modes (n^i_m) in full and pruned systems, in the peak regime of $\langle\Delta\rangle$. The solid curves display data for the number of zero modes in the full symmetric (n_s) and generic systems (n_g); the dashed curves display data for the pruned symmetric (n'_s) and generic systems (n'_g). We show data for system sizes (a) $N = 20$ and (b) $N = 30$.

panel (d) we show the deviations rescaled by N^2, which leads to a reasonable collapse for large N. Our data thus suggests that the difference between $\langle\Delta\rangle$ and $\langle\Delta'\rangle$ is extensive. Altogether, these findings demonstrate that pruned systems accurately capture the excess zero modes of full systems, which allows us to study the nature and multiplicity of the excess zero modes in the simpler pruned systems.

To further show that pruned systems well describe the full systems, we also display some ensemble distributions of Δ and Δ' as CDFs in Fig. 4.5, for ρ that corresponds to the peak location of $\langle\Delta\rangle$. For both system sizes that are shown in Fig. 4.5, we observe a very similar translation of n_s and n_g as a result of pruning. These findings therefore further support that the number of excess zero modes in the full and pruned system are very close.

Finally, to gain insight in the cause of deviations between Δ and Δ', and the role of self-connectors, consider the three examples shown in Fig. 4.6. First, the strip-like cluster in panel (a) displays a typical example illustrating why we do not prune self-connectors; by keeping this self-connector, we find that $\Delta = \Delta' = 1$. In contrast, where we to prune

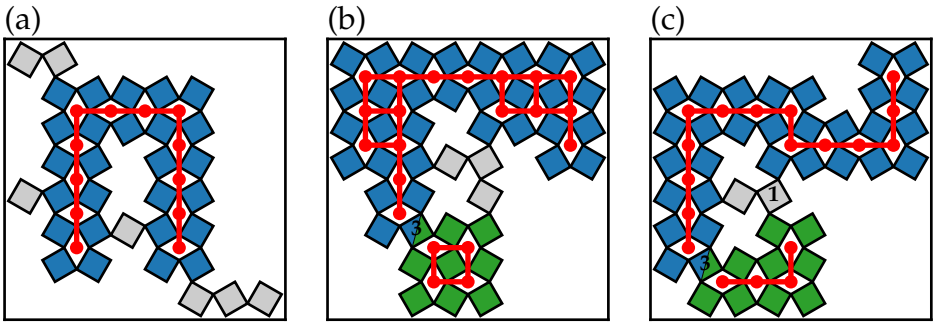

Figure 4.6: Cutting patterns that illustrate the role of self-connectors and deviations between Δ and Δ'. (a) Pruning this system while keeping the self-connector yields $\Delta = \Delta'$, whereas we would find $\Delta'' = 0$ if we additionally prune the self-connector. This illustrates that self-connectors rigidify hinging generic strips. (b-c) Cutting patterns for which pruning of remaining grey quads changes the number of excess zero modes ($\Delta \neq \Delta'$). Note that these examples do not contain self-connectors, such that $\Delta' = \Delta''$.

all remaining quads, we would find that $\Delta'' = 0$. The second and third example [panel (b-c)] display cutting geometries that do not contain self-connectors, but for which $\Delta \neq \Delta'$. These have to do with more complex clusters of remaining quads that, in general, will be hard to detect. Nonetheless, as we have shown in Fig. 4.4, deviations such as as displayed in Fig. 4.6(b-c) are rare and Δ and Δ' are very close.

4.4 Topology based counting argument

We now explain how we develop a counting argument to estimate n_s' from the topology of the clusters and their connectors. To establish this counting argument we will utilize the generalized Maxwell count [Eq. (3.3)], which can be adapted to the context of clusters (explicitly indicated by the superscript c) as:

$$N^c_{\text{dof}} - N^c_{\text{con}} = n^i_m + 3 - n^c_{ss}. \tag{4.1}$$

In the above equation, N^c_{dof} represents the number of internal and rotational/translational degrees of freedom of the clusters and N^c_{con} the

number of constraints of their associated connectors†. We note that large clusters feature a significant number of internal states of self-stress, but these we will ignore. However, when clusters are coupled, new states of self-stress may arise and we count these by n_{ss}^c. Hence, we see the clusters as 'black boxes' that can rotate, translate, and hinge — all internal states of self-stress are ignored. Thus, n_{ss}^c describes the number of redundant inter-cluster constraints given by the type 1,2 and 3 connectors. Lastly, recall that n_m^i expresses the number of internal zero modes, with $n_m^i = n_s'$ for symmetric systems. We furthermore note that Eq. (4.1) also applies to generic systems with $n_m^i = n_g'$, but as motivated already, our main focus is to count the number of zero modes in symmetric systems, rather than the zero modes in generic systems where exact algorithms to count these already exist [47, 48].

Below we first discuss how the number of (excess) zero modes can be determined in the simplest scenario in which connectors are absent ($N_{\text{con}}^c = n_{ss}^c = 0$). Subsequently, we consider systems in which connectors are present and explain how to correctly take the N_{con}^c constraints and n_{ss}^c self-stresses into account. Finally, we present a step-wise analysis of a (randomly diluted) system that exemplifies how our methodology can be applied to successfully count n_s'.

4.4.1 Counting without connectors

In the simplest scenario, there are no connectors between clusters and inter-cluster self-stresses are absent ($n_{ss}^c = 0$) — this typically occurs at either low [Fig. 4.3(d)] or high cutting fractions [Fig. 4.3(a)]. We now show that — in the absence of connectors — we can estimate the number of (excess) zero modes from a simple argument based on the multitude of clusters.

Using Eq. (4.1) we first note that $n_m^i = N_{\text{dof}}^c - 3$ in the absence of connectors. For symmetric systems, all clusters are hinging, and each cluster contributes 4 degrees of freedom. For a randomly diluted system that contains n_c clusters, we therefore immediately obtain $n_s' = 4n_c - 3$. To determine the number of *excess* zero modes we also need to obtain n_g'. In principle, our goal is not to develop a counting argument for n_g', since

†The constraints introduced by the connectors should not be confused with the internal constraints of an individual cluster. The internal constraints constitute clusters, but do not constrain the motion between adjacent clusters.

this is already captured by the pebble game [47, 48]. However, in the very special case where constraints and inter-cluster self-stresses are absent, we can approximate the outcome of the pebble game using a simple argument — which helps us to gain insight into the nature and multiplicity of the excess zero modes. We therefore proceed by providing an approximate counting method for n'_g.

To determine n'_g we need to distinguish hinging clusters and rigid clusters. For generic systems we have discussed before that while strips have one hinging mode, larger clusters are rigid; generic clusters can therefore either contribute 3 (rigid clusters) or 4 (hinging clusters) degrees of freedom. Assuming that of the n_c clusters, n_r are rigid and n_h are hinging, we obtain $n'_g = 4n_h + 3n_r - 3$. Combining the expressions for n'_s and n'_g, this yields the rigid-cluster-based estimate $\Delta_r = n'_s - n'_g = n_r$, demonstrating that the number of excess zero modes is tantamount to the number of rigid clusters in the absence of connectors. Hence, excess modes are induced as a result of sufficiently large clusters that are rigid in the generic case, but retain their hinging mode in the symmetric case.

Comparison of Δ_r and exact results — We now systematically compare the rigid-cluster-based prediction $\langle \Delta_r \rangle(\rho, N)$ (dashed lines) and the exact results $\langle \Delta \rangle(\rho, N)$ (solid lines) based on the Hessian matrix, as obtained from a large number of independent simulations. These simulations are based on two assumptions: First, we did not prune these systems to reduce numerical computation time, as this circumvents the need to detect remaining quads and connectors. This does not affect the outcome for the number of rigid clusters. Second, counting of the number of rigid clusters is approximate. A sufficient condition for rigidity is that somewhere in the cluster dual nodes are connected in a loop. We detect the most common loop that occurs in dense clusters, where one or more primitive loops of length four occur [Fig. 4.1(c)]. In principle, the smallest loop on the dual grid can be longer than four [e.g. as in Fig. 3.3(c)]. However, these cases are rare and yield a negligible overestimation of n_r, which therefore does not affect the main findings we will discuss now.

In Fig. 4.7 we display $\langle \Delta_r \rangle(\rho, N)$ (dashed lines) and the exact results $\langle \Delta \rangle(\rho, N)$ (solid lines). As can be observed from Fig. 4.7(a), $\langle \Delta_r \rangle$ qualitatively captures the peak behaviour of $\langle \Delta \rangle$ and yields, as expected, an excellent approximation for large cutting fractions; in this regime the sys-

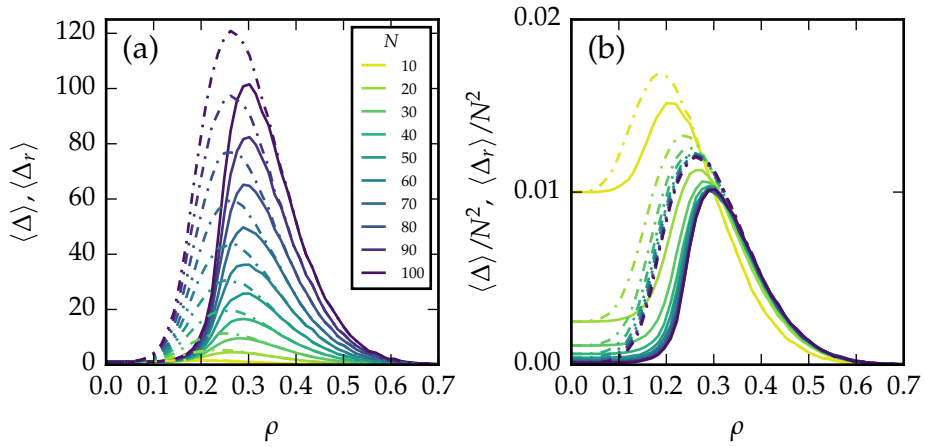

Figure 4.7: Comparison of the average number of excess zero modes predicted by exact calculations $\langle\Delta\rangle(\rho)$ (solid lines) and the simultaneous rigid-cluster-based approximation $\langle\Delta_r\rangle(\rho)$ (dashed lines), for a range of system sizes. We have used the same ensemble size for each N as in Fig. 3.12 for the data shown. (a) $\langle\Delta_r\rangle$ shows excellent agreement with $\langle\Delta\rangle$ for cutting fractions beyond the peak location of $\langle\Delta\rangle$. (b) Scaling collapse of $\langle\Delta\rangle$ and $\langle\Delta_r\rangle$ in the $N \to \infty$ limit for the data shown in panel (a).

tems contain a few rigid clusters as well as numerous isolated hinging clusters and floppy groups of loosely connected quads — only the rigid clusters contribute to Δ. In contrast, $\langle\Delta_r\rangle$ displays significant deviations from $\langle\Delta\rangle$ for intermediate ρ. Comparison of the large N asymptotics of $\langle\Delta_r\rangle$ —which similar to $\langle\Delta\rangle$ scale as N^2— with the asymptotics of $\langle\Delta\rangle$ shows that these deviations are characterized by a peak value which is consistently overestimated, and a peak location with a slight offset to the left [Fig. 4.7(b)].

The observed deviations expose the effect of connectors. For intermediate ρ the clusters are strongly entangled by connectors of type 1,2 and 3 and consideration of their associated self-stresses is necessary to accurately estimate n_s' and n_g'. As discussed before, in the generic case there exists an exact algorithm to determine n_g' and the self-stresses. However, in the symmetric case this algorithm is not suitable and our main goal therefore is to develop an approximate counting argument for the number of (excess) zero modes and self-stresses in these systems.

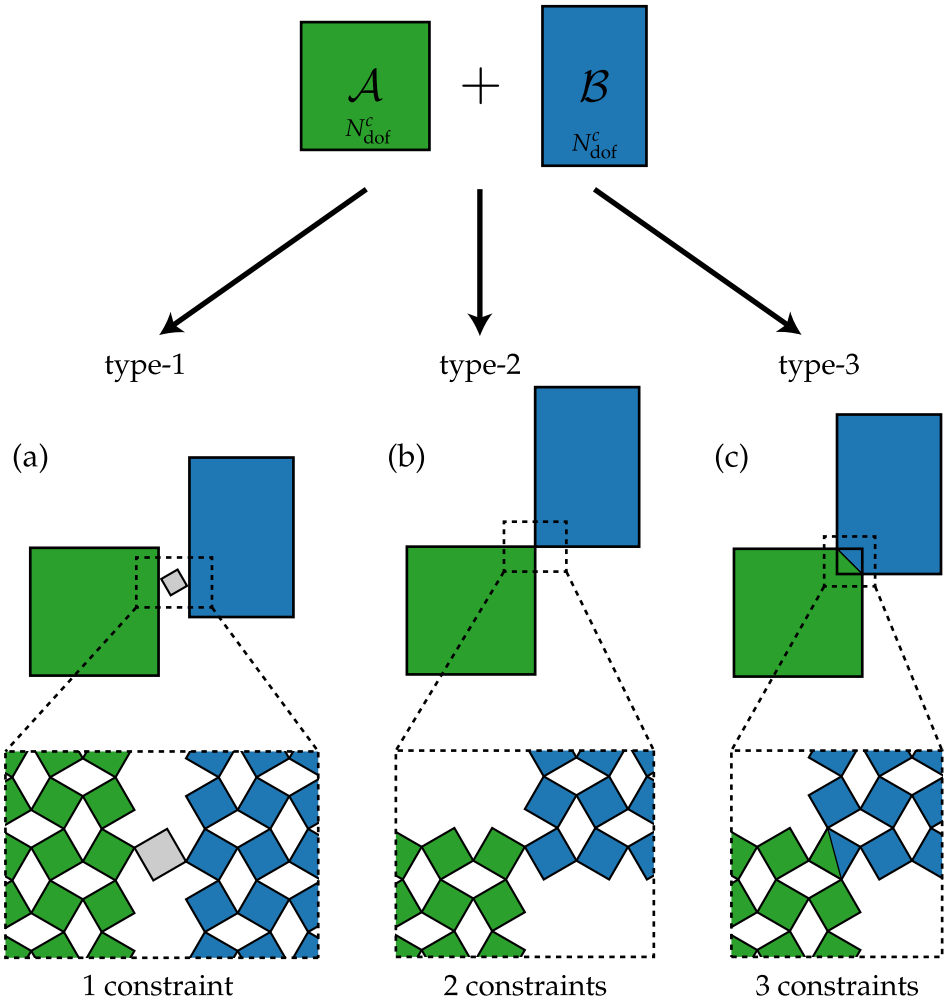

Figure 4.8: Cluster connector types. There exist three distinct manners in which clusters $\mathcal{A}$ and $\mathcal{B}$ may be connected, namely via a type-1 (left), type-2 (middle) or type-3 (right) connector. These connectors, according to their names, introduce precisely 1, 2 or 3 constraints, as is indicated at the bottom of the figure. In the main text we provide a derivation for the number of constraints associated with each connection type.

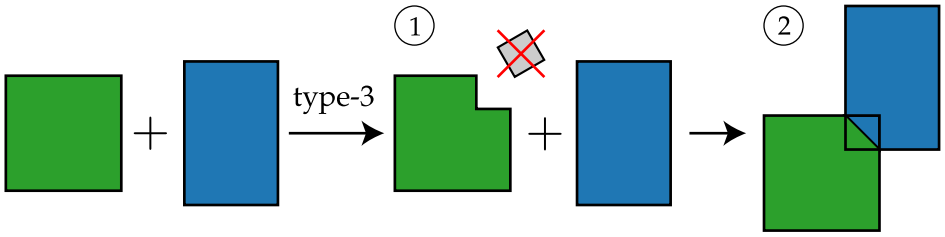

Figure 4.9: Step-wise explanation of connection type-3. In step 1, the upper right quad of the green cluster is removed and two associated bonds are cut. In step 2 both clusters are connected and two bonds are restored, leading to no net change in the number of bonds and the net removal of 3 degrees of freedom. Hence, the number of zero modes as a result of the type-3 connection is lowered by 3.

4.4.2 Connectors, constraints and inter-cluster self-stresses

Before presenting a counting argument that takes into account connectors, we now first show that connectors of type 1,2 and 3, conform to their names, introduce precisely 1, 2 and 3 constraints when connecting a pair of clusters. Moreover, we explain the appearance of inter-cluster self-stresses as a result of these connectors.

Connectors and constraints. — To evidence the mentioned number of constraints associated with each connection type, we consider a system that consists of two clusters, $\mathcal{A}$ and $\mathcal{B}$, and determine how n_s of this two-cluster system is lowered for each connection type. In Fig. 4.8 we display a detailed graphical representation of the three distinct manners in which cluster $\mathcal{A}$ and $\mathcal{B}$ can be connected. Before the clusters are connected $n_s = N^c_{\text{dof}} - 3 = 5$, where each cluster contributes 4 degrees of freedom. These 5 zero modes should be interpreted as the independent hinging of both clusters (2 zero modes) and relative translations or rotations of the clusters (3 zero modes). After the clusters are connected by a type 1, or 2 or 3 connector, some of their relative motions become constrained, and n_s will respectively be lowered by one, two or three — which is equivalent to the presence of one, two or three constraints. To understand these connectors add one, two or three constraints, consider the zoomed areas in Fig. 4.8. First, for a type-1 connection the grey quad adds 3 degrees of freedom and 2 extra bonds, hence we lose one zero mode. Second, a

type-2 connection introduces a single additional bond, hence we lose two zero modes. Third, a connection type-3 is more subtle and can be seen as removing a single quad (three degrees of freedom), without adding new bonds, thereby lowering the number of zero modes by three. This is illustrated in more detail in Fig. 4.9.

Inter-cluster self-stresses. — Above we have illustrated the lowering of n_s as a result of a *single* type-1, *or* type-2, *or* type-3 connector. To understand the role of self-stresses we now consider the same two-cluster in the presence of *multiple* connectors and note that the lowering of n_s is bound to the minimum $n_s = 1$; coupled symmetric clusters always attain a global hinging mode, regardless of the number of constraints these share. As an example, consider the two-cluster system shown in Fig. 4.10 that contains two type-3 connectors (6 constraints). For this system one can readily envision that the connectors constrain all relative motions of the green cluster with respect to the blue cluster (and vice versa); the system only allows for a single dependent internal zero mode in which the motions of both clusters are coupled — global hinging. The cluster pair is dependent, acts as a single cluster with $n_s = 1$, and from Eq. (4.1) we accordingly find that the clusters share 2 inter-cluster self-stresses ($n_{ss}^c = 2$). Hence, the system contains 2 *degenerate* (redundant) constraints.

The example discussed above illustrates a general observation which is most easily understood by the consideration of Table 4.1. Here we tabulated n_s and n_{ss}^c as function of the number of constraints (N_{con}^c), where

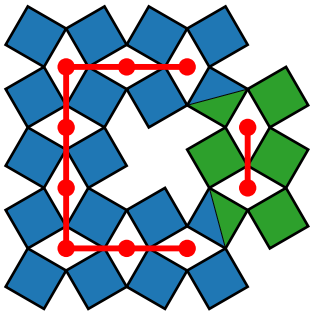

Figure 4.10: The clusters in this system exhibit no relative motions ($n_s = 1$) due to two type-3 connectors and effectively thus behave as a single cluster.

two-cluster system		
N^c_{con}	n_s	n^c_{ss}
0	5	0
1	4	0
2	3	0
3	2	0
$\downarrow n_s$ constant $\downarrow$		
4	1	0
5	1	1
6	1	2

Table 4.1: Number of zero modes (n_s) and inter-cluster states of self-stress (n^c_{ss}), as function of the number of constraints (N^c_{con}) for a two-cluster system. For $N^c_{\text{con}} \geq 4$ the number of internal zero modes becomes constant, as indicated by the shaded region.

we note that N^c_{con} is related to the number of type-1,2 and 3 connectors via

$$N^c_{\text{con}} = n_1 + 2n_2 + 3n_3, \tag{4.2}$$

with n_1,n_2 and n_3 the number of type 1,2 and 3 connectors. Starting from two unconnected clusters ($n_1 = n_2 = n_3 = 0$), Table 4.1 clearly shows that for each constraint added, n_s is lowered by one. The lowering of n_s persists until all relative motions of the clusters become constrained ($n_s = 1$), which requires precisely 4 constraints. From this point, the presence of any additional constraints does no longer affect n_s, but leads to the development of inter-cluster self-stresses instead.

4.4.3 Counting including connectors

In this section we present a counting argument which is suitable for symmetric systems and that takes into account the presence of connectors. The challenge in counting n'_s is that that the number of inter-cluster self-stresses generally is non-zero: Whereas the left hand side of Eq. (4.1) is easily calculated based on the number of clusters and constraints of the connectors, we need to know n^c_{ss} in order to calculate n'_s. In what follows we therefore develop a counting method that *iteratively* eliminates

the inter-cluster self-stresses, yielding an hierarchy of predictions for n_s', based on the following methodology:

(i) Given a randomly diluted system, we first partition the system into clusters, connectors and remaining quads, and subsequently prune the system.

(ii) We merge clusters that share four or more constraints. This cluster coarsening eliminates (some of the) the inter-cluster self-stresses.

(iii) After the merging of clusters, we may find that new clusters have formed that again share four or more constraints — we iterate until no more clusters can be merged.

(iv) In each iteration step, we calculate the estimate for n_s' from the topology of the clusters and their connectors, according to

$$n_{s,i} = 4n_{c,i} - N^c_{\text{con},i} - 3, \tag{4.3}$$

where $n_{c,i}$ and $N^c_{\text{con},i}$ are the number of clusters and constraints associated with iteration step i.

We will now first clearly motivate this methodology, and then provide a worked-out example. Step (i) is straightforward and already explained in Sections (4.2-4.3). The merging of cluster pairs as described in steps (ii) and (iii) ensures inter-cluster states of self-stress become internal and do no longer contribute to n^c_{ss}, which circumvents the need to explicitly count these. For example, we could treat the system in Fig. 4.10 as two distinct clusters with 8 degrees of freedom, 6 constraints, 4 zero modes and 2 self-stresses, but the simpler approach is to merge the clusters into a single (unconstrained) cluster that features 4 degrees of freedom and zero inter-cluster self-stresses. A sufficient condition for merging is that two clusters share 4 (or more constraints); this condition is based on Table 4.1, which we used previously to motivate that clusters with four or more constraints are dependent and develop inter-cluster self-stresses. Finally, Eq. (4.3) as given in step (iv) is based on Eq. (4.1), where we note that the number of inter-cluster self-stresses does not appear in Eq. (4.3) as we tentatively assume $n^c_{ss} = 0$ after the cluster merging in each iteration step.

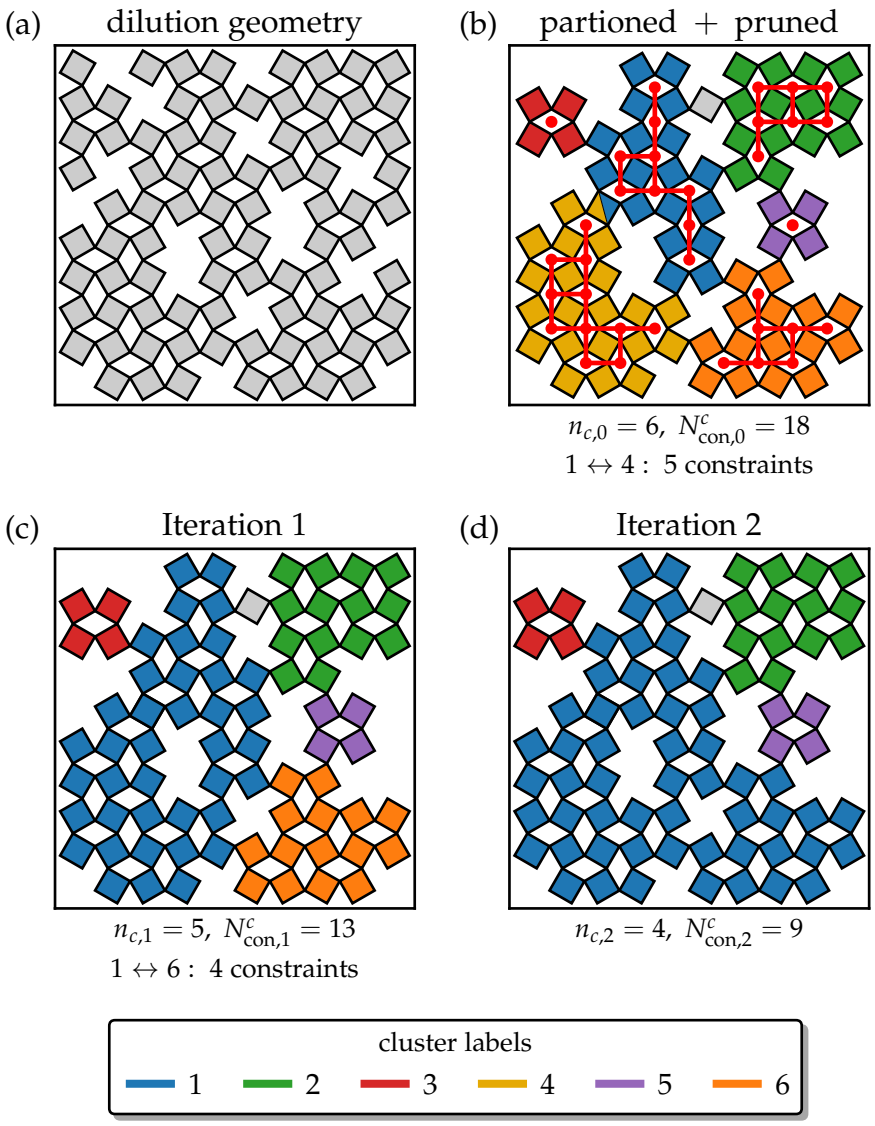

Figure 4.11: Visualization of the iterative merging of clusters to estimate n'_s (see main text). At the bottom of each of the respective panels we provide the number of clusters $n_{c,i}$ and constraints $N^c_{\text{con},i}$. The text '$\mathcal{A} \leftrightarrow \mathcal{B}$' additionally indicates the labels of cluster pairs $\mathcal{AB}$ that share sufficient constraints for merging. (a) Initial random dilution geometry. (b) Partitioned and pruned system. (c-d) Subsequent merging of clusters. For simplicity, the dual grid is only shown in panel (b).

Worked-out example. — To gain intuition for the iterative counting argument, we finalize this section by calculating the hierarchy of predictions $n_{s,i}$ for the system shown in Fig. 4.11(a). We will show that this system requires two merging iterations and that the result as obtained from the final iterated topology of the clusters and their connectors is in agreement with exact results based on the Hessian matrix, namely $n_s' = 4$.

initial estimate (no iterations) $(n_{s,0})$**. —** To obtain the initial estimate for n_s' we partition the randomly diluted system and prune the remaining quads [Fig. 4.11(b)]. From the resulting topology of the clusters and their connectors, we note that there are $n_{c,0} = 6$ clusters and $N^c_{\text{con},0} = 18$ constraints, tantamount to 1 type-1 connector, 7 type-2 connectors and 1 type-3 connector, and determine accordingly that $n_{s,0} = 4 \times 6 - 18 - 3 = 3$.

estimate for one iteration $(n_{s,1})$**. —** By inspection of the clusters and their connectors in Fig. 4.11(b) we find that clusters 1 and 4 are qualified for merging; these share 5 constraints (one type-2 and 3 connector), whereas all other cluster pairs share less than 4 constraints. We therefore merge cluster 1 and 4 by assigning these the same colour, which leads to the (new) topology of the clusters and their connectors as displayed in Fig. 4.11(c). Due to the cluster merging we lose 1 cluster, 1 type-2 connector and 1 type-3 connector, such that we now obtain that $n_{c,1} = 5$ and $N^c_{\text{con},1} = 13$. The new estimate therefore yields $n_{s,1} = 4 \times 5 - 13 - 3 = 4$.

estimate for two iterations $(n_{s,2})$**. —** Due to the formation of new clusters in the previous iteration [Fig. 4.11(c)], we now also find that clusters 1 and 6 qualify for merging. We therefore also merge these clusters to obtain the topology as shown in Fig. 4.11(d), for which we accordingly find that $n_{c,2} = 4$ and $N^c_{\text{con},2} = 9$. This yields the estimate $n_{s,2} = 4 \times 4 - 9 - 3 = 4$, which is also the final estimate since Fig. 4.11(d) contains no more clusters that satisfy the merging rule. We thus find that our result as obtained via counting is in agreement with the exact result based on the Hessian matrix, demonstrating that we can count n_s' from the topology of the clusters and their connectors. In the next section we test the general accuracy of this approach.

4.5 Counting of (excess) zero modes

In this section we compare exact results for the number of (excess) zero modes in symmetric systems, based on the Hessian matrix, against the topology based estimates that have been developed above. We in particular focus on the sequence of predictions for Δ', rather than n_s' itself, since the main goal of this chapter is to gain insight in the nature and multiplicity of the *excess* zero modes. We therefore define $\Delta_i = n_{s,i} - n_g'$, where $n_{s,i}$ follows from the iterative counting argument and n_g' from exact Hessian based calculations, and show that Δ_∞ (final iterated result) yields a tight lower bound on the exact Δ'. Finally, to understand deviations between Δ_∞ and Δ', we compare $n_{s,\infty}$ to n_s' and discuss a number of examples for which our iterative counting method is inaccurate.

4.5.1 Results

Here we test the accuracy of Δ_i as function of system size N and cutting fraction ρ, by comparing the average estimates $\langle \Delta_i \rangle(\rho, N)$ to the exact results $\langle \Delta' \rangle(\rho, N)$. In order to obtain $\langle \Delta_i \rangle$, we have numerically implemented the iterative counting method. This Python-based code automatically partitions each randomly diluted systems into clusters, connectors[‡] and remaining quads, and subsequently iteratively merges clusters that satisfies the merging rule. In this way we acquire Δ_i for a large number of independent randomly diluted systems, allowing us to obtain reasonably well statistically converged data for $\langle \Delta_i \rangle$ (we used the same ensemble sizes as described in section 3.3.2). Recall that $i = 0$ corresponds to the initial system for which no clusters have been merged yet. For $i = 1$ we have merged the clusters once, for $i = 2$ twice, etc. The final topology, which contains no more clusters that satisfy the merging rule, is denoted by $i = \infty$.

In Fig. 4.12 we show the average iterative predictions $\langle \Delta_0 \rangle(\rho)$, $\langle \Delta_1 \rangle(\rho)$, $\langle \Delta_2 \rangle(\rho)$ and $\langle \Delta_\infty \rangle(\rho)$ and the exact result $\langle \Delta' \rangle(\rho)$ for two system sizes, $N = 10$ and $N = 90$. For clarity we do not display the estimates that fall in between $i = 2$ and $i = \infty$. As can be observed, the most naive approximation that neglects inter-cluster self-stresses, $\langle \Delta_0 \rangle$, accurately captures

[‡]In appendix 4.A we explain how each connection type is numerically detected using the adjacency matrix of the system and of individual clusters.

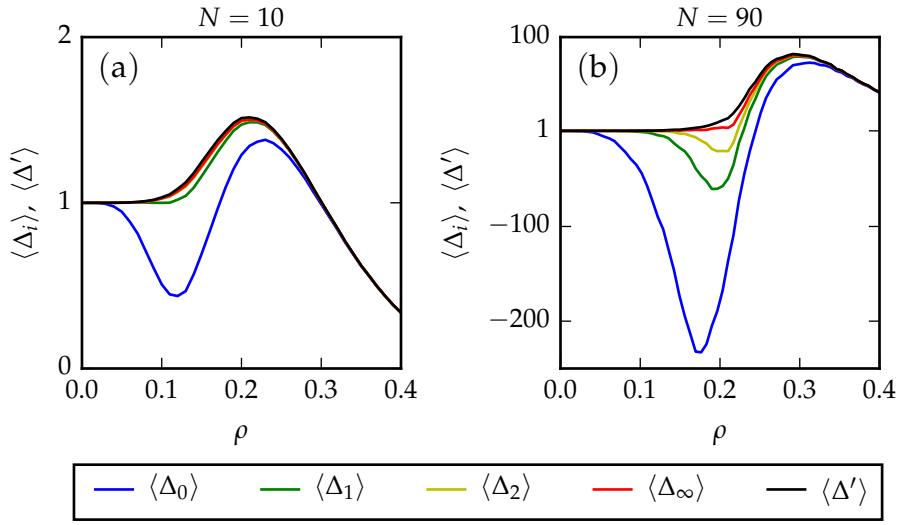

Figure 4.12: Iterative topology based estimates $\langle\Delta_0\rangle, \langle\Delta_1\rangle, \langle\Delta_2\rangle$ and Δ_∞ compared against the exact Hessian-based result $\langle\Delta'\rangle$. We show results for (a) $N = 10$ and (b) $N = 90$.

$\langle\Delta'\rangle$ for large cutting fractions, but displays significant deviations for intermediate cutting fractions. This finding is fully consistent with the expectation that inter-cluster self-stresses are present for strongly connected clusters (intermediate ρ), but absent for clusters that share few to none connectors (large ρ). The estimate after one merging iteration, $\langle\Delta_1\rangle$, already leads to a significantly improved estimate for the number of excess zero modes for each of the system sizes shown; the cluster merging eliminates a large number of self-stresses such that $\langle\Delta_1\rangle$ appears much closer to $\langle\Delta'\rangle$ than $\langle\Delta_0\rangle$. However, the new topologies of the clusters and connectors as obtained after one iteration again contain self-stresses such that $\langle\Delta_1\rangle$ still deviates from $\langle\Delta'\rangle$. The second merging iteration again eliminates these self-stresses, and as can be seen, $\langle\Delta_2\rangle$ yields an improved estimate for $\langle\Delta'\rangle$. The described improvement persists for the iterations thereafter ($i = 3,4..$) (not shown) up to the point where all of the clusters are merged, yielding the final estimate $\langle\Delta_\infty\rangle$. Fig. 4.12 demonstrates that we find very close correspondence of $\langle\Delta'\rangle$ and $\langle\Delta_\infty\rangle$ (which we will show to be a lower bound below) for each of the system sizes shown. Nonethe-

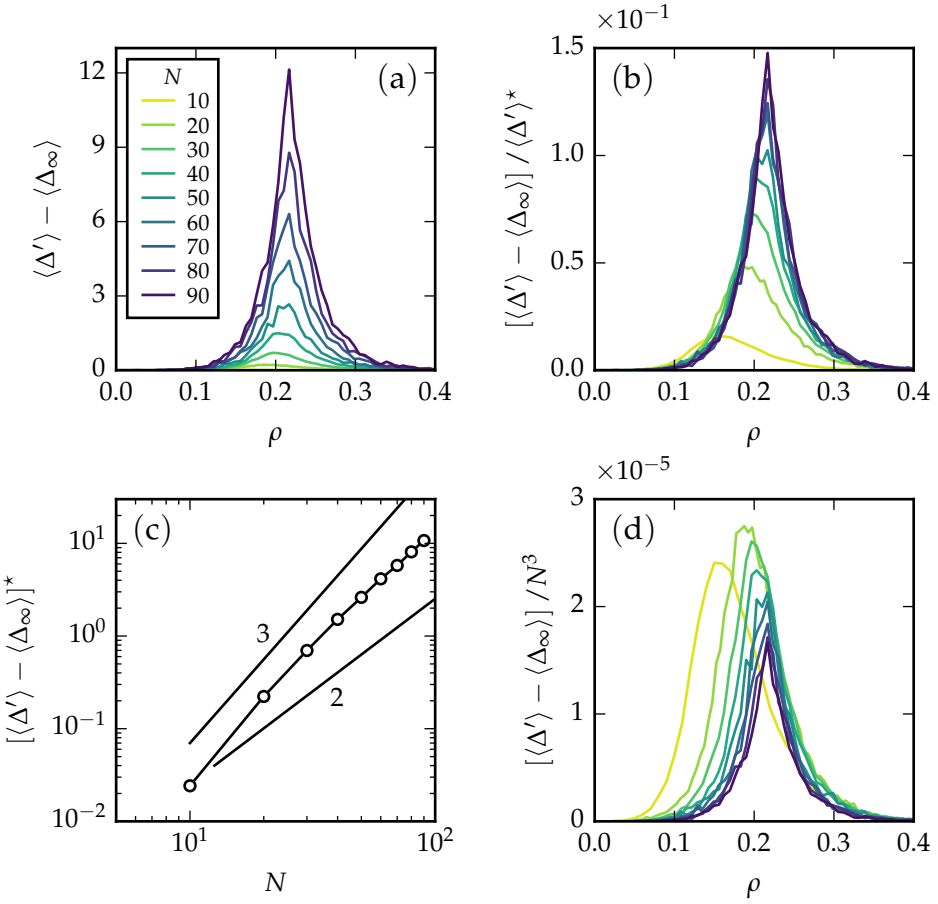

Figure 4.13: Comparison of $\langle\Delta'\rangle$ and $\langle\Delta_\infty\rangle$ for a range of system sizes. (a) The final deviations, $\langle\Delta'\rangle - \langle\Delta_\infty\rangle$ as a function of ρ. (b) Deviations as compared to the peak value of $\langle\Delta'\rangle$, $\langle\Delta'\rangle^\star$, show that relative deviations are maximally 15% for the largest system size considered ($N = 90$). (c) The peak value of $\langle\Delta'\rangle - \langle\Delta_\infty\rangle$ as function of N. (d) Deviations rescaled by N^3.

less, we observe small deviations between $\langle\Delta_\infty\rangle$ and $\langle\Delta'\rangle$, which implies the final iterated topologies contain remaining inter-cluster self-stresses.

In Fig. 4.13 we systematically compare $\langle\Delta'\rangle$ and $\langle\Delta_\infty\rangle$ for system sizes $N = 10, 20, \ldots, 80, 90$. Panel (a) shows that the final difference between our iterative counting method and the exact number of excess zero modes,

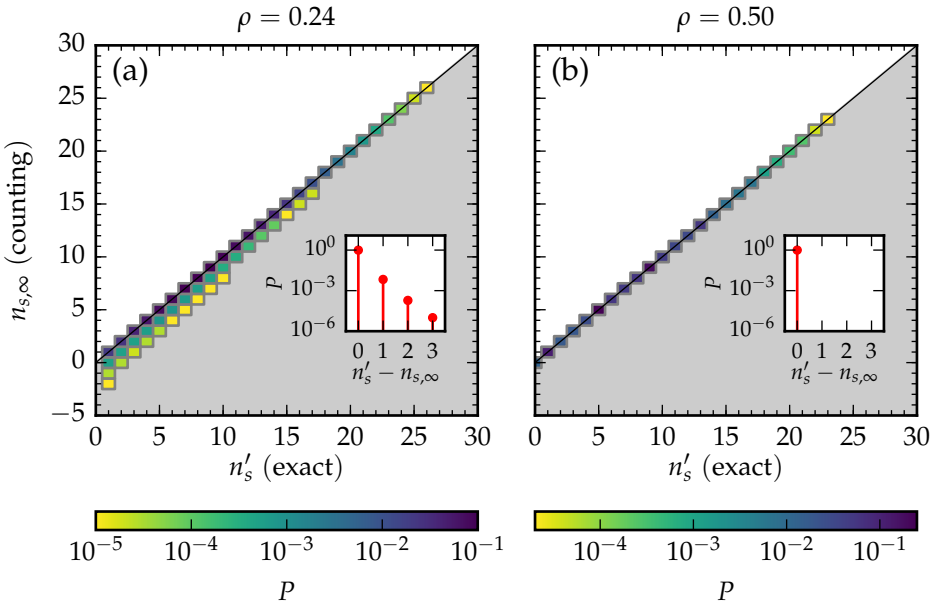

Figure 4.14: Discrete joint probability distributions as function of n_s' and $n_{s,\infty}$, for $N = 10$ and an ensemble size of 10^5 independent simulations. The solid black lines describe $n_s' = n_{s,\infty}$ and the shaded areas indicate the regions $n_s' > n_{s,\infty}$. (a) For $\rho = 0.24$ our counting method is exact for 99% of the simulations and yields a lower bound on n_s'. (b) For $\rho = 0.5$ our counting argument is exact for all simulations. Inset graphs: Discrete probability distribution of $n_s' - n_{s,\infty}$.

$\langle \Delta' \rangle - \langle \Delta_\infty \rangle$, grows with system size and is strongest for $\rho \approx 0.2$ (significantly below the peak location at $\rho \approx 0.3$). As can be seen in panel (b), we find that the relative deviations $[\langle \Delta' \rangle - \langle \Delta_\infty \rangle] / \langle \Delta' \rangle^\star$, with $\langle \Delta' \rangle^\star$ the peak average of $\langle \Delta' \rangle$, are maximally 15% for the largest system size considered. Panel (c) shows that the peak value of the deviations, $[\langle \Delta' \rangle - \langle \Delta_\infty \rangle]^\star$, initially grows faster than N^3, and then slowly crosses over to a smaller effective exponent; our data shows that this exponent is smaller than 3 [see panel (d)], and it is conceivable that ultimately deviations grow as N^2, i.e., are extensive, similar to the data in, e.g., Fig. 4.4.

Comparison of n_s' and $n_{s,\infty}$. — To understand the deviations between $\langle \Delta_\infty \rangle$ and $\langle \Delta' \rangle$ we now consider the discrete probability distributions as function of n_s' and $n_{s,\infty}$ in Fig. 4.14. Panel (a) demonstrates a general

and important feature of our iterative counting argument: $n_{s,\infty}$ yields a strict lower bound on n_s'. Deviations are rare, and occur when one or more connectors are degenerate (examples are discussed below), resulting in complex inter-cluster self-stresses that are not eliminated by our iterative merging method. According to Eq. (4.1) these remaining self-stresses yield an estimate for n_s' which is too low. Therefore, $n_{s,\infty}$ yields a lower bound on n_s', and accordingly, Δ_∞ yields a lower bound on $\langle \Delta' \rangle$. Nonetheless, our counting method provides an excellent approximation to n_s' — for $N = 10$ and $\rho = 0.24$ we find that our counting method is accurate in 99% of the cases [Fig. 4.14(a)]. Note that strongly diluted systems typically do not contain remaining self-stresses such that our counting method is exact for all simulations in that case [Fig. 4.14(b)].

We finally discuss two dilution patterns to illustrate the role of degenerate connectors that lead to complex inter-cluster self-stresses that are not eliminated by the iterative merging procedure (Fig. 4.15). In panel (a) we show 3 clusters that are connected in a loop and in this system none of the clusters satisfies the merging rule. Therefore, we estimate $n_{s,\infty} = 12 - 9 - 3 = 0$ (we count 4 clusters and 9 constraints). However, the correct answer is $n_s' = 1$; the system attains a global hing-

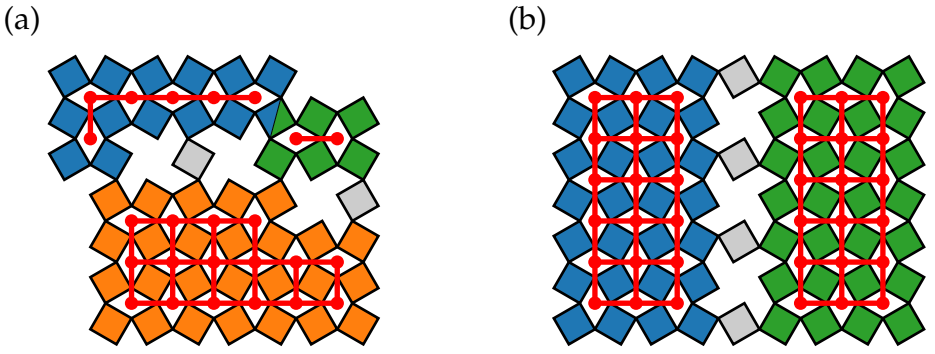

Figure 4.15: Dilution examples that contain special cases of degenerate connectors. (a) In this example the upper left type-1 connector is degenerate, but not eliminated by the iterative merging. This system has one excess zero mode $(n_s' = 1, n_g' = 0)$. (b) The two clusters shown satisfy the merging rule in principle, but can nonetheless be sheared with respect to each other. This system has three excess zero modes $(n_s' = 3, n_g' = 0)$.

ing mode. Apparently, the system contains one redundant constraint, and we have numerically confirmed the upper left type-1 connector (connector between the blue and orange cluster) is redundant; its removal does not change the numerically obtained value for n_s'. In panel (b) we show two clusters that share 4 type-1 connectors on row 1,3,5 and 7. In principle, these two clusters satisfy the merging rule, but one can readily understand that the connectors for this special geometry do not constrain all of the relative motions — the clusters can always shear with respect to each other. Therefore, the estimate $n_{s,\infty} = 1$ is incorrect, and we numerically confirmed that $n_s' = 3$ (global hinging + shear + a quartic mode). We moreover numerically found that one can add an infinite amount of type-1 connectors on every odd row (e.g. 1,3,5,7,9,...) without reducing the number of zero modes; the minimum number of type-1 connectors for $n_s' = 3$ is two, every connector added thereafter is degenerate. Both of the discussed geometries show examples of complex inter-cluster self-stresses that are not correctly taken into account by the iterative merging. Nonetheless, as we have shown in Fig. 4.13, deviations introduced by such special dilution geometries are relatively small.

4.6 Conclusions

We developed an approximate counting method for the number of (excess) zero modes in systems of hinging squares. We therefore first presented a procedure to partition any randomly diluted system into clusters, connectors and remaining quads. We then showed that pruning of the remaining quads does not significantly affect the number of excess zero modes, which allowed us study the nature and multiplicity of the excess zero modes in the simplest possible setting, where quads that are irrelevant for Δ are removed. To develop the counting argument we subsequently treated the clusters as 'black boxes' that can rotate, translate and hinge, showed how each of the three types of connectors constrains the number of zero modes and developed an iterative cluster algorithm to take into account the inter-cluster self-stresses that appear between sufficiently strongly connected clusters. Finally, we compared the predictions of our iterative counting argument against exact Hessian-based results for a large number of independent randomly diluted systems, and showed that we obtain a tight lower bound on $\langle \Delta' \rangle$.

Appendix

4.A Detecting cluster-constraints

In this appendix we explain the algorithms devised to detect type 1,2 and 3 constraints [Figs. (4.16-4.18)]. To demonstrate these algorithms we will use a simple system that consists of two clusters, denoted $\mathcal{A}$ and $\mathcal{B}$. Quads that belong to cluster $\mathcal{A}$ are coloured blue, quads that belong to cluster $\mathcal{B}$ green, and quads that belong to neither of those are coloured grey.

4.A.1 Type 3 constraint

Type 3 constraints are fairly easy to detect in comparison to type 2 and 1 constraints. This is because two clusters connected through a type 3 constraint share a quad, as depicted in Fig. 4.16. Therefore, it suffices to find the intersections of the set of quads that belong to cluster $\mathcal{A}$ and $\mathcal{B}$. For the example shown, the set of quads for both clusters are $\mathcal{A} = \{0,1,2,3\}$ and $\mathcal{B} = \{3,4,5,6\}$, such that the intersection equals $\mathcal{A} \cap \mathcal{B} = \{3\}$. The number of elements in the intersection then equals the number of type 3 constraints. When $\mathcal{A}$ and $\mathcal{B}$ are disjoint (no quads in common), the clusters encompass no type 3 constraints.

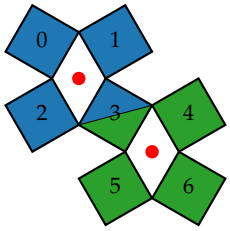

Figure 4.16: Type 3 constraint. Quad number 3 is shared between the blue and green cluster, as indicated by the bi-coloured quad.

4.A.2 Type 2 constraint

To detect type 2 constraints, we use the adjacency matrix of the system, C. This matrix indicates whether pairs of quads are adjacent: Element C_{ij} is set to one when quad i is adjacent to quad j, and zero otherwise. For

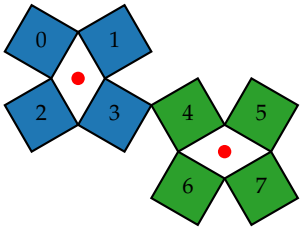

Figure 4.17: Type 2 constraint. The blue cluster ($\mathcal{A}$) and the green cluster ($\mathcal{B}$) are connected via quads 3 and 4.

the system shown in Fig. 4.17, the adjacency matrix is

$$C = \begin{array}{c} \text{label} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{cccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{bmatrix} \times & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ & \times & 0 & 1 & 0 & 0 & 0 & 0 \\ & & \times & 1 & 0 & 0 & 0 & 0 \\ & & & \times & 1 & 0 & 0 & 0 \\ & & & & \times & 1 & 1 & 0 \\ & & & & & \times & 0 & 1 \\ & & & & & & \times & 1 \\ & & & & & & & \times \end{bmatrix} \end{array}, \tag{4.4}$$

where we only provide the upper triangular part of the symmetric matrix for clarity. Furthermore, the crosses on the main diagonal indicate that quads cannot be connected to themselves.

The goal then is to extract all elements from the adjacency matrix that connect cluster $\mathcal{A}$ to $\mathcal{B}$, which, in this case, is C_{34}. In order to find such elements, we define the adjacency sub-matrices $C^{\mathcal{A}}$ en $C^{\mathcal{B}}$ that are constructed from C as follows. We define a sub-matrix $\mathcal{A}$ for which we retain only the rows and columns of C that belong to $\mathcal{A}$, and we follow a similar procedure to construct $\mathcal{B}$. We then find that

$$C^{\mathcal{A}} = \begin{bmatrix} \times & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ & \times & 0 & 1 & 0 & 0 & 0 & 0 \\ & & \times & 1 & 0 & 0 & 0 & 0 \\ & & & \times & 1 & 0 & 0 & 0 \\ & & & & \times & 0 & 0 & 0 \\ & & & & & \times & 0 & 0 \\ & & & & & & \times & 0 \\ & & & & & & & \times \end{bmatrix} \text{ and } C^{\mathcal{B}} = \begin{bmatrix} \times & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & \times & 0 & 0 & 0 & 0 & 0 & 0 \\ & & \times & 0 & 0 & 0 & 0 & 0 \\ & & & \times & 1 & 0 & 0 & 0 \\ & & & & \times & 1 & 1 & 0 \\ & & & & & \times & 0 & 1 \\ & & & & & & \times & 1 \\ & & & & & & & \times \end{bmatrix}. \tag{4.5}$$

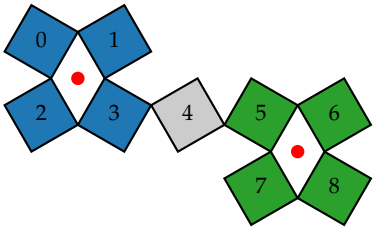

Figure 4.18: Type 1 constraint. The blue cluster ($\mathcal{A}$) and the green cluster ($\mathcal{B}$) are connected via a single quad (4), that is neither a member of $\mathcal{A}$ nor of $\mathcal{B}$.

Finally, two clusters share a type 2 constraint for all ij that satisfy $C_{ij}^{\mathcal{A}} = C_{ij}^{\mathcal{B}} \neq 0$.

4.A.3 Type 1 constraint

We now discuss type 1 constraints [Fig. 4.18], which are most difficult to detect. To find this constraint type, we first determine the system's adjacency matrix, yielding

$$C = \begin{array}{c} \text{label} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} \begin{array}{c} \begin{array}{ccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \left[\begin{array}{ccccccccc} \times & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & \times & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & \times & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & \times & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \times & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \times & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & \times & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \times & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & \times \end{array}\right] \end{array}. \tag{4.6}$$

Subsequently, we eliminate all internal bonds of cluster $\mathcal{A}$ and $\mathcal{B}$. Put differently, we cut all bonds between quads that constitute cluster $\mathcal{A}$ and $\mathcal{B}$. This is equivalent to setting all columns of C associated with quads in $\mathcal{A}$ and $\mathcal{B}$ to zero. The resulting reduced matrix then leaves us with all connections to quads that neither are in $\mathcal{A}$ nor in $\mathcal{B}$. Performing this procedure on C reveals that only elements C_{34} and C_{54} would persist. Since quad 4 is connected to quad 3, which is in $\mathcal{A}$, and quad 5, which is in $\mathcal{B}$, these clusters must be connected through a type 1 constraint.