Supplementary Information for

The emergence of interstellar molecular complexity explained by interacting networks

Miguel García-Sánchez, Izaskun Jiménez-Serra, Fernando Puente-Sánchez and Jacobo Aguirre

Jacobo Aguirre
E-mail: jaguirre@cab.inta-csic.es

This PDF file includes:

Supplementary text
Figs. S1 to S5
References for SI reference citations

Supporting Information Text

S1. NetWorld's algorithm

Starting with an initial set of $n(t)$ networks ($n(0) = N$ isolated nodes in the first step of our simulations), the algorithm randomly selects two networks that will interact. If this interaction leads to the formation of a new network, the process will continue into the next time step with $n(t) - 1$ networks. On the other hand, if the union between those networks is not possible, the algorithm selects two new networks from the remaining set. After a successful union, all networks have a partition probability that depends on their topological stability μ (the second smallest eigenvalue associated with the Laplacian matrix of the network, i.e, the Fiedler eigenvalue (1)), and the environment parameter β. This process of union and partition of networks continues until no further union is possible or until a limit number of successful unions, $T_{max} = 10^4$ time steps, is reached. A general sketch of the algorithm is shown in Fig. S1.

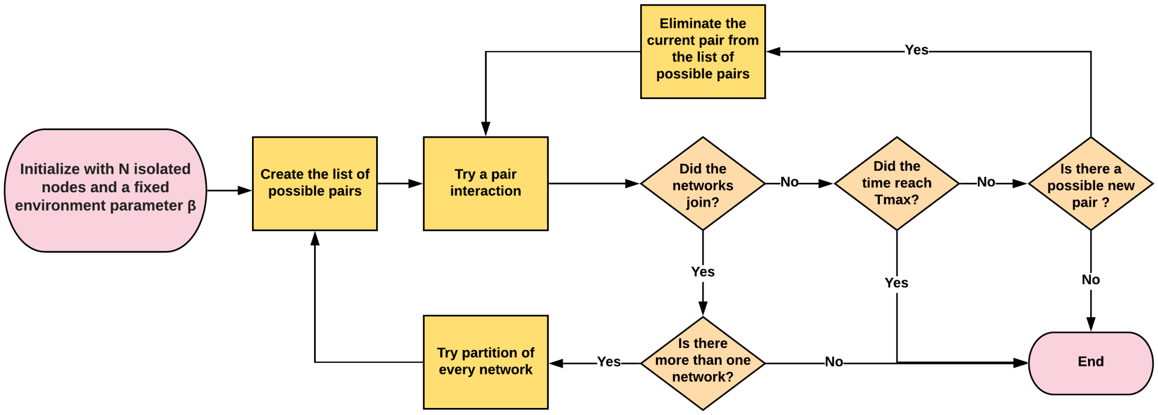
Initialize with N isolated nodes and a fixed environment parameter β
Create the list of possible pairs
Try a pair interaction
Did the networks join?
No
Did the time reach Tmax?
No
Is there a possible new pair ?
Yes
Eliminate the current pair from the list of possible pairs
No
Yes
Yes
Is there more than one network?
Yes
Try partition of every network
No
End

Fig. S1. Flow chart of the algorithm.

In summary, the main structure of our computational environment NetWorld is to repeat two routines in a loop: (i) selecting a pair of networks and reproducing its interaction, and (ii) analyzing the potential division of all the networks. The details are presented here:

A. Pair network interaction. Given a set of $n(t)$ networks, the algorithm tries the interaction of randomly chosen pairs of networks until one pair joins together moving into the next step with $n(t) - 1$ networks or until no further interaction is possible. In order to save computation time, the algorithm never repeats a failed interaction.

The interaction between the two chosen networks A and B is as follows. Note that the pre-existing links of each network (*intra-links*) are not modified during the interaction. In each step of this interaction, the algorithm randomly selects two nodes, $a \in A$ and $b \in B$ (*connector nodes*), and connects them through an undirected link (*connector link*). For simplicity, only one connector link per node is allowed, and therefore any pre-existing connector links associated with those nodes a and b are erased. The new state of this interaction is only accepted if both connector nodes increase their centrality measured as a dynamical importance plus a minimum payoff ω taken as $\omega = 10^{-6}$ (a small payoff is advisable to avoid computational noise, but the results are qualitatively the same if $\omega = 0$). If any of the two nodes does not increase its centrality, the system returns to its previous state and two new nodes are chosen. Note that the *dynamical importance* of each node l is measured as $I_l = \lambda_1 u_l$, where λ_1 is the largest eigenvalue of the adjacency matrix of the new network formed by A, B and the connector links added so-far during the interaction, and $\vec{u}$ represents its associated eigenvector, normalized such that $\sum u_l = 1$. u_l is the eigenvector centrality of node l, and measures the importance of a node based on how well connected it is and how important its neighbors are (1).

This interaction continues connecting A and B through different connector links until any of these three situations happens:

1. **Nash equilibrium:** The interaction reaches a state in which there are not new links that would satisfy the conditions for being incorporated, that is, any new link or rewiring of an existing link would make at least one of the involved connector nodes lose centrality and therefore would not be accepted.

2. **Cycle of states**: The interaction gets stuck in a situation where it is continuously changing among a small number of different states in a cycle. Once a cycle is detected, we pick a number from a Poisson variable with mean 5. This number determines the number of new successful connector links (or successful rewirings) accepted until we end the interaction. This process chooses preferentially the states that are more frequent in the cycle.

3. **Limit of successful tries**: As the size of the networks increases, the number of possible connector links rapidly increases. In order to reduce the computation time that might take the interaction, we impose a maximum limit of accepted

connector links. Once this limit is reached, the interaction stops in the last configuration. This limit is set as a function of the sizes of the networks. If two networks of sizes N_A and N_B are interacting, our limit of successful tries is $(N_A + N_B)^3$, which is in general several orders of magnitude larger than the average time to reach the end of the interaction through situations 1 or 2.

Finally, it could happen that no links have been accepted between A and B when all $N_A \cdot N_B$ potential connector links have been tried. We suppose then that A and B do not react and both remain unchanged. In this case, the time is not increased and the interaction between two other networks starts. Note that if two networks are not able to connect once, they will not do it in a future try. Therefore, if they are again chosen to interact, this selection is discarded to save computation time.

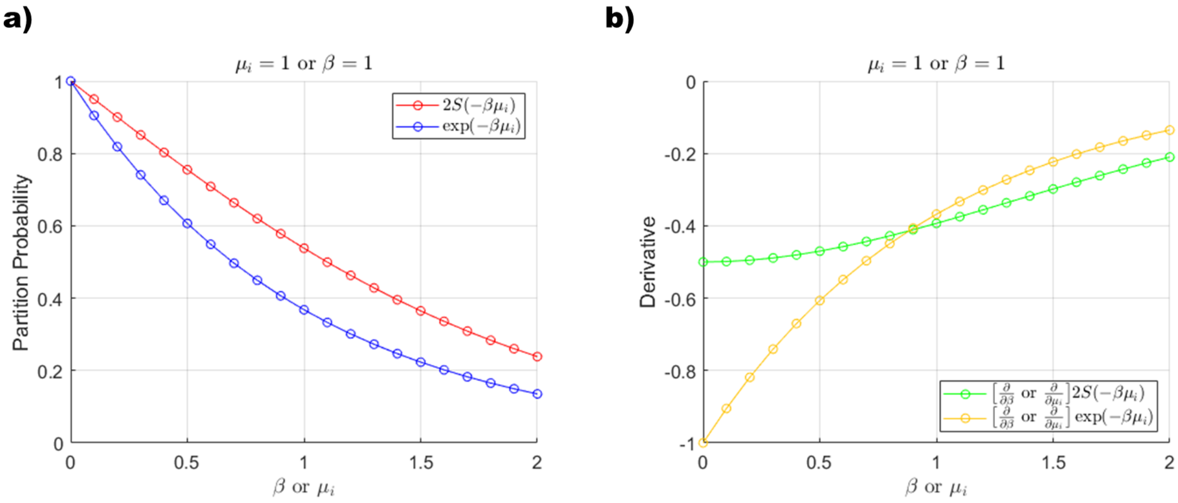

Fig. S2. Comparison of the partition probabilities $P = 2S(-\mu_i\beta)$ and $P' = \exp(-\mu_i\beta)$. Dependence of these two quantities (a) and their derivatives (b) on the environment parameter β for a fixed stability parameter of the network μ_i and on μ_i for a fixed β.

B. Partition of the networks. After a successful union of two networks, every network i in the ensemble is susceptible to break into smaller structures with a partition probability

$$P(\mu_i, \beta) = 2S(-\mu_i\beta) = \frac{2}{1 + \exp(\mu_i\beta)}\,, \qquad [1]$$

where (i) μ_i is the second smallest eigenvalue associated with the Laplacian matrix of each network and therefore is a proxy for its resistance to be divided in communities, and (ii) β is the environment parameter and is the only global parameter of the system. Note that when $\mu_i = 0$ network i is already divided into at least two components ($P(0, \beta) = 1$), and as μ_i increases $\lim_{\mu\to\infty} P(\mu_i, \beta) = 0$. The selection of the sigmoid function presented in Eq. 1 over the in principle simpler exponential function $P'(\mu_i, \beta) = \exp(-\mu_i\beta)$ is due to the difference in the sensitivity of both functions to the parameters. It is straightforward that

$$\begin{aligned}\frac{P}{P'} &= \frac{2}{1 + \exp(-\mu_i\beta)} > 1 \;\forall \mu_i, \beta > 0\,, &[2]\\ \frac{P}{P'} &= 1 \text{ for } \beta = 0 \text{ or } \mu = 0\,. &[3]\end{aligned}$$

Also,

$$\frac{\partial P/\partial\mu_i}{\partial P'/\partial\mu_i} = \frac{\partial P/\partial\beta}{\partial P'/\partial\beta} = \left[\frac{1}{1 + \exp(-\mu_i\beta)}\right]^2, \qquad [4]$$

which yields that $\partial P/\partial\mu_i > \partial P'/\partial\mu_i$ and $\partial P/\partial\beta > \partial P'/\partial\beta$ for $\mu_i\beta < -\ln(\sqrt{2} - 1) \sim 0.88$.

As a consequence, the sigmoid function allows the partition probability to be smaller and less sensitive to variations in the value of μ or β. For example, for low μ, i.e. for very unstable networks, the partition probability of networks with very similar stability μ would be much more different with P' than with P; in a similar way, little changes in the environment for low β would also perturb the system more drastically with P' than with P (see Fig. S2 for a numerical example). Let us note, however, that the phenomenology presented here would be equivalent using any of these two expressions for the partition probability.

In order to calculate the substructures in which each network can be divided at each time step, we use a community detection algorithm (2) based on the maximization of the modularity parameter

$$Q = \frac{1}{4m} \sum_{j,l} M_{jl} s_j s_l\,. \qquad [5]$$

The modularity matrix $\mathbf{M}$ of a network has elements

$$M_{jl} = G_{jl} - \frac{k_j k_l}{2m}\,, \qquad [6]$$

where G_{jl} are the elements of the adjacency matrix of the network, k_j is the degree (number of neighbors) of each node j, m is the total number of links and $s_j = 1$ or $s_j = -1$ depending on the community to which node j belongs to in a potential division. Note that Q increases with the number of links within communities and decreases with the number of links between communities (connector links), and thus is a measure of the goodness of a partition (3).

The method we have used is based on the spectral decomposition of the modularity matrix $\mathbf{M}$, and rewrites the modularity parameter as

$$Q = \frac{1}{4m}\vec{s}^{T}\mathbf{M}\vec{s} = \sum_{j=1}^{n}(\vec{v}_j \cdot \vec{s})^2\gamma_j\,, \qquad [7]$$

where n is the number of nodes of the network. Therefore Q can be computed as a function of $\mathbf{M}$'s eigenvalues γ_j and its eigenvectors $\vec{v}_j$. A first approximation to the maximum value of Q is to take the partition vector $\vec{s}$ to be parallel to the eigenvector $\vec{v}_1$ with largest eigenvalue. However, the entries of $\vec{s}$ can only take the values ± 1, therefore $s_j = 1$ if $v_{1j} \geq 0$ and $s_j = -1$ if $v_{1j} < 0$ where v_{1j} is the j-th entry of the leading eigenvector $\vec{v}_1$ of $\mathbf{M}$. Finally, note that this method does not necessarily make all networks divide into two smaller communities, as the two subdivisions created by the algorithm might not be fully connected (for example, a star-network of n nodes would be divided into the central node against the rest, but as the latter are not connected, the final action of the algorithm would be to split the network into n isolated nodes).

S2. Computation of the diversity and the relative abundance of the different configurations for a fixed environment in NetWorld

All different configurations detected at the end of any time step for the several realizations of the system with a fixed environment parameter β, even if they were destroyed later, represent its diversity, and the relative abundance of each configuration is given by the probability of finding it in the set of networks accumulated during all times and realizations. As a clarifying example, after 25 simulations of 10^4 time steps of the process for a certain β, (i) the diversity will be given by the number of different structures detected at the end of any time step throughout those 25 simulations, N_{conf}, and (ii) the relative abundance of each configuration will be equal to the ratio between the number of times that it was detected during all simulations and the total number of networks detected during all simulations. Note that if at a certain time t a configuration is present more than once, this multiplicity will be taken into account in the calculation of the relative abundance.

To obtain the number of possible configurations that can be generated in NetWorld for a certain value of β, we need a measure to determine whether or not two networks are equal. In our computational environment, networks have *unlabeled* nodes, that is, we consider that two networks represent the same configuration if their adjacency matrices coincide under a certain permutation of rows and columns (as an example, two square-networks are always considered the same configuration, no matter how we permute the nodes). Identifying and quantifying dissimilarities among networks is still a challenging problem of practical importance in many fields of network science, in special when networks are of different sizes (4, 5). As we only need to assess whether two networks are identical or not, we map each network into a feature vector $\vec{w} = (N, \lambda_1, \lambda_2, \mu, \bar{k}, H)$ composed of several topological quantities (the size of the network N, the first two largest eigenvalues of the adjacency matrix of the network λ_1 and λ_2, the second smallest eigenvalue of the Laplacian matrix of the network or Fiedler parameter μ, the mean degree $\bar{k}$ and the Shannon degree entropy H –calculated as $H = -\sum_{i=1}^{K} p_i \times \log_2(p_i)$, where p_i is the fraction of nodes of degree i, and K the maximum degree–), and we define the distance between two networks with associated vectors $\vec{w}_1$ and $\vec{w}_2$ as the Euclidean distance

$$d(\vec{w}_1, \vec{w}_2) = \|\vec{w}_1 - \vec{w}_2\|_2\,, \qquad [8]$$

assessing that two networks represent the same configuration if their distance is less than a threshold $d_{min} = 10^{-12}$.

S3. Computation of the number of paths to reach a configuration in NetWorld

In this work we have shown that the relative abundances of the different structures created in the NetWorld environment correlate with the number of paths identified to create them. Here we explain how we compute the number of different paths that lead to the creation of each configuration. For simplicity, we will focus on the case where there is no network partition, $\beta = \infty$, which is the one studied in the main manuscript (Fig. 4(f)).

We can define a path r that leads to the creation of a network as a succession $\{r_1, r_2, \cdots, r_{n-1}, r_n \mid r_n = r\}$, where each step of the path, r_i, is the resulting network of the union of a pair of networks (see Fig. S3 for an illustrative example with $N = 5$ initial nodes). For each number of initial nodes $N \leq 10$ and $\beta = \infty$, we simulated 10^4 realizations of the process, and for each configuration obtained, we selected all the paths that eventually ended on it. In order to obtain which paths were repeated in the list of potential paths for a certain configuration and should not be taken into account, we compared all paths two by two as follows. Let us suppose we must compare paths p_1 and p_2 for a certain configuration, being $p_1 = \{r_1, r_2, \cdots, r_{n-1}, r_n \mid r_n = r\}$ and $p_2 = \{r'_1, r'_2, \cdots, r'_{m-1}, r'_m \mid r'_m = r\}$. The first step is obviously to check that their lengths are equal, $n = m$. If this is so, we compare each step of the two paths. Since each element of the succession is represented by a network, we use the distance between networks defined in the Supporting Information Text S2 to compare them, and only when two paths are equal step by step will we erase one of them from the list of paths. The number of remaining paths to create one configuration after comparing all paths is considered the number of paths identified to create it.

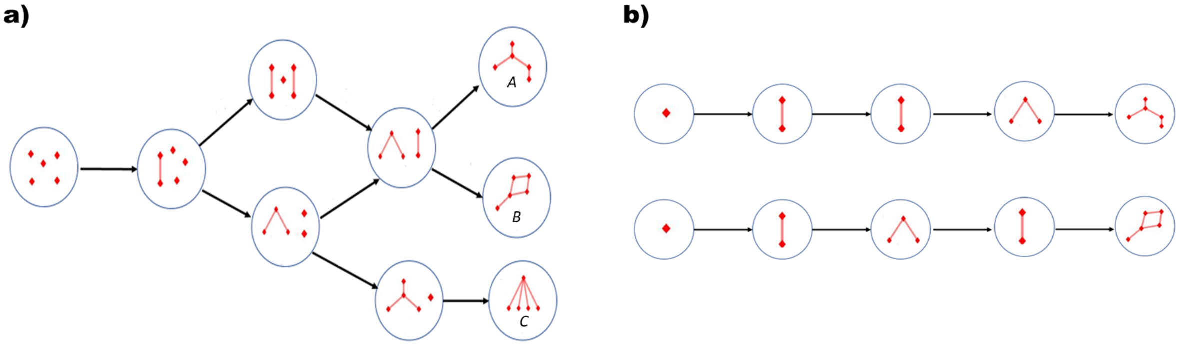

Fig. S3. a) Possible configurations and paths for $N = 5$ initial nodes and $\beta = \infty$ (networks can not be divided). Two paths can be identified for network A, two for network B, and one for network C. b) Examples of paths for configurations A and B as the succession of unions of networks.

S4. Dependence of the computation time with NetWorld's parameters

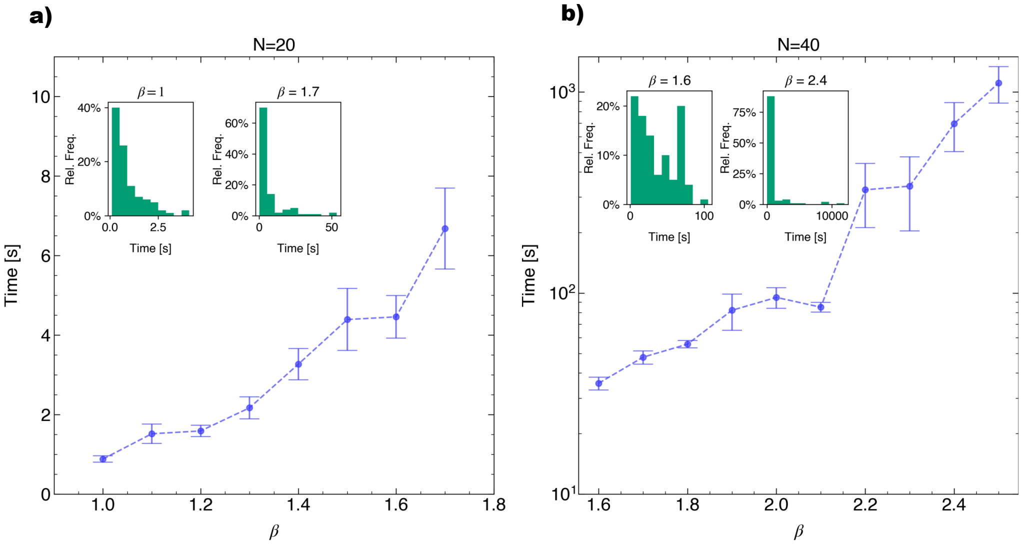

Fig. S4. Dependence of the computation time with the environment parameter β for $N = 20$ and $N = 40$ initial nodes. Embedded histograms show the distribution of computation times for two values of β. 100 simulations were done for each value of N and β with $T_{\max} = 10000$. Server specifications: Microsoft Windows Server 2019 Standard with an Intel(R) Xeon(R) Silver 4116 CPU and 128 GB of RAM.

For a given set of parameters N (number of initials nodes), β (environment parameter) and $T_{\max}$ (limit number of successful unions), the computation time strongly fluctuates due to the inner stochasticity of NetWorld. The most demanding process in terms of computation time present in NetWorld's algorithm is the union between two networks (described in the Supporting Information Text S1.A). The reason is that the computation time of a simulation will mostly depend on the number of interactions between large networks that occur during that realization, which ultimately depends on the number of initials nodes N and the environment parameter β, as shown in Fig. S4, where the relationship between the computation time and β for $N = 20$ and $N = 40$ is plotted. 100 simulations were calculated for each value of N and β with $T_{\max} = 10000$.

For low values of β, independently of the number of initial nodes, the computation time is low because the partition probability, given by Eq. 1, is very high and most networks remain small throughout the simulation. As β increases, however, the average of network sizes increases and therefore also does the probability of interactions between large networks. Obviously, this fact is strongly enhanced when N grows.

Finally, it is remarkable that the distributions of computation times plotted in Fig. S4 show that, even for large values of β and N, most of the simulations of the system are completed within short times, except for some very expensive realizations –those that face unions between especially large networks–, which make the average time grow substantially.

S5. Algorithm availability

For further insight on the algorithm, a Matlab(R2020b) implementation with description documentation and instructions of use is available at the following public repository https://github.com/MiguelGarciaSanchez/NetWorld

S6. Molecular abundances measured toward diffuse molecular, translucent and dense clouds

In this section we describe how we have obtained the molecular abundances plotted in Figs. 4(a-e) of the main manuscript for the cases of diffuse molecular, translucent and dense clouds. We note that all the molecular abundances were obtained with respect to molecular H_2, and the abundances that were presented as an upper bound were discarded.

As diffuse molecular cloud, we have selected the interstellar cloud ζ Ophiuci, whose visual extinction is $A_v = 1.06$ mag. Its molecular abundances with respect to H_2 were extracted from Table 2 of (6) and corrected by the fact that only 56% of the gas is in H_2. For translucent clouds, we have chosen the study carried out toward the cloud located in the foreground of the SgrB2 K4 ultracompact HII region (7, 8). Its visual extinction is A_v=2.0 mag. We have used the molecular column densities reported in Table 1 of (7) for the translucent cloud at $V_{LSR}\sim$3.4 km s^{-1} and a H_2 column density of 1.9×10^{21} cm^{-2} inferred by (9) for the same velocity component (see Table B.2 in (7)). The remaining of molecular abundances toward SgrB2 K4 were extracted from Table 4 of (8) for the velocity component at V_{LSR}=0 km s^{-1} after correcting them by the H_2 column density obtained by (9) (of 1.9×10^{21} cm^{-2}). Note that this component is the same as the one reported by (7) at $V_{LSR}\sim$3.4 km s^{-1}, because the velocity resolution of the data of (8, 10) was very poor (of 6-10 km s^{-1}). Finally, for the dense clouds L134N (Serpens) and TMC-1 (Taurus), with $A_v > 10$ mag, we extracted the molecular abundances directly from Table 4 reported in (11).

S7. Statistical analysis of the data associated with molecular clouds TMC-1 and L134N

Here we develop a statistical analysis of the relationship between the abundance of the molecules detected in interstellar clouds TMC-1 (Taurus) and L134N (Serpens) (data obtained from (11)), the number of reactions that have them as products (data obtained from the astrochemical reaction dataset KIDA (12)), and the molecular size, understood as the number of atoms contained within a molecule.

Figure S5 shows the relationship between these three magnitudes for both dense clouds. The Pearson correlation coefficients r and the corresponding p-values between the logarithms of the molecular abundance, the number of reactions and the molecular size for TMC-1 are

$$r = \begin{pmatrix} 1 & 0.57 & -0.47 \\ 0.57 & 1 & -0.53 \\ -0.47 & -0.53 & 1 \end{pmatrix}, \; p = \begin{pmatrix} 0 & 2\cdot10^{-6} & 10^{-4} \\ 2\cdot10^{-6} & 0 & 10^{-5} \\ 10^{-4} & 10^{-5} & 0 \end{pmatrix}. \qquad [9]$$

According to these results, the molecular abundance in TMC-1 correlates with the number of reactions (see Fig. S5(a) and Fig. 4(e) of the main manuscript) following the same functional dependence as NetWorld's simulations ($y \propto x^{\alpha}$, where $\alpha = 1.0 \pm 0.2$, $r = 0.57$, $p = 2\cdot10^{-6}$). A similar calculation for the data associated to the cloud L134N and plotted in Fig. S5(d) and Fig. 4(e) yields the same functional dependence between these two magnitudes ($y \propto x^{\alpha}$, $\alpha = 1.2 \pm 0.3$, $r = 0.54$, $p = 6\cdot10^{-4}$).

In probability theory, the partial correlation measures the degree of association between two random variables, removing the effect of a set of controlling random variables. For this statistical method to be specially accurate, it is convenient that the variables are normally distributed, are related linearly and do not have important outliers. All of these assumptions are satisified by our data, as far as we work with the logarithms of the datasets. In particular, all three magnitudes studied here show log-normal distributions, which means that the logarithms of the data are normally distributed (log-normal fit with correlation coefficient $r = 0.998$ for the molecular abundance, $r = 0.898$ for the number of reactions and $r = 0.933$ for the molecular size).

The Pearson partial correlation coefficients r' and the corresponding p-values between the molecular abundance, the number of reactions and the molecular size for TMC-1 are

$$r' = \begin{pmatrix} 1 & 0.42 & -0.26 \\ 0.42 & 1 & -0.36 \\ -0.25 & -0.36 & 1 \end{pmatrix}, \; p' = \begin{pmatrix} 0 & 8\cdot10^{-4} & 0.06 \\ 8\cdot10^{-4} & 0 & 4\cdot10^{-3} \\ 0.06 & 4\cdot10^{-3} & 0 \end{pmatrix}. \qquad [10]$$

Note that the partial correlation between the molecular abundance and the number of reactions shown in Fig. S5(a) and Fig. 4(e) is statistically significant (partial correlation coefficient $r' = 0.42$ and $p' = 8\cdot10^{-4}$), which means that they are correlated even when the effect of the molecular size is removed. Regarding the interstellar cloud L134N, similar calculations to those presented above for TMC-1 yield that the correlation between the molecular abundance and the number of reactions shown in Fig. S5(d) and Fig. 4(e) is also statistically significant when we remove the effect of the molecular size (partial correlation coefficient $r' = 0.45$ and $p' = 6\cdot10^{-3}$).

In summary, from a preliminary statistical study of the interaction between these three magnitudes, we infer that the abundances of the chemical compounds detected toward interstellar clouds TCM-1 and L134N show a correlation with the number of reactions that have them as a product that is independent of potential biases due to the molecular size.

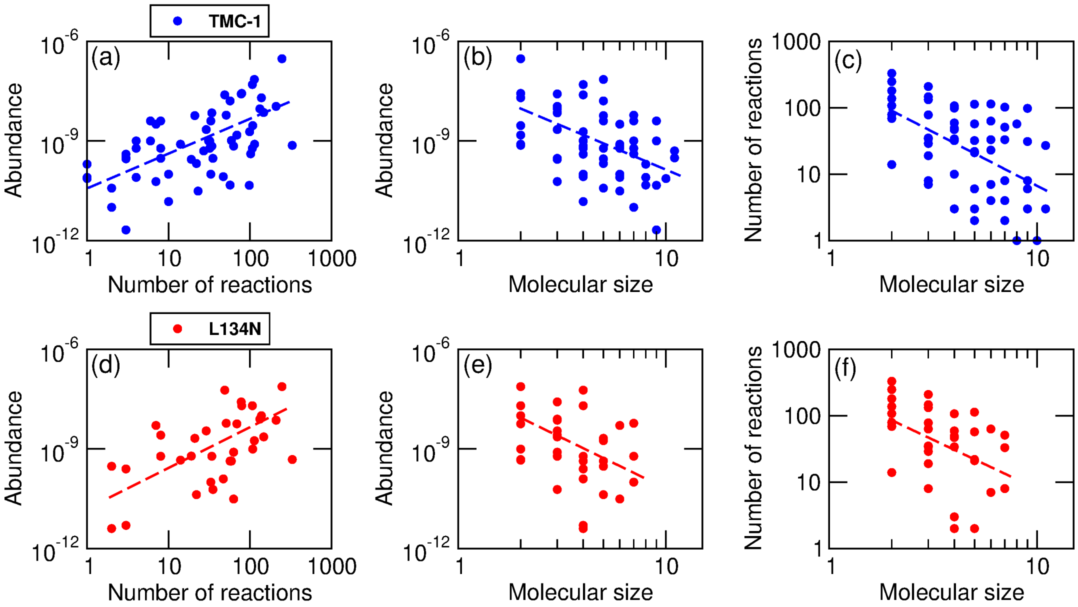

Fig. S5. Mutual dependence between the abundances of the molecules, the number of astrophysical reactions that show them as products, and the molecular size understood as the number of atoms contained within a molecule. The data correspond to interstellar clouds TMC-1 in Taurus (blue circles) and L134N in Serpens (red circles) (11). Curve fits to power-laws ($y \propto x^{\alpha}$) are plotted as dashed lines. Note that CO abundances were out of range for clarity in (a,b,d,e) but were considered in the fits.

References

1. Newman MEJ (2010) *Networks: An Introduction.* (Oxford University Press, Inc., New York, NY, USA).
2. Newman MEJ (2006) From the Cover: Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103:8577–8582.
3. Newman MEJ, Girvan M (2004) Finding and evaluating community structure in networks. *Physical Review E* 69(2):026113.
4. Schieber TA, et al. (2017) Quantification of network structural dissimilarities. *Nature Communications* 8(1).
5. Martínez JH, Chavez M (2019) Comparing complex networks: in defence of the simple. *New Journal of Physics* 21(1):013033.
6. Snow TP, McCall BJ (2006) Diffuse atomic and molecular clouds. *Annual Review of Astronomy and Astrophysics* 44(1):367–414.
7. Thiel, V., Belloche, A., Menten, K. M., Garrod, R. T., Müller, H. S. P. (2017) Complex organic molecules in diffuse clouds along the line of sight to Sagittarius B2. *A&A* 605:L6.
8. Corby, J. F., McGuire, B. A., Herbst, E., Remijan, A. J. (2018) The molecular chemistry of diffuse and translucent clouds in the line-of-sight to sgr b2: Absorption by simple organic and inorganic molecules in the gbt primos survey. *Astronomy & Astrophysics* 610:A10.
9. Winkel, B., et al. (2017) Hydrogen in diffuse molecular clouds in the milky way - atomic column densities and molecular fraction along prominent lines of sight. *Astronomy & Astrophysics* 600:A2.
10. Corby JF, et al. (2015) An ATCA survey of Sagittarius B2 at 7 mm: chemical complexity meets broad-band interferometry. *MNRAS* 452(4):3969–3993.
11. Agúndez M, Wakelam V (2013) Chemistry of dark clouds: Databases, networks, and models. *Chem. Rev.* 113(12):8710–8737.
12. Wakelam V, et al. (2015) The 2014 KIDA network for interstellar chemistry. *ApJS* 217(2):20.