

Fast Minimal Cutset Evaluation in Cyclic Undirected Graphs for Power Transmission Systems

A. Gaun, H. Renner, G. Rechberger

Abstract-- The evaluation of network reliability is an important topic in the planning, design, and operation of power systems. The aim of this publication is to investigate and tune up the speed of calculating 2-terminal minimal cutsets (MC). Two fast and well known algorithms are compared with a new developed MC algorithm concerning the speed of computation. The new designed algorithm is based on a novel graph reduction method, and on an adapted recursive merge method. Eleven benchmark-networks are used to analyze all three MC algorithms. Experimental results show that the new developed MC algorithm has a linear dependency between the computation time and the graph density of a network for a fixed number of nodes. Furthermore it is shown that the proposed algorithm is faster than the minimal path based algorithms and the currently best available MC algorithm for 2-terminal reliability in complex power transmission networks. A representative 57 node power transmission network demonstrates that the new proposed algorithm reduces the computation time for all relevant MC by 96.2 %.

Index Terms-- 2-terminal network reliability, computation time, graph reduction, induced cycle, minimal cutset, power transmission system, recursive merge;

I. ACRONYM, NOTATIONS, NOMENCLATURE AND ASSUMPTIONS

A. Acronym

CI	cycle-incidence matrix
MC	minimal cut(s) / minimal cutset(s)
MP	minimal path(s) / minimal path set(s)
s	specified source node
t	specified sink node

B. Notations

$G(V, E)$ A connected network (graph) G with the node set $V = \{s, t, x_0, x_1, \dots, x_{k-2}\}$ and the edge set E [1]. For example, Fig. 1 center part is a network. This paper treats only planar cyclic undirected networks.

$e_{x_1x_2}$ $e_{x_1x_2} \in E$ is an arc between nodes x_1 and x_2 .

$ \cdot $	The number of elements of \cdot , e.g., $ V $ is the number of nodes in V .
P	A non-empty graph $P = (V, E)$ with $V = \{x_0, x_1, \dots, x_k\}$ and $E = \{e_{x_1x_2}, e_{x_2x_3}, \dots, e_{x_{k-1}x_k}\}$ [2]. P is called a path.
C	If $P = x_0 \dots x_{k-1}$ is a path and $k \geq 3$, then the graph $C = P + e_{x_{k-1}x_0}$ is a cycle [2].
SS	A connected source set of nodes with $s \in SS$ [3].
$G * x_1$	Node x_1 is merged into SS of G by eliminating any edge connecting x_1 and SS [3].

C. Nomenclature

unavailability	Is defined as $\lambda_{MC} / (\lambda_{MC} + \mu_{MC})$ in h / year with the failure rate λ_{MC} and the repair rate μ_{MC} for each minimal cut of the MC [4], [5].
cut order	The cut order is defined as the number of elements that must fail, to disconnect the sink t and the source s . In this paper written as $o(MC)$.
node degree	The node degree is the number of all edges that touch the node, written as $d(x_1)$; $\delta(V)$ is the minimum degree of V with $\delta(V) := \min\{d(x_i) \mid \forall x_i \in V\}$ [2].
adjacent node	A node x_1 is said to be adjacent to a subgraph of G if there exists an undirected edge $e_{x_1x_2}$ leading from this subgraph to the node x_1 [3]. If G contains G' ($G' \subseteq G$), G' is a subgraph of G .
redundant node	A redundant node is a node which is adjacent to SS and has no P to t without going through any node in SS [3]. For example see node 10 in Fig. 4 Step 4. A redundant loop contains only redundant nodes (see Fig. 4 Step 4 loop 18-8-13-18).
MC	A 2-terminal cutset is a set of arcs such that, by removing these arcs, there is no P in G from node s to node t . A MC is a cutset with no subset of it is a cut. For example, see Fig. 4 Step 4 MC1. MC are also expressed in terms of node sets [6].

A. Gaun is with the Institute of Electrical Power Systems, Graz University of Technology, A-8010 Graz (e-mail: alexander.gaun@tugraz.at).

H. Renner is with the Institute of Electrical Power Systems, Graz University of Technology, A-8010 Graz (e-mail: herwig.renner@tugraz.at).

G. Rechberger is with the Institute of Electrical Power Systems, Graz, University of Technology, A-8010 Graz (e-mail: georg.rechberger@tugraz.at).

MP A path P between s and t is minimal, if no node or branch is traversed more than once in this path.

induced cycles An induced cycle C in G is a cycle in G forming an induced subgraph that has no chords. A chord of a C is an edge that joins two nodes of C but is not itself an edge of C [2]. For example see chordless cycles I-VI Fig. 4 Step 2. A graph consists of $|E| - |V| + 1$ induced (chordless) cycles.

D. Assumption

The networks satisfy following assumptions [6], [7]:

- 1) Perfectly reliable nodes.
- 2) Connected planar cyclic undirected graph with no parallel branches.
- 3) Each edge is either in working or failed state with known probability.
- 4) All power flows in the network obey the conservation law.
- 5) All nodes have to be numbered in increasing order starting with s and ending with t .

II. INTRODUCTION

POWER transmission network reliability is an important topic in the planning, design, and operation of power systems. Several algorithms for evaluation of terminal reliability evaluation are proposed and classified in the literature [8]. Among them the MC approach is one of the most popular used techniques to evaluate 2-terminal network reliability [1], [9]. MC provide a list of events that cause network failures, a disconnection between the 2-terminals s and t , and therefore MC are preferred to calculate the network reliability. The determination of MC or MP is necessary to reduce the sum of disjoint product terms and, hence, the overall reliability computation time. For some networks it is simply impractical to enumerate all MP [3]. For instance, a 2×49 lattice network (see Fig. 7 (11)) contains 2^{49} paths. Although it has a huge number of MP, it only contains 2500 MC. Hence, the computation of all MC and the reliability of a network is a time intensive operation and can grow exponentially with the number of nodes (NP-hard), even if no MP are calculated [6], [10], [11]. The aim of this publication is to investigate and tune up the speed of calculating MC for power transmission networks.

Usually elements of power transmission networks have a very small unavailability. It is proven, that in this case MC up to order plus one of the lowest cut order of the investigated network have to be considered [5], [12]. In power transmission networks in the majority of cases the lowest MC order of a network is equal to one, if substations are considered [12], or two. Thus it is sufficient in the majority of the cases to calculate MC up to the 2nd order for reliability analysis, especially in the context with transmission grid reliability optimisation, e.g. with Genetic Algorithms [13] where hundreds and thousands of these operations have to be done in large networks. In this paper three 2-terminal reliability algorithms are benchmarked with eleven planar

cyclic undirected benchmark-networks [3], [14] (see Fig. 7). Two of the algorithms are well known from the literature [6], [15] and one is a new proposed algorithm that is based on a novel network reduction method and on an adapted recursive merge algorithm.

This paper is organized as follows: in Section III a short overview of related researches is given and a short description of the used algorithms is provided. Section IV contains a detailed report about the network reduction, including the relevant theorem to show the accurateness of the Algorithm and the extended Algorithm itself. The third main part in Section IV deals with a representative example. Section V contains the benchmark of the three different networks with remarks to the computation time and the performance with different sizes of networks. With a 57 node sample network the effectiveness of the new proposed algorithm is demonstrated in computation all MC. Concluding remarks are presented in Section VI.

III. DESCRIPTION OF THE USED ALGORITHMS

An overview of related researches in determining (all) MC in graphs up to the year 2003 can be found in [6]. Within the last five years a new approach to calculate MP has been published in [7]. This algorithm requires fewer calculations to generate MP and is more effective in generating MP without duplicates and unfeasible MP [7]. Although this algorithm provides very good calculation results it is not used in this paper for the MP estimation, due to a higher calculation time for MP compared to the implemented MP-algorithm. This disadvantage is caused by the implementation in MATLAB and not by the algorithm itself. Further work has been done on the improved search for all MC in modified networks [16], which is useful if modifications on networks are performed in planning processes, reinforcement evaluation and network expansion.

Reference [6] deals with the authors current best known algorithm for the MC problem between all node pairs and between two special nodes. This algorithm has a time complexity $O(|V| \cdot 2^{|V|})$ for the MC problem between two special nodes and also all node pairs [6]. It is based on some simple intuitive theorems that characterize the structure of the MC using a node set and it can only be used for undirected graphs. This algorithm is easy to understand and to implement and it has the advantage that it can calculate all MC within reasonable time [6]. This algorithm has the disadvantage that it cannot deal with graphs, where the minimal node degree $\delta(V)$, except for the source node s and the sink node t , is equal to one. The edge adjacent to these nodes with node degree $d(x_1) = 1$ can be removed from the network in a preprocessing step, without losing any MC, before starting the MC estimation. Another disadvantage of the algorithm can be seen in the fact that it can not deal with self loops [6]. For the further investigations in this paper this algorithm is called Alg. A. Alg. A can also be used to check the accuracy of the other two implemented algorithms.

Alg. B [15] is based on MP and can therefore be utilized with directed and undirected graphs. It has an exponential worst time complexity in the number of minimal paths [10]. This estimation of all MP is NP-hard [11] and requires high

memory demand. The MP-algorithm in this paper is very simple. Firstly the algorithm estimates all paths from the source with a specified path length with a breadth first search approach. Since every connected graph $G(V, E)$ contains at least one path of length $\min\{2\delta(V), |E|-1\}$ [2], the specified path length is $|E|-1$ in the first step so that at least all edges are explored once. Secondly all paths with the sink as endpoint are chosen as MP. Once all MP are deduced an incidence matrix is constructed ($|MP| \times |E|$). Columns that have all non-zero entries are MC of order one and must be eliminated from the incidence matrix. In the 2nd step all combinations of two columns are compared to find columns with non-zero entries. This MC are cuts of the 2nd order. To find higher order cuts the 2nd step has to be repeated with combinations of 3, 4, 5, ... columns and any cuts of lower order are eliminated [4]. Alg. B has the advantage that it can find MC up to a specified order. This reduces the computation time and Alg. B can deal with graphs having nodes with degree equal to one, due to the fact that they are ignored by the MP. Alg. B has the disadvantage that calculating all MP is simply impractical for special types of networks (e.g. Fig. 7 (11)), although a wide range of transmission power grids can be evaluated with this method [10], [15].

Alg. C, which is based on [3], is a new recursive algorithm that uses a merge approach and a novel intuitive graph reduction in a special preprocessing step to evaluate MC up to the 2nd order. This algorithm combines the major advantages of Alg. A and Alg. B. Experimental results in [3] show that, this algorithm has a linear running time for different graph densities with a given number of nodes and an exponential running time with different numbers of nodes. The MC can be evaluated up to the 2nd order without calculating the MP. The new designed algorithm impresses by a very high computation speed, even higher than Alg. B and it is not limited by the network size and structure, as it would be, if a MP approach would be used for the MC estimation.

IV. THE NEW PROPOSED ALGORITHM ALG. C

A. Network reduction

To calculate the reliability of power transmission networks, due to the low probability of occurrence, MC of $o(MC) \geq 3$ are not incorporated. Thus a novel intuitive method is proposed to reduce the network graph to estimate first and second order MC. This method is fundamentally based on the following lemmas and theorem.

Lemma 1: The order of a MC is at least equal or greater three, provided that an edge $e_{x_1x_2}$ of an induced cycle is a member of the MC.

Proof: Consider a simple cycle appears with three nodes $x_1, x_2,$ and x_3 and three edges $e_{x_1x_2}, e_{x_2x_3}$ and $e_{x_3x_1}$. Every MC between e.g. $s = x_1$ and $t = x_3$ in this simple cycle has $o(MC) = 2$ (see Fig. 1, left part). Consider a second simple cycle with three nodes $x_4, x_2,$ and x_3 and three edges $e_{x_4x_2}, e_{x_2x_3}$ and $e_{x_3x_4}$. Now, with $s = x_1$ and $t = x_4$ those two MC ($\{e_{x_1x_2}, e_{x_2x_3}\}$ and $\{e_{x_1x_2}, e_{x_3x_4}\}$), that

include the edge $e_{x_2x_3}$ have $o(MC) = 3$ and the other two MC have $o(MC) = 2$ (see Fig. 1, center part). Since every other connected cyclic network can be reduced to such kind of network with a merging approach (see Algorithm in [17]), it is obvious that the degree is equal or greater three. In other words this means every MC that cuts an edge of two chordless cycles has at least twice the minimal degree of a cut of a simple cycle minus one, which is equal or greater three. \square

To consider the special case if s or t is a member of two or more induced cycles following Lemma 2 is defined.

Lemma 2: All edges touching 2-connected ($d(x_i) \leq 2$) nodes x_i , with $x_i \neq s \vee t$ and x_i is exclusively connected over 2-connected nodes with $s \vee t$ or adjacent to $s \vee t$, of a chordless cycle C , with $s \vee t \in C, x_i \in C$ and $d(s \vee t) = 2$, are excepted from the network reduction subroutine. If $d(s \vee t) = 1$ and there exists $x_{s,t}$, with $d(x_{s,t}) = 3$, where s or t and $x_{s,t}$ is connected over exclusively one P , then all edges touching 2-connected x_i or $x_{s,t}$, with x_i is exclusively connected over 2-connected nodes with $x_{s,t}$ or adjacent to $x_{s,t}$, of an induced cycle C , with $x_i \wedge x_{s,t} \in C$, are also excepted from the network reduction. For example see the single-connected node t in Fig. 7 (9), where all edges between to the two dashed line nodes and the before mentioned t have to be considered for MC with $o(MC) \leq 2$.

Since single-connected nodes are never a member of an induced cycle and therefore are not considered by the network reduction, the following important theorem derives from Lemma 1 and 2.

Theorem 1: Each edge $e_{x_2x_3}$ that is an edge of two different induced cycles C and C' , and not an edge of Lemma 2, can be eliminated by merging together node x_2 and x_3 into node x_3 without deleting any MC with $o(MC) \leq 2$ (see Fig.1 right part).

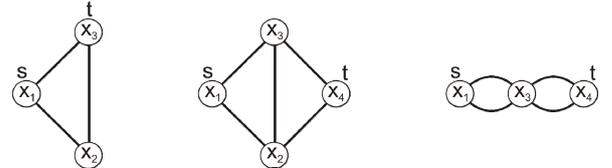


Fig. 1. An example network to explain the network reduction algorithm with two simple (chordless) cycles.

B. Detailed description of the algorithm

The algorithm consists of four main subroutines. The first subroutine deletes all nodes x_i from the graph with $d(x_i) < 3$ excluding node s and t and stores the information of deleted edges $e_{x_i x_k}$ and the belonging new edges. The second subroutine calculates all chordless cycles of the reduced graph. This is done with a shortest path approach based on the following proposition [2].

Proposition 1: Every G contains a P of length $\delta(G)$ and C of length at least $\delta(G) + 1$ (provided that $\delta(G) \geq 2$)

Proof: The longest P in G is $x_0 \dots x_k$. Then all the neighbours of x_k lie on P . Hence $k \geq d(x_k) \geq \delta(G)$. If $i < k$ is minimal with $e_{x_i x_k} \in E$, then $x_i \dots x_k x_i$ is a C with length

$C \geq \delta(G) + 1$ [2]. \square

The shortest paths are calculated with the well known algorithm of Dijkstra [18]. Afterwards the cycles are extended with the information of the deleted edges of subroutine one. Hence the original network is reconstructed and the gained information of chordless cycles can be used to generate a cycle-incidence matrix $CI = (|E| \times |C|)$.

In subroutine three the network is reduced with Theorem 1. Each node of an edge, that has a least two non-zero entries in the edge-row of CI , is merged into the graph in that way, that one of two nodes is replaced by the other one and the touching edge is deleted. Thus the original network is reduced to a graph that contains nodes with $d(x_i) > 3$ and nodes with $d(x_i) \leq 3$. The earlier group is also connected to one or more loops that may contain nodes or be a self-loop, see Fig. 4 Step 4. This reduced graph is thus analyzed with the extended algorithm of [3] in subroutine four.

The extended recursive merge algorithm is based on the obvious idea that s is prevented from arriving at t , if all edges emitting from s are deleted. If one of these edges is not deleted, then s has a MP to t since G is connected and these edges are a MC of G . The basic idea is to build a SS – adjacent nodes to s are merged one by one into this set –, where all edges emitting from this SS are a MC. Furthermore redundant nodes are also considered by merging this kind of nodes into SS before a MC is calculated.

Due to the fact, that the network reduction creates also redundant nodes (loops) that are not relevant for the MC estimation, all self loops are eliminated. To verify that this algorithm enumerates all first and second order cuts correct, consider Lemma 3 and 4.

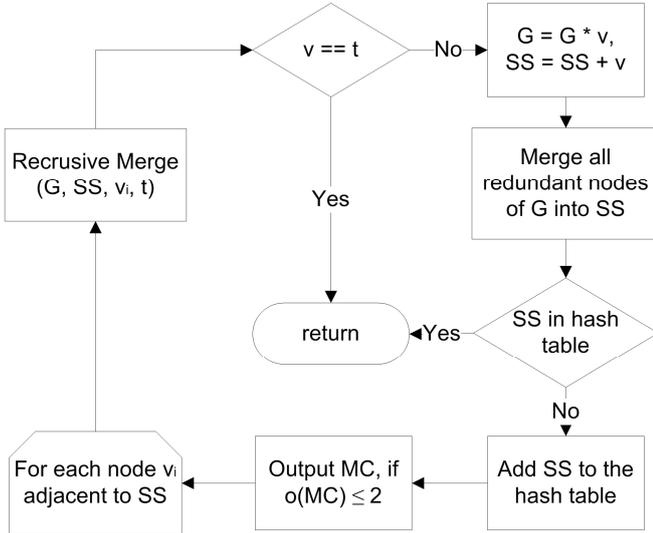


Fig. 2. Flowchart of subroutine four; the extended recursive merge algorithm based on [3].

Lemma 3: Both, the network reduction and the extended Alg. C do not produce isolated nodes in any subroutine.

Proof: In the network reduction subroutine, which is the only subroutine in the pre-processing steps that eliminates edges and nodes, only nodes are merged together. Since all

nodes are connected and the merging-process does not disconnect nodes, no isolated nodes occur (see Lemma 1 in [3]). \square

Lemma 4: Any first and second order MC generated by Alg. C can also be produced by the Algorithm of [3].

Proof: Since the Algorithm of [3], which itself is verified on accuracy by the algorithm in [17], is except two special requests concerning the order of MC and redundant loops, exactly the same as Alg. C, and since edges of a MC with $o(MC) = 1$ are not eliminated by the network reduction, Alg. C evaluates all MC correct up to the 2nd order. \square

In Fig. 2 the flowchart of the extended recursive merge Alg. C is depicted.

C. An Example

A moderate size network is chosen to demonstrate the methodology of the new proposed algorithm. This simple network is evaluated with the algorithm to find the MC between s and t . Not all steps are depicted detailed. Those steps that eliminate more than one edge are a series of single edge deleting and node replacing processes.

Consider the initial network G in Fig. 4 Step 0. After deleting all nodes x_i with degree $d(x_i) < 3$, the graph can be displayed with Fig. 4 Step 1. The edge-deleting-subroutine is necessary to estimate all induced cycles with the shortest path approach. This approach is based on the idea (see Proposition 1), that the shortest path between node e.g. 1 and 2, if edge (1) is deleted, includes edge (2) and (3) (see Fig. 7 (7)). This operation is repeated until all $|E| - |V| + 1$ induced cycles are found. CI of the example is shown in Fig. 3.

	<i>I</i>	<i>II</i>	<i>III</i>	<i>IV</i>	<i>V</i>	<i>VI</i>
(1)	1	0	0	0	0	0
(2)	1	0	0	0	0	0
(3)	1	1	0	0	0	0
(4)	0	1	1	0	0	0
(5)	0	1	0	0	0	0
(6)	0	1	1	0	0	0
(7)	0	0	1	1	0	0
(8)	0	0	1	1	0	0
(9)	0	0	1	0	0	0
(10)	0	0	0	1	1	0
(11)	0	0	0	1	0	0
(12)	0	0	0	0	1	0
(13)	0	0	0	1	0	0
(14)	0	0	0	1	1	0
(15)	0	0	0	1	0	1
(16)	0	0	0	0	1	0
(17)	0	0	0	0	1	1
(18)	0	0	0	0	0	1
(19)	0	0	0	0	1	1
(20)	0	0	0	0	0	1
(21)	0	0	0	0	1	0
(22)	0	0	0	0	1	1
(23)	0	0	0	0	0	1
(24)	0	0	0	0	0	1
(25)	0	0	0	0	0	1
(26)	0	0	0	0	0	1

Fig. 3. The CI of the example with 26 nodes (first column) and 6 induced cycles (upper row). The deleted edges in subroutine three are labeled bold.

In the next subroutine the graph is reduced as schematically

depicted in Fig. 4 Step 3.1 to Step 3.6. Step 3.0 in Fig. 4 is the initial state. The algorithm in subroutine three firstly replaces node 2 by node 3 and eliminates the edge (3). Next node 3 is replaced by node 5 and the edges (4) and (6) are deleted as well as node 4, which is not depicted in Fig. 4. This node is the offspring of edge (4) and (6) and is therefore replaced by node 5 (Theorem 1).

With this operation, a self-loop of node 5 with edge (5) is created. It is obvious, that this self-loop is never a member of a MC due to the definition of MC. In Step 3.3 node 5 is replaced by node 7 and edges (7) and (8) are eliminated. In this step the second self-loop at node 7 with edge (9) appears. Step 3.4 shows how node 7 is replaced by the adjacent node 11 and edges (10) and (14) are deleted. The following Step 3.5 merges node 12 into node 11. This creates a new graph without edge (15) and a third loop with node 10 (see Step 4 in Fig 2.) and edges (11) and (13). As in Step 3.2 it is obvious that the edges of this loop are never a member of a MC because they are not on a path from s to t . Step 3.6 finishes the network reduction subroutine three by merging node 18 into node 12 and eliminating edges (17), (19) and (22). Subroutine four (for results see Step 4 in Fig. 4) operates recursive (see Fig. 2 and [3]). The main difference between the algorithm in [3] and its extended version used here is that the extended version in subroutine four can deal with self-loops. Redundant

loops, as the loop depicted in Fig. 4 Step 4 with nodes 18-13-18, are merged into SS in one step and not one by one as in [3] before creating a new MC. For example consider node 8 in Fig. 4 Step 4. If $SS = \{1, 18\}$, node 8 is adjacent to SS and on a redundant loop. The touching edges are never a MC for the connection between s and t , respectively node 1 and 21. Therefore $G * 8$ and $G * 13$ in one step, without creating a MC. Both Alg. B and Alg. C provide the same results for the graph example: six 2nd order MC MC 1 ... MC 6 with the edges (see Fig. 4 Step 4) $\{(1), (2)\}$, $\{(25), (18)\}$, $\{(25), (20)\}$, $\{(25), (23)\}$, $\{(25), (24)\}$ and $\{(25), (26)\}$.

V. BENCHMARK OF THREE DIFFERENT ALGORITHMS

A. Processing time of Alg. C.

The new proposed algorithm was tested with randomly generated planar cyclic undirected grid-graphs G with fixed numbers of nodes. The time to enumerate all MC was measured and Fig. 5 depicts the results of running time versus graph density, which is defined as $2 \cdot |E| / (|V| \cdot (|V| - 1))$. The number of nodes ranges from 24 to 40 at a step size of 4 and the graph density ranges from 0.1 to the maximal size of planarity at a step size of 0.005. The maximal size of the planar graphs was calculated with [20]. The minimal size results from the constraint that the graph has to be connected

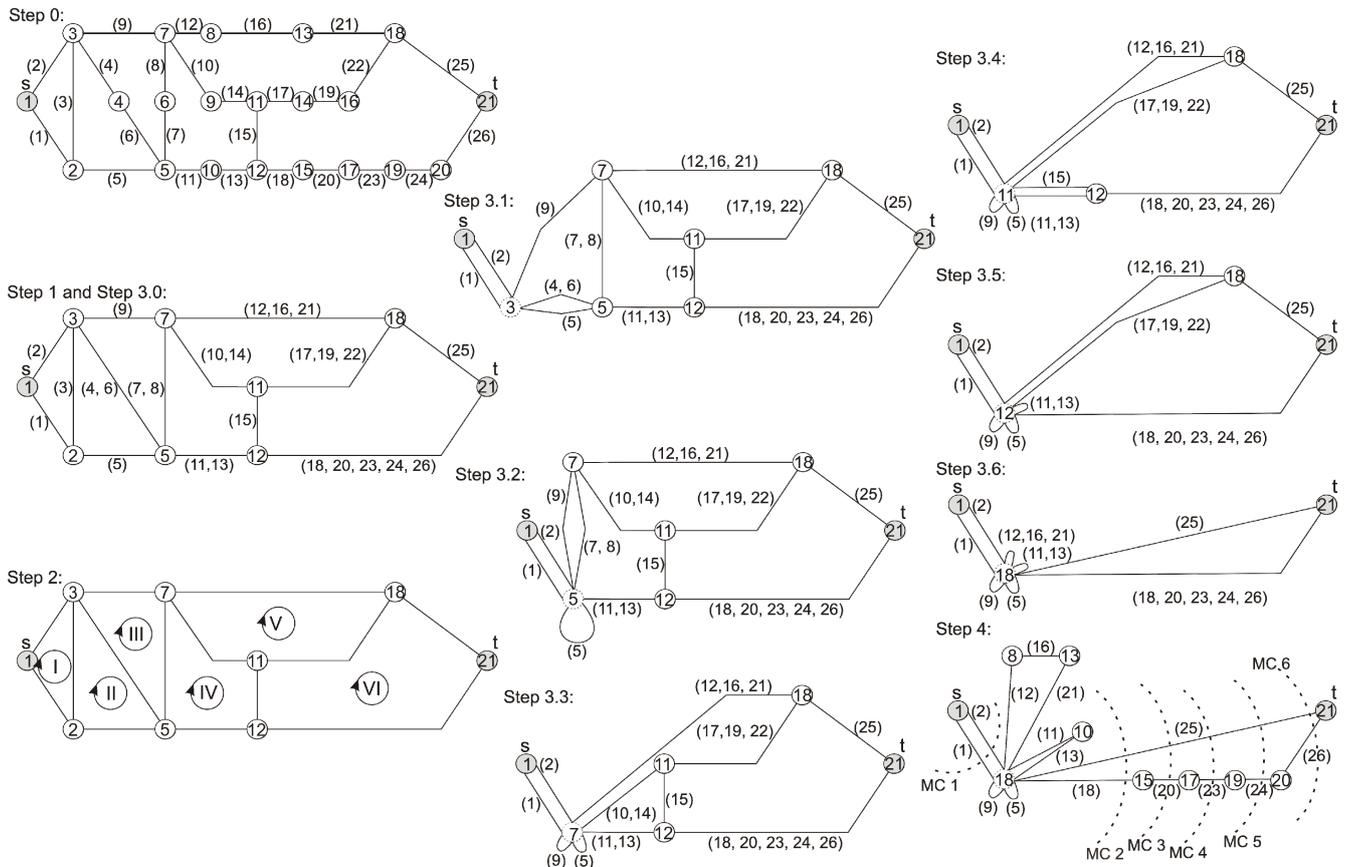


Fig. 4. Example network with source s and sink t from [3]; Step 0: Initial network; Step 1: result after subroutine 1; Step 2: result after subroutine 3; Step 3 and Step 4: The different steps of the network reduction subroutine three and the six second order MC (see dotted curves in the figure with title Step 4), when the extended Algorithm of subroutine four (Alg. C) is applied.

without any intersections. This is guaranteed for the considered range with a graph density exceeding 0.1. To test either the randomly generated graph is planar or not algorithm form [20], [21] was used.

Due to the stochastic behavior concerning the number of, the induced cycles of the random graphs, the induced cycle depended computation time versus the graph density was fitted with a robust fit regression. This regression mode is used, if outliers have to be considered with lower importance, which is appropriate in this case. The results of this regression for the considered random networks are also depicted in Fig. 5. It is evident that there is a linear relationship between the graph density and the computation time for the 2-terminal MC. Furthermore the computation time for the 2-terminal MC per fixed number of nodes is increasing with the graph density. This gradient of the linear function is unequal for different number of nodes which is already pointed out in [3].

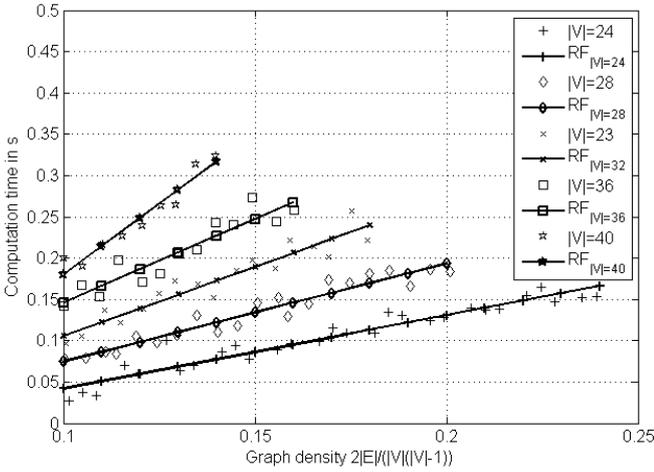


Fig. 5. The computation time versus graph density $2|E|/(|V|*(|V|-1))$ with fixed number of nodes. RF ... Robust Fit regression for the relevant range of planar cyclic undirected networks

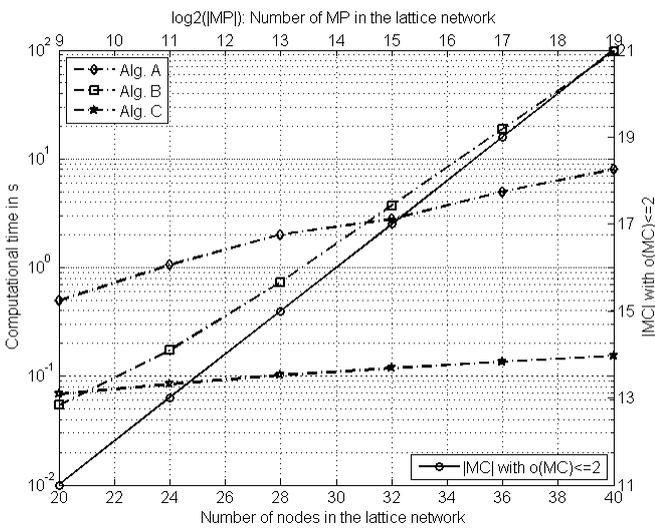


Fig. 6. The computation time for 2-terminal MC versus the increasing node number of lattice networks is shown for all three used algorithms. Additional, in the figure the number of 2nd order MC and the number of MP is depicted in the left y-axis and upper y-axis respectively.

In Fig. 6, six lattice networks (for example see Fig. 7 (11)), ranging from 20 nodes to 40 nodes in steps of 4 nodes are compared concerning the computation time. Alg. A, Alg. B and Alg. C are used to show the exponential computation time behaviour of the problem. All three algorithms were implemented in MATLAB on an Intel® Core™2 Duo CPU with 3.33 GHz.

It's worthless noting, that Alg. C is the fastest algorithm of all three presented ones, if the network exceeds a certain size. Alg. A, which enumerates all MC is the second best algorithm in terms of computation time, if lattice networks with more than 32 nodes are considered. Due to the huge computational effort to calculate the MP, e.g. the lattice network with 32 nodes has $2^{15} = 32768$ MP, Alg. B is the slowest for lattice networks with more than 32 nodes, even if MC up to the 2nd order are estimated.

B. Benchmark with test grids

The intermeshing degree v in Table I is defined as $|E|/|V|$. In power networks the intermeshing degree is normally between 1 and 2, in power transmission networks, which are focussed in this paper, it is between 1 and 1.5 [19]. Depending on the number of nodes, for example the before mentioned range for power transmission network ranges around a graph density of 0.1 for a network with 30 nodes.

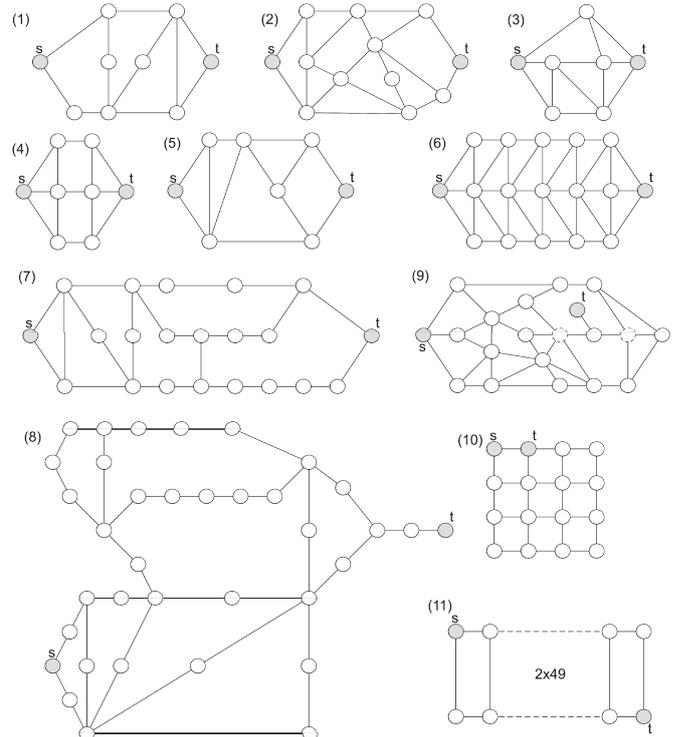


Fig. 7. Eleven planar Benchmark-networks with source node s and sink node t based on [3], [14]; Network (7) is used for the example to explain Alg. C. Network (9) is used for the example in Lemma 2

In Table I the normalized computation time, related to the respective fastest algorithm based on the best value out of ten calculations, for each benchmark-network in Fig. 7 is presented. The faster one of the two compared has a 1.00

TABLE I
NETWORK INFORMATION TABLE AND RESULTS OF THE BENCHMARK

Network	Intermeshing degree v	1 st order Cuts	2 nd order Cuts	3 rd order Cuts	Unavailability ^a	Normalized computation time Alg. A vs. Alg. B ^b	Normalized computation time Alg. B vs. Alg. C		
${}_{9}^{12}G(1)_{28}^{13}$	1.33	0	3	7	$3.43 \cdot 10^{-8}$	1.98	1.00	1.00	4.24
${}_{12}^{20}G(2)_{128}^{150}$	1.66	0	2	3	$2.28 \cdot 10^{-8}$	2.42	1.00	1.00	2.18
${}_{7}^{12}G(3)_{20}^{25}$	1.74	0	0	2	$2.61 \cdot 10^{-14}$	1.02	1.00	1.00	4.14
${}_{8}^{13}G(4)_{29}^{29}$	1.63	0	0	3	$3.91 \cdot 10^{-14}$	1.43	1.00	1.00	3.35
${}_{8}^{12}G(5)_{19}^{24}$	1.50	0	2	5	$2.28 \cdot 10^{-8}$	1.13	1.00	1.00	3.62
${}_{17}^{36}G(6)_{560}^{22401}$	2.12	0	0	2	$2.61 \cdot 10^{-14}$	1.00	3.11	19.44	1.00
${}_{21}^{26}G(7)_{328}^{44}$	1.24	0	6	66	$6.85 \cdot 10^{-8}$	13.50	1.00	1.00	1.76
${}_{36}^{43}G(8)_{2666}^{56}$	1.19	2	8	16	0.02	39.77	1.00	1.00	1.08
${}_{20}^{35}G(9)_{2545}^{4008}$	1.75	1	1	2	0.01	8.80	1.00	4.66	1.00
${}_{16}^{24}G(10)_{105}^{98}$	1.50	0	1	2	$1.14 \cdot 10^{-8}$	1.00	3.11	1.00	2.28
${}_{100}^{148}G(11)_{2500}^{299}$	1.48	0	51	98	$5.82 \cdot 10^{-7}$	1.00	*	*	1.00

^c $G(z)_c^p$ Benchmark-network number z with v nodes, e edges, p MP and c MC.

* Due to memory problems no calculation could be performed.

^a In h / year; it is the same if all MC up to the highest order are considered or if MC up to the 2nd order are considered, $\lambda = 0.01$ 1 / year and $\mu = 1$ h.

^b The normalized computation time, if Alg. B evaluates all MC up to the 3rd order and Alg. A evaluates all MC.

entry in the table. Alg. B considers MC up to the 3rd order for comparison with Alg. A and up to the 2nd order for comparison with Alg. C (see Table I). Alg. C determines all MC up to the 2nd order. As a result of the investigation it can be pointed out, that in the case of an intermeshing degree v exceeding 2, the Alg. A is better for the MC calculation from the point of view of computation time compared to Alg. B if 3rd order cuts or higher are analyzed. If Alg. B for 1st and 2nd order MC is compared to Alg. C, it can be seen, that for all sample graphs except graph (6), (9) and (11), Alg. B has lower computational time. Alg. C is the fastest for these graphs that have a high number of MP, which is increasing with the number of nodes and branches, respectively with the graph density. The system unavailability in Table I is calculated with the frequency and duration approach [4], [5]. The failure rate λ of all components is 0.01 1 / year and the repair rate μ of all components is 1 h. The results in Table I show that only 1st order cuts have a major impact in the reliability evaluation if λ and μ are considered.

C. Computation of all MC

With the network in Fig. 8, that is more representative for a real world power transmission network, the benefit of the new proposed algorithm is shown when all MC up to the 2nd order of a network have to be calculated.

The network in Fig. 8 is from [14] and has 57 nodes and 78 edges. It consists of five sources, 52 loads, 710 2nd order MC and five 1st order MC. The network has an intermeshing degree of 1.37, a graph density of 0.049 and 11497979 MP. All MC up to the 2nd order are evaluated with Alg. B and Alg. C. Alg. B needs 5242.1 seconds and Alg. C needs 198.7 seconds. This is a saving in computation time of more than 5043.3 seconds or in other terms, Alg. C needs only 3.8 % of the computation time of Alg. B.

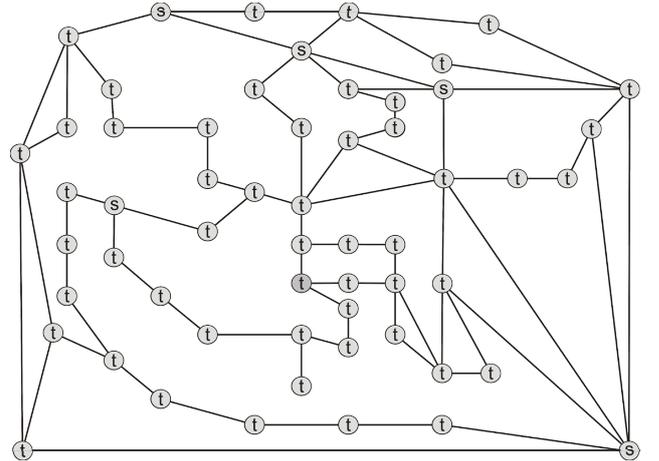


Fig. 8. Test network for the computation of all MC up to the 2nd order based on [14] with 57 nodes (five sources and 52 loads).

VI. CONCLUSION

It is sufficient to calculate MC up to the 2nd order for 2-terminal power system transmission reliability analyses, especially in the context with transmission grid reliability optimisation with Genetic Algorithms, where hundreds and thousands of these operations have to be done. Since the estimation of reliability in transmission grids is NP-hard, research interest focuses on fast MC algorithms to reduce the computation time to a minimum. The proposed algorithm, which is based on a novel intuitive network reduction and on a recursive merge approach, calculates all MC up to the 2nd order in satisfying time and even faster as the currently best know algorithm. Depending on the number of nodes respectively on the graph density, it is also faster as a simple, but for small networks powerful, MP algorithm that generates all MC up to the 2nd order. Furthermore with a 57 node test network it is demonstrated that the new proposed algorithm

can calculate all MC up to the 2nd order. In terms of computation time this is a reduction by 96.2 % compared to the MP algorithm.

VII. REFERENCES

- [1] W.-C. Yeh, "Search for all MCs in networks with unreliable nodes and arcs," *Journal of Reliability Engineering and System Safety* 79 (2003), pp. 95-101.
- [2] R. Diestel, "Graph Theory," Electronic Edition 2000. New York: Springer, 2000, pp. 4-166.
- [3] H.-Y. Lin, S.-Y. Kuo, and F.-M. Yeh, "Minimal Cutset Enumeration and Network Reliability Evaluation by Recursive Merge and BDD," 2003, in *Proc. of the Eighth IEEE International Symposium on Computers and Communication (ISCC'03)*, 6 pages.
- [4] R. Billinton, and R. N. Allan, "Reliability Evaluation of Engineering Systems, Concepts and Techniques," 2nd ed., New York: Plenum Press, 1992.
- [5] R. Billinton, and R. N. Allan, "Reliability Evaluation of Power Systems," 2nd ed., New York: Plenum Press, 1996.
- [6] W.-C. Yeh, "A simple algorithm to search for all MCs in networks," *European Journal of Operational Research* 174 (2006), pp. 1694-1705
- [7] W.-C. Yeh, "A Simple Heuristic Algorithm for Generating All Minimal Paths," *IEEE Trans. On Reliability*, vol. 56, No. 3, pp. 488-494, September 2007.
- [8] S. Soh, and S. Rai, "Experimental Results on Preprocessing of Path/Cut Terms in Sum of Disjoint Products Technique," *IEEE Trans. On Reliability*, vol. 42, No. 1, pp. 488-494, March 1993.
- [9] M. Fotuhi-Firuzabad, R. Billinton, T.S. Munian, and B. Vinayagam, "A Novel Approach to Determine Minimal Tie-Sets of Complex Network," *IEEE Trans. On Reliability*, vol. 53, No. 1, pp. 61-70, March 2004.
- [10] D. R. Shier and D. E. Whited, "Iterative Algorithms for Generating Minimal Cutsets in Directed Graphs," *Networks*, Vol. 16, (1986), pp.133-147.
- [11] M. O. Ball, "Computational Complexity of Network Reliability Analysis: An Overview," *IEEE Trans. On Reliability*, vol. R-35, No. 3, pp. 230-239, August 1986.
- [12] H.-J. Haubrich, "Zuverlässigkeitsberechnung von Verteilungsnetzen, Grundlagen – Verfahren – Anwendungen," ABEV Band 36, Verlag der Augustinus Buchhandlung, 1996.
- [13] A. Gaun, H. Renner, and G. Rechberger, "A reliability and environmental based Multi-Criteria Decision Making genetic algorithm (MCDMGA)," in 6th *Power Quality and Supply Reliability Conference, 2008. PQ 2008*, pp. 217-223, Aug 2008.
- [14] A. R. Abdelaziz, "A New Approach for Enumeration Minimal Cut-sets in a Network," 2000, in *Proc. of 7th IEEE International Conference on Electronics, Circuits and Systems, (ICECS 2000)*, pp. 693-696.
- [15] R. N. Allan, R. Billinton, and M. F. De Oliveira, "An Efficient Algorithm for Deducing the Minimal Cuts and Reliability Indices of a General Network Configuration," *IEEE Trans. On Reliability*, vol. R-25, No. 4, pp. 226-233, October 1976.
- [16] W.-C. Yeh, "An improved algorithm for searching all minimal cuts in modified networks," *Reliability Engineering & System Safety*, vol. 93, Issue. 7, pp.1018-1024, July 2008.
- [17] S. Tsukiyama, I. Shirakawa, and H. Ozaki, "An Algorithm to Enumerate All Cutsets of a Graph in Linear Time Per Cutset," *Journal of A.C.M.*, vol. 27, No. 4, pp. 619-632, October 1980.
- [18] K. P. Murphy, "The Bayes Net Toolbox for Matlab," Department of Computer Science, University of California, Berkeley. Available: <http://HTTP.CS.Berkeley.EDU/~murphyk/Papers/bnt.ps.gz>, October 2001.
- [19] D. Oeding, and B. R. Oswald, "Elektrische Kraftwerke und Netze," 6. Auflage, Springer-Verlag Berlin Heidelberg New York, 2004.
- [20] D. Gleich, "MatlabBGL A Matlab Graph Library", Institute for Computational and Mathematical Engineering, Stanford University. Available: http://www.stanford.edu/~dgleich/programs/matlab_bgl/, October 2008.
- [21] J. M. Boyer, and W: J. Myrvold, "On the Cutting Edge: Simplified O(n) Planarity by Edge Addition", *Journal of Graph Algorithms and Applications*, vol. 8, No. 3, pp. 241-273, October 2004.

VIII. BIOGRAPHIES

Alexander Gaun was born in Kufstein, Austria, in 1979. He received his diploma degree in 2005 at Graz University of Technology, Graz. He is currently a scientific assistant at the Institute of Electrical Power Systems. His research interests include electrical power system planning, reliability computation, genetic algorithms and electromagnetic compatibility.

Herwig Renner was born in Graz, Austria, in 1965. He received his doctoral degree in 1995 at Graz University of Technology, Graz, where he currently holds a position as associate professor at the Institute of Electrical Power Systems. His main work in research and teaching is in the field of electrical power system analyses with special emphasis on quality, supply reliability and power system control and stability. In 2007 he was employed as visiting professor at Helsinki University of Technology.

Georg Rechberger was born in Linz, Austria, in 1978. He received his diploma degree in 2005 at Graz University of Technology, Graz, where he currently holds a position as scientific assistant at the Institute of Electrical Power Systems. His main work in research and teaching is in the field of electrical power system planning.