

Efficient Extreme Event Screening for Power Systems Using Constrained and Unbalanced Partitioning

Cristinel Ababei, *Member, IEEE*, and Rajesh Kavasseri, *Member, IEEE*

Abstract—In this paper, we design and implement an efficient algorithm for extreme event screening for power systems based on constrained and unbalanced partitioning. The proposed methodology directly addresses the cutsize and power-imbalance oriented partitioning problem. This problem has two objectives: minimization of the cutsize and maximization of the power-imbalance between partitions. The proposed algorithm uses the well known hMetis partitioner as a core partitioning engine. The main advantages of the proposed screening methodology are scalability, efficiency, and ease of implementation. The average CPU runtime is less than 0.1 s and 1.5 s for systems with 2,383 buses and 43,501 buses, respectively. The quality of the partitioning solution is similar to that achieved by the algorithm studied in [7] for the 30-bus system.

I. INTRODUCTION

MULTIPLE cascading or simultaneous contingencies may result in extreme events in electric power systems such as the 2003 blackout [1]. While the $N - 1$ operation criterion of the power grid is practical in addressing single contingencies, the combinatorial nature of the general case of $N - k$, $k = 2, 3, \dots$, becomes very challenging computationally. Therefore, prediction tools that can be used to efficiently anticipate such events are highly desirable [2] and have recently started to attract a lot of interest. Previous work on this topic can be classified in two major categories. Methods in both categories, simplify the general $N - k$ operation criterion problem in that they restrict the search to scenarios with severe consequences that may arise from relatively few contingencies.

A first category of methods are based on the calculation of the minimum distance from an operating point to the feasibility boundary [3]-[6]. Donde et al. [3], proposed a two-stage method to identify a few transmission lines which, when cut, cause the power flow equations to not have a solution, possibly leading to a failure. In the first screening stage, the system graph is partitioned into subgraphs to identify a preliminary set of lines cut during the partitioning. In the second stage, a subset of these lines - whose outage would result in a loss of feasibility - are isolated for further detailed analysis. Their method considered only active power flow constraints, and was then extended to include reactive power in a mixed integer nonlinear programming formulation [4]. A nonlinear optimization based technique to identify the most critical lines, failure of which can cause severe blackouts was

proposed in [5]. Information on criticality of lines can be used to identify multiple contingencies by selectively exploring multiple combinations of broken lines.

A second category of methods are based on the multiobjective partitioning of the power system such that the cutsize is minimized while the power-imbalance between partitions is maximized [7]-[9]. Lesieutre et al. [7], proposed a simplified graph theoretic technique based on searching pathological partitions (causing severe power imbalance) with the least effort (fewest cuts). This type of partitioning is called the *inhibiting bisection problem* [9] and solved using spectral partitioning based techniques [7],[8] and techniques based on the formulation as a maximum-flow/minimum-cut problem [9].

Algorithms based on nonlinear optimization or spectral partitioning may become computationally expensive especially for large sizes of power systems. For example, the approach in [8] has a worst case computational complexity of $O(MN^3)$ ¹. Spectral graph partitioning is known to be slower than other partitioning approaches [10]. The algorithm proposed in this paper uses hMetis - a multi-level move-based partitioner - which is typically 4-10 times faster than other partitioning schemes [11]. The computational complexity of previous studies represents the motivation of this paper: we implement and study a technique for fast preliminary extreme event screening for power systems based on minimum cutsize and maximum power-imbalance partitioning. The main merits of the proposed algorithm are: (a) scalability, (b) efficiency - the proposed technique provides very short computational runtimes to construct power-imbalance/cutsizes ratio curves on large scale power systems, (c) simplicity - ease of implementation.

We applied the proposed algorithm to power systems with sizes of up to 43,501 buses. The CPU runtime to construct the power-imbalance/cutsizes ratio curve with thirty data points² is 2.41 s for a system with 2,383 buses. The CPU runtime to construct the power-imbalance/cutsizes ratio curve with ten data points for a system with 43,501 buses is 15.12 s. The quality of the partitioning solution is similar to that achieved by the algorithm proposed in [7] for the 30-bus system.

C. Ababei and R. Kavasseri are with the Department of Electrical and Computer Engineering, North Dakota State University, Fargo, ND, 58105, USA e-mail: {cristinel.ababei, rajesh.kavasseri}@ndsu.edu

¹Where M is the number of lines and N is the number of buses in the system.

²Thirty data points represent thirty different runs of the proposed algorithm.

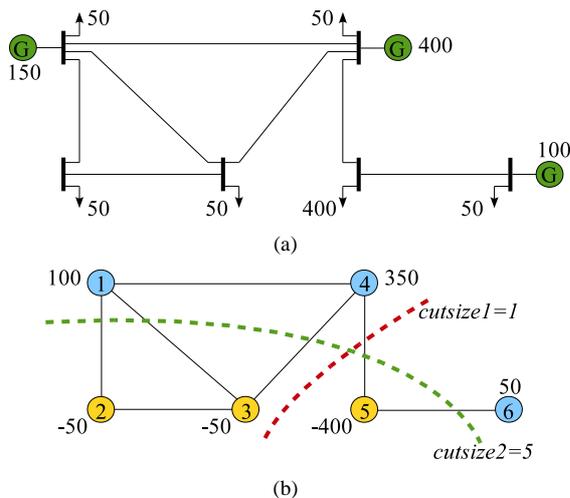


Fig. 1. (a) One line diagram of an example power system with three generators. (b) The associated system graph used for partitioning has six nodes and seven arcs.

II. CUTSIZE AND POWER-IMBALANCE ORIENTED PARTITIONING

In this section, we discuss the problem of partitioning for minimum cutsize and maximum power-imbalance that was introduced in [7].

This problem is defined using a *system graph* $G(V, A)$ (where V and A are the sets of nodes and arcs/edges) associated with the power system under study. For example, the system graph associated with the power system from Fig. 1.a is shown in Fig. 1.b. In this example, we also present on the system graph the equivalent power injections, which are positive for generators and negative for power sinks (or loads).

Typically, graph partitioning is the operation of dividing (or bisecting) the nodes of V into two *balanced* sets V_1, V_2 such that the number of cut arcs³, also referred to as *cutsize*, is minimized. This type of partitioning is called *bi-partitioning*, and it can be generalized to *k-way partitioning* on general graphs that have hyperedges with more terminals. Because an in-depth treatment of the problem of graph partitioning (as a graph theoretical problem) is outside the scope of this paper, the interested reader is referred for details (including spectral partitioning studied in [7]) to [13],[14].

While most commonly the main objective of graph partitioning is the minimization of the cutsize, the minimum cutsize and maximum power-imbalance oriented partitioning addressed in this work is a multiobjective problem, and so with no single best solution. For example, the partitioning of the graph from Fig. 1.b with a *cutsize1=1* has a power-imbalance⁴ of 700, while the partitioning with a *cutsize2=5* has a power-imbalance of 1000. In such situations, the best solution has to be identified based on the relative importance (or weight) of each of the individual objectives. In the partitioning problem addressed in this paper, one can use the ratio

³An arc is cut if it has one of its two terminals in V_1 and the other terminal in V_2 .

⁴Power-imbalance is calculated as the difference between the total cumulated powers of the nodes from the two partitions V_1, V_2 .

between power-imbalance and cutsize to evaluate different partitioning solutions. For example, in the case from Fig. 1.b, the first partitioning offers a ratio of 700/1 that is greater than 1000/5 of the second partitioning. The physical intuition in this example is that the first partitioning is a better indicator of the arcs that work in the most extreme conditions.

Due to the maximum power-imbalance between partitions, the transmission lines that represent the minimum cutsize will carry high power flows. Failure of these critical lines can lead to cascading outages, which can potentially result in a blackout. The existence of reserves can alleviate this problem. This may be possible only if the reserves themselves would not be isolated due to the failed transmission lines. However, the effect of reserves is not modeled in this paper.

III. PROPOSED PARTITIONING METHODOLOGY

A. Preliminaries

In this section we briefly discuss multi-level partitioning that is the main algorithm implemented by hMetis [11] and present the assumptions that we make in order to simplify the problem. The efficiency of the hMetis partitioner is due to its multi-level partitioning technique. This technique is illustrated in Fig. 2. It consists of three phases: coarsening, initial partitioning, and uncoarsening and refinement. During the coarsening phase, a sequence of successively smaller (coarser) graphs is constructed. Then, during the initial partitioning phase, a bisection of the coarsest graph is computed. This is very efficient because of the smaller size of the coarse graph. Finally, during the uncoarsening and refinement phase, the initial bisection is successively projected to the next finer level, and at each level an iterative refinement technique is used to further improve the bisection. It has to be noted that this partitioning approach can be done with the constraint of fixing selected nodes in each of the two partitions. The fixed nodes will act as fixed anchors in each of the two partitions during the coarsening and uncoarsening phases.

In this paper, we assume a lossless power system network similarly to [5],[7]. The main reason for that is to avoid the use of a power flow solver, which may become computationally expensive considering the very large combinatorial space of line outages. Because of this assumption, thermal and voltage limits on transmission lines are not directly considered. The proposed partitioning-based screening technique is intended to be used as a first stage to identify critical lines, which

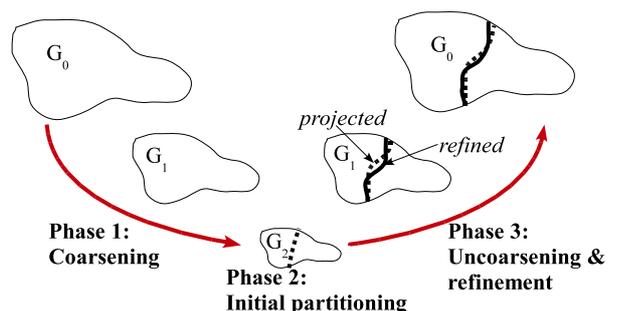


Fig. 2. Illustration of the three-phase multi-level partitioning [11].

Partitioning()

Given: $G(V, A)$ and all power injections and loads for all nodes
Sort and store in S_1 nodes with injections in nonincreasing order of
injected power

Sort and store in S_2 nodes with loads in nonincreasing order of
absolute extracted power

N_{DP} = Number of desired data points or partitionings

for (as many data points as N_{DP}) {

Fix first $k\%$ nodes of S_1 in partition V_1

Fix first $k\%$ nodes of S_2 in partition V_2

Do partitioning using hMetis partitioner without balance constraints

Return: Partitioning solution with maximum *power_imbalance/cutsize ratio*, lines with highest frequency of being cut during the N_{DP} partitionings

Fig. 3. Pseudocode of the proposed partitioning methodology. By default, k is varied between 1-10 in order to collect the default $N_{DP} = 10$ data points. However, the number of desired data points N_{DP} is a parameter that can be specified by the user.

should be then considered for a more detailed analysis in a second stage. The second stage can be more accurate by considering voltage and thermal limits as well, while being computationally efficient because it would focus only on a small subset of lines - identified in the first stage by the proposed technique.

B. Proposed Partitioning Methodology

The proposed minimum cutsize and maximum power-imbalance oriented partitioning methodology uses as a core algorithm the hMetis partitioner, which is publicly available at [12]. In order to generate multiple partitioning solutions we vary the fraction of pre-fixed nodes in each of the two partitions and then use the hMetis partitioner with a totally relaxed *unbalance factor*⁵. These partitioning solutions represent the data points that we use to create the power-imbalance/cutsize ratio curves, which one can use as an indicator to identify the arcs that work in the most extreme conditions (similar to the example of Fig. 1.b). The pseudocode of the proposed partitioning methodology is presented in Fig. 3.

The key aspect of the proposed algorithm is fixing an increasing percentage of nodes from the sorted list S_1 of injection nodes (positive power numbers in Fig. 1.b) in the partition V_1 and from the sorted list S_2 of load nodes (negative power numbers in Fig. 1.b) in the partition V_2 . Because the first nodes that will be fixed during this process are the nodes with maximum power injections (fixed in partition V_1) and the nodes with maximum power loads (fixed in partition V_2), this will lead to maximizing the power-imbalance between partitions in the final partitioning solution. These fixed nodes will act as *fixed anchors* and will not be allowed to move between partitions during the actual partitioning done by hMetis. Moreover, because the unbalance factor of hMetis is completely relaxed, the partitioner will find the unbalanced

⁵The unbalance factor is relaxed so that the resulting partitioning solution is not constrained to be balanced like in the general graph partitioning. The unbalance factor can be set directly using the arguments of the hMetis partitioner.

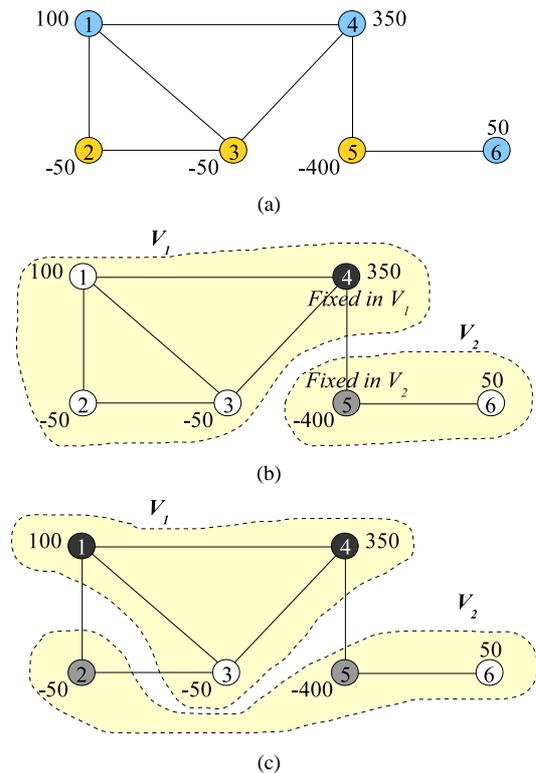


Fig. 4. (a) Initial system graph $G(V, A)$. (b) First partitioning solution obtained in the first iteration of the algorithm shown in Fig. 3: *cutsize1=1*, *power_imbalance1=700*. Node 4 is fixed in V_1 and node 5 is fixed in V_2 before hMetis is called. (c) Second partitioning solution: *cutsize2=3*, *power_imbalance2=800*. Nodes 4,1 are fixed in V_1 and nodes 5,2 are fixed in V_2 before hMetis is called.

partitioning solution that has the minimum cutsize. The output of the proposed methodology consists of complete details on each of the N_{DP} partitionings, the maximum power-imbalance/cutsize ratio, and a list of the lines with highest frequency of being cut. Further details will be discussed in the experimental results section.

For the example from Fig. 1.b, the two sorted lists are: $S_1 = \{350, 100, 50\}$ and $S_2 = \{-400, -50, -50\}$. In this case, the result of the first partitioning in the proposed methodology (the first iteration of the for loop from Fig. 3) is shown in Fig. 4.b. Here, only the node index 4 (Fig. 4.a) is fixed in partition V_1 , while the node index 5 is fixed in partition V_2 . The result of the second partitioning (with nodes 4,1 fixed in V_1 and nodes 5,2 fixed in V_2) is shown in Fig. 4.c.

IV. EXPERIMENTAL RESULTS

We implemented the proposed partitioning methodology in C++ and simulations were performed on a Linux machine running on a 2.8 GHz Intel Quad processor with 2 GB memory.

A. Results

We report results on four testcases with sizes ranging from 57 to 43,501 buses, presented in Table I. The system *bus_43501* is the Eastern interconnect system. The power-imbalance over cutsize ratio is reported as the largest among

TABLE I

RESULTS OBTAINED USING THE PROPOSED PARTITIONING METHODOLOGY. THE POWER-IMBALANCE/CUTSIZE RATIO IS THE MAXIMUM VALUE AMONG ALL TEN DATA POINTS. CUTSIZE IS THE VALUE CORRESPONDING TO THE MAXIMUM POWER-IMBALANCE/CUTSIZE RATIO.

| Testcase | Maximum Power-imbalance/Cutsize ratio | Cutsize | Most critical lines | CPU runtime [s] |
|-----------|---------------------------------------|---------|---------------------|-----------------|
| bus_57 | 307.3 | 3 | 15 12 14 | 0.0511 |
| bus_118 | 283.13 | 3 | 175 132 101 | 0.0992 |
| bus_2383 | 145.05 | 46 | 763 1493 89 | 0.8372 |
| bus_43501 | 11025.85 | 63 | 9840 1969 12639 | 15.12 |

all ten data points collected for each testcase as described in Fig. 3. The cutsize corresponding to this ratio is reported in the third column. The CPU runtime represents the computational runtime to collect all $N_{DP} = 10$ data points for each testcase (i.e., processing runtime for running the proposed algorithm ten times). For example, the CPU runtime for *bus_43501* is 15.12 s, which translates into 1.512 s for a single run of the proposed algorithm. However, the number of data points N_{DP} can also be specified by the user. In particular, if N_{DP} is selected to be sufficiently large, then all possible data points will be collected, until all injection (or generator) nodes will be fixed in partition 1 and all load nodes will be fixed in partition 2. For example, we present in Fig. 5, the power-imbalance/cutsize ratio curve for *bus_2383* using $N_{DP} = 30$ data points that were collected in 2.41 s. We also plot in Fig. 5, the percentage of power imbalance⁶, which has similar variation to that reported in [9].

As mentioned above, we advocate the use of the maximum power-imbalance/cutsize ratio (reported in Table I) as an indicator for the fewest lines that form the cutsize with the largest average power imbalance per line. Additionally, our algorithm counts the number of occurrences of each line (arc in the system graph) in the cut-set of each partitioning solution from all ten⁷ collected data points. Intuitively, if a line has a higher frequency of being cut, then it must have assigned a higher criticality (we define criticality similarly to [5]). Therefore, at the end of the partitioning methodology, the lines that formed the cut-set with the largest power-imbalance/cutsize ratio and the lines with the highest cut frequency represent the lines that should be used to study multiple contingencies in a second detailed analysis stage, similarly to [3],[7].

B. Comparison with Previous Work

In this section, we apply the proposed partitioning methodology to the 30-bus system with the modified power injections and loads used in [7]. The modified power values are duplicated here and presented in Table II. The partitioning algorithm proposed in [7] is based on spectral partitioning, which suffers from long computational runtimes for increasing sizes of power systems. Our algorithm finds exactly the same partitioning solution as in [7] in the second iteration of the

⁶Out of the maximum possible in the system when all nodes with power injections are in one partition and all nodes with loads are in the other partition, irrespective of the cutsize.

⁷Or among all N_{DP} partitionings required by the user, like in the example from Fig. 5 where $N_{DP} = 30$.

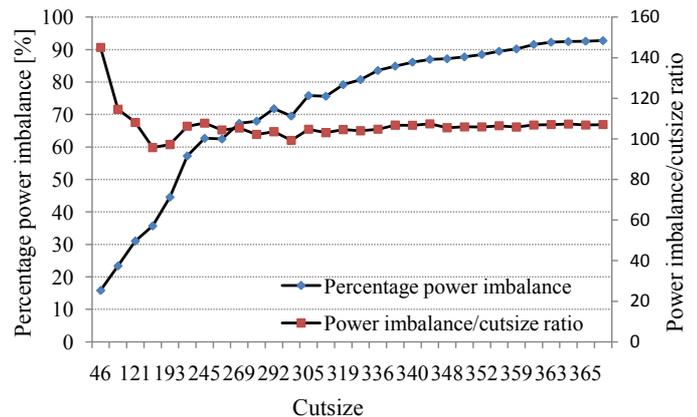


Fig. 5. The percentage of power imbalance and the power-imbalance/cutsize ratio variations with the cutsize for *bus_2383* testcase, based on $N_{DP} = 30$ data points collected in 2.41 s.

TABLE II

POWER INJECTIONS (POSITIVE VALUES) AND LOADS (NEGATIVE VALUES) USED FOR THE 30-BUS SYSTEM. VALUES TAKEN FROM [7].

| Bus | P | Bus | P | Bus | P | Bus | P |
|-----|-------|-----|------|-----|-------|-----|-------|
| 1 | 145.3 | 9 | 0 | 17 | -90 | 25 | 0 |
| 2 | -7.3 | 10 | -58 | 18 | -32 | 26 | -35 |
| 3 | -24 | 11 | 0 | 19 | -95 | 27 | 469.1 |
| 4 | -76 | 12 | -112 | 20 | -22 | 28 | 0 |
| 5 | 0 | 13 | 410 | 21 | -175 | 29 | -24 |
| 6 | 0 | 14 | -62 | 22 | 315.9 | 30 | -106 |
| 7 | -228 | 15 | -82 | 23 | 310 | | |
| 8 | -300 | 16 | -35 | 24 | -87 | | |

partitioning methodology (the second data point), when nodes 13,27 are fixed in partition V_1 and nodes 7,8 are fixed in partition V_2 . This partitioning solution is shown in Fig. 6. The CPU runtime of our algorithm is less than 1 s for all ten iterations necessary to collect all ten data points.

V. CONCLUSION

We proposed an efficient partitioning based methodology for power system extreme event screening. The main merits of the proposed approach are scalability, efficiency, ease of implementation, and capability of generating power-imbalance/cutsize ratio curves. The average CPU runtime is

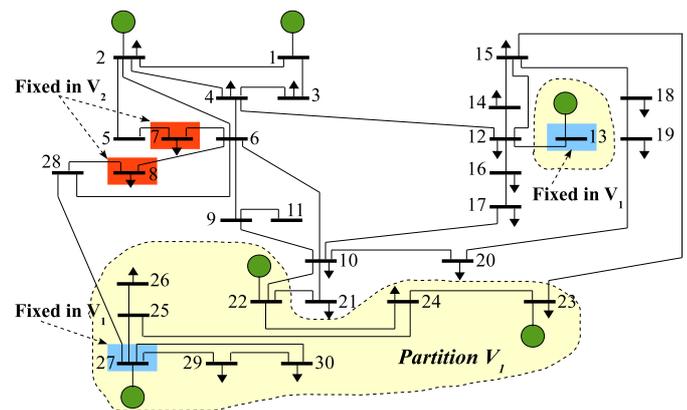


Fig. 6. Partitioning solution of the 30-bus system obtained using the proposed methodology with nodes 13,27 fixed in V_1 and nodes 7,8 fixed in V_2 .

less than 0.1 s and 1.5 s for systems with 2,383 buses and 43,501 buses, respectively. The partitioning quality is similar to that achieved by the algorithm studied in [7] for the 30-bus system. The tool proposed and implemented in this paper is publicly available at [15].

REFERENCES

- [1] Final Report on the August 14th Blackout in the United States and Canada, 2004, [Online]. Available: <https://reports.energy.gov/BlackoutFinal-Web.pdf>
- [2] I. Dobson, B.A. Carreras, V.E. Lynch, D.E. Newman, "Complex systems analysis of series of blackouts: Cascading failure, criticality, and self-organization," *AIP Chaos*, vol. 17, no. 2, June 2007.
- [3] V. Donde, V. Lopez, B. Lesieutre, A. Pinar, Y. Chao, J. Meza, "Identification of severe multiple contingencies in electric power networks," *IEEE North American Power Symposium (NAPS)*, pp. 59-66, Oct. 2005.
- [4] V. Donde, V. Lopez, B. Lesieutre, A. Pinar, Y. Chao, J. Meza, "Severe Multiple Contingency Screening in Electric Power Systems," *IEEE Trans. Power Systems*, vol. 23, no. 2, pp. 406-417, May 2008.
- [5] A. Pinar, A. Reichert, B. Lesieutre, "Computing Criticality of Lines in Power Systems," *IEEE Int. Symposium on Circuits and System (ISCAS)*, pp. 65-68, May 2007.
- [6] A. Pinar, J. Meza, V. Donde, B. Lesieutre, "Optimization Strategies for the Vulnerability Analysis of the Electric Power Grid," *Lawrence Berkeley National Laboratory Technical Report*, 2008.
- [7] B.C. Lesieutre, S. Roy, V. Donde, A. Pinar, "Power System Extreme Event Screening using Graph Partitioning," *IEEE North American Power Symposium (NAPS)*, pp. 503-510, Sep. 2006.
- [8] B. Lesieutre, A. Pinar, S. Roy, "Power system extreme event detection: the vulnerability frontier," *IEEE Hawaii Int. Conf. on System Sciences (HICSS)*, Jan. 2008.
- [9] A. Pinar, Y. Fogel, B. Lesieutre, "The Inhibiting Bisection Problem," *ACM Symposium Parallel Algorithms and Architectures (SPAA)*, 2007.
- [10] R. Leland, B. Hendrickson, "An empirical study of static load balancing algorithms," *Scalable High-Performance Computing Conference*, pp. 682-685, May 1994.
- [11] G. Karypis, R. Aggarwal, V. Kumar, S. Shekhar, "Multilevel Hypergraph Partitioning: Applications in VLSI," *IEEE Transactions on VLSI Systems (TVLSI)*, vol. 7, no. 1, pp. 69-79, 1999.
- [12] G. Karypis, hMetis, 2009, [Online]. Available: <http://glaros.dtc.umn.edu/gkhome/metis/hmetis/download>
- [13] C.J. Alpert, A.B. Kahng, "Recent Developments in Netlist Partitioning: A Survey," *Integration: the VLSI Journal*, vol. 19, no. 1-2, pp. 1-81, 1995.
- [14] U. Elsner, "Graph partitioning - a survey," *Technical Report*, TU Chemnitz, SFB393/97-27, 1997.
- [15] C. Ababei, R. Kavasseri, PROPART, 2009, [Online]. Available: <http://venus.ece.ndsu.nodak.edu/~cris/software.html>