Fully Distributed Power Routing for an Ad hoc Nanogrid

Amos Brocco
Institute for Information Systems and Networking
Department of Innovative Technologies
University of Applied Sciences of Southern Switzerland
Email: amos.brocco@supsi.ch

***Abstract*—This paper details a model for an ad hoc self-organized nanogrid based on distributed energy sources. The proposed concept draws inspiration from ad hoc computer networks and aims at creating an adaptive, scalable, and reliable power network to support energy provisioning with limited planning when the main grid is unavailable or severely damaged. Such a system could be used to support electrification and energy sharing in isolated rural areas, or to help emergency response crews in disaster relief situations. To control the operation of the nanogrid, we detail an autonomous fully distributed protocol that enables on demand dynamic configuration of power transmission paths between power sources and load devices across a mesh of intelligent routing nodes. The adaptiveness, resiliency, and scalability of our solution is demonstrated through simulation experiments.**

I. INTRODUCTION

Smart grids represent the future of power networks, where production, transmission, and consumption of energy meet with communication technologies and intelligent control [1]. A key role in smart grids is played by Distributed Energy Resources (DER), which require flexible and dynamic power networks that are able to cope with highly unpredictable events and fluctuating power flows resulting from variations in demand and offer [2]. Alongside with distributed generation, other novel concepts are deemed essential to smart grids, namely microgrids [3], which can operate autonomously from the main grid and aim for storage and consumption of energy in the vicinity of the generation sites [4], [5], [6], and nanogrids [7], represent small-scale systems that rely solely on distributed generation. These technologies change the role of the demand side from a passive to an active one, bringing both challenges and benefits. On the one hand, unstable sources such as photovoltaic, wind, or tidal generators [8], upon which microgrids and nanogrids typically rely, add further complexity to the controlling and management tasks [9]. On the other hand, distributed generation simplifies on demand provisioning, because additional power sources can be rapidly activated to support ancillary services and balance production and consumption [10], increases the reliability of the power grid [5], and generally opens up a wide range of new research directions. In this regard, we consider novel use case scenarios and applications for microgrids and nanogrids, in particular within rural areas and in disaster relief situations. More specifically, our aim is to implement the concept of *ad hoc-ness* in the realm of power grids by exploiting advances in power electronics, and by drawing inspiration from solutions to similar challenges adopted in computer networks. Hence we propose a model for a nanogrid (or isolated microgrid) that can be deployed with little forethought and is able to operate almost autonomously. In the following a crucial component of such a system is presented, namely an on demand provisioning protocol based on a fully distributed power routing algorithm. This protocol helps creating and maintaining energy provisioning paths between power sources and loads in situations where the main grid is not available or severely damaged. The rest of this paper is organized as follows: in Section II we describe the ad hoc nanogrid model and the considered application scenarios; in Section III we detail the basic hardware of our system, namely the smart power routing node. In Section IV we discuss some pitfalls when routing power, whereas in Sections V and VI we present the routing algorithm and evaluation results respectively. Finally, Section VII presents some related research work, and Section VIII provides conclusions and discusses future works.

II. AD HOC NANOGRID

In computer science and engineering the latin expression *ad hoc*, meaning *for this purpose*, is typically associated with computer networks where hosts can communicate wirelessly with each other without a fixed infrastructure [11]. In a similar way, we define an ad hoc nanogrid as a mesh of interconnected devices and distributed energy resources that coordinate to provide consumers with the power they need. As with ad hoc wireless networks, nanogrids are also concerned with topology control (i.e discovery which nodes are spatially close or linked together) and routing between consumers and providers to provide *energy on demand*. In contrast to an *always connected* power transmission network (as in a *traditional* grid), on demand power routing supports the creation of disjoint transmission paths with different QoS levels (as long as sufficient transmission lines are present in the network). In these situations, high quality paths could be used to connect critical loads to non-volatile energy sources, whereas low quality paths would be employed to connect less important loads to intermittent sources such as photovoltaic. Paths with different QoS levels would not interfere with each other, guaranteeing that voltage fluctuations do not propagate to the whole network. Another benefit of on demand provisioning concerns situations where demand exceeds offer: priorities associated with each path would then allow the system to keep only crucial devices active, while (temporarily) disconnecting the least important ones. The purpose of an ad hoc nanogrid is to support energy provisioning with limited infrastructure planning, no central control, and in dynamic conditions. In particular we consider two situations where infrastructure, namely the main grid, is not available or cannot be used anymore: electrification of rural areas and support to emergency response crews in disaster relief activities. In the first scenario we aim at creating a nanogrid system which is scalable and simple to manage even by non-professionals, whereas in the second scenario we target the creation of a robust emergency power network that can be easily deployed and maintained as well as reconfigured to suit environmental changes. This nanogrid model relies on distributed generation and power transmission networks, the topology of which might rapidly change or scale. In both scenarios, centralized control must be avoided in favor of fully distributed approach. Furthermore, to ease the maintenance of the system, autonomous and self-organized behaviors must be implemented to reduce the need for human intervention to a minimum. Conforming to these requirements, in the following we describe the required hardware and an autonomous provisioning protocol based on a fully distributed power routing algorithm.

III. SMART POWER ROUTING NODE

The concept of ad hoc nanogrid is based on a power switching device called *smart power routing node*, or *smart node*. A smart node acts as an intelligent power router consisting of several input/output ports where power sources, loads, or other smart nodes can be connected to. At this stage of our research we consider nodes with four ports, in order to limit the complexity of the power switching circuit of an hardware testbed, but future revisions might include nodes with different configurations as well. Each port can be connected to at most one device, and by means of a crosspoint switch each port can be internally connected to any of the remaining ports. Smart nodes are responsible for creating and maintaining power routing paths (referred to as *provisioning agreements*) between loads and generators. More specifically, they must ensure that all loads receive the required amount of power, that generators do not become overloaded, and that optimal transmission paths are chosen. Accordingly, beside the power switching hardware, smart nodes are also equipped with voltage and current sensing circuitry for each port, and an embedded computer that runs the routing and control software. To perform this task, nodes must also know the capabilities of connected energy sources (for example, photovoltaic of gasoline generators) and the requirements of connected loads (i.e. power required): currently these value need to be defined by a human operator, but the integration of intelligent devices that can interact directly with a node is envisioned as future work. Smart nodes can communicate with each other by means of an ad hoc wireless network. Because control of the power network is based on a fully distributed protocol, coordination between nodes and understanding of the network topology play important roles. In this regard, wireless communication is sided with a low-level signaling protocol that allows each smartnode to know the identity of neighbor nodes connected to its ports and construct a local view of the nanogrid.

IV. POTENTIAL PITFALLS WHEN ROUTING POWER

As stated before, ad hoc nanogrids bear similarities with ad hoc computer networks, and it might make sense to consider routing algorithms developed for computer networks in power networks too. However significant differences between the two makes such solutions infeasible. In particular, at a high-level, routing algorithms for packet switched network know exactly where each piece of information is transmitted to, and communication between peers can be performed in a deterministic point-to-point manner. In contrast, power routing has to deal with *electrons* that freely flow in the circuit between power sources and all connected loads through paths of least resistance, which might differ from the routes intended by an algorithm. Accordingly we are forced to consider a number of side-effects that can result from switching a connection between two nodes on or off. Among all possible situations, in this section we discuss two examples of pitfalls that a control system must deal with in order to ensure reliable provisioning of energy in the power network. The problems presented here result from accidental sharing of the same transmission line between multiple routing paths: when this happens, without a global overview of the network it becomes difficult to sort out dependencies between loads and generators, and an inappropriate response might occour in the event of a failure. In the proposed system, a solution to these issues is implemented by continuously monitoring the network and by adjusting routing decisions to account for the differencies between expected and actual power flows. In order to keep our discussion as simple as possible, we omit here the details of how routing decisions are taken: the exact operation of the provisioning mechanism will be discussed in Section V.

A. Route leaking

The first situation (Figure 1) considers four smart nodes A, B, C and D, two generators G_1 and G_2 which can provide at most $15A$ (*amps*), and a load L_1 requiring $2.5A$. For the sake of clarity, ports on each node are numbered clockwise starting from the one on the top edge. To fulfill the requirements of L_1, a provisioning path is created between port 2 on node C, to port 4 on node A, passing through node B. Subsequently, load L_2, which requires $15A$ is connected to port 4 on node B. The provisioning mechanism employs generator G_2, because G_1 cannot provide more than $12.5A$. Accordingly, $15A$ are reserved on G_2 and the path from port 4 on B to port 3 on D is enabled. Unfortunately this decision joins two provisioning paths and $6A$ (instead of $15A$) and $11.5A$ (instead of $2.5A$) are drawn from G_2 and G_1 respectively [1]. In this situation L_1 and L_2 now rely on both generators, but only have an *agreement* with either one of them.

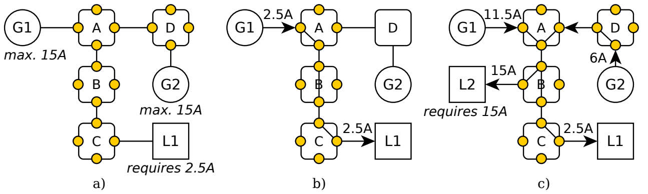

Fig. 1. Example of a power routing pitfall: route leaking.

---

[1] Values shown in these examples have been obtained using the Gnucap circuit simulator, and serve the purpose of illustrating situations where an incongruency between the power flow decided by the routing algorithm and the real one exists.

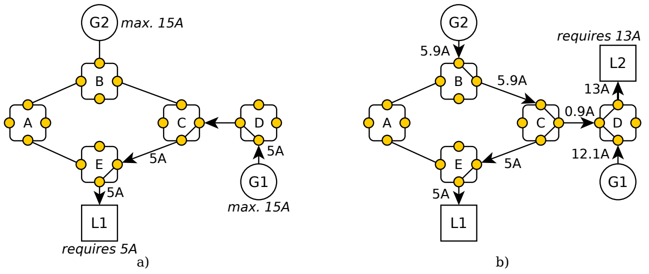

Fig. 2. Example of a power routing pitfall: flow inversion.

B. Flow inversion

In the second situation (Figure 2) we consider a load L_1 (requiring $5A$), connected to port 3 on a node D, which is fed by generator G_1, connected to port 3 on a node E. Current flows from G_1 to L_1 through the power routing path across nodes C, D, E. Subsequently, another load L_2 is attached to the system. The requirements for this load exceed the leftover capacity of G_1 ($10A$), and power has to be provided by G_2. Accordingly, a provisioning path is created between port 1 on node B and port 1 on node D. However, this decision also creates a link to generator G_1. Because electricity flows through the path of least resistance, L_2 now draws its power mostly from G_1 ($12.1A$) and only in some part from G_2 ($0.9A$). The current flow on the transmission line between nodes C and D reverses its direction, resulting in L_1 not being fed anymore by G_1 but by G_2 instead. However, no provisioning agreement between L_1 and G_2 exists.

V. FULLY DISTRIBUTED PROVISIONING PROTOCOL

One of the primary goals of an ad hoc nanogrid is to ensure that all connected loads are fed with the amount of power required for them to work, while ensuring that generators are not overloaded, i.e. remain within their power rating. The provisioning algorithm must operate in an evolving power network, where the topology could dynamically change as result of rearrangement of existing transmission lines, creation of new lines, or removal of existing ones (either voluntary or accidental). Furthermore the composition and scale of the system could change with time, and user-defined constraints could be set at any time (for example, in the form of load priorities). Accordingly, the provisioning algorithm must be able to adjust power routing decisions to continuously improve the quality of the solutions found and to overcome system failures. The proposed approach is divided into several phases: power provisioning, path maintenance, accounting, and path improvement. Each phase employs different messages (as shown in Figure 3) that are exchanged by nodes over an ad hoc wireless network. In the following the operations performed by each node in each phase will be detailed.

A. Provisioning Phase (1)

The purpose of the provisioning phase is to check connected loads, verify that their requirements are fulfilled, and possibly start a discovery process to find routing paths toward power sources in the network and create provisioning agreements. Allocation of a new routing path is performed using a *check-then-act* sequence, where a pre-allocation of the resources is required prior to a confirmation of the routing decisions. The node connected to the load initiates and coordinates the discovery, pre-allocation, and confirmation steps.

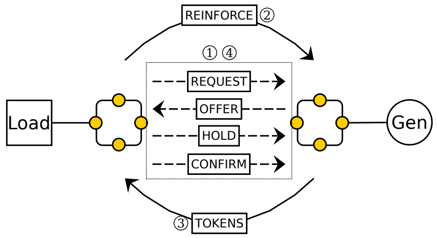

Fig. 3. Autonomous power routing, protocol messages.

1) Request: By means of a request message a node can query the network to discover power sources. The initiating node starts by inquirying local generators. If the requirements cannot be fulfilled locally, the request message is forwarded for a limited number of hops in the network using an iterative deepening strategy that follows the topology determined by transmission lines. Even though a wireless broadcast would be possible, we employ point-to-point communication in order to collect and store useful information about discovered paths that helps determining their cost: for example, current load, impedance, the number of routing paths sharing each transmission line, etc. This controlled forwarding also enables the definition of forbidden paths, for example through transmission lines that are reaching their maximum capacity, are about to be voluntarily disconnected or that are registered to a different QoS class.

2) Offer: When a node receives a request message, the leftover capacity of local generators (if any) is checked. If available power satisfies the requirements set in the query, the node replies to the initiating node with an offer message.

3) Hold: The initiating node waits for some limited amount of time for incoming offers, and evaluates them according to a cost function. Several cost functions can be considered: impedance of the routing path, amount of leftover capacity on the generator, number of other routing paths sharing the same transmission lines, etc. The cheapest offer is selected and hold messages are issued to pre-allocate the resources on the routing path: each node on the path thus receives all required information (such as the name of the initiating node, the concerned generator, the amount of power requested, etc.). In contrast to request and offer messages, hold messages (along with confirm messages) are transmitted using a reliable protocol to ensure that all recipients are correctly notified.

4) Confirm: After all nodes on the routing path have received an hold message, a confirm message from the initiating node can follow. Nodes on the path activate the corresponding switches to allow electricity to flow from the generator toward the load. Furthermore, the initiating node switches to the maintenance phase for this provisioning agreement, and becomes responsible for continuously monitoring the current flowing to the load device and for verifying that the path remains active.

B. Maintenance Phase (2)

Each confirmed agreement has to be maintained in order to solve all unexpected failures and ensure robust operation of the nanogrid. Each node monitors the paths for locally connected loads by means of a signaling protocol that mimics the behavior of ants. Each agreement stored on a node is associated with a numerical value called *artificial pheromone*. This pheromone is set to an initial concentration value of 1 when the agreement is confirmed. Periodically each node simulates evaporation of local pheromones, by decrementing their values according to a decay function (typically exponential). When the pheromone is lower than a certain threshold, the agreement is discarded and the associated routing configuration on the node is disabled (if not used by any other agreement). To avoid deletion, for each confirmed agreement a reinforce message is periodically forwarded along the corresponding path starting from the demand side node (i.e. connected to the load). On each node encountered while traveling forward toward the generator, transmission lines are verified to ensure that power correctly flows across the network. When the reinforce message reaches the last node (i.e. the one connected to the generator) it changes its direction, heading back to the starting node. This time on each node the pheromone value is incremented to its maximum value. If for some reason the reinforce message cannot complete its journey while traveling forward (for example because the corresponding agreement is not registered anymore on a node), a negative reinforcement is employed on the way back. Using artificial pheromone trails to maintain paths in the network draws inspiration from the foraging behavior of ants. In nature, pheromone is employed as an indirect communication mechanism, called stigmergy, which enables individuals in a colony to exploit optimal paths between the nest and food sources. An important branch of computational intelligence, namely ant colony optimization (ACO) [12], is based on the pheromone paradigm. Similar to other swarm intelligence [13] and bio-inspired approaches, ant algorithms are an ideal candidate for solving network related problems, because they are inherently distributed and do not require direct communication between agents. In particular, indirect communication through stigmergy has been successfully applied to routing problems in ad hoc networks [14], [15].

C. Accounting and Adjustement Phase (3)

In order to maintain control over all resources, accurately measure power consumption, and ensure appropriate response in the event of failure, it is necessary to track which generator is feeding which load. In particular, when multiple paths cross, determining the source of energy might not be easy (as explained in Section IV). As such, an important phase of our protocol is devoted to determining how power is propagated from generators toward loads. The accounting and adjustment phase is divided in three steps.

1) Token generation step: To account for the amount of current entering a node the concept of *incoming tokens* is used. Each token associates some amount of power to the identifier of a generator. Smart nodes store tokens separately for each port. Nodes connected to generators measure the amount of current flowing through the corresponding port (in *amperes*) and generate a proportional number of *incoming tokens*. Additional *incoming tokens* might be received from neighbor nodes, if a positive amount of power is provided.

2) Token propagation step: All subsets of connected ports are considered in this step: the total amount of *incoming tokens* is divided between all outbound ports, and a number of *outgoing tokens* proportional to the amount of current exiting the node through that port is generated. *Outgoing tokens* are then either propagated to connected nodes or used to determine the source of energy for connected loads.

3) Adjustment step: The adjustment step is required when the energy flow determined during the propagation step does not match the agreements made during the provisioning phase. If an existing confirmed agreement needs to be updated to match the actual power drawn from a source an adjustment request is piggybacked to reinforcement messages: if the target node does not accept the update, the agreement might be revoked. Conversely, if power is received from a generator but no agreement exists, one has to be created. Confirmed agreements can be canceled if the actual power drawn is negligible.

D. Improvement Phase (4)

A node responsible for a provisioning agreement also issues proactive requests to look for alternate paths toward generators, which are evaluated using the same cost function as in the provisioning phase. Improvement queries are only initiated when the requirements for a load are fulfilled. When a better path is found the node can perform a switchover by simultaneously confirming the new agreement and unconfirming the superseded one. The improved path will be activated, and thanks to normal path maintenance the old one will subsequently disappear because reinforcement would not take place anymore.

VI. EVALUATION

To achieve an initial validation of our fully distributed power routing algorithm we set up a dynamic scenario using a software simulator based on Gnucap. This simulation serves as a *proof-of-concept*, but we understand the need for real-world experiments on a hardware testbed (which are nonetheless planned for the near future) to fully understand the dynamics of the system. For this paper, the considered initial setup, which is depicted in Figure 4, consists of 5 smart nodes, one load ($L1$) and a generator ($G1$). For simplicity, all considered loads require $2.5A$ and each generator can feed at most $15A$. The topology of the nanogrid is changed dynamically during the simulation. To evaluate the scalability of our solution, after 250 seconds into simulation, the network is expanded with 6 nodes, an additional load and a generator; after 500 seconds, 6 additional nodes are connected, along with 3 loads. Finally, after 750 seconds, 6 nodes, 3 loads and 1 generator are connected to the nanogrid completing its expansion. To validate the robustness of the system, after 1000 seconds 3 nodes are disconnected from the grid, followed by the removal of another 2 nodes at the 1250 seconds mark; simulations are stopped at 1500 seconds. Two disconnection policies are considered: the first one assumes *graceful* disconnection of nodes, and the second one uses *abrupt* disconnection. Graceful disconnection enables the system to resolve an alternative path before the node is removed, whereas abrupt disconnection simulates a failure. Because nodes execute asynchronously and independently from each other, routing decisions might vary slightly across different runs. Accordingly the presented results represent an average over 10 simulation runs. The cost

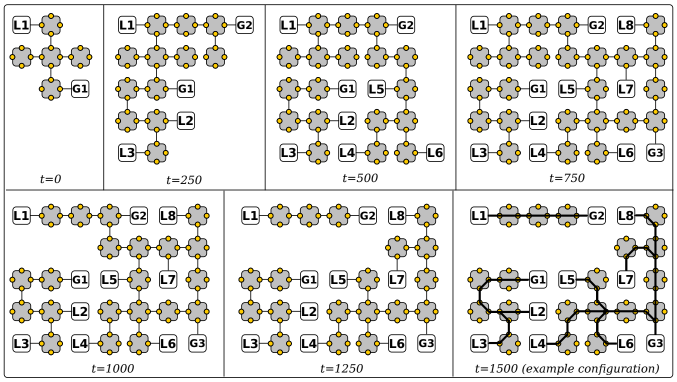

Fig. 4. Evaluation scenario with multiple loads and generators: nodes are added and removed throughout the simulation. At t=1500, bold lines indicate active paths determined by the provisioning protocol.

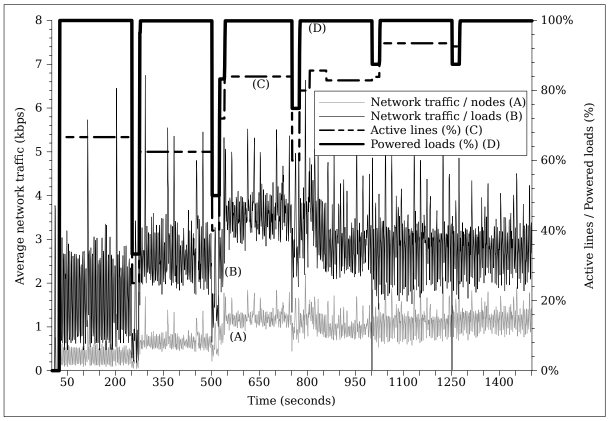

Fig. 5. Powered loads, active transmission lines, and network traffic (abrupt disconnection)

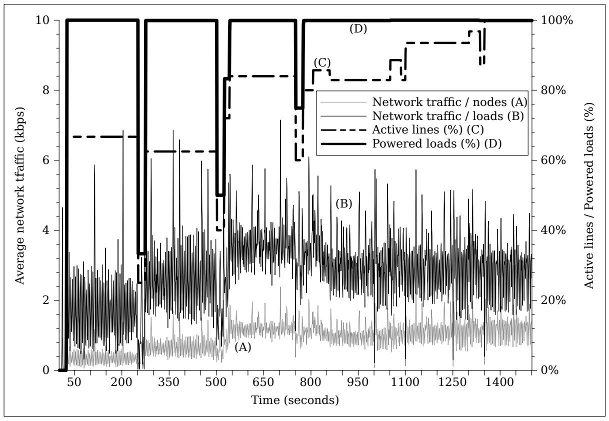

Fig. 6. Powered load, active transmission lines, and network traffic (graceful disconnection)

function employed in all experiments aims at minimizing the distance between loads and generators, namely the number of transmission lines activated by the routing protocol.

A. Resilience and Adaptiveness

The resilience and adaptiveness of our protocol is determined by the ability to provide power to each load through short routing paths. In this regard, Figures 5 and 6 illustrate the percentage of the power received by loads (100% meaning that the required power is provided) as well as the fraction of active transmission lines (transmission line that carry power from generators to loads). When the network is expanded, new loads must wait until provisioning paths are created. The speed of this process depends on the time required for discovering and activating a new path, which in turn depends on the complexity of the topology and the number of trials required (as iterative deepening is used to propagate request messages). For this evaluation, nodes typically need 15 to 45 seconds to perform the request, hold, and confirm phases (depending on the distance between the load and the generator). When nodes are disconnected, loads are either not affected (graceful disconnection) or minimally affected (abrupt disconnection). In the latter case, new transmission paths are discovered quickly (in a real network a battery backup would be sufficient to eliminate downtime). Although not shown in the figures, provisioning paths are not only reconfigured dynamically to adapt to topology changes (for example, due to node disconnection) but also to exploit newly discovered shorter paths to the available generators when the network is expanded. For example, at $t = 500$ loads $L4$, $L5$ and $L6$ connect to generator $G2$, but after $t = 750$ a shorter path to $G3$ is discovered and exploited. Similarly, $L1$ initially connects to $G1$, but is forced to find an alternative, namely $G2$ at $t = 1000$, because the original provisioning path is interrupted.

B. Network traffic and scalability

The scalability of the provisioning protocol is determined by the amount of network traffic generated by the algorithm. The average traffic per node (thin grey line) is about 1kbps, whereas the traffic per load (thin black line) varies from 2kbps to about 4kbps. The traffic generated by all phases of the protocol clearly depends on the size of the network and the number of transmission lines, however results demonstrate that the protocol is able to scale well. Traffic peaks are evident when the network is expanded or shrinked, as provisioning requests are executed, but there is no significant difference between abrupt and graceful disconnection, because both require the affected nodes to find alternative paths. It should be noted that a further traffic reduction could be achieved at the expense of a less robust network by lowering the communication rate of the maintenance and adjustement phases.

VII. RELATED WORK

In this section we briefly present some of the research literature related to the two main topics covered by our research: power routing and autonomous control of micro and nanogrids. The problem of power routing is tackled in a number of scientific research papers. In [16] routing is related to maximizing the total revenue (for both power suppliers and power customers), whereas intelligent power routers and distributed coordination have been proposed in [17] and [18] to increase the robustness and flexibility of the distribution network. Similar studies have been conducted in [19], where graph routing algorithms are proposed to efficiently route power, and in [20], where a distributed market based allocation technique that minimizes the overall power generation cost is presented. In contrast to these solutions, our approach does not deal with energy cost but focuses on providing *best effort* performance in a dynamic environment. The second important aspect concerns autonomous operation, which requires self-management and adaptive behaviors, as suggested in [21]. In [22] an intelligent and self-configurable microgrid is presented: the proposed approach is based on a centralized control unit in each microgrid that performs automated load management and can isolate loads to reduce the overall demand. To deal with the problems that arise when transitioning from passive to active electric distribution networks, authors in [23] advocate the need for distributed, flexible, and intelligent management. In this respect an agent-based approach and a distributed optimal routing algorithm are presented. In [24] an agent-oriented system for a self-healing smart grid is discussed: energy is dynamically rerouted toward optimal paths depending on the state of the system, in order to overcome disruptions that can lead to unserved demands for power. In [25] decentralized control through peer-to-peer interaction between components is deemed essential for microgrids, because it increases robustness and survival chances in the event of failures.

VIII. CONCLUSIONS

In this paper we presented the concept for an ad hoc nanogrid, along with an autonomous on demand power routing algorithm. The system implements decentralized control by means of intelligent power routing nodes that communicate and collaborate with each other over an ad hoc wireless network. All power routing decisions are determined using a fully distributed algorithm run by each node, depending on locally collected information about the state of the network. The goal of the proposed system is to create an adaptive, scalable, and reliable nanogrid that does not require supervision or control from a central component. The considered application scenarios aim at situations where the main grid is unavailable or severely damaged: electrification of rural areas and support for disaster relief operations. Both scenarios rely on distributed energy sources and present highly dynamic conditions and unexpected failures that could affect power transmission. Current research focuses on the development of a low-voltage hardware testbed platform that will enable an in depth evaluation and validation of a small-scale system. This platform also aim at becoming a research tool and teaching aid to help exploring smart grid technologies and experiment with power monitoring and distributed routing algorithms. On the software side, the main concern is security, which includes authentication mechanisms and trust management. Finally, integration of intelligent load and generator devices in the presented provisioning protocol, support for storage devices, and an in-depth sensitivity analysis of the parameters of the provisioning algorithm are also considered.

REFERENCES

[1] H. Farhangi. The path of the smart grid. *Power and Energy Magazine, IEEE*, 8(1):18 –28, january-february 2010.

[2] Faycal Bouhafs and Michael Mackay. Active control and power flow routing in the smart grid. *IEEE Smart Grid*, December 2012.

[3] S. Mizani and A. Yazdani. Design and operation of a remote microgrid. In *Industrial Electronics, 2009. IECON '09. 35th Annual Conference of IEEE*, pages 4299 –4304, nov. 2009.

[4] R H Lasseter. Microgrids and distributed generation. *Journal of Energy Engineering*, 133(3):144, 2007.

[5] G. Pepermans, J. Driesen, D. Haeseldonckx, R. Belmans, and W. D'haeseleer. Distributed generation: definition, benefits and issues. *Energy Policy*, 33:787–798, 2005.

[6] T. Ackermann, G. Andersson, and L. Sder. Distributed generation: a definition. *Electric Power Systems Research*, 57:195–204, 2001.

[7] Duke R. Bryan, J. and S. Round. Decentralised control of a nanogrid. In *Australasian Universities Power Engineering Conference*, Christchurch , New Zealand, 2003.

[8] G. Joos, B.T. Ooi, D. McGillis, F.D. Galiana, and R. Marceau. The potential of distributed generation to provide ancillary services. In *Power Engineering Society Summer Meeting, 2000. IEEE*, volume 3, pages 1762 –1767 vol. 3, 2000.

[9] F. Blaabjerg, R. Teodorescu, M. Liserre, and A.V. Timbus. Overview of control and grid synchronization for distributed power generation systems. *Industrial Electronics, IEEE Transactions on*, 53(5):1398 – 1409, oct. 2006.

[10] C. Yuen and A. Oudalov. The feasibility and profitability of ancillary services provision from multi-microgrids. In *Power Tech, 2007 IEEE Lausanne*, pages 598 –603, july 2007.

[11] Stefano Basagni, Marco Conti, and Silvia Giordano. *Mobile Ad Hoc Networking*. Wiley-IEEE Press, August 2004.

[12] Marco Dorigo and Thomas Stuetzle. *Ant Colony Optimization*. Bradford Book, 2004.

[13] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm intelligence: from natural to artificial systems*. Oxford University Press, Inc., New York, NY, USA, 1999.

[14] F. Ducatelle, G. Di Caro, and L.M. Gambardella. Ant agents for hybrid multipath routing in mobile ad hoc networks. In *Wireless On-demand Network Systems and Services, 2005. WONS 2005. Second Annual Conference on*, pages 44 – 53, jan. 2005.

[15] S. Kumar, R. Chaudhary, and Nitin. Optimization of routing algorithms in ad-hoc networks using swarm intelligence. In *Information and Communication Technologies (WICT), 2011 World Congress on*, pages 677 –681, dec. 2011.

[16] Husheng Li and Weiyi Zhang. Qos routing in smart grid. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1 –6, dec. 2010.

[17] A. A. Irizarry-Rivera, M. Rodriguez-Martinez, B. Velez, B. Velez-Reyes, A. Ramirez-Orquin, E. O'Neill-Carrillo, and J. Cedeno. Intelligent power routers: a distributed coordination approach for electric energy processing networks. *International Journal of Critical Infrastructures*, 3(1-2):20–57, 2007.

[18] P.H. Nguyen, W.L. Kling, and P.F. Ribeiro. Smart power router: A flexible agent-based converter interface in active distribution networks. *Smart Grid, IEEE Transactions on*, 2(3):487 –495, sept. 2011.

[19] PH Nguyen, WL Kling, G Georgiadis, M Papatriantafilou, LA Tuan, and L Bertling. Application of the graph theory in managing power flows in future electric networks. *IEEE Transactions on Microwave Theory and Techniques*, 2012.

[20] B. HomChaudhuri, M. Kumar, and V. Devabhaktuni. Market based approach for solving optimal power flow problem in smart grid. In *American Control Conference (ACC), 2012*, pages 3095 –3100, june 2012.

[21] Matthias Baumgarten and Maurice D. Mulvenna. Towards intelligent and self-evolving network infrastructures for energy management. In *SASO Workshops*, pages 72–75, 2010.

[22] A.A. Zaidi and F. Kupzog. Microgrid automation - a self-configuring approach. In *Multitopic Conference, 2008. INMIC 2008. IEEE International*, pages 565 –570, dec. 2008.

[23] Phuong Nguyen, Wil Kling, Giorgos Georgiadis, Marina Papatriantafilou, Tuan Le, and Lina Bertling. Distributed routing algorithms to manage power flow in agent-based active distribution network. *IEEE PES Conference on Innovative Smart Grid Technologies Europe, Gothenburg, Sweden, October 10-13, 2010*, 2010.

[24] S. Bou Ghosh, P. Ranganathan, S. Salem, Jingpeng Tang, D. Loegering, and K.E. Nygard. Agent-oriented designs for a self healing smart grid. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 461 –466, oct. 2010.

[25] P. Piagi and R.H. Lasseter. Autonomous control of microgrids. In *Power Engineering Society General Meeting, 2006. IEEE*, page 8, 0-0 2006.