State-of-the-Art of Topology Processors for EMS and PMU Applications and Their Limitations

Mostafa Farrokhabadi and Luigi Vanfretti
KTH Royal Institute of Technology, School of Electrical Engineering, Electric Power Systems Department
E-mail: mostafaf@kth.se, luigiv@kth.se

***Abstract*- This article provides a review of the current openly available works in the field of topology processing. The most important works in this area are critically scrutinized, and their limitations are identified. The drawbacks are identified and fully discussed, and their effect on the output of the topology processor is investigated. To support the discussion, the IEEE Reliability Test System 1996 is simulated in real-time to show the deficiencies with the current available topology processor that uses PMU data. The real-time simulation is performed using an eMegaSim Opal-RT real-time simulator which is part of the "SmarTS Lab" at KTH Royal Institute of Technology. Finally, possible potential solutions are briefly proposed as a conclusion.**

I. INTRODUCTION

Topology Processor plays a crucial role in all of the Energy Management Systems (EMS) applications. All the processes and methods in EMS applications are highly dependent on the outcome of the system topology processor. Unfortunately, the openly available documented works of the current topology processors and their corresponding implementations show a lack of rigorosity. Actually, the documentation and algorithms are from being in a state that allows for possible independent implementation.

In addition, the application of phasor measurement units in different power systems functions is proliferating [8]. A good example is the introduction of PMU-only state estimators which are largely dependent on robust topology processors [6, 7]. PMU applications of this sort have increased the need for having a fast and robust topology processor which has the ability of using PMU measurements simultaneously with conventional data. However, there is just one previous work which discusses the application of PMU for determining the topology of a power system [4]. Unfortunately, this TP is a rudimentary algorithm and lacks enough regorosity for its application.

The openly available documented topology processors are quite limited, and most of the other related works provide only a basic description of what a topology processor is. No clear algorithm or execution method is provided. In this article the authors attempt to cover most of the related works in this area and to provide a clear discussion about them.

The reminder of this paper is as follows. In section II, a brief description of a typical topology processor is provided. All the necessary parts involved in a topology processor are introduced. Section III fully scrutinizes the current openly available topology processors. The significant works in the area are discussed and their major drawbacks are identified. Then in section IV, the results from the emulation of the algorithms are demonstrated and discussed. Real-time simulation results of IEEE Reliability Test System 1996 [10] are provided. The aim is to use this simulation to discuss the work about the application of PMUs in topology processing. In addition, some potential remedies to the drawbacks identified are briefly discussed. The article ends with conclusion in Section V.

II. NETWORK TOPOLOGY PROCESSOR

As mentioned before, topology processing is one of the key steps in any EMS application. It uses switch statuses and network connectivity data to determine the network topology. In addition, it transforms the bus-section/switching-device model of a power network to a bus/branch model.

A typical topology processor encompasses many different processes. All these processes play an important role in the topology processing procedure. Overlooking any of these processes may have a considerable effect on the final output of a TP. This section provides a brief description of a typical topology processor along all its necessary parts.

A. Processing of Input Data

The topology processing unit receives the breakers status and pre-defines network connectivity data as inputs. The network connectivity data contains information about the system components, and how they are connected to each other initially.

Traditionally, the breaker statuses are telemetered through analog transmitters using TCP/IP or ICCP protocols. The arrival of PMUs has made it possible to send breaker statuses using digital channels of the PMU with much higher rates. This transmission of this digital data is included in the IEEE Std C37.118.2-2011 [9].

B. Substation Analysis

Topology processing might appears as a straightforward task; an open breaker always means that a line is disconnected. However, this situation is much more complex in reality.

In fact, there are many different substation configurations. These configurations can be classified in 5 different types. All

the other configurations are a modification or combination of these 5 original types:

- Single bus configuration
- Main and a transfer bus configuration
- Ring configuration
- Switch and a half configuration
- Double-bus double-switch configuration

Each substation configuration has its own characteristics. In each configuration, the station components are connected to each other in a unique way.

The concept of different substation configurations has made the topology processor to be a complex application. In fact, in any configuration rather than single bus, an open breaker does not necessarily mean that a line is disconnected. This is due to the fact that in most of the configurations, a line is connected to buses in more than one closed path.

In addition, there may be situations in which one entire substation is split into two or more different nodes. These split nodes may be later merged back to form a unified station.

All these phenomena make the topology processor procedure complex. The task of the substation analysis is to scrutinize the stations configurations. It should correctly identify the disconnected and connected components and also report if there are slitting/merging nodes.

C. *Islanding Analysis*

An electric power system consists of many different substations which are connected to each other via transmission lines. However, in some situations it may occur that the link between two stations is disconnected due to switching in the system. This situation may result in one or more parts of the system to have no link with the other parts. In this case, it is said that islanding has occurred, and those disconnected parts are called islands.

Obviously, there may be situations in which previously separated islands will join together again and form a new island. A topology processor should be able to detect and report all these changes correctly. In addition, it should be able to detect if an island is energized, or de-energized. An island is energized if it has at least one working generator station.

The identification of energized and de-energized islands is important as well. Actually, those parts of the system which are de-energized should not participate in any EMS calculations.

III. CURRENT TOPOLOGY PROCESSORS AND THEIR LIMITATIONS

This section is dedicated to the scrutiny of the current and openly available topology processors. This section attempts to fully discuss their limitations and drawbacks.

A. *Automatic Power System Network Topology Determination [1]*

This work is the first published in this area. It is the corner stone for several other works in the field of topology processing. Although the idea introduced in this article is a fascinating one, the procedure is exposed to a number of drawbacks as follows.

The algorithm presented in this article is developed using explanations in a purely descriptive manner. No mathematical or flowchart representation is provided to support these descriptions. In addition, the explanations provided are so complex and far from enabling an executable form. Only, general descriptive explanations are given and no further detailed instructions are provided to help to implement or code the algorithm. This is a considerable drawback. In fact, many important parts of the method are left without any well-developed explanation.

Just as an example, regarding the islanding analysis, it is states that "*when one or more substations have become separated, it is necessary to introduce phantom unavailable circuits that connect the separated parts of the substation*". No information is provided regarding how and where to introduce these circuits, and how to correlate them with input and output data matrices. Also no algorithm is provided regarding the correct procedure to introduce these phantom circuits. There are lots of other explanations like this which makes the entire algorithm far from reaching an executable form.

For the algorithm to work properly, it needs to perform topology processing (for the whole system) each time a change occurs. In fact, each time that a breaker switching happens, it has to compare all the system breaker statuses with the initial states; in this way it determines the topology for a substation in which some changes from initial status have taken place. The reason is that the algorithm does not correctly track nor correlates the changes from one cycle to the next one. As no clear automatic numbering is provided for splitting/merging nodes, the TP has to perform the topology for the whole system again. Otherwise it may fail to assign numbers to those nodes correctly.

However, there is another problem even if the algorithm successfully performs the topology processing for the whole system again. In this case, the numbers previously assigned to some nodes have changed completely even if those nodes have had no breaker status changes inside. This makes the tracking of the system impossible.

Moreover, no information is saved regarding the original breaker statuses in the 'Configuration Matrix' from the offline data. This is a considerable drawback. Suppose that a station which has been previously split into n nodes is split into m nodes in the current cycle. In this case, the algorithm is incapable of re-defining the Configuration Matrix. This is a major defect because this matrix is also used for islanding analysis.

In addition, the algorithm suffers from several limitations. For example, it only deals with changes within a substation configuration. There is no consideration for changes of breakers which are not located in a substation configuration but are located on the lines interconnecting substations.

Also, the algorithm is tested on a system that only has combinations of "breaker and a half" and "double bus double

breaker" configurations. The circuits in the system are numbered using a numbering rule which fails to function when it comes to configurations such as "ring-bus".

In addition, the algorithm has no ability to detect if an island is energized or de-energized. This adds to the previous limitations of the proposed method.

Another limitation of this TP is that it completely overlooks the effect of disconnectors in the system. Considering the disconnectors will result in a major drawback in the algorithm basic rules. In fact, the test system is designed in a way to comply with the algorithm rules, therefore ignoring the presence of disconnectors.

There are also some other minor drawbacks such as unnecessary redundancies in offline data, specifically the Configuration and Index matrices. These redundancies make data interpretation unnecessarily complex.

The above mentioned drawbacks as well as the algorithm highly descriptive presentation, endangers the method's efficiency. The algorithm is not in the form that allows execution of independent implementation, and suffers from many limitations.

B. Real-time Modeling of Power Networks [2]

Real-time modeling of power networks refers to a group of applications which result in a computer-based representation of a power network.

These different applications are usually a topology processor, observability analysis, state estimation, bad data detection and external network modeling. [2] offers a general description for each part of the real-time modeling process. The presentation of the topology processor is more conceptual than technical. It defines the concepts of topology processing, giving a general definition of the necessary parts in a typical topology processor. No implementation code or algorithm is provided, and the description does not provide the technical specification for implementing the concept. Therefore, this article provides no insight for a practical implementation of a real topology processor.

C. A Topology Processor That Tracks Network Modifications Over Time [3]

What is proposed in this article is not an actual independent TP. Instead, it proposes a method to modify a topology processor's output so as to reduce the EMS applications' computational burden.

To develop the idea, the work is completely dependent on the conventional topology processing [1]. Actually, that part of the method which is related to the topology processing is completely the same as the one of the conventional topology processing [1]. Therefore, it is exposed to the same drawbacks and limitations.

In [3], no any specific algorithm is provided and the reader is just limited to purely descriptive explanations. This makes the verification of the method almost impossible. Moreover, the test results only document the computation time, and do not show results on the topology processor itself. Therefore, this article has no practical value for the implementation of topology processor by third parties.

D. A New Algorithm of Topology Analysis Based on PMU Information[4]

This work is the only one in the area that has considered the involvement of PMU data in a topology processor. The proposed algorithm is also completely different from a conventional topology processor. It uses graph theory to define the topology in a power network. Besides, it uses phasor current magnitudes computed by PMUs and compares them with a pre-defined threshold. The result of this comparison is the base for further decisions.

In addition, the approach to the topology processing part is quite rudimentary. Almost all the necessary parts of a topology processor are overlooked. No information is provided regarding different substations configuration, automatic numbering, or islanding. Moreover, the algorithm approach to splitting/merging nodes is inefficient. There is no algorithm on how to assign numbers to newly formed or merged nodes and islands; no information is provided on how to find the corresponding lines (circuits) of the nodes and how to involve possibly new formed nodes in the topology.

As mentioned before, the algorithm verifies switching (or detects switching failures to operate) using the current magnitude. It compares the difference of current magnitudes for two consecutive cycles with a threshold to find out if a switching has happened. It also compares the current magnitude through a line with a threshold to find out if it is connected or not. However, this approach disregards the fact that changes in one part of the system may affect other parts also. No discussion or proof is provided to support the proposed approach. For example, there may be cases in which a line disconnection results in currents in another line to be extremely low. This drawback is demonstrated in Section IV.

As a conclusion, this article only provides rudimentary methods to involve PMU data in a topology processor. The part related to topology processing itself is quite incomplete. In addition, no rigorous algorithm or discussion is conducted to support the method. In addition, no serious testing is carried on to demonstrate how the method works subject to different tests. Therefore, the article has probably low-to-zero efficiency in any practical implementation.

E. A New Approach to Initializing and Updating The Topology of An Electrical Network [5]

The algorithm in this method also uses graph theory to solve for the network topology. First, the power system is transformed into a graph. Then using a specific set of rules, a sparse matrix is built according to the graph. The mathematical methods to solve for this sparse matrix are then used to determine the topology.

Although the approach presented here is quite interesting, the style of presentation does not allow any extension, implementation, or further development by independent researchers. It has been documented more like an industrial report rather than a work which allows for further research.

For example, to extract the network topology from the sparse matrix, the reader is just limited to a few number of mathematical equations.

In addition, the approach of the method towards dealing with different substation configurations is quite vague. It is not clear how the topology processor deals with splitting nodes, islanding, and their corresponding numberings. Unfortunately, the test results are also just limited to documenting the computation time; the behavior of algorithm subject to different situations is not discussed at all.

The discussion regarding the outputs of the topology processor is not well-developed. As mentioned before, the topology is determined by solving the sparse matrix. However, there is not a rigorous algorithm on how to interpret the results, and make it interpretable for other applications.

As a conclusion, considering all the existing issues, it is not possible to implement a practical topology processor using only the information provided in this article. As mentioned before, this article can be considered more like an industrial report.

IV. Demonstration of the Identified Drawbacks

In the previous section, the five major openly available works were scrutinized and their respective drawbacks were determined. Out of the five works, [2] has no practical implementation value. [3] is completely based on [1] when it comes to determining the topology of the system. The problem with [5] is that it is quite intricate and provides vague information. As such it is impossible to implement it independently or further develop it for research purposes. Due to this fact, no clear drawback can be demonstrated for [5].

Therefore, in this section the identified drawbacks with [1] and [4] are demonstrated. For [4] meticulous testing is carried out and it is shown why the procedure is erroneous. However, for [1] it is not possible to provide clear strict results as the algorithm is not in an executable format; it is purely descriptive. Therefore, the demonstration of the drawbacks is limited to situations in which the algorithm of [1] does not work properly (based on the descriptive rules and procedure).

A. Drawback demonstration for "Automatic Power System Network Topology Determination" [1]

The same test model as one introduced in [1] is shown in Fig. 1 and used for illustrating of the drawbacks. In this figure some disconnectors are considered for the system (they are not considered in [1]).

First, the algorithm would not work if the disconnectors are considered as shown in Fig. 1. The basic rules of the algorithm can't comply with it. For example, there is a rule that every circuit breaker should be connected to two and only two circuits. Apparently this rule can't be satisfied in many conditions. In fact, [1] introduces phantom circuit breakers in the case that a breaker is connected to more than two circuits. However, there is no consideration for situations

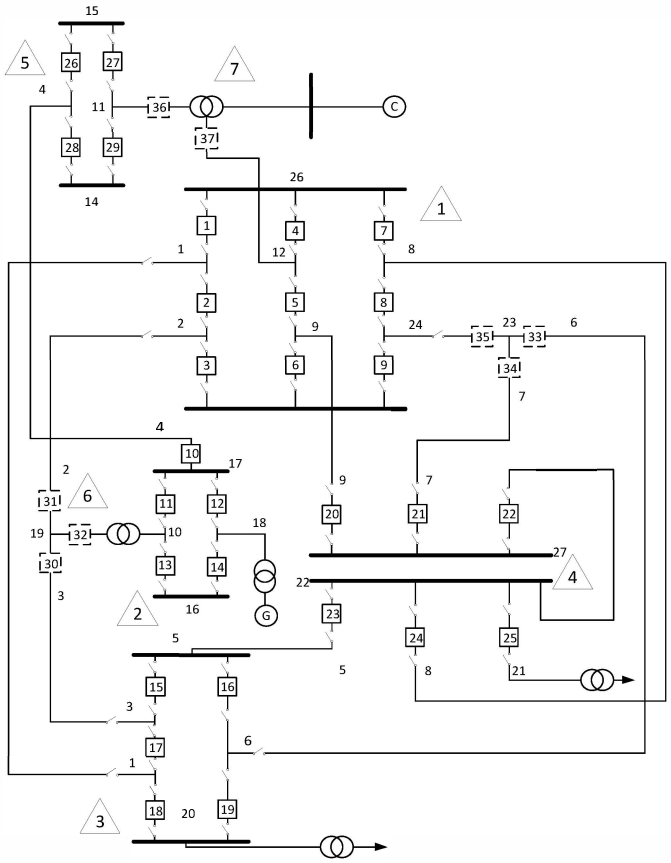

Fig. 1. Modified fictitious test system

in which the breaker is connected to less than two circuits (due to series switches, either breaker or disconnector).

Please note that disconnector and a breaker are equally important when it comes to carrying out topology processing. Therefore, it is very important to consider the presence of disconnectors.

In addition, the algorithm has not considered the possibility of having switches (either breaker or disconnector) on transmission lines (such as lines 1, or 2 in Fig. 1).

Now, suppose that breakers 1, 3, 15, 16, and 18 are opened (disregard the disconnectors). This is the same test scenario in [1]. Station 1 will be split to nodes 1 and 8. Station 3 will be split to nodes 3 and 9. Note that this number assignation is based on the assumption that numbers will be assigned incrementally; no concrete number assignation algorithm is provided in [1]. There will be two islands; one consists of nodes 8, 4 and 9. The other one consists of nodes 1, 2, 3, 5, 6, and 7. According to [1], the algorithm can report these changes correctly. Note that it provides no information regarding the energization state of these islands, and that it is assumed that the algorithm works as described (which is not the reality, because the description of the procedure is far from executable format; refer to Section III part A).

Now suppose that breakers 7 and 9 are also opened. First, if the algorithm wants to determine the topology for station 1, it will find out that the station is split to 3 nodes, and it will give the numbers 1, 8, and 9 to the nodes. However, number 9 is previously assigned to station 3. Actually, no clear method is

available on how to order the numbers, and how to avoid possible interference.

Therefore, the algorithm should be carried out for the whole system again and compare the breaker statuses with the initial ones. In this case, station 1 will be split to nodes 1, 8, and 9 and station 3 will be split to nodes 3 and 10. But, when the algorithm searches the 'Configuration Matrix' to replace the node number 3 to 10 for the corresponding measurements (which previously belong to node number 9) it will get lost. The reason is that there are two indexes for each circuit, and the algorithm should search to find that index with the matching station number. Previously, it searched for 3, and replaced it with 9. Now, it again searches for 3 to replace it with 10, but it cannot find the correct index as it is previously replaced by 9.

This drawback exists even with the assumption that automatic numbering works perfectly. Suppose that the number assignation works perfectly by some method (which is not possible by using only the information provided in [1]). Breakers 1, 3, 15, 16 and 18 are open. Now breakers 1, and 3 are closed. It determines the topology for station 1 and finds out that it consists of only one node. There are some measurements which previously belong to node 8 in 'Configuration matrix'. Those measurements remain untouched although they now belong to node 1 again. The reason is the same as explained in last paragraph. This drawback will severely affect the islanding analysis.

In addition, as said before, there are some other drawbacks; however, there is no space to discuss all of them. For example, the test system is designed in a way to comply with the rules of the algorithm; as such the rules can't comply with some other configurations such as ring-bus arrangement with two series breakers.

The idea of finding a closed path within the substations is useful. However, the algorithm in [1] is quite inefficient in achieving a practical TP. Potential remedies could be the introduction of original substation numbers for both input matrices, and reformation of those matrices. The rules should also be modified to comply with all kinds of substation configuration (including disconnectors). In addition, an efficient automatic numbering algorithm should be provided. The topology processing algorithm itself should change to a more rigorous one. This can be done by describing the algorithm through flowcharts and providing detail of instructions for all the parts. Also the possibility of having switches on the lines should be considered. A possible solution can be to classify the breakers into different types. Then a different algorithm can be provided for each sub-process. In addition, all the necessary processes of a typical topology processor should be included (such as the islands' energy state).

B. Drawbacks with A New Algorithm of Topology Analysis Based on PMU Information[4]

To show the drawbacks of this article a modified version of IEEE Reliability Test System 1996 [10] is simulated in real-time. The modeling platform is MATLAB/Simulink. The simulated model is shown in Fig. 2.

In Fig. 2 the red numbers are those assigned to the breakers; the blue numbers are those assigned to the lines. The green numbers corresponds to busbars in the system.
The aim of the real-time simulation is to produce the measurements similar to those of PMUs. Therefore, current and voltage phasors for the entire systems are captured at 50 samples per second. Therefore, it is possible to verify the procedure proposed in [4].

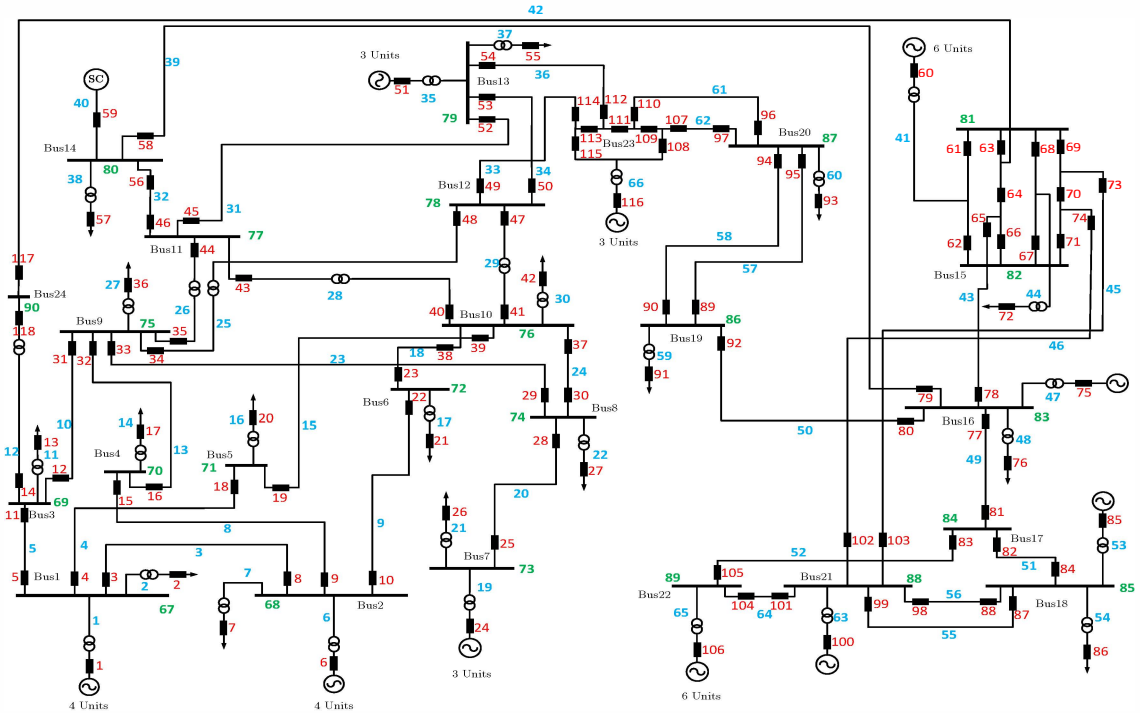

Fig. 2. Modified IEEE Reliability Test System 1996

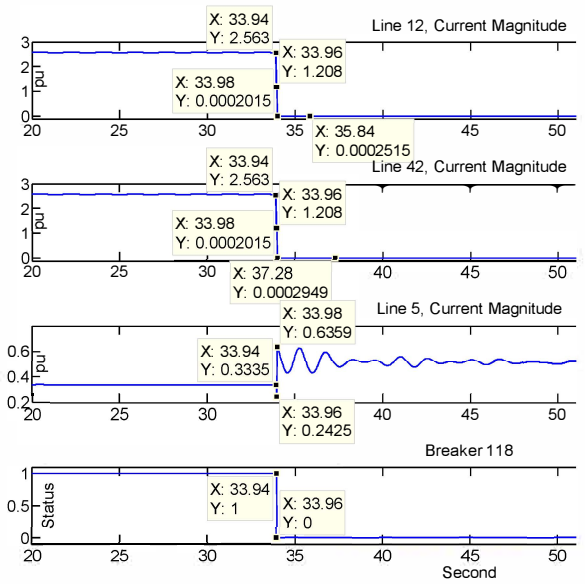

Fig. 2. Switching of breaker 118 and current magnitudes through the lines

The focus here is on the usage of PMU current phasor measurements in [4]. The drawbacks with the topology processor itself are discussed in part D of Section III.

The general procedure of [4] is that it detects the switching using the following formula:

$$\left|\dot{I}_m^{(k+1)} - \dot{I}_m^{(k)}\right| > 0.3\,pu \qquad (1)$$

After a switching is detected, it decides if the corresponding line is connected or disconnected using the following formula:

$$I_m > 0.08\,pu \qquad (2)$$

Now suppose that breaker 118 (located at the left side of Fig. 2) is opened. The currents through lines 5, 12, and 42 along breaker 118 status are shown in Fig. 3. From this figure it can be observed that a switching has happened at time *t=33.96* seconds. It is observed that the currents through all these lines are highly affected by the switching of breaker 118. Actually currents through lines 12 and 42 satisfy (1) at *t=33.94-33.96*, and *t=33.96-33.98*. Current through line 5 also satisfies (1) at *t=33.96-33.98*. Therefore, using the procedure in [4], in addition to breaker 118, breakers 117, 14, 11, and 5 are "wrongly detected" as being switched. Please note that there are several other lines highly affected which can't be shown due to space limitations. So by the switching of one single breaker many breakers have been detected as switched.

In addition, although only line 12 is disconnected, both line 12 and 42 satisfies (2); this means that both line are identified as disconnected.

Therefore, the method proposed in [4] is apparently not efficient enough to exploit PMU measurements. Possible remedies could be the consideration of voltages phasors along currents phasors. Another possible solution can be to consider the current phasor for more than two consecutive instances. In addition, consecutive instances can be compared to each other rather than using a pre-defined threshold. It is also possible to build equations using the current phase angles.

V. CONCLUSION.

In this paper the openly available and major works in the area of topology processing were fully scrutinized. It is shown that although TP is an important tool for power systems operation, all the proposed methods suffer from several limitations.

The drawbacks with current TPs are identified and critical discussions are provided on their effects. Specifically, real-time simulations are performed to demonstrate the defects with one of the approaches. In addition, it is shown that the algorithms lack enough rigorosity in their approach and documentation. This has made any independent implementation almost impossible. Besides, none of the proposed methods are developed to the extent that they can be considered as a complete topology processor. All of them have neglected one or more important parts involved in a practical TP.

As a conclusion, there is an apparent need for having a topology processor which is well-developed and well-documented. It should be in a form to provide to opportunity of being implemented independently. In addition, it should be robust enough to allow for further developments and usage in academic or practical EMS applications such state estimators. In [11], the authors have proposed such a topology processor. In [12], the authors have proposed a solution to involve the PMU measurements in topology processing efficiently.

REFERENCES

[1] A.M. Sasson, S.T. Ehrmann, P. Lynch, and L.S. Van Slyck, "Automatic Power System Network Topology Determination," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-92, no. 2, pp. 610-618, 1973.

[2] A. Bose and K.A. Clements, "Real-time Modeling of Power Networks," *Proceedings of the IEEE*, vol. 75, no. 12, pp. 1607-1622, 1987.

[3] A. Bose and M. Prais, "A Topology Processor That Tracks Network Modifications Over Time," *IEEE Transactions on Power Systems*, vol. 3, no. 3, pp. 992-998, 1988.

[4] C. Qian, Z. Wang, and J. Zhang, "A New Algorithm of Topology Analysis Based on PMU Information," *IEEE 5th International Conference on Critical Infrastructure (CRIS), 2010*, pp. 1-5, 2010.

[5] R. Kussel, R. Chrustowski, and C. Jaborowicz, "The Topology Engine: A new Approach To Initializing And Updating The Topology Of An Electrical Network", ABB Netzleittechnik GmbH, Germany.

[6] L. Vanfretti, J.H. Chow, S. Sarawgi, and B.B. Fardanesh, "A Phasor-data-based State Estimator incorporating phase bias correction," *IEEE Transactions on Power Systems,* vol. 26, no. 1, pp. 111-119, 2011.

[7] A.G. Phadke, J.S. Thorp, and K. Karimi, "State Estimation with Phasor Measurements," *IEEE Transactions on Power Systems*, vol. 1, no. 1, pp. 233-238, 1986.

[8] A.G. Phadke and J.S. Thorp, *Synchronized Phasor Measurements and their Applications*. Springer, New York, 2008.

[9] "IEEE for Synchrophasor Data Transfer for Power Systems," IEEE Std C37.118.2-2011 (Revision of IEEE Std C37.118-2005) , vol., no., pp.1-53, Dec. 28 2011

[10] C. Grigg *et al*, "The IEEE Reliability Test System-1996. A report prepared by the Reliability Test System Task Force of the Application of Probability Methods Subcommittee", *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 1010-1020, 1999.

[11] M. Farrokhabadi, and L. Vanfretti, "An Efficient Automated Topology Processor for HV Transmission Networks State Estimators," submitted for review, *Electric Power System Research* (Elsevier), April 2012.

[12] M. Farrokhabadi, and L. Vanfretti, "Phasor Assisted Automated Topology Processing for State Estimators," submitted for review, *IEEE Transactions on Power Systems*, April 2012.