



SOFTWARE TOOL ARTICLE

REVISED Network-based Observability and Controllability Analysis of Dynamical Systems: the NOCAD toolbox [version 2; peer review: 2 approved]

Dániel Leitold ^{1,2}, Ágnes Vathy-Fogarassy ^{1,2}, János Abonyi²

¹Department of Computer Science and Systems Technology, University of Pannonia, Egyetem u. 10, Veszprém, 8200, Hungary

²MTA-PE Lendület Complex Systems Monitoring Research Group, University of Pannonia, Egyetem u. 10, POB. 158, Veszprém, 8200, Hungary

v2 First published: 09 May 2019, 8:646 (<https://doi.org/10.12688/f1000research.19029.1>)
 Latest published: 18 Sep 2019, 8:646 (<https://doi.org/10.12688/f1000research.19029.2>)

Abstract

The network science-based determination of driver nodes and sensor placement has become increasingly popular in the field of dynamical systems over the last decade. In this paper, the applicability of the methodology in the field of life sciences is introduced through the analysis of the neural network of *Caenorhabditis elegans*. Simultaneously, an Octave and MATLAB-compatible NOCAD toolbox is proposed that provides a set of methods to automatically generate the relevant structural controllability and observability associated measures for linear or linearised systems and compare the different sensor placement methods.

Keywords

Dynamical systems, Complex networks, Controllability and observability analysis, Robustness, MATLAB toolbox

Open Peer Review

Reviewer Status

	Invited Reviewers	
	1	2
REVISED		
version 2	report	report
published 18 Sep 2019		
version 1		
published 09 May 2019	report	report

- 1 **Gilles Didier** , University of Montpellier, Montpellier, France
- 2 **Arthur Montanari** , Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil
- Ercílio Moreira** , Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil
- Luis A. Aguirre** , Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

Any reports and responses or comments on the article can be found at the end of the article.

Corresponding author: János Abonyi (abonyij@fmt.uni-pannon.hu)

Author roles: **Leitold D:** Methodology, Software, Writing – Review & Editing; **Vathy-Fogarassy Á:** Formal Analysis, Methodology, Writing – Review & Editing; **Abonyi J:** Conceptualization, Methodology, Software, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This research was supported by the National Research, Development and Innovation Office NKFIH, through the project OTKA-116674 (Process mining and deep learning in the natural sciences and process development) and the EFOP-3.6.1- 16-2016- 00015 Smart Specialization Strategy (S3) Comprehensive Institutional Development Program. Dániel Leitold was supported by the ÚNKP-18-3 New National Excellence Program of the Ministry of Human Capacities.

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2019 Leitold D *et al.* This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Leitold D, Vathy-Fogarassy Á and Abonyi J. **Network-based Observability and Controllability Analysis of Dynamical Systems: the NOCAD toolbox [version 2; peer review: 2 approved]** F1000Research 2019, 8:646 (<https://doi.org/10.12688/f1000research.19029.2>)

First published: 09 May 2019, 8:646 (<https://doi.org/10.12688/f1000research.19029.1>)

REVISED Amendments from Version 1

According to the invaluable remarks of the reviewers, we made the following changes in our manuscript: the most significant change is that we rewrote the section 'Use cases', and we changed the example network to the frontal neural network of *C.elegans* from standard biological datasets in order to highlight the potential of the NOCAD toolbox in the field of biology. According to this modification Figures 2, 3, 4, 5, 6 and 7 were removed from and two tables were added to the manuscript, that contains the new results. We extended the section 'Methods' with the basic notations of the methodology, such as the definition of controllability, observability and maximum matching and introduce them through a subnetwork of the frontal neural network of *C.elegans*. According to this modification, two new equations and a new Figure about the subnetwork were added to the section. Furthermore, we also made minor changes to improve the quality of our manuscript: the abstract and the introduction were improved, existing software was introduced in a separate chapter, the grammar of the paper was improved by a native speaker and 10 new references were added to the manuscript.

Any further responses from the reviewers can be found at the end of the article

Introduction

In the life sciences, the determination of driver nodes in networks that play a significant role in the emergence or treatment of diseases is an intensively researched field¹. The importance of determining the proper driver nodes, i.e. the ones that ensure physically feasible controllability with the minimum cardinality and energy requirement, in biological networks, or more generally in any dynamical system, is unequivocal, and the amount of research concerning network science has increased rapidly. A detailed study of the control principles in biological networks has already been published². A review about the utilisation of the network science-based determination of driver nodes has also been published that introduced the results of the analysis of the protein-protein interaction (PPI) networks, *Caenorhabditis elegans* neuronal network, neurochemical rat brain network, *Saccharomyces cerevisiae* cell cycle networks, Epithelial Mesenchymal Transition (EMT) network, myeloid differentiation regulatory network and Th differentiation network, moreover, the identification of drug targets was also presented³.

The network science-based analysis of dynamical systems has spread rapidly as it provides simple and efficient tools to analyse the structural controllability of any linear or linearised system¹. In terms of controlling the human signalling network, the role of different proteins was also systematically analysed with the toolset of network controllability in 4 to highlight the role of cancer-associated genes. Target control with objective-guided optimisation (TCO) was introduced to control a set of variables (or targets) of interest while the number of drivers and constrained nodes were minimised and maximised, respectively. This method is capable of determining the leading phenotype transitions in biological networks that can be identified as drug targets⁵. In large-scale human liver metabolic networks (HLMN), the driver metabolites have essential functions, moreover, the role of transport reactions and extracellular metabolites in terms of controlling HLMN have revealed

the importance of the environment of human liver metabolism with regard to the health of the liver⁶. Using statistical analysis, a subset of critical control nonprotein-coding RNAs (ncRNAs) enriched by human disease can also be determined⁷. In intra-cellular networks, to understand the information flow, a natural control system was utilised and the robustness of such a control was analysed⁸.

The contribution of this paper is to introduce the novel toolbox, NOCAD⁹, and its applicability in the life sciences through the example of the local network of 131 frontal neurons of *Caenorhabditis elegans*¹⁰. The proposed toolbox is also suitable for the comprehensive analysis of any linear or linearised dynamical systems through their static network representation¹¹⁻¹³. Although in the literature the phrase dynamical network is commonly used, it does not mean that the nodes or connections are temporal but refers to the network of dynamical systems. In the nonlinear case, the methodology needs further clarification because for small nonlinear examples the results can be incorrect¹⁴ and the cardinality of the assigned sensors underestimated¹⁵. As a result, this toolbox deals with only the linear case, nonlinear system-related methods will be implemented later. In the following sections, the representation of linear systems as well as their structural controllability and observability are introduced. Then the theoretical background of the methodology is presented and the implemented functions and measurements introduced through the network of rostral ganglia of *C.elegans*.

Existing software

Although considerable research has utilised this method¹⁶, a flexible software tool which may be used to support research in this field has yet to be designed. Parallel studies have resulted in a collection of applications, toolboxes, plug-ins and scripts that analyse and determine several structural properties of genes, protein-protein interactions and even social or urban networks. Most of these applications only analyse the structural properties of static networks and just a handful of them utilise these structural properties to draw conclusions concerning the dynamics of the system investigated. As our toolbox belongs to the second group, in the following section, the available applications and programs of this group are elaborated on.

A brief summary of the available tools with expanded functionalities is given in Table 1. Applications or software packages implemented in Python and capable of analysing the controllability and observability of dynamical systems are: graph-control¹⁷ and WDNfinder¹⁸. The advantage of Python-based development lies in its widespread use and the countless methods and packages implemented in this language, including the tools developed for network analysis¹⁹. Although in Python the focus is on developing a broad software package for complex systems analysis, this has yet to be fulfilled and all of the available solutions have limitations. The graph-control toolbox only analyses the impact of network topology on the number of inputs and implements the fast matching algorithm²⁰. Even though WDNfinder only determines the minimum driver node set (MDS) and classifies nodes based on MDS, it is incapable of facilitating extended analysis.

Table 1. Toolboxes that implement some functions for the dynamical analysis of complex systems based on their structural analysis.

Software	Language	Applied on	GUI	Ref.	Last updated
netctrl	C++	General networks	No	21	January 8, 2015
CONTEST	MATLAB	General networks	No	22	February, 2009
CytoCtrlAnalyser	Java	Biomolecular networks	Yes	23	May 25, 2017
graph-control	Python	General networks	No	17	December 16, 2015
WDNfinder	Python	Biological networks	No	18	June 24, 2018
enaR	R	Ecological networks	No	24	May 18, 2018

Additionally, the CytoCtrlAnalyser²³ plug-in for Cytoscape²⁵ has been developed, which was implemented in Java and offers graphical user interfaces as well. It evaluates control centrality, control capacity and classifies nodes for biomolecular networks. Furthermore, the Ecological Network Analysis in R software package (enaR) provides some dynamical analysis functions and can generate models to analyse ecological networks in the R environment²⁴. As can be seen, both software packages deal with special kinds of networks. The netctrl program can determine the driver nodes and switchboard dynamics model for any complex network²¹. CONTEST is a MATLAB toolbox which can analyse the dynamics of complex systems, but these dynamics do not cover the structural controllability and observability properties²² of the analysed system. Although the presented software packages ensure the design of a controllable and observable system, they do not provide the opportunity to analyse the designed system exhaustively. These functions are helpful in terms of supporting the work of experts, but are insufficient for the sophisticated analysis of systems.

Methods

In the background of the toolbox the linear systems and their structural controllability and observability properties are studied²⁶. A linear time-invariant (LTI) system is commonly described by its state-space representation that consists of the state equation (Eq. 1) and the output equation (Eq. 2).

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu} \quad (1)$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du} \quad (2)$$

In the state-space representation, \mathbf{x} stands for state variables, \mathbf{u} represents the inputs, i.e. the actuators, and \mathbf{y} denotes the vector of outputs, i.e. the sensors of the system. Matrices \mathbf{A} and \mathbf{B} define how state variables and inputs influence changes to the state variables, while matrices \mathbf{C} and \mathbf{D} define how state variables and inputs influence the outputs, respectively. The cardinality of state variables, inputs and outputs are noted by N , M and K , respectively.

A dynamical system is said to be controllable if it can be driven from any initial state to any desired final state within a finite time with properly selected inputs. Observability is the

mathematical dual of controllability. A system is said to be observable if its state can be determined at a given time by a finite set of measured input and output variables. Kalman's rank criterion²⁷ was used to determine the structural controllability and observability as follows: if the rank of the controllability matrix is equal to the number of state variables, $\text{rank}(\mathbf{C}) = N$, then the system is structurally controllable, where $\mathbf{C} = [\mathbf{B}, \mathbf{AB}, \dots, \mathbf{A}^{N-1}\mathbf{B}]$. Analogously, if the rank of the observability matrix is equal to the number of state variables, $\text{rank}(\mathbf{O}) = N$, then the system is structurally observable, where $\mathbf{O} = [\mathbf{C}^T, (\mathbf{CA})^T, \dots, (\mathbf{CA}^{N-1})^T]^T$.

To ensure controllability (or observability) using a minimum number of inputs (or outputs), a brute force approach should generate $2^N - 1$ configurations of matrix \mathbf{B} (or \mathbf{C}). To solve this challenging task, the maximum set of disjoint edges is generated by the maximum matching algorithm¹. Two edges are disjoint if they do not share a common starting point or endpoint. The matched nodes are the endpoints of the edges that are a member of the maximum set of disjoint edges, the others are unmatched. Then the unmatched nodes that are generated based on \mathbf{A} are the sensor nodes, where outputs should be placed to grant structural observability, while the unmatched nodes generated based on \mathbf{A}^T are the driver nodes, where inputs should be placed to grant structural controllability. \mathbf{A}^T is also the adjacency matrix of the network representation that is the input of the toolbox. It is very important to note that the result of maximum matching is not unique, and it is possible that the matching is perfect, i.e. no unmatched nodes have resulted. In our implementation, the canonical decomposition of Dulmage-Mendelsohn was utilised to calculate maximum matching²⁸.

For a better understanding, we illustrate the aforementioned definitions by a small example in Figure 1 that contains the command interneurons AVAL, AVAR, AVBL, AVBR, AVDL and AVDR from the frontal neural network of neurons and synapses in *C. elegans*.

With the help of the presented Octave- and MATLAB-compatible toolbox, experts can create, analyse and improve any type of dynamical systems. As the structure of the dynamical systems is generally represented by their adjacency matrix and linear

dynamical systems can be described by the state-space model that contains the dynamical, input, output and feedthrough matrices, the Octave/MATLAB programming language is a perfect environment to handle these matrices and provide comprehensive functionalities based on them. With the use of NOCAD⁹, experts and researchers can effectively determine the input and output matrices of state-space models, calculate system-specific qualitative measurements (e.g. diameter, relative degree, control centrality and robustness of the system, etc.) and improve the system to satisfy the relative degree-based requirements. The workflow of the toolbox can be seen in Figure 2.

Implementation

According to the aforementioned approach, the implemented functions of the toolbox were divided into three modules as follows: (1) network mapping module, (2) system characterisation module and (3) improvements and robustness module. The input of the first module is the adjacency matrix of the network to be analysed. The second module requires the matrices of the dynamical system generated by the first module. The result

of the second module is a structure that is also the input of the third module.

The *network mapping* module creates a dynamical system from a given network structure, i.e. the necessary matrices of the state-space model are generated for the topology in such a way, that the created system is structurally controllable and structurally observable. The determination of the input and output matrices can be achieved by the path finding and signal sharing methods¹¹, which modify the result of the maximum matching algorithm.

The *system characterisation* module performs the calculation of 49 numerical measures to qualify the dynamical system based on its structure. The implemented measures, on the one hand, are well-known static measures (e.g. the number of nodes and edges, closeness and betweenness centralities), and, on the other hand, measures that characterise the dynamics of the system (e.g. structural controllability, observability, control centrality and relative degree). This module can also be used for the purpose of simple network analysis.

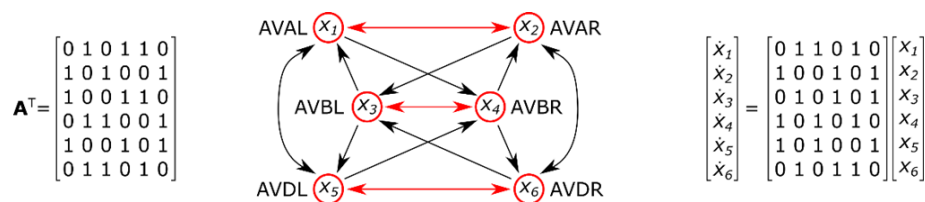


Figure 1. Representation of the command interneurons as a linear dynamical system. The adjacency matrix of the command interneurons, their network representation and the state equation without assigned input. In this example, due to the symmetric edge pairs between the nodes, the matching is perfect, i.e. all the nodes are matched. In this case, structural controllability and observability can be granted by selecting any node as a driver node and any node as a sensor node.

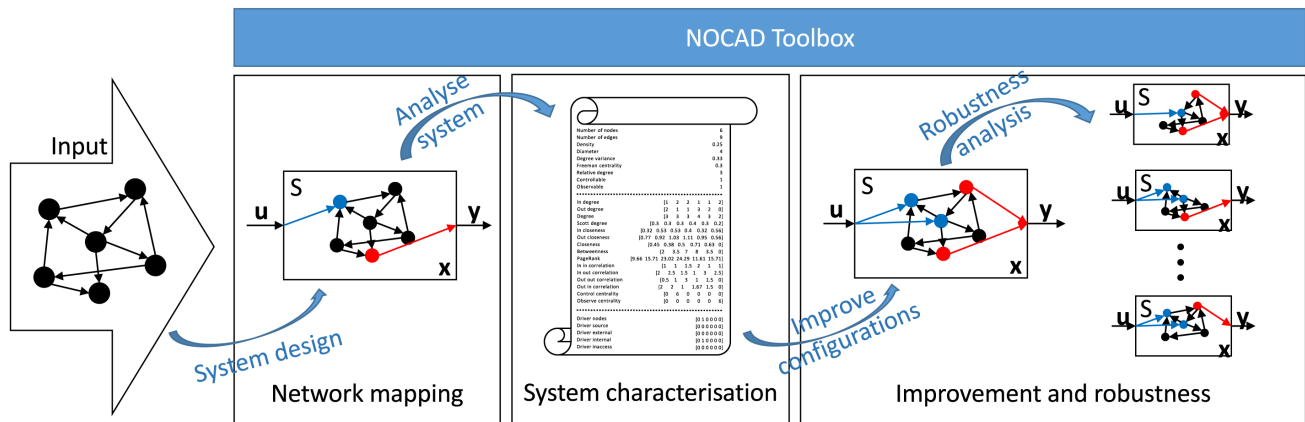


Figure 2. Workflow of the utilisation of the NOCAD toolbox. The network mapping module provides two methods to create a dynamical system based on the topology of the state variables. The system characterisation module generates more than 49 measures to analyse, classify and characterise the developed system. The improvement and robustness module offers five algorithms to improve the system with additional inputs (observers) as well as outputs (controllers) and can analyse the robustness of the designed system.

The *improvement and robustness module* integrates two main functions. On the one hand, it enables the input and output configurations of the system to be extended in such a way that the relative degree of the modified system does not exceed the initially defined threshold. For this purpose, this module implements five methods, namely the set covering-based grassroot and retrofit methods¹², the centrality measures-based method¹², the modified Clustering Large Applications based on Simulated Annealing algorithm (mCLASA), and the Geodesic Distance-based Fuzzy c-Medoid Clustering with Simulated Annealing algorithm (GDFCMSA)^{12,13}. On the other hand, this module allows users to examine the robustness of the extended configurations by removing nodes from the network representation and by checking the structural controllability and structural observability of the damaged system.

Although the last module seems to be out of line at first, its existence is reasonable. The importance of the controllability of a complex system has already been addressed¹. In terms of control theory, the relative degree is an important measure to describe how fast the system can be influenced or how sluggish it is. In the field of biology, this “speed” is also important, e.g. the time elapsed between taking a painkiller and feeling its effect. The implemented methods are introduced in detail in the cited articles and the manual of the NOCAD toolbox.

Operation

In order to use the NOCAD toolbox⁹, installation of Octave or MATLAB is required. Then the directories of the toolbox must be copied into the working directory, or the directories of the toolbox must be added to the paths. The functions were implemented in Octave 5.1.0 and MATLAB R2016a on a Windows 64-bit system. On other operating systems, or with other Octave or MATLAB versions, proper operation is not guaranteed. Our toolbox is independent of other Math-Works toolboxes, it uses only the octave-networks-toolbox²⁹ and the greedy set covering implementation³⁰.

Use cases

In this section, the main functionalities of the NOCAD toolbox⁹ are presented through the analysis of the local network of 131 frontal neurons of *Caenorhabditis elegans*. The first step in the workflow is to create a state-space model based on the adjacency matrix that presents the structural description of the system that, in this case, has the size of 131×131 according to the 131 frontal neurons.

Two methods, path finding and signal sharing are proposed that were implemented to correct the insufficient result of maximum matching. Both methods are modified versions of the maximum matching algorithm. The maximum matching method determined the following 12 neurons to be driver nodes: RMEL, RMER, SIADL, SIADR, SIAVL, SIAVR, SIBDL, SIBDR, SIBVL, SIBVR, SMDDR and URYDR, moreover, determined 12 sensor nodes that correspond to the following neurons: AINL, ASHL, ASIR, ASJR, AWAL, IL2DL, IL2DR, IL2L, SIBDL, URBL, URBR and URYDL. As no critical strongly connected components were present, the results

were identical in the case of both the path finding and signal sharing methods.

After utilising the second module of the toolbox, the measures that qualify the whole network with one value are introduced, as presented in Table 2. The network contains 131 neurons and 764 synapses. The density shows that the number of edges is less than a twentieth of the possible maximum, and the diameter of the system, namely the longest shortest path in the network that presents its structure, is 9. The degree variance is 44.3299 which is relatively high given the size of the network, while the Freeman’s centrality is 0.2057. The relative degree of the system is also 4. The Pearson correlation coefficient shows that the in-in, in-out and out-out correlations are slightly assortative in nature, while the out-in correlation is likely to be disassortative. The system is controllable and observable. As no loop is present in the network, the percentage of loops relative to edges is 0%. As 77 symmetrical connections are present between 687 connected node pairs, the percentage of the symmetric edge pairs is 11.2082%.

The second module generates node centrality measures that can reveal structurally important nodes. Since the generated measures can be presented by large tables, they are attached in Excel format to the toolbox⁹. This analysis shows that one of the most important values is the highest degree of the nodes, which belongs to RIAR, an interneuron located in the nerve ring³¹. As Scott’s centrality is a normalised degree, the most important node is once again RIAR. The closeness of node x_i is calculated as the ratio of the number of nodes reachable from x_i to the sum of their distances from x_i . The higher value indicates

Table 2. Centrality measures of the system generated for the neural network of *C. elegans*.

Measure	Value
controllability	1
observability	1
number of nodes	131
number of edges	764
density	0.0445
diameter	9
Freeman's centrality	0.2057
degree variance	44.3299
relative degree	4
Pearson in-in	0.0426
Pearson in-out	0.0048
Pearson out-out	0.1694
Pearson out-in	-0.1524
percentLoops	0
percentSym	11.2082

the more central position of the node, and now RIAL is the most central element. The betweenness centrality shows how many shortest paths intercept the given node. If a node has a high value, then it is a critical node in the structure. The highest value belongs to neurotransmitter RIH that is a serotonin³². The PageRank assigns a percentage value to each node, based on their centrality roles if Markov-chains are modelled. The measure referred to as correlation shows the proportion of the number of edges of neighbours' and the number of neighbours. This information is useful when determining the assortativity of the system. The control centrality and observe centrality measures determine how many state variables can be influenced or observed by the nodes.

The determined driver and sensor nodes can be classified into four groups³³. According to these groups, four phenomena can provide driver or sensor nodes. Firstly, *source nodes* when the node has no incoming edges, thus, a dedicated input is needed. Secondly, *dilation*, when the generated set of child nodes has higher cardinality than the number of parent nodes. A distinction is made between *internal dilation* and *external dilation*, in the former the child node is not a leaf, i.e. it has children, while in the latter the child is a leaf node, i.e. it has no children. The last type is the *inaccessible nodes* when the node has an incoming edge and no dilation is present, but the node is not reachable by a directed path from any of the inputs. These types are important properties, e.g. the existence of dilation or inaccessibility is detrimental to complete structural controllability³. The controlling and observing matrices are sparse matrices as only the columns of drivers and sensors contain nonzero values. The values show the number of derivations necessary to influence or observe a state variable in the system. Next, the similarity of the driver and sensor nodes is presented. This similarity is based on how similar the set of nodes is, which can be reached for driving or observing. Furthermore, the necessary derivation to influence or observe them is also part of the comparison. R_c and R_o are the simple reachability matrices. They show which nodes can be controlled or observed by a given node in its structural meaning, i.e. the existence of a directed path between the nodes is shown. In R_c , the i^{th} column shows which nodes can control node i . From the other viewpoint, elements in row i highlight those nodes which can be controlled by node i . It is very important that R_c is only a reachability matrix, the structural controllability of the reachable nodes is not granted by a node that can reach them, but in some cases the structural controllability problem can be reduced to a reachability problem³⁴. The R_o matrix can be interpreted analogously with regard to observability.

Finally, measures of edge centrality are generated by the system characterisation module. The betweenness has the same meaning as in the case of nodes, that is, it yields the number of shortest paths that intercept the edge³⁵. From this perspective, the most critical synapsis is the one between the command interneuron AVAL and amphid ADLL with a value of 640.5833. The endpoint similarity shows how similar the influenced and observed sets of the state variables with regard to the endpoints

of edges are. This metric has a high value if the edge is part of a cycle or creates a bridge in the network. As no bridges are present in this network, only cycles can be recognised by this measure. The edge similarity shows how similar the roles of edges are, and it allows redundancies, to be located.

For the demonstration of the last module, four plus one methods were applied to the neural network of *C. elegans*. The set covering-based grassroot method (SetCovGr) optimises the placement of driver nodes and sensor nodes to provide an initially demanded relative degree, but this method does not take into account the original input and output configurations also, thus, structural controllability and observability is not granted also. The other four methods grant controllability and observability by expanding the minimal configurations. They are the centrality measures-based (CentMeas) retrofit, set covering-based retrofit (SetCovRet), modified Clustering Large Applications based on Simulated Annealing (mCLASA) and Geodesic Distance-based Fuzzy c-Medoid Clustering with Simulated Annealing algorithm (GDFCMSA) methods¹³. These methods were utilised with the following parameters: the required relative degree was set at 2, while the alpha parameter of the cost function was set at 0.5¹³. The results can be seen in Table 3. The number of assigned driver nodes varies significantly when different methods are applied. The centrality measures-based method assigned the most driver nodes to the system. Thus, this method results in the smallest cost, but the difference is irrelevant, most of the methods resulted in a cost of 1.5. The increase of the number of the driver nodes decreases the mean relative degree, which is the lowest in the case of the centrality measures-based method.

The robustness of the configuration was also analysed. In each scenario, a node was removed from the network. Using the leave-one-out strategy, the network with the altered configuration remains controllable in 115 scenarios. As for the sensor nodes, the difference is not as significant between the methods as in the case of the driver nodes. Critical nodes were also generated. A node is critical if the system becomes uncontrollable or unobservable if the node is removed. The determined critical nodes and the names of selected driver and sensor nodes can be found in the Excel file attached to the toolbox.

Conclusions

Although numerous papers have utilised the network-based determination of driver and sensor nodes, a flexible toolbox that may be used to support the analysis has yet to be designed. To fill this gap, in this article the Octave- and MATLAB-compatible NOCAD toolbox⁹ was proposed to support the network-based controllability and observability analysis of dynamical systems, and through the analysis of the neural network of *C. elegans*, the applicability of the toolbox in the life sciences was presented. The toolbox offers two methods to design a structurally controllable and observable system based on the adjacency matrix (A^T). The designed system can be analysed by 49 qualitative measures both from structural and dynamical points of view. The toolbox serves five methods to improve the

Table 3. Improved input and output configurations for the neural network of *C. elegans* with the required relative order of 2.

	CentMeas	SetCovRet	SetCovRet	mCLASA	GDFCMSA
number of drivers	27	17	16	21	19
cost	1.4580	1.5610	1.5496	1.5382	1.5649
relative degree	2	2	2	2	2
mean of rel. deg.	0.9160	1.1221	1.0992	1.0763	1.1298
input robustness	117	116	116	120	117
input robustness (%)	0.8931	0.8855	0.8855	0.9160	0.8931
number of sensors	23	16	12	19	20
cost	1.4924	1.5687	1.6336	1.5382	1.5267
relative degree	2	2	2	2	2
mean of rel. deg.	0.9847	1.1374	1.2672	1.0763	1.0534
output robustness	121	121	120	121	121
output robustness (%)	0.9237	0.9237	0.9160	0.9236	0.9236

designed system by adding new inputs and outputs to it, thus, its relative degree can be decreased. Then the robustness of the individual designs can also be evaluated. The modular structure of the toolbox supports the facile improvement of the modules by adding new functions and the toolbox can be extended by new modules as well. Even though the modules are built on each other, most of their functions can also be used independently from each other.

Although our goal in this paper is to draw the attention of researchers of life sciences to the services provided by the NOCAD toolbox, it can be utilised in practice in various fields of sciences as well, for example, it enables social networks to be controlled in the economy, transaction networks to be analysed in finance or dynamical systems to be designed in engineering.

Data availability

All data underlying the results are available as part of the article and no additional source data are required.

Software availability

Source code available from: <https://github.com/abonyilab/NOCAD>.

Archived source code at time of publication: <https://doi.org/10.5281/zenodo.2656674>⁹

License: GNU General Public License v3.0

Author contributions

Dániel Leitold reviewed the literature on network science, developed the algorithms, implemented the Octave and MATLAB functions, designed as well as performed the experiments, and wrote the related sections. Ágnes Vathy-Fogarassy participated in the formalisation of the methodology. János Abonyi developed the algorithms, implemented the Octave and MATLAB functions and proofread the paper.

References

- Liu YY, Slotine JJ, Barabási AL: **Controllability of complex networks.** *Nature.* 2011; **473**(7346): 167–73.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Li M, Gao H, Wang J, *et al.*: **Control principles for complex biological networks.** *Brief Bioinform.* 2018.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Wu L, Li M, Wang JX, *et al.*: **Controllability and Its Applications to Biological Networks.** *J Comput Sci Technol.* 2019; **34**(1): 16–34.
[Publisher Full Text](#)
- Liu X, Pan L: **Identifying driver nodes in the human signaling network using structural controllability analysis.** *IEEE/ACM Trans Comput Biol Bioinform.* 2015; **12**(2): 467–72.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Guo WF, Zhang SW, Shi QQ, *et al.*: **A novel algorithm for finding optimal driver nodes to target control complex networks and its applications for drug targets identification.** *BMC Genomics.* 2018; **19**(Suppl 1): 924.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Liu X, Pan L: **Detection of driver metabolites in the human liver metabolic**

- network using structural controllability analysis.** *BMC Syst Biol.* 2014; **8**(1): 51.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
7. Nacher JC, Akutsu T: **Controllability methods for identifying associations between critical control ncRNAs and human diseases.** *Methods Mol Biol. In Computational Biology of Non-Coding RNA.* 2019; **1912**: 289–300.
[PubMed Abstract](#) | [Publisher Full Text](#)
 8. Ravindran V, Nacher JC, Akutsu T, *et al.*: **Network controllability analysis of intracellular signalling reveals viruses are actively controlling molecular systems.** *Sci Rep.* 2019; **9**(1): 2066.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 9. Abonyi J: **abonyilab/nocad v2.0.** 2019.
<http://www.doi.org/10.5281/zenodo.2656674>
 10. Kaiser M, Hilgetag CC: **Nonoptimal component placement, but short processing paths, due to long-distance projections in neural systems.** *PLoS Comput Biol.* 2006; **2**(7): e95.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 11. Leitold D, Vathy-Fogarassy Á, Abonyi J: **Controllability and observability in complex networks—the effect of connection types.** *Sci Rep.* 2017; **7**(1): 151.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 12. Leitold D, Vathy-Fogarassy Á, Abonyi J: **Evaluation of the complexity, controllability and observability of heat exchanger networks based on structural analysis of network representations.** *Energies.* 2019; **12**(3): 513.
[Publisher Full Text](#)
 13. Leitold D, Vathy-Fogarassy Á, Abonyi J: **Network distance-based simulated annealing and fuzzy clustering for sensor placement ensuring observability and minimal relative degree.** *Sensors (Basel).* 2018; **18**(9): pii: E3096.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 14. Sun J, Cornelius SP, Kath WL, *et al.*: **Comment on "Controllability of Complex Networks with Nonlinear Dynamics"**. *arXiv preprint arXiv:1108.5739.* 2011.
[Reference Source](#)
 15. Letellier C, Sendiña-Nadal I, Aguirre LA: **Nonlinear graph-based theory for dynamical network observability.** *Phys Rev E.* 2018; **98**(2–1): 020303.
[PubMed Abstract](#) | [Publisher Full Text](#)
 16. Liu YY, Barabási AL: **Control principles of complex systems.** *Rev Mod Phys.* 2016; **88**(3): 035006.
[Publisher Full Text](#)
 17. Chaturvedi V: **Controllability of networks.**
[Reference Source](#)
 18. Chu Y, Wang Z, Wang R, *et al.*: **Wdnfinder: A method for minimum driver node set detection and analysis in directed and weighted biological network.** *J Bioinform Comput Biol.* 2017; **15**(5): 1750021.
[PubMed Abstract](#) | [Publisher Full Text](#)
 19. Zinoviev D: **Recognize-Construct-Visualize-Analyze-Interpret.** Pragmatic Bookshelf. 2018.
[Reference Source](#)
 20. Faradonbeh MKS, Tewari A, Michailidis G: **Optimality of fast-matching algorithms for random networks with applications to structural controllability.** *IEEE Trans Control Netw Syst.* 2017; **4**(4): 770–780.
[Publisher Full Text](#)
 21. Nepusz T, Vicsek T: **Controlling edge dynamics in complex networks.** *Nat Phys.* 2012; **8**(7): 568–573.
[Publisher Full Text](#)
 22. Taylor A, Higham DJ: **Contest: A controllable test matrix toolbox for matlab.** *ACM Trans Math Softw.* 2009; **35**(4): 26.
[Publisher Full Text](#)
 23. Wu L, Li M, Wang J, *et al.*: **Cytoctrlanalyser: a cytoscape app for biomolecular network controllability analysis.** *Bioinformatics.* 2018; **34**(8): 1428–1430.
[PubMed Abstract](#) | [Publisher Full Text](#)
 24. Borrett SR, Lau MK: **enaR: an R package for ecosystem network analysis.** *Methods Ecol Evol.* 2014; **5**(11): 1206–1213.
[Publisher Full Text](#)
 25. Shannon P, Markiel A, Ozier O, *et al.*: **Cytoscape: a software environment for integrated models of biomolecular interaction networks.** *Genome Res.* 2003; **13**(11): 2498–504.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 26. Lin CT: **Structural controllability.** *IEEE Trans Automat Contr.* 1974; **19**(3): 201–208.
[Publisher Full Text](#)
 27. Kalman RE: **Mathematical description of linear dynamical systems.** *Journal of the Society for Industrial and Applied Mathematics, Series A: Control.* 1963; **1**(2): 152–192.
[Publisher Full Text](#)
 28. Pothén A, Fan CJ: **Computing the block triangular form of a sparse matrix.** *ACM Trans Math Softw.* 1990; **16**(4): 303–324.
[Publisher Full Text](#)
 29. Bounova G: **Octave networks toolbox.** 2015.
[Reference Source](#)
 30. Gori F, Folino G, Jetten MS, *et al.*: **MTR: taxonomic annotation of short metagenomic reads using clustering at multiple taxonomic ranks.** *Bioinformatics.* 2011; **27**(2): 196–203.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 31. Aamodt EJ: **The Neurobiology of C. elegans.** Elsevier. 2006; **69.**
[Reference Source](#)
 32. Hope IA (Ed.): **C. elegans: a practical approach.** OUP Oxford. 1999; **213.**
[Reference Source](#)
 33. Ruths J, Ruths D: **Control profiles of complex networks.** *Science.* 2014; **343**(6177): 1373–1376.
[PubMed Abstract](#) | [Publisher Full Text](#)
 34. Varga El, Hangos KM, Szigeti F: **Controllability and observability of heat exchanger networks in the time-varying parameter case.** *Control Eng Pract.* 1995; **3**(10): 1409–1419.
[Publisher Full Text](#)
 35. Freeman LC: **A set of measures of centrality based on betweenness.** *Sociometry.* 1977; **40**(1): 35–41.
[Publisher Full Text](#)

Open Peer Review

Current Peer Review Status:  

Version 2

Reviewer Report 03 October 2019

<https://doi.org/10.5256/f1000research.22645.r54089>

© 2019 Aguirre L et al. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



Arthur Montanari 

Graduate Program in Electrical Engineering, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

Ercílio Moreira 

Graduate Program in Electrical Engineering, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

Luis A. Aguirre 

Graduate Program in Electrical Engineering, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

The revised version is clearly more focused and directed to researchers in life science, although its application is much wider. It has been shortened and some figures removed. Discussions are more objective and some technicalities corrected.

Some more general discussions that have been suggested in the first review were simply added as assertions. This helps keep the paper compact, which is understandable.

As a second notice, though, we still think that the third module could be discussed in more depth. The toolbox provides five (“four plus one”) methods to analyze the network robustness (from a structural controllability/observability point of view) and to place driver/sensor in such a way that the relative degree of the network system does not exceed a specified threshold. Although the relevance of the third module is now more well-justified, it is not yet clear what is the difference between these five methods. What are the pros and cons of each method? How can this be perceived in the studied neural network of *C. elegans*?

We found no compatibility issues with Octave in this version.

Minor problems:

1. In the definition of the observability matrix there is a “^” missing. The last T within the square brackets should be a transpose.

2. "Since the generated measures can be presented by large tables, they are attached in Excel format to the toolbox [9]." Ref. [9] is a link to the old toolbox (in Zenodo website). It would be nice to link the site with current version (GitHub).

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Control theory, control of networked, nonlinear dynamics.

We confirm that we have read this submission and believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 30 September 2019

<https://doi.org/10.5256/f1000research.22645.r54088>

© 2019 Didier G. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



Gilles Didier 

Institut Montpellierain Alexander Grothendieck (IMAG), CNRS, University of Montpellier, Montpellier, France

Thanks for your revised manuscript which addressed most of my comments. In particular, the structure of the paper was improved and the notions used in your toolbox are now introduced. I also appreciated the new example which provides a better illustration of the possibilities of your package. Though I am not a native speaker myself, I feel that some issues remained in your text. Please find below suggestions to improve the writing.

Abstract:

I suggest the following alternative:

"The network science-based determination of driver nodes and sensor placement has become increasingly popular in the field of dynamical systems over the last decade. We developed an Octave and MATLAB-compatible NOCAD toolbox which implements various methods to compute relevant structural, controllability and observability measures associated to linear or linearised systems and to compare different sensor placement methods.

We illustrated the use of our toolbox in life sciences by applying it to the analyse the neural network of *Caenorhabditis elegans*."

Introduction:

An alternative first paragraph:

"In the life sciences, the determination of nodes that play a significant role in networks (e.g., in the emergence or the treatment of diseases) is an intensively researched field¹. In particular, determining the proper driver nodes, i.e. the ones that ensure physically feasible controllability with the minimum cardinality and energy requirement, in biological networks, or more generally in dynamical systems, is an important question. We refer to 2 for a detailed study of control principles in biological networks. A recent review paper presents presents the application of the network science-based determination of driver nodes to the analysis of various biological networks (protein-protein interaction -PPI- networks,

Caenorhabditis elegans neuronal network, neurochemical rat brain network, Saccharomyces cerevisiae cell cycle networks, Epithelial Mesenchymal Transition -EMT- network, myeloid differentiation regulatory network and Th differentiation network) and to the identification of drug targets³."

I am not sure to well understand your sentence: "This method is capable of determining the leading phenotype transitions in biological networks that can be identified as drug targets 5"

Do you mean something like: "This method is capable of determining genes controlling phenotype transitions in biological networks, thus may help to identify drug target candidates⁵" ?

Please reformulate:

"In large-scale human liver metabolic networks (HLMN), the driver metabolites have essential functions, moreover, the role of transport reactions and extracellular metabolites in terms of controlling HLMN have revealed the importance of the environment of human liver metabolism with regard to the health of the liver⁶. Using statistical analysis, a subset of critical control nonprotein-coding RNAs (ncRNAs) enriched by human disease can also be determined⁷. In intra-cellular networks, to understand the information flow, a natural control system was utilised and the robustness of such a control was analysed⁸."

The paragraph is a bit confusing and the relation with your work is not obvious. I guess that the examples presented was analysed with the same approaches as those implemented in your toolbox but this should be stated somewhere.

Find below, an alternative suggestion for the last paragraph. I am not sure that it is the right place for the second sentence.

"The aim of this paper is to introduce the novel toolbox, NOCAD⁹, and to illustrate its applicability in the life sciences through the analyse of the local network of 131 frontal neurons of Caenorhabditis elegans¹⁰. The NOCAD toolbox is also suitable for the comprehensive analysis of any linear or linearised dynamical systems through their static network representation^{11–13}.

We emphasise that in our context, the expression "dynamical network" does not implies that the nodes or connections change with time but refers to networks arising from dynamical systems.

The current version of NOCAD deals with only the linear case. Nonlinear system-related methods will be implemented later. In the nonlinear case, the methodology needs further clarification because for small nonlinear examples the results can be incorrect¹⁴ and the cardinality of the assigned sensors underestimated¹⁵.

Theoretical background and formal definitions of the representation of linear systems and of their structural controllability and observability are provided below."

Other comments:

- page 4 col 1 first paragraph "provide the opportunity" -> "allow".
- page 4 col 1 second paragraph "stood" -> "fixed".
- page 4 col 2 first paragraph "if the rank of the controllability matrix is equal to the number of state variables, $\text{rank}(C) = N$ " -> "if the rank of the controllability matrix C is equal to the number of state variables, i.e., $\text{rank}(C) = N$ " (same suggestion for the sentence about the observability matrix).
- page 4 col 2 "For a better understanding, we illustrate the aforementioned definitions by a small example in Figure 1 that contains the command interneurons AVAL, AVAR, AVBL, AVBR, AVDL and AVDR from the frontal neural network of neurons and synapses in *C. elegans*." -> "Figure 1 illustrates the aforementioned definitions with a small example which contains the command interneurons AVAL, AVAR, AVBL, AVBR, AVDL and AVDR of the *C. elegans*' frontal neural

network."

- page 5 col 1 end of paragraph 1: "can be seen" -> "is displayed".
- page 5 col 2 par 2: remove the comma after "in such a way".
- page 6 col 1 last par.: your last sentence is too long. ", moreover," -> ". It also".
- page 6 col 2 last par.: "can be presented by" -> "are stored in".
- page 6-7: "The higher value indicates the more central position of the node, and now RIAL is the most central element." -> "The closeness of a node measures its centrality. In our dataset, RIAL is the most central element under this criteria."
- page 7 col 1 par 1: "if Markov-chains are modelled" -> "under Markov models".
- page 7 col 1 par 1: "the proportion of the number of edges of neighbours' and the number of neighbours" -> "the proportion of the number of edges of neighbours to the number of neighbours".
- page 7 col 1 par 1: "when determining" -> "to determine".
- page 7 col 1 par 2: "source nodes when the node has no incoming edges, thus," -> "source nodes are nodes with no incoming edges, for which".
- page 7 col 1 par 2: "dilation, when" -> "dilation occurs when".
- page 7 col 1 par 2: "between internal dilation and external dilation, in the former the child node is not a leaf, i.e. it has children, while in the latter the child is a leaf node, i.e. it has no children." -> "between internal dilation nodes and external dilation nodes according to whether their child node is an internal node, i.e., has children, or a leaf, i.e., has no children, respectively."
- page 7 col 2 last par.: "Although numerous papers used the network-based determination of driver and sensor nodes, a flexible toolbox that may be used to support the analysis has yet to be designed. To fill this gap, in this article the Octave- and MATLAB-compatible NOCAD toolbox 9 was proposed to support the network-based controllability and observability analysis of dynamical systems, and through the analysis of the neural network of C.elegans, the applicability of the toolbox in the life sciences was presented." -> "Although numerous papers have utilised the network-based determination of driver and sensor nodes, a flexible toolbox implementing this approach has yet to be designed. To fill this gap, we propose the Octave- and MATLAB-compatible NOCAD toolbox 9 which performs the network-based controllability and observability analysis of dynamical systems. The applicability of the toolbox in the life sciences was demonstrated through the analysis of the neural network of C.elegans."

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: My main area of research is applied mathematics. I worked on biological networks (regulatory and interaction networks).

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Version 1

Reviewer Report 25 June 2019

<https://doi.org/10.5256/f1000research.20856.r48912>

© 2019 Aguirre L et al. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Arthur Montanari**

Graduate Program in Electrical Engineering, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

Ercílio Moreira

Graduate Program in Electrical Engineering, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

Luis A. Aguirre

Graduate Program in Electrical Engineering, Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil

In this paper the authors describe the NOCAD toolbox developed for the analysis of some aspects of networks as, for instance, structural controllability and observability. The topic of the paper is not only very interesting but also timely as issues relating to controllability and observability of networks could be fundamental in a number of practical situations. Hence to have a nice set of tools to analyze dynamical networks is welcome.

The toolbox described is useful in a specific range of problems. Here we focus on the controllability and observability properties of networks, as prompted by the title. The tools presented focus on structural controllability and observability of linear systems. The assumption of linearity is not mentioned in the abstract. As a matter of fact, the opening phrase of the last paragraph in the introduction should be copied to the abstract.

The nonlinear case and other definitions of controllability and observability, such as, dynamical and symbolical are not mentioned in the paper nor are handled by the toolbox. This is an important remark as it is now known that some algorithms underestimate the cardinality of the set of sensor nodes when applied to nonlinear systems.

The paper

In the second paragraph of the Introduction, the authors mention the importance of “determining the proper driver nodes”. We wonder if the average reader would know what that is. In clarifying this point the authors would like to address first what is “a” proper set of driving nodes (e.g. one that will guarantee full

controllability). However, in order to specify “the” proper set, possibly some more detailed measure of controllability should be employed. For instance, suppose two sets of driving nodes S_1 and S_2 , both with the same cardinality, such that the network is fully controllable either from S_1 or from S_2 . Hence S_1 is “a” proper set of driving nodes (assuming that by proper the authors are referring to full controllability) and S_2 is another one. Now, it could well be that using S_1 less energy is required to drive the network from one state $x(t_i)$ to another $x(t_f)$ when compared to S_2 . In that case, S_1 and S_2 are not totally equivalent. Structural controllability and observability are unable on their own to provide this distinction.

Following the same vein, we wonder if the average reader of F1000Research would know the distinction of static and dynamic networks. A word about this would be profitable, especially because the toolbox refers mainly to the second class.

In some parts of the paper the authors refer to matrix A as the “state transition” matrix. In continuous-time this is incorrect. In discrete-time this is only correct for a transition time of one sampling period. Matrix A is called the “dynamical matrix”. State transition matrix is something else.

Relevance to the journal

The MS introduction is written in such a way that it instigates interest from the target audience of this journal (mathematical biology). However, the techniques implemented in the toolboxes are usually found around the network science community, which includes several other fields that range from power systems to social networks. To show more coherence to this journal, the authors should provide some example of application on a “real-world network” under a biological context. This is also interesting to show how the results provided by the toolbox can be useful to draw conclusions under a practical context other than a “toy problem” as presented.

Background

The MS content is not sufficient to understand the techniques implemented in the toolbox. We know that the paper does not aim at providing extensive background, but some could be helpful. A reader coming from a control theory background might expect the toolbox to provide results based on Kalman’s definition of controllability and observability. However, the toolbox is based on Liu and coworker’s maximum matching algorithm which is based, in turn, on Lin’s structural definition (1974) (which is not even mentioned in the MS). Which definition of controllability and observability the toolbox is based on should be crystal clear in the main text and abstract. Moreover, the notions of structural controllability and observability should be presented to the reader and how they interplay with the more well-known notion in Kalman’s sense (e.g. Lin’s definition is only a necessary condition for Kalman’s definition). Although we think that the definitions of controllability and observability should be mentioned on the main text, the authors provide some background on the implemented maximum matching algorithms in the toolbox. This, however, should be mentioned explicitly in the MS.

Examples

The paper furnishes examples to illustrate some of the features of the new toolbox. The network topology is the same in each example, which facilitates understanding and comparison. In some cases, more discussion would be welcome. For instance, in the last example (Figure 7) not all the indices are clearly defined to the user, for instance the critical nodes are given as x_2 , x_4 and x_7 . From the context it seems that if any of these nodes is lost, the network would become uncontrollable, but this is not directly stated.

In referring to Figure 5 the authors speak in terms of Reachability matrices: R_c and R_o . It is clear from Figure 5 that R_c shows which nodes have a path to node i (the i th column). As for R_o , it seems that we should look row-wise instead of column-wise, is that right? Anyhow, we do not think it is adequate to say that R_c shows which nodes can be controlled from another one. The word “control” is perhaps too general. We would suggest just to say that R_c shows which nodes can be reached from another one.

Toolbox

The toolbox is subdivided in three modules. The first one, “network mapping module”, implements the maximum matching algorithm (and related modifications) to return a structurally controllable (observable) network with the smallest set of driver (sensor) nodes. This is a welcome feature, especially for a MATLAB environment, which we are unaware of any alternative.

The second module, “system characterization module”, provides several graph measures that are available on other MATLAB-based toolboxes, but are indeed useful to assess the network controllability and observability properties of a system. Thus, its presence is justifiable.

The third module, “improvements and robustness module”, is a set of functions which specifically implement previous results of the authors (e.g. Refs. 20¹, 21²). It seems quite specific, but nevertheless the toolbox relevance is justifiable in great part for its module one.

The toolbox seems fast and no bugs were found in its implementations as far as the MATLAB environment is concerned. See further comments on some compatibility issues when using Octave.

The target audience of the toolbox

The toolbox is applicable to any kind of network (graph), be it directed or undirected, weighted or unweighted, and so on. However, although general, the authors should discuss in the MS what are the kinds of networks where a structural analysis of controllability and observability are more useful. For instance, a linearization of a power system model modelled by interconnected Kuramoto oscillators yield a dynamical matrix “ A ” whose corresponding adjacency graph is not only highly connected but also undirected. Consequently, the toolbox points out that only one driver node is needed to structurally control the network, independently of its size “ n ” (which is true in this case according to Lin’s definition). However, this does not give insight to the problem since basically any node can be chosen as a driver node, and only one node being sufficient seems quite unrealistic. The question is: For what kind of networks is the structural approach more interesting (and hence the toolbox)? It could be that the toolbox could be extra-helpful in the context of more sparse and directed networks (with higher hierarchy).

Some nitpicking:

Sometimes it is not clear whether the authors refer to the dynamical matrix “ A ” or the adjacency matrix “ A^T ”. We recommend that different nomenclatures be used, such as “ A ” for the dynamical matrix and “ A_{Adj} ” for the adjacency matrix. This should be changed in the MS and manual.

Regarding the installation of the NOCAD toolbox. We noticed that the NOCAD already comes with a “octave-network-toolbox” folder which does not have all the necessary functions to use Module 2. Thus, we had to download the “octave-network-toolbox-master” folder in Ref. 22 to have access to all needed functions. Is this necessary or is the NOCAD toolbox really missing some functions?

Some statements in Section “Use Cases”, Paragraph 5, are redundant with the information already present in Fig. 4. Instead of repeating the same information, it might be more interesting to make some comments on how useful some of these network measures can be to design better and more robust networks from a control and observation point-of-view.

There are some issues in the Octave version of this toolbox. For instance, function “heatmaps” has some bugs in Octave but works well in MATLAB. This happens because the function “colormap” from Octave accepts the argument “hot” but not “Hot”. The authors should do a careful review of the Octave toolbox and check for further compatibility issues.

References

1. Leitold D, Vathy-Fogarassy A, Abonyi J: Evaluation of the Complexity, Controllability and Observability of Heat Exchanger Networks Based on Structural Analysis of Network Representations. *Energies*. 2019; **12** (3). [Publisher Full Text](#)
2. Leitold D, Vathy-Fogarassy A, Abonyi J: Network Distance-Based Simulated Annealing and Fuzzy Clustering for Sensor Placement Ensuring Observability and Minimal Relative Degree. *Sensors (Basel)*. 2018; **18** (9). [PubMed Abstract](#) | [Publisher Full Text](#)

Is the rationale for developing the new software tool clearly explained?

Partly

Is the description of the software tool technically sound?

Partly

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Partly

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Control theory, control of networked, nonlinear dynamics.

We confirm that we have read this submission and believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however we have significant reservations, as outlined above.

Reader Comment 10 Sep 2019

Janos Abonyi, University of Pannonia, Veszprém, Hungary

Dear Reviewers,

We are grateful for your useful remarks.

In the following, we provide a detailed report about how we improved the paper based on your valuable comments and suggestions.

We hope that the modifications have significantly improved the understandability of the paper.

Sincerely yours,
Janos Abonyi

In this paper the authors describe the NOCAD toolbox developed for the analysis of some aspects of networks as, for instance, structural controllability and observability. The topic of the paper is not only very interesting but also timely as issues relating to controllability and observability of networks could be fundamental in a number of practical situations. Hence to have a nice set of tools to analyze dynamical networks is welcome.

The toolbox described is useful in a specific range of problems. Here we focus on the controllability and observability properties of networks, as prompted by the title. The tools presented focus on structural controllability and observability of linear systems. The assumption of linearity is not mentioned in the abstract. As a matter of fact, the opening phrase of the last paragraph in the introduction should be copied to the abstract.

The nonlinear case and other definitions of controllability and observability, such as, dynamical and symbolical are not mentioned in the paper nor are handled by the toolbox. This is an important remark as it is now known that some algorithms underestimate the cardinality of the set of sensor nodes when applied to nonlinear systems.

- Thank you for this invaluable remark, we have emphasised in both the *Abstract* and *Introduction* that the toolbox is applicable to the analysis of linear and linearized systems rather than nonlinear systems for which incorrect results may be suggested.

The paper

In the second paragraph of the Introduction, the authors mention the importance of “determining the proper driver nodes”. We wonder if the average reader would know what that is. In clarifying this point the authors would like to address first what is “a” proper set of driving nodes (e.g. one that will guarantee full controllability). However, in order to specify “the” proper set, possibly some more detailed measure of controllability should be employed. For instance, suppose two sets of driving nodes S_1 and S_2 , both with the same cardinality, such that the network is fully controllable either from S_1 or from S_2 . Hence S_1 is “a” proper set of driving nodes (assuming that by proper the authors are referring to full controllability) and S_2 is another one. Now, it could well be that using S_1 less energy is required to drive the network from one state $x(t_i)$ to another $x(t_f)$ when compared to S_2 . In that case, S_1 and S_2 are not totally equivalent. Structural controllability and observability are unable on their own to provide this distinction.

- Thank you for your valuable remark, we have expanded the relevant part of the paper and emphasized that during optimisation the cardinality and energy demand should be minimised, while the configurations should provide structural controllability and observability.

Following the same vein, we wonder if the average reader of F1000Research would know the distinction of static and dynamic networks. A word about this would be profitable, especially because the toolbox refers mainly to the second class.

- We are grateful for this suggestion, in the *Introduction* we have clarified the nomenclature as well as system and network classes used in the manuscript.

In some parts of the paper the authors refer to matrix A as the “state transition” matrix. In continuous-time this is incorrect. In discrete-time this is only correct for a transition time of one sampling period. Matrix A is called the “dynamical matrix”. State transition matrix is something else.

- Thank you for this invaluable remark, we have corrected the incorrect nomenclature.

Relevance to the journal

The MS introduction is written in such a way that it instigates interest from the target audience of this journal (mathematical biology). However, the techniques implemented in the toolboxes are usually found around the network science community, which includes several other fields that range from power systems to social networks. To show more coherence to this journal, the authors should provide some example of application on a “real-world network” under a biological context. This is also interesting to show how the results provided by the toolbox can be useful to draw conclusions under a practical context other than a “toy problem” as presented.

- We are grateful for this useful suggestion. We have improved the *Introduction* and, with the aid of references, introduced the networks that are already utilised in network-based structural controllability and observability analysis. In addition, the example problem has been replaced by the frontal neural network of *C. elegans*, a well-known standard biological dataset.

Background

The MS content is not sufficient to understand the techniques implemented in the toolbox. We know that the paper does not aim at providing extensive background, but some could be helpful. A reader coming from a control theory background might expect the toolbox to provide results based on Kalman’s definition of controllability and observability. However, the toolbox is based on Liu and coworker’s maximum matching algorithm which is based, in turn, on Lin’s structural definition (1974) (which is not even mentioned in the MS). Which definition of controllability and observability the toolbox is based on should be crystal clear in the main text and abstract. Moreover, the notions of structural controllability and observability should be presented to the reader and how they interplay with the more well-known notion in Kalman’s sense (e.g. Lin’s definition is only a necessary condition for Kalman’s definition).

Although we think that the definitions of controllability and observability should be mentioned on the main text, the authors provide some background on the implemented maximum matching algorithms in the toolbox. This, however, should be mentioned explicitly in the MS.

- We are grateful for your useful remarks and have improved the section entitled *Methods*, moreover, extended it with the theoretical background to dynamical systems, definitions of structural controllability as well as observability, and the applied methods. The proposed works from the literature have been cited in the relevant sections and for this suggestion we would like to express our sincere gratitude.

Examples

The paper furnishes examples to illustrate some of the features of the new toolbox. The network topology is the same in each example, which facilitates understanding and comparison. In some cases, more discussion would be welcome. For instance, in the last example (Figure 7) not all the indices are clearly defined to the user, for instance the critical nodes are given as x2, x4 and x7. From the context it seems that if any of these nodes is lost, the network would become uncontrollable, but this is not directly stated.

In referring to Figure 5 the authors speak in terms of Reachability matrices: R_c and R_o . It is clear from Figure 5 that R_c shows which nodes have a path to node i (the i th column). As for R_o , it seems that we should look row-wise instead of column-wise, is that right? Anyhow, we do not think it is adequate to say that R_c shows which nodes can be controlled from another one. The word "control" is perhaps too general. We would suggest just to say that R_c shows which nodes can be reached from another one.

- We are grateful for your valuable remark and have expressed the results provided by the toolbox as well as inserted a more thorough introduction to the results. According to the reachability matrices, control and reach derived from several dynamical systems overlap to some extent when the problem with regard to controllability was reduced to one concerning reachability.

Toolbox

The toolbox is subdivided in three modules. The first one, "network mapping module", implements the maximum matching algorithm (and related modifications) to return a structurally controllable (observable) network with the smallest set of driver (sensor) nodes. This is a welcome feature, especially for a MATLAB environment, which we are unaware of any alternative.

The second module, "system characterization module", provides several graph measures that are available on other MATLAB-based toolboxes, but are indeed useful to assess the network controllability and observability properties of a system. Thus, its presence is justifiable.

The third module, "improvements and robustness module", is a set of functions which specifically implement previous results of the authors (e.g. Refs. 201, 212). It seems quite specific, but nevertheless the toolbox relevance is justifiable in great part for its module one.

The toolbox seems fast and no bugs were found in its implementations as far as the MATLAB environment is concerned. See further comments on some compatibility issues when using Octave.

- We are grateful for your useful remarks and have highlighted in the section entitled *Implementation* the importance of the third module for dynamical systems in general and biological systems specifically.

The target audience of the toolbox

The toolbox is applicable to any kind of network (graph), be it directed or undirected, weighted or

unweighted, and so on. However, although general, the authors should discuss in the MS what are the kinds of networks where a structural analysis of controllability and observability are more useful. For instance, a linearization of a power system model modelled by interconnected Kuramoto oscillators yield a dynamical matrix “A” whose corresponding adjacency graph is not only highly connected but also undirected. Consequently, the toolbox points out that only one driver node is needed to structurally control the network, independently of its size “n”(which is true in this case according to Lin’s definition). However, this does not give insight to the problem since basically any node can be chosen as a driver node, and only one node being sufficient seems quite unrealistic. The question is: For what kind of networks is the structural approach more interesting (and hence the toolbox)? It could be that the toolbox could be extra-helpful in the context of more sparse and directed networks (with higher hierarchy).

- Thank you for your valuable remark. We believe no “better” types of networks for structural analysis are known since the importance of a node in a complex system from a structural point of view can be excessive or critical with regard to reliability, independent of the directedness in the network. The hardest aspect of these analyses is the evaluation of the results that should be provided by experts, moreover, this may cause undirected or directed and sparser or denser networks to be preferred.

Some nitpicking:

Sometimes it is not clear whether the authors refer to the dynamical matrix “A” or the adjacency matrix “A^T”. We recommend that different nomenclatures be used, such as “A” for the dynamical matrix and “A_{Adj}” for the adjacency matrix. This should be changed in the MS and manual.

- Thank you for this invaluable remark, we have used the nomenclature “A^T” to emphasise the transposition, i.e. the connection between the dynamical and adjacency matrices.

Regarding the installation of the NOCAD toolbox. We noticed that the NOCAD already comes with a “octave-network-toolbox” folder which does not have all the necessary functions to use Module 2. Thus, we had to download the “octave-network-toolbox-master” folder in Ref. 22 to have access to all needed functions. Is this necessary or is the NOCAD toolbox really missing some functions?

- We are grateful for the thorough testing of the toolbox. Unfortunately, some function from the octave-networks-toolbox was excluded during its upload, but this has been corrected.

Some statements in Section “Use Cases”, Paragraph 5, are redundant with the information already present in Fig. 4. Instead of repeating the same information, it might be more interesting to make some comments on how useful some of these network measures can be to design better and more robust networks from a control and observation point-of-view.

- Thank you for your remark, we have replaced the example network, therefore, the diagrams have been removed and the relevant parts improved. We hope that this problem has been resolved.

There are some issues in the Octave version of this toolbox. For instance, function “heatmaps” has some bugs in Octave but works well in MATLAB. This happens because the function “colormap” from Octave accepts the argument “hot” but not “Hot”. The authors should do a careful review of the Octave toolbox and check for further compatibility issues.

- Thank you for your remark. Interestingly, our system did not alert us to this problem. We have corrected the source code as suggested since both parameters were applied to our configurations, however, we kindly ask you to check your version of Octave as the problem can also be caused by the use of an older version of this program.

Competing Interests: No competing interests were disclosed.

Reviewer Report 28 May 2019

<https://doi.org/10.5256/f1000research.20856.r48269>

© 2019 Didier G. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



Gilles Didier 

Institut Montpelliérain Alexander Grothendieck (IMAG), CNRS, University of Montpellier, Montpellier, France

This software tool article presents the Octave/MATLAB toolbox "NOCAD" for networks analysis. NOCAD implements various standard measures on networks and their nodes (density, betweenness, centrality...). Its main novelty stands in the "dynamical-systems-based" tools to analyse (static) networks that it provides. This article can be seen as a companion paper of the reference 19¹ from the same authors (It could be stated somewhere in the text).

The ideas implemented in the toolbox are interesting, but I feel that their presentation has to be improved. First, I suggest you get your text proof-read by a native speaker. Second, many concepts used in the manuscript should be at least briefly recalled. Third, the article lacks a detailed description of the input (in the general sense, including user interaction) of the toolbox. The basic input seems to be the incidence matrix of a directed network but, if I well understood, the toolbox allows to set input and output nodes and has network design capabilities.

Another major concern is that though one of the aims of the article is to draw attention of life science researchers, it does not present clearly how the toolbox can be used on biological networks and on which kind of biological networks it can be applied. Real biological examples should be actually more than welcome in the "Use cases" section, which contains only toy examples. Without such examples, I am afraid that the article fails to draw attention of its expected audience. In particular, I am wondering in what extent the toolbox can be used on standard biological datasets, e.g., PPI networks can have thousands vertices.

Please find below some detailed comments:

Abstract

The abstract has to be extended in order to give a better idea of the content of the article.

Introduction

The introduction does not reflect well what follows and the topic of the paper. You should start by briefly introducing the general ideas of dynamical analysis of networks, precise in what sense you use the term

"driver node", which is sometimes used in a biological context etc. First mention of dynamical system comes at the end of the second paragraph and could be developed (e.g. by briefly recalling ideas of [7]²) while biological examples may not need to be as much detailed.

The description of previous software sounds good. It may take place a dedicated section.

Method

The method section should recall the notions of controllability and observability and the general principles of "dynamical analysis" (it can be brief, something like the two first subsections of the section "Results" of [19]¹). In the current version, I don't think that this part can be understood without reading your reference [19]¹.

You should state explicitly which are the inputs of your toolbox, the accepted formats etc.

I found the paragraph about the modularity of the toolbox is not very useful. It could be shortened or even discarded.

Subsection "Operation" ("Installation"?) is OK.

Use cases

If I well understood, the input is the incidence matrix A from which are computed the matrices B, C and D as illustrated with the first example. It should be interesting to apply the approach on a biological network, for instance a protein-protein interaction network or a regulatory network, in order to illustrate the interest of the results obtained in this context.

In the second example (Figure 3), input and output nodes seem to be set arbitrarily in order to get a system complex-enough to show the function of the second module of the toolbox (incidence matrix is the same as that of the first example). Is it possible to illustrate the toolbox with an example in which the input and outputs are not fixed? I am actually concerned with the fact that the use cases do not really show how the toolbox can be used on a real dataset, which may not have natural input and outputs.

Figures 4, 5 and 6 displays the results of the toolbox on the second example. These results are briefly discussed in the text which also recalls the definitions of some of the measures. Applying the toolbox on a real dataset would make the discussion of the results of the toolbox much easier to interpret. Moreover, a real dataset could provide evidence of the relevance of the results obtained by relying on the nature of nodes distinguished by the toolbox.

The different types of nodes displayed in Figure 5 should be defined rather than just refer to [24]³.

Could you please give an idea about what kind of biological networks can be studied by the means of your third module? This part is not very intuitive from a biological point of view.

References

1. Leitold D, Vathy-Fogarassy Á, Abonyi J: Controllability and observability in complex networks – the effect of connection types. *Scientific Reports*. 2017; **7** (1). [Publisher Full Text](#)
2. Li M, Gao H, Wang J, Wu FX: Control principles for complex biological networks. *Brief Bioinform*. 2018. [PubMed Abstract](#) | [Publisher Full Text](#)
3. Ruths J, Ruths D: Control profiles of complex networks. *Science*. 2014; **343** (6177): 1373-6 [PubMed Abstract](#) | [Publisher Full Text](#)

Is the rationale for developing the new software tool clearly explained?

Partly

Is the description of the software tool technically sound?

Partly

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Partly

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

No

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: My main area of research is applied mathematics. I worked on biological networks (regulatory and interaction networks).

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Reader Comment 10 Sep 2019

Janos Abonyi, University of Pannonia, Veszprém, Hungary

Dear Reviewer,

We are grateful for your useful remarks.

In the following, we provide a detailed report about how we improved the paper based on your valuable comments and suggestions.

We hope that the modifications have significantly improved the understandability of the paper.

Sincerely yours,

Janos Abonyi

This software tool article presents the Octave/MATLAB toolbox "NOCAD" for networks analysis. NOCAD implements various standard measures on networks and their nodes (density, betweenness, centrality...). Its main novelty stands in the "dynamical-systems-based" tools to analyse (static) networks that it provides. This article can be seen as a companion paper of the reference 191 from the same authors (It could be stated somewhere in the text).

- Thank you for your valuable remark. We have cited the works related to the functions of the toolbox. The sections entitled *Introduction* and *Methods* have been rewritten and improved in order to highlight the applicability of the toolbox in the life sciences and provide additional contribution.

The ideas implemented in the toolbox are interesting, but I feel that their presentation has to be improved. First, I suggest you get your text proof-read by a native speaker. Second, many concepts used in the manuscript should be at least briefly recalled. Third, the article lacks a detailed description of the input (in the general sense, including user interaction) of the toolbox.

The basic input seems to be the incidence matrix of a directed network but, if I well understood, the toolbox allows to set input and output nodes and has network design capabilities.

- Thank you for this remark. We used Grammarly and asked a professional proofreader to correct the second version of the manuscript. We have tried our best to minimize the number of grammatical errors.
- At the beginning of the section entitled *Methods*, the representation of linear systems, definitions of structural controllability as well as observability, and introductions to the methodology in addition to maximum matching were included, therefore, a review of the cited literature was deemed unnecessary. The inputs of the modules were clarified in the section entitled *Implementation*.

Another major concern is that though one of the aims of the article is to draw attention of life science researchers, it does not present clearly how the toolbox can be used on biological networks and on which kind of biological networks it can be applied. Real biological examples should be actually more than welcome in the "Use cases" section, which contains only toy examples. Without such examples, I am afraid that the article fails to draw attention of its expected audience. In particular, I am wondering in what extent the toolbox can be used on standard biological datasets, e.g., PPI networks can have thousands vertices.

- We are grateful for your valuable remark and have cited the latest articles to highlight the importance of the methodology in the field of the life sciences. In the section entitled *Use cases*, the example network was replaced by a frontal neural network of *C. elegans* from the standard biological dataset that has been comprehensively analysed and is sufficiently well-known to be regarded as an interesting example for scientists from the fields of the life as well as network sciences.

Please find below some detailed comments:

Abstract

The abstract has to be extended in order to give a better idea of the content of the article.

- Thank you for this remark. We have improved the abstract based on the reviews and extended it with the information provided to develop the introduction with regard to the content and the main goal of the paper, namely the applicability of the methodology to the life sciences.

Introduction

The introduction does not reflect well what follows and the topic of the paper. You should start by briefly introducing the general ideas of dynamical analysis of networks, precise in what sense you use the term "driver node", which is sometimes used in a biological context etc. First mention of dynamical system comes at the end of the second paragraph and could be developed (e.g. by briefly recalling ideas of [7]2) while biological examples may not need to be as much detailed.

- We are grateful for your useful remarks. We have overviewed the *Introduction* and rewritten it to draw attention to the importance of the methodology in the life sciences. Furthermore, in the section entitled *Methods*, we introduced the theoretical background to network-based structural controllability and observability analysis.

The description of previous software sounds good. It may take place a dedicated section.

- Thank you for this remark, we decided to dedicate a separate section to the existing software.

Method

The method section should recall the notions of controllability and observability and the general

principles of "dynamical analysis" (it can be brief, something like the two first subsections of the section "Results" of [19]1). In the current version, I don't think that this part can be understood without reading your reference [19]1.

- We are grateful for this useful suggestion. As previously mentioned, we have included a theoretical introduction to dynamical systems and their controllability as well as observability in the section entitled *Methods*.

You should state explicitly which are the inputs of your toolbox, the accepted formats etc.

- We are grateful for your remark. The introduction to dynamical systems emphasises the input of the toolbox, namely the dynamical matrix of the state-space representation. In addition, in the section entitled *Implementation*, each of the inputs to the modules was described separately.

I found the paragraph about the modularity of the toolbox is not very useful. It could be shortened or even discarded.

- We are grateful for your valuable suggestion and have deleted this paragraph.

Subsection "Operation" ("Installation"?) is OK.

Use cases

If I well understood, the input is the incidence matrix A from which are computed the matrices B, C and D as illustrated with the first example. It should be interesting to apply the approach on a biological network, for instance a protein-protein interaction network or a regulatory network, in order to illustrate the interest of the results obtained in this context.

- Thank you for your valuable remark. We have replaced the example network with the frontal neural network of *C. elegans* to draw attention to the easily understandable measures rather than the useful results. Therefore, the results from the analysis of the frontal neural network of *C. elegans* are presented, while in the manual the example network is still available.

In the second example (Figure 3), input and output nodes seem to be set arbitrarily in order to get a system complex-enough to show the function of the second module of the toolbox (incidence matrix is the same as that of the first example). Is it possible to illustrate the toolbox with an example in which the input and outputs are not fixed? I am actually concerned with the fact that the use cases do not really show how the toolbox can be used on a real dataset, which may not have natural input and outputs.

- Since we replaced the example network with the frontal neural network of *C. elegans*, it is not necessary to apply the fixed inputs and outputs, therefore, we hope that this problem has been resolved.

Figures 4, 5 and 6 displays the results of the toolbox on the second example. These results are briefly discussed in the text which also recalls the definitions of some of the measures. Applying the toolbox on a real dataset would make the discussion of the results of the toolbox much easier to interpret. Moreover, a real dataset could provide evidence of the relevance of the results obtained by relying on the nature of nodes distinguished by the toolbox.

- As the example network was changed, we hope this remark has been acted on.

The different types of nodes displayed in Figure 5 should be defined rather than just refer to [24]3.

- Thank you for this invaluable remark. We have expanded this part of the paper and, with references, mentioned the importance of these types in biological networks.

Could you please give an idea about what kind of biological networks can be studied by the means of your third module? This part is not very intuitive from a biological point of view.

- We are grateful for this useful suggestion. We have dedicated a paragraph to the importance of this module in the section entitled *Implementation*. Given the improved theoretical background, we hope that the presence of this module is more suitable.

Is the rationale for developing the new software tool clearly explained?

Partly

- As no widespread and approved tool for the network-based structural controllability and observability analysis of dynamical systems is known, we propose our toolbox for this purpose, and its applicability is introduced with the aid of a real network from a standard biological dataset.

Is the description of the software tool technically sound?

Partly

- We have improved the introduction to the theoretical background, therefore, the paper can be regarded as an independent work that introduces the advantages of the methodology and implemented toolbox.

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Partly

- We have stated the input used in this article and the toolbox, moreover, a well-known example was used in order of maximal reproducibility.

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

No

- We have included a more detailed introduction to the theoretical background in order to provide sufficient information concerning the expected outputs and results.

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

- We hope that the frontal neural network of *C. elegans* and its results represent well the applicability of the toolbox, and believe in the importance of such a novel toolbox, even though numerous publications utilise this methodology.

Competing Interests: No competing interests were disclosed.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research

