An Interactive Interface for Super Nested Arrays

Chun-Lin Liu

Department of Electrical Engineering, California Institute of Technology

E-mail: cl.liu@caltech.edu

What is this?

In array processing, mutual coupling between sensors has an adverse effect on the estimation of parameters (e.g., DOA). While there are methods to counteract this through appropriate modeling and calibration, they are usually computationally expensive, and sensitive to model mismatch. On the other hand, sparse arrays, such as nested arrays, coprime arrays, and minimum redundancy arrays (MRAs), have reduced mutual coupling compared to uniform linear arrays (ULAs). With N denoting the number of sensors, these sparse arrays offer $O(N^2)$ freedoms for source estimation because their difference coarrays have $O(N^2)$-long ULA segments. But these well-known sparse arrays have disadvantages: MRAs do not have simple closed-form expressions for the array geometry; coprime arrays have holes in the coarray; and nested arrays contain a dense ULA in the physical array, resulting in significantly higher mutual coupling than coprime arrays and MRAs.

The super nested array [1-4] has all the good properties of the nested array, and at the same time achieves reduced mutual coupling. There is a systematic procedure to determine sensor locations. For fixed N, the super nested array has the same physical aperture, and the same hole-free coarray as does the nested array. But the number of sensor pairs with small separations ($\lambda/2$, $2\times\lambda/2$, etc.) is significantly reduced. The last property helps to reduce mutual coupling effects.

This interactive interface assists users to understand the basics of super nested arrays and to create new array configurations of their own. With this interface, users can place the sensors arbitrarily on the 2D representation, show the associated difference coarray, and calculate the weight functions.

How to use this interactive interface?

Installation

- Extract `SN.zip` to a working directory. (Click here to download SN.zip)

Run the interactive interface for the first time

- Open MATLAB and run `interactive_interface.m` from the working directory. The following figure should pop up on your screen.

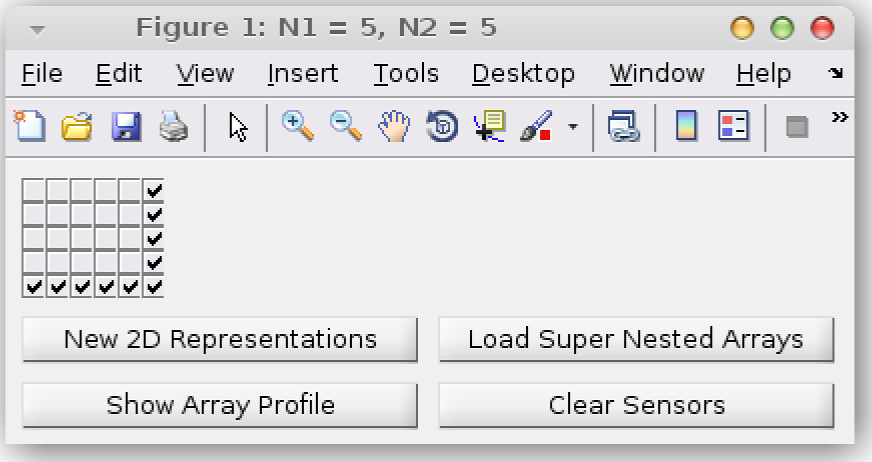


Fig. 1: The interactive interface with N1=5 and N2=5. Note that the actual appearance might be slightly different due to operating systems. This program was running under Debian GNU/Linux 8.5 (Jessie), MATE Desktop Environment 1.8.1, and MATLAB R2015b (GLNXA64).

The meaning of these controls are clarified as follows.

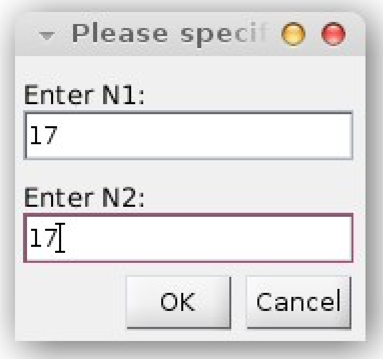
- The **title** (Figure 1: `N1=5, N2=5`) shows the parameters `N1` and `N2` of this 2D representation. 2D representations are defined in [1], or page 13 in our presentation slides (Click here to download).
- The **check boxes** denote all the grid points on 2D representations. If a check box is checked (✓), it means there is a sensor at this position and vise versa. For example, the array configuration shown in the Fig. 1 has sensors located at 1, 2, 3, 4, 5, 6, 12, 18, 24, 30.
- The usage of the push buttons are
  - **New 2D Representations**: Create another interactive interface with specified `N1` and `N2`.
  - **Load Super Nested Arrays**: Prompt the user to enter the parameter `Q` for super nested arrays. Then the sensor locations will be generated according to `N1`, `N2`, and `Q`.
  - **Show Array Profile**: Show the 1D representation, the 2D representation, and the weight function in another figure.
  - **Clear Sensors**: Make all the check boxes unchecked on the 2D representation.

Super nested arrays as an example

Consider the super nested array with `N1=17`, `N2=17`, and `Q=3`. We can use the interactive interface to show the sensor locations, the difference coarray, and the weight function.

Step 1

On the interactive interface, click “New 2D Representations.” A separate window prompts the user to enter `N1` and `N2`. Please fill in the blanks with 17, as shown in Fig. 2, and then click OK.



Enter N1:
17

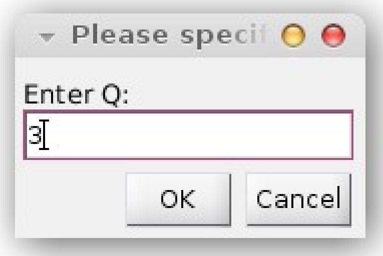
Enter N2:
17

Fig. 2: Specify N1 and N2 for the new 2D representation.

Note: One can also enter `interactive_interface( 17, 17 )` in the MATLAB command line.

Step 2

You will see another interactive interface with `N1=17` and `N2=17`. Please click “Load Super Nested Arrays” and there will be a separate window to enter the parameter `Q`, as shown in Fig. 3. Please enter 3 and then click OK.



Enter Q:
3

Fig. 3: Specify Q for the super nested array.

Step 3

Now the check boxes correspond to the sensor locations for the super nested array of the specified parameters, as depicted in Fig. 4.

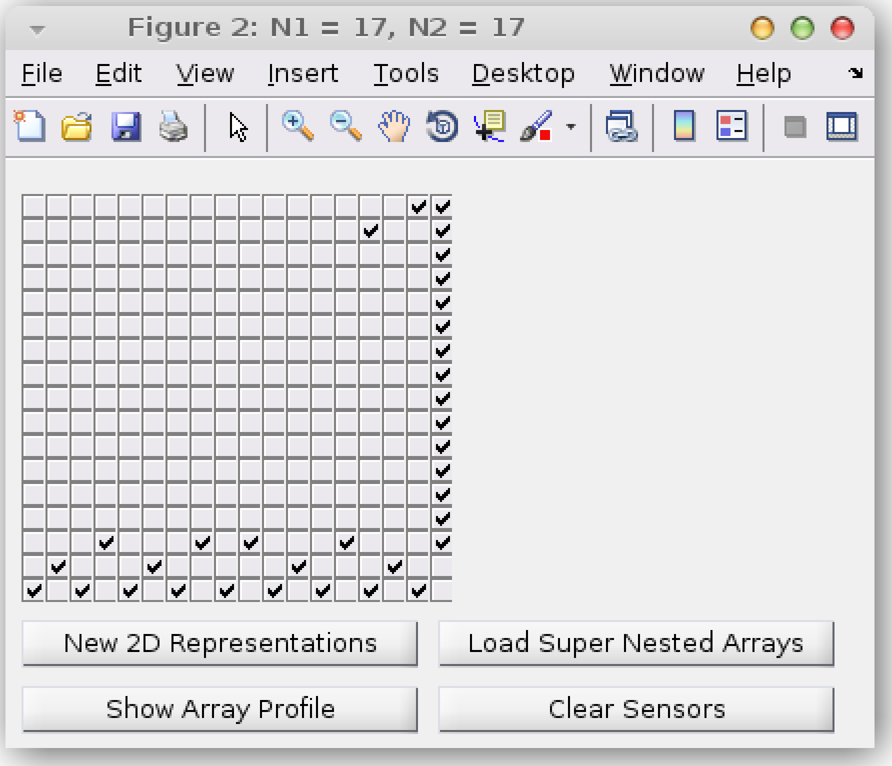


Fig. 4: The interactive interface for the super nested array with N1=17, N2=17, and Q=3.

Next click "Show Array Profile." A separate figure, as shown in Fig. 5, will appear on the screen. The meaning of the four subplots is as follows:

- Top left: The 1D representation of the array in Fig. 4.
- Top right: The 2D representation of the array in Fig. 4.
- Bottom left: The weight function $w(m)$ for the coarray location $m \geq 0$.
- Bottom right: The indicator function of $w(m) > 0$, which is similar to Fig. 4 of [2]. If this plot has all ones, then the associated array is a restricted array (or with hole-free difference coarray). If it contains some zeros, they indicate the location of holes.

At the same time, some messages about the coarray are dumped to the MATLAB command line. The first line and the second line of the messages show the number of sensors and their locations, respectively. The third line is the weight function $w(m)$ for $1 \leq m \leq 8$. The fourth line and the fifth line correspond to the number of holes and where these holes are.

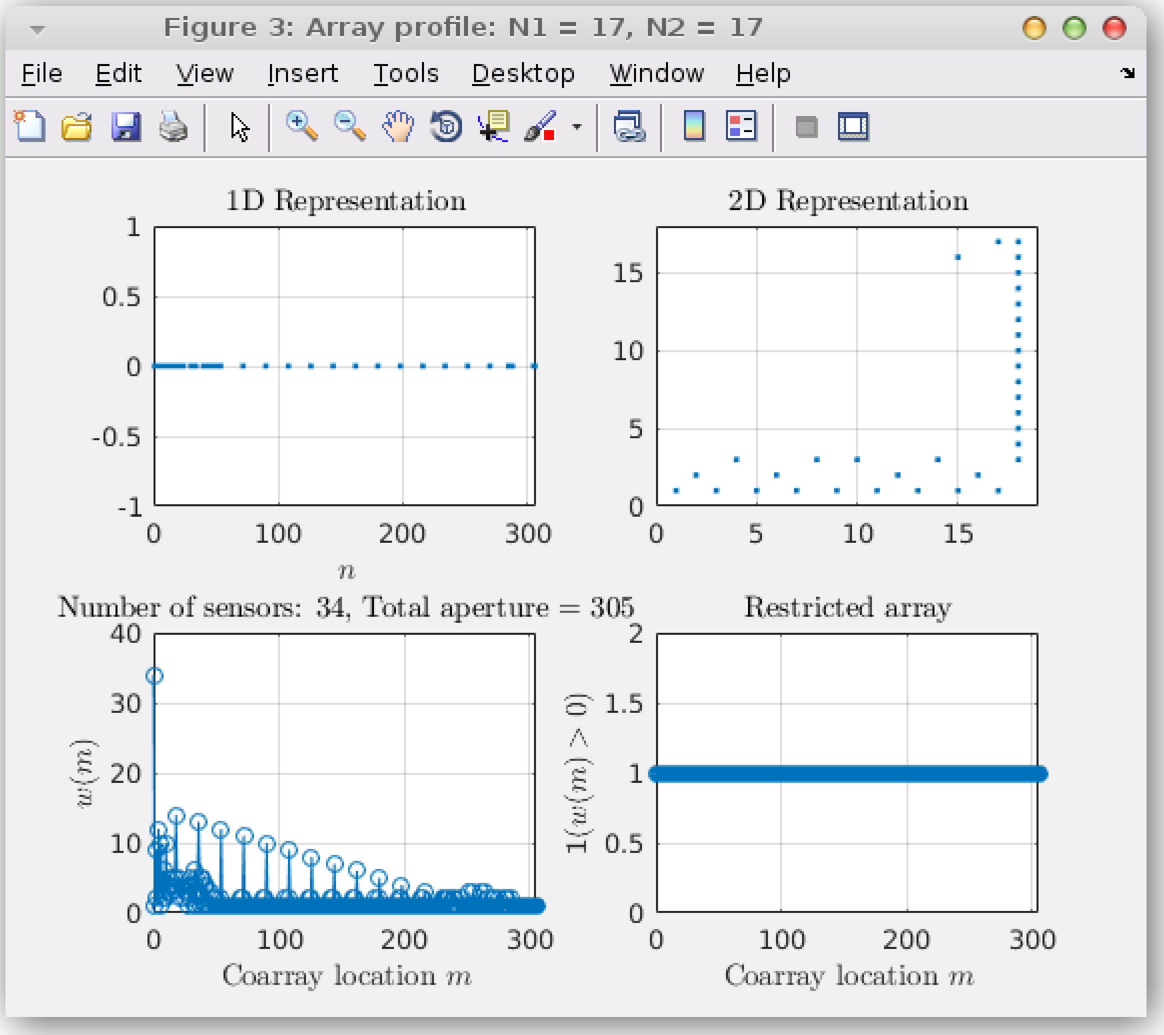


Fig. 5: Some information about the array configuration in Fig. 4.

Design array configurations on 2D representations

This program also allows users to design new arrays by adjusting the check boxes on the interface. For instance, Fig. 6 shows the interactive interface and the associated information of a new array, whose second and third layers are different from those in super nested arrays with `Q=3` (Fig. 4). However, it can be seen that this new array is a restricted array (or with hole-free difference coarrays).

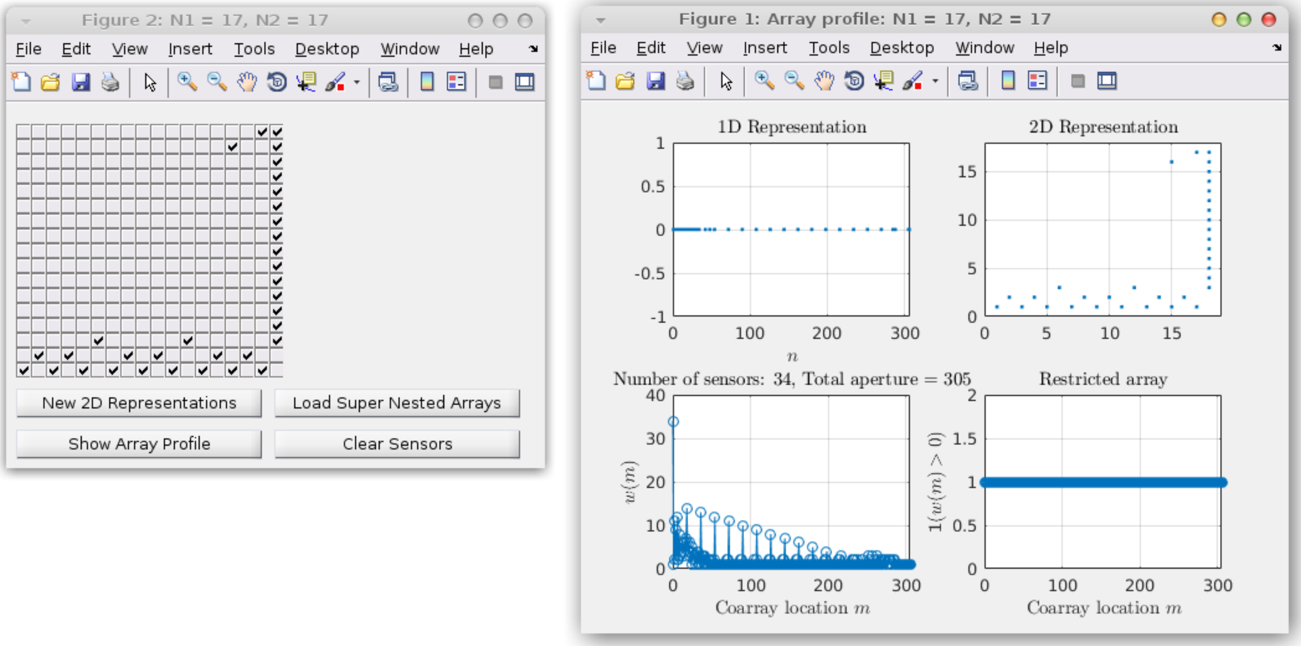


Figure 2: N1 = 17, N2 = 17
File Edit View Insert Tools Desktop Window Help
New 2D Representations
Load Super Nested Arrays
Show Array Profile
Clear Sensors
Figure 1: Array profile: N1 = 17, N2 = 17
File Edit View Insert Tools Desktop Window Help
1D Representation
n
2D Representation
Number of sensors: 34, Total aperture = 305
$w(m)$
Coarray location m
Restricted array
$1(w(m) > 0)$
Coarray location m

Fig. 6: Another array configuration that leads to a hole-free difference coarray.

Contact information

Chun-Lin Liu

E-mail: cl.liu@caltech.edu
Project website: http://systems.caltech.edu/dsp/students/clliu/SuperNested.html

References

1. **C.-L. Liu** and P. P. Vaidyanathan, “Super Nested Arrays: Linear Sparse Arrays with Reduced Mutual Coupling - Part I: Fundamentals,” *IEEE Transactions on Signal Processing*, vol. 64, no. 15, pp. 3997-4012, Aug. 2016.
2. **C.-L. Liu** and P. P. Vaidyanathan, “Super Nested Arrays: Linear Sparse Arrays with Reduced Mutual Coupling - Part II: High-Order Extensions,” *IEEE Transactions on Signal Processing*, vol. 64, no. 16, pp. 4203-4217, Aug. 2016.
3. **C.-L. Liu** and P. P. Vaidyanathan, “Super Nested Arrays: Sparse Arrays with Less Mutual Coupling than Nested Arrays,” in *Proc. of 2016 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP 2016)*, pp. 2976-2980, Shanghai, China, Mar. 2016.
(Best student paper award)
4. **C.-L. Liu** and P. P. Vaidyanathan, “High Order Super Nested Arrays,” in *Proc. of the Ninth IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM 2016)*, Rio de Janeiro, Brazil, Jul. 2016.
(Best student paper award)

Last update: Aug 14, 2016.