Title: enaR: An R package for Ecosystem Network Analysis

Running Title: R ecosystem network analysis package

Word Count: 3162

Authors: Stuart R. Borrett, Matthew K. Lau

Addresses:

- SRB: Department of Biology and Marine Biology, University of North Carolina Wilmington, Wilmington, NC 28403
- MKL: Department of Biological Sciences and the Merriam-Powell Center for Environmental Research, Northern Arizona University, 617 S. Beaver St., Flagstaff, AZ 86011
- Current Address: Harvard Forest, Harvard University, 324 N. Main St., Petersham, MA 01366

Contact Details:

- Email: borretts@uncw.edu
- Phone: 910.962.2411
- Fax: 910.962.4066
**enaR: An R package for Ecosystem Network Analysis**

Stuart R. Borrett\(^a\)\(^,*\) and Matthew K. Lau\(^b\)

\(^a\) Department of Biology and Marine Biology, University of North Carolina Wilmington, Wilmington, NC 28403  
\(^b\) Department of Biological Sciences and the Merriam-Powell Center for Environmental Research, Northern Arizona University, 617 S. Beaver St., Flagstaff, AZ 86011  
Current Address: Harvard Forest, Harvard University, 324 N. Main St., Petersham, MA 01366  
\(^*\) Corresponding author, borretts@uncw.edu

September 25, 2014

---

**Abstract**

- Network analysis is a useful approach for investigating complex and relational datasets in many fields including ecology, molecular, and evolutionary biology.

- Here, we introduce **enaR**, an R package for Ecosystem Network Analysis (ENA). ENA is an analytical tool set rooted in ecosystem ecology with over 30 years of development that examines the structure and dynamics of matter and energy movement between discrete ecological compartments (e.g., a food web).

- In addition to describing the primary functionality of the package, we highlight several features including a library of 100 empirical ecosystem models, the ability to analyze and compare multiple models simultaneously, and connections to other ecological network analysis tools in R.

**KEYWORDS:** network analysis, ecosystem, open-source software, network environment analysis, ascendency, input–output analysis, food web, urban metabolism, Ecopath, NETWRK, WAND
1 Introduction

Network ecology – the study of ecological systems using network models and analyses to characterize their structure, function, and evolution – is a large and rapidly growing area of ecology (Proulx et al., 2005). For example, Ings et al. (2009) discovered that a notable fraction of 2008 publications in 11 select journals were related to food webs (≈2.4%), mutualistic networks (≈0.9%), and host-parasitoid networks (≈0.06%). Likewise, Borrett et al. (2014) found that the percent of ecology and evolutionary biology papers indexed by Web of Science that could be classified as network ecology increased from 1.3% in 1991 to more than 5% in 2012. This rise of network ecology contributes to, mirrors, and builds on the more general growth of network sciences (Barabási, 2012; Borgatti & Foster, 2003; Freeman, 2004; Newman, 2003; Wasserman & Faust, 1994).

Ecosystem Network Analysis (ENA) is a branch of network ecology that has been used to address a range of key ecosystem questions (Borrett et al., 2012; Fath & Patten, 1999; Ulanowicz, 1986). For example, in the food web of Big Cypress National Preserve (Florida, USA) Bondavalli & Ulanowicz (1999) found evidence of an indirect mutualism between the American alligator and some of its prey items. Applications of ENA have also lead to new insights into the classic trophic questions of “What limits food-chain length?” (Ulanowicz et al., 2014) and “Are food webs modular?” (Allesina et al., 2005; Borrett et al., 2007; Krause, 2004). Hines et al. (2012) used ENA to quantify the relative importance of coupling between biogeochemical processes (e.g., nitrification) in the Cape Fear River estuary sedimentary nitrogen cycle. Further, scientists have used ENA to investigate differences in urban sustainability (Bodini & Bondavalli, 2002; Bodini et al., 2012; Chen & Chen, 2012; Zhang et al., 2010). Collectively, this work consistently shows the power of a transactional network to generate unexpected ecological relationships that then influence the system function and evolution (Jørgensen et al., 2007; Patten, 1991; Ulanowicz, 1997).

enaR is an open-source software to facilitate ENA. The currently available ENA software pack-
ages (Allesina & Bondavalli, 2004; Christensen & Walters, 2004; Fath & Borrett, 2006; Kazanci, 2007; Ulanowicz & Kay, 1991) each have critical limitations, which led us to three primary design objectives for enaR. The first objective was to collect the major ENA functions into a single software package. While multiple investigators have contributed to algorithmic development (e.g., Allesina & Bondavalli, 2003; Fath & Patten, 1999; Finn, 1976; Ulanowicz, 1986; Ulanowicz & Kay, 1991), the broad set of tools is not available in a single existing software. The second objective was to increase the availability and extensibility of the software. We chose to use R in part because of its increasing popularity as an analytical tool in the biological sciences (e.g., Dixon, 2003; Metcalf et al., 2012; Revell, 2012). Further, users can freely download a stable version of the package from the CRAN website (http://cran.r-project.org/web/packages/enaR), and the code for every function in R is available from within R (e.g., edit(function_name)). In addition, enaR development is being managed via GitHub (https://github.com/TheSeeLab/enaR) to encourage collaborative development. The third design objective was to enable enaR users access to network analysis tools from other disciplines. To enable this, enaR was designed to work directly with two existing R network analysis packages: network (Butts, 2008a) and sna (Butts, 2008b). In summary, the aim of the enaR package is to make ENA tools more available and easier to use, adapt, and extend.

In this paper, we present an overview of enaR and highlight some of its functionality. A full description of the ENA algorithms and their use and interpretation is beyond the scope of this short paper, but we refer interested readers to a selection of reviews as an entry point to ENA (Fath & Borrett, 2006; Fath & Patten, 1999; Jørgensen et al., 2007; Schramski et al., 2011; Ulanowicz, 1997). For a more comprehensive description on how to use the enaR package, please refer to the package vignette: http://cran.r-project.org/web/packages/enaR/vignettes/enaR-vignette.pdf.
2 Overview of enaR

ENA is an agglomeration of algorithms developed to analyze network models of energy or matter movement in ecosystems (e.g., Fath & Patten, 1999; Hannon, 1973; Ulanowicz, 1986), but it can generally be applied to any Input-Output system that follows a thermodynamically conserved unit among the compartments. Thus, it is a family of related algorithms to analyze the ecosystem from several perspectives including its structure, flow, storage, and utility. Together, these analyses function as a “macroscope” to investigate (1) whole system organization, (2) the direct and indirect effects among system components, and (3) the processes that create and sustain ecological systems.

In this section we provide an overview of the algorithms and tools included in the enaR software. After describing the required model information, we highlight the primary ENA algorithms included in enaR. We then walk through an example application of the enaR Flow analysis.

2.1 Data Requirements and Input

ENA is a data-intensive methodology. The system is modeled as a set of compartments or network nodes that represent species, species-complexes (i.e., trophic guilds or functional groups), or non-living components of the system in which energy or matter is stored. These nodes are connected by a set of direct energy or matter transactions among the nodes, termed directed edges or links. These models also have energy–matter inputs into the system and output losses from the system. In summary, the full set of data required includes: (1) internal flows, (2) boundary inputs, (3) boundary exports, (4) boundary respiration, (5) boundary outputs, which may be the sum of exports and respiration, (6) biomass or storage values, and (7) designation of living status of each node. While all seven elements are required for a full analysis, the specific data requirements varies among the ENA algorithms.

The primary ENA algorithms in enaR assume the model data is presented as an R network data
object defined in the network package. Given the data elements, users can use the pack function
to combine the data elements into the R network data object. While a standard data format for an
ENA model does not yet exist, there are two commonly used formats. First, there is the Scientific
Committee for Ocean Research (SCOR) format that is the required input to NETWRK (Ulanowicz
& Kay, 1991), and the second format is the Excel sheet formatted data that is the input to WAND
(Allesina & Bondavalli, 2004). The enaR package includes a read.scor and a read.wand function
to read in these common data formats (Table 1).

2.2 Visualization

Visualization of network models can be an essential analytical tool (Lima, 2011; Moody et al.,
2005). Because enaR is built specifically to use the network package and data type, it is possible to
quickly create network plots of the model internal structure. Fig. 1a shows an example visualization
of Dame & Patten’s (1981) Oyster Reef ecosystem model. The network package includes three
network layout algorithms: circle, Fruchterman-Reingold, and Kamada-Kawai. The Fruchterman-
Reingold algorithm used here is the default. The R script to generate this visualization is included
in the online supplementary information (Item S1).

2.3 Algorithm Overview

enaR includes many of the most commonly used ENA algorithms (Table 2), along with a number
of work flow tools and specialty analyses (Tables 1 and 3). The nine primary ENA functions begin
with the prefix ‘ena’ followed by the specific analysis name (see Table 2). There are a total of 34
functions in the enaR package. Comparison of the enaR package to previous implementations of
ENA algorithms (i.e., NETWRK, NEA.m, EcoNet) shows high agreement in function output and
significant expansion of the available ENA algorithms (Table S1 online).
Scharler & Fath (2009) identify two schools of ENA. The first school is based on the work of Robert Ulanowicz and colleagues at the University of Maryland (Ulanowicz, 1986, 1997, 2009). Primarily focused on trophic ecology, this approach uses information theory and the ascendency concept to characterize ecosystem growth and development (Ulanowicz, 1986, 1997). This work is often referred to as “Ecological Network Analysis” as it predates many other types of network ecology. The second school is based on the work of Bernard Patten at the University of Georgia (Fath & Patten, 1999; Matis & Patten, 1981; Patten, 1982; Patten et al., 1976). Steeped in dynamic equations, simulations, and systems analysis, this approach developed around the environ concept that formalizes the concept of environment (Patten, 1978), and has often been referred to as “Network Environ Analysis.” enaR currently captures all of the Patten School algorithms previously implemented in NEA.m (Fath & Borrett, 2006). Presently, the Ulanowicz School algorithms are more limited, including the ascendency calculations (Ulanowicz, 1997) and mixed trophic impacts analyses (Ulanowicz & Puccia, 1990); however, we expect the package capabilities to continue to grow, especially with the assistance of new users. This combination of the Patten and Ulanowicz schools of analyses is rare in existing software.

2.4 Example Application

Given a network model, applying ENA algorithms with enaR is straightforward. We demonstrate how to use the package with an example Flow analysis on Dame & Patten’s (1981) model of energy flow in an Oyster Reef ecosystem. Figure 2 shows the example script. The analysis involves: (1) loading the model data, (2) checking and balancing the model if necessary, and (3) inputting the balanced model into the analysis function. The final step is interpreting the analytical output. This is a typical workflow for ENA.

After loading the enaR package, the next step is to enter the model data. Here, we extract the
model information from the paper and create a vector of node names, the flow matrix (F), inputs
(z), outputs (y), and the logical vector indicating whether or not the nodes are living (Fig. 2). We
then use the \texttt{pack} function to create the required network data object. The next step is to apply
the \texttt{ssCheck} function ensure that the model is at steady-state, which is one of the assumptions of
the flow analysis (Fath & Borrett, 2006; Finn, 1976). If the model had not been at steady-state,
we could have then applied one of four automated balancing algorithms (AVG, Input-Output,
Output-Input, AVG2; Allesina & Bondavalli, 2003) to force the model into a steady-state. We then
apply the \texttt{enaFlow} function to the model to perform the desired ENA flow analysis. As shown
with the \texttt{attributes} function, this analysis returns 4 matrices (G, GP, N, NP) and two vectors
(throughflow, T, and a vector of 20 whole-network statistics, ns).

Interpreting the ENA results is the final challenge. Here, we provide a few illustrative interpre-
tations of the Flow analysis. Starting with the whole-network flow statistics, we see that the total
system throughflow (TST) of the oyster reef model is 83.6 Kcal m$^{-2}$ d$^{-1}$. TST is a measure of
the total activity of the system, which is often referred to as the size or power of the system. The
Finn Cycling Index (FCI) indicates that 11% of this activity was generated by recycling. Further,
the average path length (APL = 2.02) shows that an average input passes over two paths before
exiting the system, and the ratio of indirect to direct flows (ID.F = 1.58) indicates that the indirect
flow exceeds the direct flow in this system. Together, these whole network indicators show the
importance of indirect interactions in the system. A next analytical step might be to apply the
Utility or Mixed Trophic Impacts analyses to determine the net relationships among the ecosystem
components when we consider the direct and indirect interactions, but this is beyond our analysis
here. More detailed guidance for how to interpret ENA results can be found in previously published
literature (Fath & Borrett, 2006; Jørgensen et al., 2007; Schramski et al., 2011).
3 Value Added Features

There are several features of the *enaR* package beyond the core analyses that add substantive value for users. In this section we highlight several of these features including a library of 100 ecosystem network models, methods for conducting batch analysis (i.e., simultaneous analysis of multiple models), and connections to other analytical software.

3.1 Model Library

To facilitate new systems ecology and network science, we included a library of 100 previously published ecosystem network models with the *enaR* package. These models each trace a thermodynamically conserved unit (e.g., C, N, P) through a particular ecosystem. The models in this set are empirically-based in that the authors attempted to model a specific system and parameterized the model to some degree with empirical estimates. While the library includes models used previously to test several systems ecology hypotheses (Borrett, 2013; Borrett & Salas, 2010; Borrett *et al.*, 2010; Salas & Borrett, 2011), and the set has a 47% overlap with the set of models previously collected by Dr. Ulanowicz ([http://www.cbl.umces.edu/~ulan/ntwk/network.html](http://www.cbl.umces.edu/~ulan/ntwk/network.html)), the full set has not previously been collected and distributed together.

We tentatively split these models into two classes. The most abundant class is the trophic network models. These models tend to have a food web at their core, but also include non-trophic fluxes generated by processes like death and excretion. The annual carbon flux model for the mesohaline region of the Chesapeake Bay is a typical example (Baird & Ulanowicz, 1989). The second class of models focuses on biogeochemical cycling. In contrast to the trophic networks, the biogeochemical cycling models tend to have more highly aggregated nodes (more species grouped into a compartment), include more abiotic nodes that could represent chemical species (e.g., ammonia in a nitrogen cycle), have a lower dissipation rate, and therefore they tend to have more
recycling (Borrett et al., 2010; Christian et al., 1996). Christian & Thomas’s (2003) models of
nitrogen cycling in the Neuse River Estuary are good examples of the class. The package vignette
has a full listing of the models included along with references to their original publications (Lau
et al., 2013).

3.2 Batch Analysis

Advances in ecosystem ecology have been made by comparing network metrics across multiple
ecosystem models. For example, Christensen (1995) applied ENA to identify and compare the
maturity of 41 ecosystem models, and van Oevelen et al. (2011) compared the organic matter
processing of food webs in three sections of the Nazaré submarine canyon. The enaR tool simplifies
the work flow for these types of comparison. Given a list of models like the model library, it is
possible to quickly analyze multiple models using R’s lapply function (see help(“lapply”)). This
facilitates the kind of comparative network analysis often of interest to ecologists (Christian et al.,
2005; Monaco & Ulanowicz, 1997; Whipple et al., 2007).

Batch analysis can be used in several additional ways. One application is for meta-analyses,
such as tests of the generality of hypothesized ecosystem properties like network non-locality (Salas
& Borrett, 2011), or to investigate how physical features might influence ENA results (Niquil et al.,
2012). Fig. 1b illustrates the rank-ordered network homogenization statistic for the 56 trophic-
based ecosystem models in the library. The homogenization statistic is greater than one in all of
these models indicating that the network of indirect interactions tend to more uniformly distribute
the resources than is obvious from the direct interactions, which extends previous results of Borrett
& Salas (2010) to include several new models. A second kind of application is the exploration of
new ENA inter-relationships. With the collection of algorithms and the library of models, we can
now investigate possible relationships among ENA indicators from different schools (Fig. 1c). The
R script to generate Fig. 1 is available as an online enhancement (Item S1). A third application of batch analysis is to investigate the previously unknown empirical ranges of ENA whole-network statistics, which may be useful for interpreting results from specific applications. Fig. 3 shows the observed distribution of values for selected network statistics from the 100 models in the library easily analyzed using `lapply` and the associated `enaR` functions.

### 3.3 New Connections

A third advantage of the `enaR` package design is that it enables network ecologists easier access to other network tools and analyses that might be useful. The `enaR` package uses the R network data structure defined in the `network` package (Butts, 2008a). This means that network ecologists using `enaR` can also use the network manipulation functions and visualization features of the `network` package. Further, the R Social Network Analysis (SNA) package, `sna`, (Butts, 2008b) also uses this network data object. This means that network ecologists can apply many of the SNA algorithms directly to their ecological network models. Fig. 1d illustrates applying the betweenness centrality function to the Chesapeake Bay trophic model (Baird & Ulanowicz, 1989) and visualizing the results using a target centrality plot (Brandes et al., 2003). This analysis highlights the central role of Sedimentary Particulate Carbon and bacteria in the Sediment Particulate Organic Carbon (POC) in the carbon flux of the estuary.

In addition, `enaR` can be a starting point for ecosystem network ecologists to use other R network tools. For example, the `igraph` package provides functions to apply classic graph theory (Csardi & Nepusz, 2006). The `limSolve` package provides capabilities to infer network model fluxes from empirical data by linear inverse modeling (Soetaert et al., 2009), which can also be used for uncertainty analyses of ENA (Kones et al., 2009). There are a wealth of additional R package that network ecologists may find useful including `bipartite` (Dormann et al., 2008), `vegan` (Dixon,
The enaR package encodes exiting ENA algorithms, and is designed to address limitations of current ENA software and facilitate wider use and development. It does this by (1) providing greater accessibility to the code (e.g., free and open source software available on multiple OS), (2) collecting a broad set of available ENA algorithms and workflow management functions, and (3) creating the potential for collaborative development (via GitHub and CRAN). Further, the software is extensible for individual needs and it lets users integrate ENA into a broader workflow in R in a way that is more challenging when using web based tools like EcoNet (Kazanci, 2007; Schramski et al., 2011). Finally, it lets users have access to other network and statistical analysis tools (e.g., social network analysis) that are already part of R. These benefits come at the cost of having a steeper learning curve (e.g., users must know R), which may make enaR more suited to advanced practitioners.

In the near future, we anticipate two initial lines of continued development for the enaR package. The first is to increase the connections between the enaR package and other modeling and analytical tools. For example, we are currently working with colleagues to enable users of Ecopath with Ecosim (Christensen & Walters, 2004) to apply the enaR tools in a seamless way. We are also developing functions to connect between enaR and the R limSolve package (Soetaert et al., 2009) for creating models using Linear Inverse Modeling and to enable uncertainty analysis (Kones et al., 2009). The second line of development is to extend the package’s capabilities. While it currently contains most of the many commonly used ENA algorithms used by ecologists, it is far from complete. For example, Ulanowicz’s (1983) decomposition of cycles is not yet included nor is his construction for the Lindeman trophic spine (Ulanowicz & Kemp, 1979). Network model construction tools, such as least-inference methods for building models from empirical data (Ulanowicz & Scharler,
243 and Fath’s (2004) algorithm for constructing plausible ecosystems models are also possible enhancements.

In conclusion, \textsf{enaR} is an R package intended to facilitate the use and the collaborative development of Ecosystem Network Analysis, a branch of network ecology. This domain is rapidly growing in part because the tools and techniques let ecologists address a wide range of relational questions at the core of ecology. We look forward to seeing new ecological discoveries made through the use of \textsf{enaR}.

5 Acknowledgments

We would like to acknowledge and thank David Hines for contributing to the initial code, and Pawandeep Singh for collecting the output for Table S1. We also thank several individuals who used the earlier versions of the software and provided helpful feedback for further development including Ursula Scharler, Shaoqing Chen, Emily Oxe, and John Mejaski. In addition, we thank the many ecosystem model authors who created, shared, and published their work. This work was funded in part by the US National Science Foundation (DEB1020944, DEB0425908), an NSF Integrative Graduate Education and Research Traineeship (MKL; DGE0549505) and a UNCW Cahill award (SRB).

References


### 6 Tables

Table 1: Selected data input, management, and export functions in enaR.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Example Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>pack</td>
<td>This function lets users combine model elements into a network data object.</td>
<td>None</td>
</tr>
<tr>
<td>unpack</td>
<td>Extracts the individual model elements (e.g., flows, inputs, outputs) from the network data object.</td>
<td>None</td>
</tr>
<tr>
<td>read.scor</td>
<td>Creates a network data object from a SCOR formatted data file.</td>
<td>Ulanowicz &amp; Kay (1991)</td>
</tr>
<tr>
<td>ssCheck</td>
<td>Checks to see if the model is at steady-state.</td>
<td>None</td>
</tr>
<tr>
<td>balance</td>
<td>Applies one of four balancing algorithms to a model not at steady-state.</td>
<td>Allesina &amp; Bondavalli (2003)</td>
</tr>
<tr>
<td>force.balance</td>
<td>Runs balancing algorithm as many times as necessary to balance the model.</td>
<td>None</td>
</tr>
<tr>
<td>write.nea</td>
<td>Writes the model data to the file format used as input for NEA.m.</td>
<td>Fath &amp; Borrett (2006)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
<td>Example Reference</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>enaStructure</td>
<td>ENA Structural analysis returns the adjacency matrix and multiple common descriptive statistics (e.g., number of nodes, connectance, pathway proliferation rate).</td>
<td>Borrett et al. (2007)</td>
</tr>
<tr>
<td>enaFlow</td>
<td>Calculates node throughflow and input and output oriented direct and integral flow intensity matrices. It also returns multiple whole network descriptive statistics including Total System Throughflow, Finn Cycling Index, and Average Path Length.</td>
<td>Finn (1976)</td>
</tr>
<tr>
<td>enaAscendency</td>
<td>Performs ascendency analysis on the model flows and returns whole-network statistics including the average mutual information, Ascendency, Capacity, and Overhead.</td>
<td>Ulanowicz (1997)</td>
</tr>
<tr>
<td>enaStorage</td>
<td>ENA Storage analysis considers how the model fluxes generate the node storage (e.g., biomass) in the system. This function returns the input and output oriented direct and integral storage matrices.</td>
<td>Matis &amp; Patten (1981)</td>
</tr>
<tr>
<td>enaUtility</td>
<td>ENA Utility analysis investigates the direct relationships among the network nodes as well as the integral relationships when all of the indirect interactions are also considered.</td>
<td>Patten (1991)</td>
</tr>
<tr>
<td>enaMTI</td>
<td>Mixed Trophic Impacts assesses the net relationships among species in a food web.</td>
<td>Ulanowicz &amp; Puccia (1990)</td>
</tr>
<tr>
<td>enaControl</td>
<td>Control analysis determines the relative control one node exerts on another through the transaction network.</td>
<td>Dame &amp; Patten (1981)</td>
</tr>
<tr>
<td>enaEnviron</td>
<td>Returns the n unit and n realized input and output environs of the model.</td>
<td>Patten (1978)</td>
</tr>
</tbody>
</table>
Table 3: Selected enaR auxiliary functions and analyses.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Example Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Specialty Analyses</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>enaAll</td>
<td>Runs all of the primary ENA algorithms.</td>
<td>None</td>
</tr>
<tr>
<td>get.ns</td>
<td>Returns the whole-network statistics from enaStructure, enaFlow, enaAscendency, enaStorage, and enaUtility.</td>
<td>None</td>
</tr>
<tr>
<td>eigenCentrality</td>
<td>Calculates the average eigenvalue centrality for any input matrix.</td>
<td>Fann &amp; Borrett (2012)</td>
</tr>
<tr>
<td>environCentrality</td>
<td>Returns the input, output, and average environ centralities for a matrix.</td>
<td>Fann &amp; Borrett (2012)</td>
</tr>
<tr>
<td>TET</td>
<td>Returns the total environ throughflows.</td>
<td>Whipple et al. (2007)</td>
</tr>
<tr>
<td>TES</td>
<td>Returns the total environ storages.</td>
<td>Matis &amp; Patten (1981)</td>
</tr>
<tr>
<td><strong>Auxiliary Functions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>get.orient</td>
<td>Determine the orientation of the results (row-to-column vs. School).</td>
<td>None</td>
</tr>
<tr>
<td>set.orient</td>
<td>Set the orientation of the results (row-to-column vs. School).</td>
<td>None</td>
</tr>
<tr>
<td>mExp</td>
<td>This function lets users calculate matrix exponents.</td>
<td>None</td>
</tr>
</tbody>
</table>
Figure 1: Example of analysis and visualizations created with enaR (a) network digraph of the internal flows of an oyster reef ecosystem model (Dame & Patten, 1981), (b) network homogenization statistic for 56 trophic ecosystem models (rank-ordered), (c) scatter plot showing the relationship between the ascendency-to-capacity ratio and the indirect flow index for the 56 trophic ecosystem models included in the package, and (d) target plot of the betweenness centrality from social network analysis calculated for the 36 nodes of the Chesapeake Bay ecosystem model (Baird & Ulanowicz, 1989).
library(enaR)  # load enaR package
> # -- ENTER MODEL DATA -- from Dame and Patten (1981)
> # node names
> names <- c("Filter Feeders","Microbiota","Meiofauna",
+  "Deposit Feeders","Predators","Deposited Detritus")
> # Internal Flows of model, as matrix (oriented row to column)
> F <- matrix(c(0, 0, 0, 0, 0, 0, 0, 0, 0,
+  0, 0, 8.1721, 0, 1.2060, 0, 0, 7.2745, 0,
+  0, 1.2060, 0.6609, 0, 0.6431, 0.5135, 0, 0,
+  0.1721, 0, 15.7910, 0, 4.2403, 1.9076, 0.3262, 0),
+  ncol=6)
> rownames(F) <- names  # add node names to rows
> colnames(F) <- names  # add node names to cols
> # boundary flows
> inputs <- c(41.47,0, 0, 0, 0, 0)
> outputs <- c(25.1650, 5.76, 3.5794, 0.4303, 0.3594, 6.1759)
> # Living
> Living <- c(TRUE,TRUE,TRUE,TRUE,TRUE,FALSE)
> # pack the model data into the R network data object
> m <- pack(flow=F,input=inputs, respiration=outputs, outputs=outputs, living=Living)
> ssCheck(m)  # check to see if the model is at steady-state
[1] TRUE
> # perform flow analysis
> F <- enaFlow(m)  # perform ENA flow analysis
> attributes(F)  # show analysis objects created
$snames
[1] "T"  "G"  "GP"  "N"  "NP"  "ns"

> F$ns  # show flow analysis network statistics
 Boundary    TST  TSTp   FCI    BFI    DFI    IFI
[1,] 41.47  83.5833  NA  2.015512  0.1101686  0.4961517  0.1950689  0.3087794
  ID.F  ID.F.I  ID.F.O  HMG.I  HMG.O  AMP.I  AMP.O  mode0.F  mode1.F
[1,] 1.582925  1.716607  1.534181  2.051826  1.891638  3  1  41.47  32.90504
 mode2.F  mode3.F  mode4.F
[1,] 9.208256  32.90504  41.47
> F$T
 Filter Feeders  Microbiota  Meiofauna  Deposit Feeders
  41.4700       8.1721       8.4805       2.5100
Predators Deposited Detritus
  0.6856       22.2651

Figure 2: Example code for applying enaR Flow analysis to Dame & Patten’s (1981) oyster reef model.
<table>
<thead>
<tr>
<th>Statistic</th>
<th>Min</th>
<th>Distribution</th>
<th>Max</th>
<th>Median</th>
<th>Mean</th>
<th>CV</th>
</tr>
</thead>
<tbody>
<tr>
<td>n</td>
<td>4</td>
<td></td>
<td>125</td>
<td>15</td>
<td>26.66</td>
<td>1.02</td>
</tr>
<tr>
<td>C</td>
<td>0.05</td>
<td></td>
<td>0.45</td>
<td>0.22</td>
<td>0.25</td>
<td>0.51</td>
</tr>
<tr>
<td>LD</td>
<td>1</td>
<td></td>
<td>3.14</td>
<td>4.58</td>
<td>0.89</td>
<td></td>
</tr>
<tr>
<td>lam1A</td>
<td>0</td>
<td></td>
<td>14.17</td>
<td>3.27</td>
<td>4.27</td>
<td>0.76</td>
</tr>
<tr>
<td>FCI</td>
<td>0</td>
<td></td>
<td>0.98</td>
<td>0.26</td>
<td>0.38</td>
<td>0.86</td>
</tr>
<tr>
<td>APL</td>
<td>1.37</td>
<td></td>
<td>186.25</td>
<td>3.67</td>
<td>20</td>
<td>1.91</td>
</tr>
<tr>
<td>IFI</td>
<td>0.04</td>
<td></td>
<td>0.99</td>
<td>0.53</td>
<td>0.56</td>
<td>0.52</td>
</tr>
<tr>
<td>HMG.O</td>
<td>1.04</td>
<td></td>
<td>13.07</td>
<td>1.78</td>
<td>2.3</td>
<td>0.83</td>
</tr>
<tr>
<td>AMP.O</td>
<td>0</td>
<td></td>
<td>323</td>
<td>6.5</td>
<td>19.77</td>
<td>1.91</td>
</tr>
<tr>
<td>AMI</td>
<td>1</td>
<td></td>
<td>2.25</td>
<td>1.57</td>
<td>1.58</td>
<td>0.21</td>
</tr>
<tr>
<td>ASC.CAP</td>
<td>0.25</td>
<td></td>
<td>0.75</td>
<td>0.39</td>
<td>0.42</td>
<td>0.28</td>
</tr>
<tr>
<td>synergism.F</td>
<td>2.41</td>
<td></td>
<td>60.51</td>
<td>3.95</td>
<td>5.69</td>
<td>1.1</td>
</tr>
<tr>
<td>mutualism.F</td>
<td>0.6</td>
<td></td>
<td>4</td>
<td>1.16</td>
<td>1.43</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Figure 3: Distributions of selected ENA network statistics from the u 100 empirically-based ecosystem models included in enaR. The results are summarized using a histogram showing the distribution of the values of each network statistic between the observed minimum and maximum values. The median, mean, and coefficient of variation (ratio of standard deviation and mean) values are also reported. The network statistics are the number of nodes (n), the connectance (C = L/n²), link density (LD = L/n), pathway proliferation rate (lam1A), Finn cycling index (FCI), average path length (APL), indirect flow intensity (IFI), output oriented network homogenization ratio (HMG.O), output-oriented network amplification ratio (AMP.O), average mutual information (AMI), the ascendency-to-capacity ratio (ASC.CAP), flow-based network synergism (synergism.F) and mutualism (mutualism.F).