Cidades Inteligentes: Uma arquitetura de Gerenciamento Autonômica no Contexto de IoT

Pablo Tibúrcio[1], **Marcelo Santos**[1], **Stênio Fernandes**[1]

[1]Centro de Informática (CIn) – Universidade Federal de Pernambuco (UFPE)
Recife – PE – Brasil

{pgst,mabs,sflf}@cin.ufpe.br

Abstract. *Network infrastructure management is a critical area for the maintenance of the network and quality of services provided by networks. This management has become quite complex as networks become larger and more heterogeneous. Thus, it increases the costs of management and the possibility of human error as well. Internet of Things (IoT) is an environment that offers these challenges. Management in IoT combines network infrastructure and node (end devices) management. Smart City (SC) is another environment with these management challenges because it is comprised by IoT, but even more complex, owing to large scale and the single administrative domain. In such regard, traditional mechanisms of network and devices management are not efficient. Thus, this work proposes AutoManIoT an architectural solution for autonomic network and devices management to the IoT and SC environments. The proposed architecture use an approach of dynamic and elastic network services provisioning through Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) technologies. These technologies improve the network control and operation efficiency and narrow the network management to application requirements.*

Resumo. *O gerenciamento de infraestrutura de redes é uma atividade fundamental para a manutenção da rede e da qualidade dos serviços providos por ela. Este gerenciamento tem se tornado bastante complexo e custoso à medida que as redes ficam maiores e mais heterogêneas. Desta forma, os custos com o gerenciamento aumentam e a possibilidade de erros humanos também. Um ambiente que oferece esses desafios é a Internet of Things (IoT). O gerenciamento na IoT envolve além da infraestrutura de rede, o gerenciamento de seus nós (dispositivos finais). Cidade Inteligente (CI) é outro ambiente como esses desafios de gerenciamento pois ele engloba a IoT, no entanto, é ainda mais complexo devido a sua grande escala e único domínio administrativo. Neste contexto, mecanismos tradicionais de gerencia de redes e dispositivos não são eficientes. Assim, esse trabalho propõe AutoManIoT, uma solução arquitetural de gerenciamento autonômico de rede e de dispositivos para o cenário de IoT e CIs. A arquitetura utiliza uma abordagem de provisionamento dinâmico e elástico dos serviços de rede através das tecnologias Redes Definidas por Software (SDN) e Virtualização de Funções de Rede (NFV). Essas tecnologias permitem melhorar a eficiência do controle e da operação da rede e ainda aproximar o gerenciamento da rede dos requisitos da aplicação.*

1. Introdução

A Internet cresceu de forma surpreendente desde sua criação há quase 50 anos anos atrás. O número de dispositivos conectados à rede cresceu de algumas dezenas para alguns bilhões. Estimativas apontam que até 2020 o número de dispositivos conectados poderá chegar a 50 bilhões[1]. Nesse contexto, o crescimento da rede provocou um aumento de sua complexidade, criando desafios de escalabilidade, manutenção, gerência de políticas e otimização do uso de recursos disponíveis. É natural que nesse ambiente coexistam dispositivos heterogêneos com diferentes capacidades e de diferentes fabricantes, o que torna ainda mais difícil a gerência de forma eficiente da rede. Estudos apontam que os custos de gerenciamento da infraestrutura já superam há algum tempo o custo com o *hardware* e *software* [Agoulmine 2011].

A revolução tecnológica que contribui para o número de dispositivos conectados à rede é conhecida como Internet das Coisas (IoT) [Atzori et al. 2010]. Dispositivos no ambiente de IoT possuem diferentes requisitos e causarão um grande impacto na forma como as pessoas irão interagir com as *coisas* ao seu redor. Atualmente, estima-se que a quantidade de dispositivos conectados é da ordem de 8 bilhões[2]. À medida que estes sistemas crescem, o seu gerenciamento também passa a ser um elemento impactante, portanto um fator que vem recebendo atenção da academia e indústria.

Diversos trabalhos destacam **heterogeneidade** e a **escalabilidade** como as principais características da IoT [Atzori et al. 2010, Borgia 2014, Gama et al. 2012, Li et al. 2015]. Desta forma, o número crescente de dispositivos na IoT exige que sua infraestrutura seja escalável, ou seja, tenha a habilidade de lidar com variações de carga enquanto faz o bom uso dos recursos disponíveis, mantendo um bom desempenho (ex: baixo tempo de resposta, baixo consumo de CPU e memória) sob diferentes situações de estresse da rede. [Borgia 2014] afirma que dispositivos na IoT serão extremamente heterogêneos em termos de suas capacidades, expectativa de vida e tecnologias de comunicação. De fato, a variedade de tecnologias e protocolos de comunicação com diferentes larguras de banda, modulação, tamanho de pacote, frequência, entre outros, torna o desenvolvimento da arquitetura para IoT um desafio. É necessário que ela seja capaz de prover interligação segura e eficiente entre os dispositivos e os diferentes tipos de redes.

Nesse contexto, Cidades Inteligentes (CIs) surgem como uma integração entre a tecnologia de informação (TI) e IoT, cujo o objetivo é tornar o espaço urbano palco de experiências de uso intensivo de tecnologias de comunicação e informação sensíveis ao contexto. Assim, é possível realizar, por exemplo, o monitoramento em tempo real de diversas atividades cotidianas, automatização de tarefas, criação de soluções sustentáveis e consequentemente a melhoria da qualidade de vida de uma população. Em outras palavras, IoT é considerada uma das tecnologias que possibilitará a realização das Cidades Inteligentes [Zanella et al. 2014]. Deste modo, como as CIs utilizam a IoT como tecnologia fundamental para seu desenvolvimento, elas irão compartilhar um conjunto semelhante de desafios, entre eles o gerenciamento de dispositivos e de sua infraestrutura. No entanto, nas CIs o administrador de rede terá um desafio ainda maior, pois terá que administrar um domínio, em geral, muito grande de dispositivos e tecnologias que fazem

[1]http://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated - *acessado em 07/04/2017*

[2]http://www.gartner.com/newsroom/id/3598917 - *acessado em 07/04/2017*

parte de um mesmo conjunto de aplicações para CI. Assim, dado a complexidade da rede e heterogeneidade dos dispositivos, gerenciar manualmente a rede é uma tarefa custosa, passível de erros e enviável na maioria dos cenários de larga escala.

Uma abordagem bastante estudada que visa diminuir esses problemas na administração de redes é a gerência autonômica de infraestrutura de redes (ANM) [Dobson et al. 2006]. Essa abordagem advém do termo computação autonômica descrita no manifesto escrito por [Horn 2001]. Ele aponta que o aumento da complexidade em sistemas computacionais é o principal problema a ser atacado pela computação autonômica. Posteriormente, [Kephart and Chess 2003] detalham a computação autonômica como um sistema que possui a capacidade de se auto-gerenciar de acordo com os objetivos definidos pelos administradores. O ponto fundamental dos sistemas autônomos é o auto-gerenciamento, que pode ser dividido em quatro aspectos: autoconfiguração, auto-otimização, autocura e autoproteção [Kephart and Chess 2003]. Essas funções autonômicas são mantidas através de uma estrutura conhecida como ciclo de controle que são configurados através de políticas criadas pelo administrador do sistema. Essas características, associadas à administração da infraestrutura de redes simplificam o processo de gerência, diminuem custo, reduzem a intervenção e desta forma miniminizam a ocorrência de erros humanos.

O auto-gerenciamento busca justamente melhorar a capacidade de gerência da rede para lidar, por exemplo, com diferentes demandas de acordo com as regras de negócio especificadas pelo administrador da rede. Isto possibilita a redução de custo, mudando o foco do administrador do processo de configuração manual de dispositivos de baixo nível, para definir apenas regras de negócio. Neste sentido, a Força de Tarefa de Engenharia da Internet (IETF) juntamente com a Força Tarefa de Pesquisa da Internet (IRTF) possuem alguns grupos de trabalho ativos em busca de avanços científicos e padronização nessa área. Podemos destacar o grupo de pesquisa em gerenciamento de redes (NMRG[3]) que trabalha na discussão de problemas de gerenciamneto de serviços de redes de alto nível. Um outro exemplo é o grupo Anima (*Autonomic Networking Integrated Model and Approach*) que surgiu com base em uma discussão na IRTF com o objetivo de desenvolver um protocolo para o auto-gerenciamento de uma rede através de funções autonômicas [Behringer et al. 2015]. Atualmente, este grupo de trabalho possui vários rascunhos (*Internet-Drafts* ID) com os detalhes funcionais de propostas de uma arquitetura autonômica. Dentre eles, se destacam o plano de controle, o protocolo genérico de sinalização autonômico (GRASP), a infraestrutura de *bootstrapping* (BRSKI) e o plano de controle autonômico (ACP). Outro grupo de trabalho que está relacionado com este tema é o SUPA[4] (*Simplified Use of Policy Abstractions*) que tem o objetivo de definir um modelo de dados para ser utilizado na representação de políticas de alto nível com função de gerenciamento de rede.

Assim, a contribuição desse artigo é propor uma arquitetura chamada AutoMan-IoT, uma solução arquitetural de gerenciamento autonômico de rede e de dispositivos para o cenário de IoT e CIs. Nesta perspectiva, este trabalho se assemelha em vários pontos da proposta do grupo Anima. Além disso, ele busca propor mecanismos para que o funcionamento autonômico da rede se adeque às características dinâmicas, heterogêneas

[3]https://irtf.org/nmrg

[4]https://trac.ietf.org/trac/supa/

e restritas da IoT. Também, é importante verificar como os paradigmas de SDN e NFV afetam a autonomia da arquitetura. Esperamos dessa forma, contribuir efetivamente para discussões em pelo menos dois grupos distintos pertencentes ao IETF/IRTF.

2. Trabalhos Relacionados

Os trabalhos que utilizam o conceito de gerência autonômica associado à IoT em geral abordam a gerência de serviços para IoT através de *middlewares*, sem se preocupar com problemas de baixo nível da rede [Savaglio and Fortino 2015].

Os trabalhos que utilizam a estratégia ANM com SDN e NFV [Chaparadza et al. 2014, Wendong et al. 2012, Qi et al. 2014] possuem abordagens ligeiramente diferentes e visam propor soluções através de arquiteturas genéricas, sem se preocupar com os desdobramentos que elas podem gerar na infraestrutura de rede . Esses trabalhos também não apresentam detalhamentos importantes como a representação de conhecimento, como as políticas de gerenciamento são implementadas na rede e como as camadas de controle se comunicam. Além disso, pouco esforço foi feito para validar e testar o desempenho e a escalabilidade das soluções.

Este trabalho avança o estado da arte propondo um novo *framework* para suportar o auto-gerenciamento de dispositivos de rede e nós da IoT e CI utilizando SDN e NFV como infraestrutura de implantação.

3. Motivação e Referencial Teórico

Os principais conceitos necessários para desenvolver uma arquitetura de gerenciamento autonômico de um infraestrutura IoT serão apresentados na seção a seguir.

3.1. Internet das Coisas

A IoT foi proposta inicialmente por Kevin Ashton em 1999 como o conceito de objetos conectados interoperáveis, identificáveis de modo único com a tecnologia RFID [Ashton 2009]. Desde então, a definição de IoT evoluiu e apesar de possuir algumas variações, em geral é definida como uma infraestrutura de rede global e dinâmica com capacidades autoconfiguráveis baseada em protocolos de comunicação padrão e interoperáveis; *things* virtuais e reais na IoT têm identidades e atributos, são capazes de utilizar interfaces inteligentes e são integradas como uma rede de informações [Li et al. 2015]. Nas arquiteturas tradicionais, os sistemas são conhecidos como silos, onde cada aplicação é construída com suporte de uma infraestrutura de Tecnologia da Informação e Comunicação (TIC) proprietária e dispositivos dedicados. Essas soluções, por falta de integração de rede e de serviços geram redundância desnecessária e aumento dos custos. Na arquitetura da IoT, as aplicações compartilham a infraestrutura e os elementos de rede.

A tecnologia IoT agrupa uma grande quantidade de sensores, dispositivos e dados de maneira que torna possível o desenvolvimento de uma gama de aplicações em muitos domínios diferentes. Em resumo, esse domínios são divididos em três grandes grupos: (i) domínio industrial, (ii) domínio de assistência médica e bem estar, e (iii) cidades inteligentes [Borgia 2014]. Ao mesmo tempo que a IoT abre novas oportunidades de criar aplicações inovadoras, a diversidade de dispositivos e domínios de atuação gera uma infinidade de desafios. A escalabilidade e heterogeneidade são desafios inerentes ao sistema

	Domínio	Segurança/ Vigilância	Assistência Médica	Edificações	Energia e Recursos Naturais	Transporte e Mobilidade	Monitoramento do Ambiente	Serviços
Requisitos de Comunicação	Pouca/Nenhuma Mobilidade			√	√		√	
	Baixa Latência		√					
	Prioridade	√						
	Baixo Consumo de Energia					√	√	
	Monitoramento e Segurança							√
	Confiabilidade		√	√				
	Pouca Rajada de Dados			√	√		√	
	Transmissão em Massa	√	√			√		

Tabela 1. Requisitos de Comunicação dos Domínios de Cidades Inteligentes

complexo e dinâmico que é a IoT [Borgia 2014]. Para lidar com esses requisitos é necessária uma solução que contemple todas as camadas. A Tabela 1 apresenta alguns destes requisitos de comunicação de acordo com um domínio específico.

Ademais, a IoT deve funcionar com pouca ou nenhuma intervenção humana. Para isso, os objetos e a infraestrutura devem ter capacidades autonômicas como autoconfiguração, auto-organização, auto-gerenciamento, dentre outros. Além disso, a IoT deve prover um ambiente seguro e com QoS para garantir os requisitos específicos das aplicações. Esses desafios listados, podem ser compreendidos como requisitos gerais da IoT [Borgia 2014].

Há atualmente um grande esforço da academia e da indústria em busca da padronização das tecnologias envolvidas na IoT. Tecnologias não padronizadas geram fragmentação das soluções e dificultam a interação entre elas. Muitos autores evidenciam como um dos principais desafios, a necessidade de uma padronização para que a IoT possa prover melhor serviço para os usuários [Gama et al. 2012, Li et al. 2012, Borgia 2014].

Desta forma, apesar do grande número de iniciativas de padronização do paradigma IoT, ele ainda está bastante fragmentado. Para que a IoT possa ser utilizada em larga escala é necessário um esforço de colaboração entre as organizações de padronização (SDOs) para que essas soluções sejam unificadas ou que elas possam interoperar. Uma das soluções neste sentido é a oneM2M[5]. Ela é uma iniciativa conjunta entre sete SDOs (p.e., ARIB, ATIS, CCSA, ETSI, TIA, TTA, e TTC) com o objetivo principal de criar uma camada de serviço que funcione com diferentes *hardwares* e *softwares* e assim possa se conectar com vários dispositivos da IoT.

É importante observar que soluções como a oneM2M, apesar de criarem uma camada de integração para que as diferentes tecnologias possam interoperar, ainda deixam em aberto um grande desafio na área de gerenciamento de infraestrutura de rede e de dispositivos. Em relação aos dispositivos, eles possuem em sua grande maioria, restrições de processamento, armazenamento e bateria. A configuração e gerenciamento desses dispositivos é, em geral, executado manualmente e individualmente. O gerenciamento da rede também é um desafio, devido ao tamanho e diversidade de tecnologias envolvi-

[5]http://www.onem2m.org

das. Isto faz com que a utilização de protocolos convencionais de monitoramento como SNMP [Case et al. 1990], RMON [Waldbusser 2006], NETCONF [Enns et al. 2011] seja complexa ou inviável. [Sehgal et al. 2012] testaram os protocolos SNMP e NETCONF em dispositivos da IoT e verificam sua viabilidade. Mas, identificam que a grande carga de controle, como estabelecimento e segurança comprometem a utilização. Nesta perspectiva, os autores identificaram uma sobrecarga na ordem de segundos para a inicialização de uma conexão segura, considerando que grande parte dos dispositivos da IoT não suporta criptografia por *hardware*. Além disso, esses mecanismos de gerenciamento consomem em torno de 15% de memória RAM e 30% de memória ROM. OMA *Device Management* é uma solução proposta para gerência de dispositivos, no entanto é restrita a dispositivos móveis. Recentemente ela foi adaptada para M2M [Klas et al. 2014] utilizando as arquitetura COAP e REST. Uma solução de gerenciamento autonômico trata e resolve grande parte desses problemas. Este tipo de solução é um dos fundamentos deste trabalho.

3.2. Cidades Inteligentes

Grande parte dos desafios da IoT estão também incorporados na área de Cidades Inteligentes (CI). Isto porque a IoT é a principal tecnologia de informação e telecomunicação (TIC) que dá suporte à CI. No contexto de CI, a literatura mostra que ainda não existe uma definição unificada, largamente aceita e que seja utilizada como padrão. Neste sentido, há alguns trabalhos que conceitualizam CI em suas diferentes dimensões. [Nam and Pardo 2011] definem CI em três dimensões: humana, institucional e tecnológica. A dimensão humana está relacionada com o comportamento humano e seu relacionamento. Nesta perspectiva, uma Cidade Inteligente é uma mistura de elementos sociais como educação, cultura, arte, comércio. A dimensão institucional envolve governo, a população e a governança, ou seja, como o governo administra a cidade. A dimensão tecnológica se refere a infraestrutura de comunicação, computação e serviços [Yovanof and Hazapis 2009]. Há ainda autores que classificam as CIs de acordo com a infraestrutura: *baseada em infraestrutura física* - transporte, água, energia; e *baseada na infraestrutura lógica* - capital social e humano, inclusão, igualdade social [Angelidou 2014].

[Naphade et al. 2011] destacam que as CIs integram e otimizam um conjunto de sistemas interdependentes públicos e privados para atingir um novo nível de efetividade e eficiência nas cidades. De fato, isto será necessário para as grandes cidades. Estudos mostram que em 2014 mais da metade da população mundial (54%) vivia em cidades e a estimativa é que em 2050 este número suba para 70%, ou seja, algumas megacidades com mais de 10 milhões de habitantes[6]. [Naphade et al. 2011] salientam que as transformações das CIs dependem da utilização de técnicas analíticas para compreender os eventos do mundo real e assim melhorar os processos de negócio urbano. Os autores dividem esses processos em *planejamento*, *gerenciamento* e *operação*. Por planejamento, entende-se, explorar as várias fontes de informação sobre o comportamento urbano para utilizar na alocação de recursos como transporte, água, energia, etc. O gerenciamento envolve a coordenação da infraestrutura da cidade, como a criação e manutenção de ruas, saneamento, sistema de distribuição de água, energia, dentre outros. A operação deve integrar várias fontes de dados para representar os domínios urbanos e as interdependências

[6]https://esa.un.org/unpd/wup/ - divisão de população da ONU - *acessado em 07/02/2017*

em tempo real.

Um dos grandes desafios das CIs, além dos já apresentados para a IoT, está centrado na camada de aplicação que deve integrar um grande conjunto de dados providos pelos dispositivos, deve minerar esses dados com técnicas como classificação, regressão dentre outras, e deve funcionar de acordo com as políticas definidas pelo gestor da cidade em conjunto com a sociedade [Rathore et al. 2016, Pan et al. 2013]. Porém, muitos sistemas de IoT implantados são incapazes de interoperar, mesmo estando na mesma rede física (silos) [Clayman and Galis 2011]. Isto dificulta a integração e/ou o compartilhamento de dados nas CIs.

Outro desafio, é o gerenciamento dos dispositivos de rede e dos dispositivos. Apesar de já ter sido tratado no contexto de IoT, o gerenciamento nas CIs possui mais alguns níveis de complexidade. Devido à variedade de áreas de atuação, a quantidade e a diversidade de dispositivos tende a ser maior que na IoT. As tecnologias de comunicação envolvidas também são diversas. Além disso, em geral, os dispositivos e a infraestrutura devem ficar sob o domínio administrativo de uma entidade central, a administração pública da cidade.

3.3. Gerenciamento Autonômico de Redes

Computação autonômica é uma evolução fundamental do gerenciamento de infraestruturas de TIC [Horn 2001]. Esta evolução faz-se necessária devido a complexidade dos sistemas computacionais, causada pelo aumento da sofisticação dos serviços oferecidos, aumento na demanda por produtividade e qualidade, aumento da heterogeneidade de dispositivos, tecnologias e plataformas. [Kephart and Chess 2003] afirmam que a complexidade de sistemas computacionais é o principal desafio a ser solucionado pela computação autonômica.

A computação autonômica também trata do gerenciamento de recursos computacionais de forma a minimizar a intervenção humana. Em sistemas complexos, a intervenção humana pode ser um ponto de inserção de falhas à medida que a gerência é aplicada. Desta forma, é possível observar a computação autonômica como um sistema capaz de gerenciar qualquer sistema de acordo com os objetivos definidos pelo administrador [Kephart and Chess 2003].

[Kephart and Chess 2003] destacam 4 requisitos para que o auto-gerenciamento seja alcançado em uma infraestrutura: (i) autoconfiguração, (ii) auto-otimização, (iii) autocura e (iv) autoproteção.

Sistemas autônomos funcionam com uma composição de um ou mais elementos autonômicos independentes e elementos/recursos gerenciados. Os elementos autonômicos realizam o auto-gerenciamento, tomando a decisão para cada elemento, mas, em busca de um melhor estado para o sistema como um todo. Elementos gerenciados são equivalente aos encontrados em sistemas não autônomos, e são mantidos por um único gerente autonômico que o controla.

A computação autonômica é estruturada de acordo com uma arquitetura simples proposta por [Kephart and Chess 2003]. Esta arquitetura é utilizada como referência em muitos trabalhos. Ela representa o elemento gerenciado e o elemento autonômico, especificando as etapas que compõem a gerência autonômica. Esta etapa é dividida em quatro

fases: (i) monitoramento, (ii) análise, (iii) planejamento e (iv) execução. Essas fases do modelo MAPE-K funcionam em um ciclo contínuo de gerência autonômica denominado ciclo de controle (*Autonomic Control Loop* (ACL)).

De forma similar à computação autonômica, que busca desenvolver computadores que auto-gerencie, as pesquisas na área de redes autonômicas estudam novas arquiteturas de redes e protocolos a fim de desenvolvê-las para que se comportem com um certo nível de liberdade e se adaptem de forma eficiente a novas situações, sem intervenção humana direta [Agoulmine 2011]. Desde que o gerenciamento de redes se torna cada vez mais distribuído, a computação autonômica é essencial para manter esses sistemas gerenciáveis [Movahedi et al. 2012]. Os mecanismos de gerenciamento de redes tradicionais mesmo com algumas estratégias de automação, possuem uma natureza estática na tomada de decisões de gerenciamento, tornando-os ineficientes em redes grandes e complexas. A Figura 1 apresenta os cinco níveis de evolução de gerência, onde o último nível, a gerência autonômica provê a auto-configuração em resposta a mudanças ou falhas, de acordo com os objetivos e políticas pré-estabelecidas.

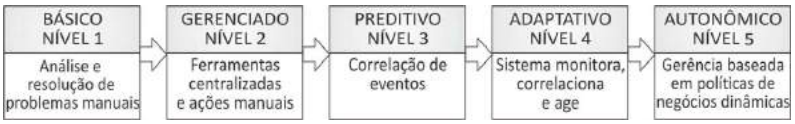

Figura 1. Níveis de evolução da gerência (Adaptado de [Kephart and Chess 2003]).

Há algumas diferenças significativas entre a computação autonômica e ANM. Enquanto a gerência de redes é mais orientada a sistemas distribuídos e serviços, a computação autonômica é voltada a aplicações e recursos computacionais. [Movahedi et al. 2012] destacam ainda que ANM é mais complexo que a computação autonômica pois os domínios heterogêneos das redes podem conflitar entre si, quando conectados. No entanto, os princípios da computação autonômica podem ser ajustados e utilizados no contexto da gerência de redes.

3.4. SDN e NFV

A infraestrutura atual da Internet enfrenta dificuldades para suprir os requisitos dos serviços atuais, dos usuários e das operadoras [Wendong et al. 2012]. Além disso ela tornou-se cara, complexa e instável [McKeown et al. 2008]. Mesmo tarefas simples como a configuração ou a aplicação de políticas exige um bom esforço, devido a falta de uma interface de controle comum para os dispositivos de rede [Nunes et al. 2014]. Neste contexto, as redes definidas por *software* (SDNs) surgem como uma nova arquitetura de rede onde o plano de controle é dissociado do plano de dados. Nesse contexto, é possível desenvolver aplicações e definir regras que executam sob uma entidade centralizada criando um ambiente altamente programável [Farhady et al. 2015]. Esse paradigma de rede permite que os administradores façam a gerência dos serviços de rede através de abstrações das funcionalidades de baixo nível dos dispositivos. [Sezer et al. 2013] destacam as diferenças entre as redes tradicionais e as redes definidas por software.

Em uma rede SDN há três partes bem definidas: aplicação, plano de controle e plano de dados [Farhady et al. 2015]. Há duas iniciativas importantes que trabalham na especificação e desenvolvimento da arquitetura SDN. Uma delas é a Open

Network Foundation (ONF) que desenvolve o protocolo OpenFlow. A outra iniciativa é do grupo de trabalho do IETF ForCES (Forwarding and Control Element Separation) [Nunes et al. 2014]. Apesar de usarem a mesma estratégia de separar o plano de controle do plano de dados, no ForCES o dispositivo de rede ainda é representado como uma entidade única. No entanto, o protocolo mais utilizado pela indústria e com maior alvo de pesquisa pela academia é o OpenFlow.

Implementar novos serviços nas redes atuais está se tornando um desafio, devido a vários fatores, como a natureza proprietária dos dispositivos de hardware, a falta de profissionais qualificados para integrar e manter esses serviços e o custo para oferecer espaço físico e energia para vários dispositivos [Han et al. 2015]. Para diminuir esses problemas a tecnologia NFV vem sendo desenvolvida, juntamente com outras tecnologias com SDN e Cloud Computing.

NFV implementa funções de rede através de técnicas de virtualização de *software* e os executa em *hardware* de propósito geral. Essa abordagem permite a redução de custos com equipamentos de rede de alto desempenho além de acelerar a implantação de novos serviços. As ferramentas virtuais na NFV podem ser instanciadas sob demanda sem a necessidade de instalar novos equipamentos [Han et al. 2015]. Vários serviços de rede podem ser virtualizados com NFV, como *Network Address Translation* (NAT), *firewall*, *Content Delivery Networks* (CDN), *Deep Packet Inspection* (DPI), *Domain Name System* (DNS), dentre outros.

4. AutoManIoT: Visão Geral da Arquitetura

Visando resolver os problemas de gerenciamento em IoT e especificamente em cidade inteligentes, está seção apresenta uma visão do funcionamento da arquitetura autonômica proposta, denominada AutoManIoT. AutoManIoT é fortemente baseada em SDN e NFV visando alcançar uma maior escalabilidade e flexibilidade na solução proposta. A Figura 2 apresenta uma visão geral da arquitetura proposta.

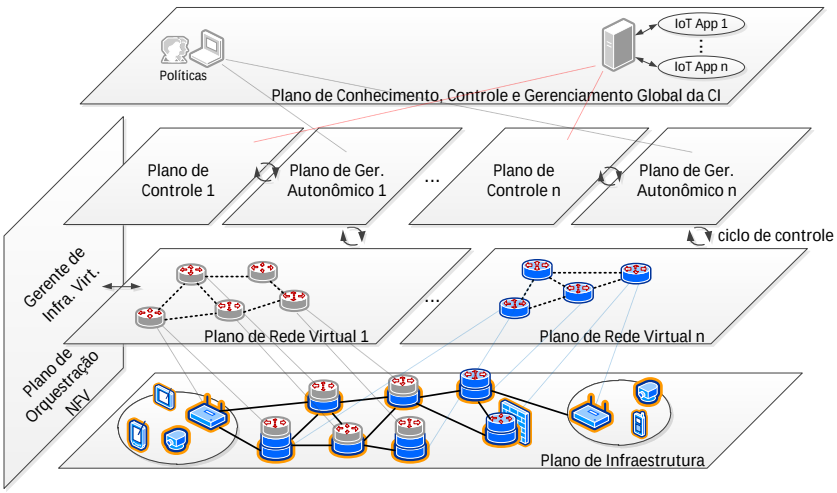

Figura 2. Visão geral da arquitetura

A arquitetura utiliza os conceitos de SDN e NFV como infraestrutura de implanta-

ção. Ao utilizar esses conceitos, os componentes: (i) *controlador SDN*, (ii) *orquestrador, gerente VNF e gerente de infraestrutura virtualizada* devem fazer parte da arquitetura.

No contexto do gerenciamento autonômico, independentemente da estratégia de autonomia utilizada, a arquitetura deve ser composta pelo (iii) *ciclo de controle autonômico* para a gerência dos recursos de rede e dispositivos. Quando se trata do gerenciamento autonômico é necessário ainda um (iv) *elemento de gerência de políticas de rede*.

Baseado nas discussões no grupo Anima da IETF, a adoção da API GRASP[7] (Generic Autonomic Signaling Protocol Application Program Interface) é utilizada como norteadora para definir a troca de mensagens entre agentes de serviços autonômicos. Além disso, o draft "An Autonomic Control Plane[8]", pertencente também ao grupo Anima, surge como base para o desenvolvimento do Plano de Gerência Autonômica da arquitetura proposta.

Por fim, para possibilitar a integração entre as aplicações IoT, independente da tecnologia utilizada e também para permitir a orquestração da rede de acordo com os requisitos das aplicações foi introduzido o modelo centrado em dados DDS (Data Distribution Service[9])

A arquitetura é apresentada em uma visão de camadas/planos descritos a seguir:

- *Plano de Infraestrutura:* Provê os recursos físicos necessários para a instanciação das funções virtuais.
- *Plano de Rede Virtual:* Este plano representa a instanciação da infraestrutura de rede virtual, atividade que será implementada no *framework* e que é controlada pelo plano de orquestração NFV. O plano de gerência virtual é gerenciado pelo gerente de infraestrutura virtual, semelhante ao modelo adotado na arquitetura ETSI NFV.
- *Plano de Orquestração NFV:* Esta camada é responsável por orquestrar as VNFs e gerenciar a infraestrutura virtualizada.
- *Plano de Controle:* Este plano provê os mecanismos de controle das funções de rede virtualizadas situadas no plano de rede virtual.
- *Plano de Gerência Autonômica:* Esta camada provê os mecanismos de auto-gerenciamento da rede. Ele monitora os dispositivos de rede virtual através da coleta de informações pertinentes à rede, processa através do ciclo autonômico e por fim aplica as devidas alterações na rede virtual.
- *Plano de Conhecimento, Controle e Gerenciamento Global da CI:* Por fim, o plano de conhecimento, controle e gerenciamento global da CI é responsável por manter e distribuir as políticas globais de gerenciamento das redes.

4.1. Detalhamento da Arquitetura

A Figura 3 apresenta o detalhamento da arquitetura através de seus componentes internos e a interação entre eles.

A camada de infraestrutura da arquitetura segue o paradigma NFV, segundo a especificação ETSI NFV [ETSI 2013]. Deste modo, a infraestrutura virtualizada da ar-

[7]https://datatracker.ietf.org/doc/html/draft-liu-anima-grasp-api
[8]https://datatracker.ietf.org/doc/html/draft-ietf-anima-autonomic-control-plane
[9]http://portals.omg.org/dds/

quitetura NFV permanece inalterada sendo gerenciada pelo gerente de infraestrutura virtualizada. As alteração ocorrem nas funções de rede virtualizadas que passam a ser controladas pelo controlador SDN na camada de controle. O orquestrador também passa a ser controlado pelo controlador SDN.

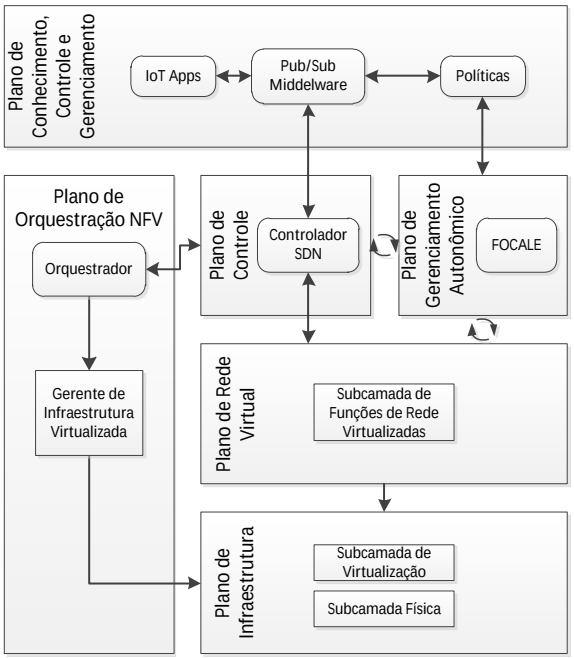

Figura 3. Arquitetura com subcamadas e componentes

O gerenciamento autonômico dos dispositivos virtuais é executado através do ciclo de controle semelhante ao ciclo do FOCALE [Strassner et al. 2006]. No entanto, como a arquitetura proposta utiliza SDN/NFV, os modelos de informação, a tradução desses modelos e a ontologia são simplificados pelas interfaces de operação das tecnologias SDN/NFV (interface SDN/NFV). Além disso, foi incluído um novo processo de decisão, no qual, se o dispositivo não suportar/comportar alguma configuração, uma nova função de rede virtualizada é criada. A Figura 4 a seguir apresenta as tecnologias SDN/NFV incorporadas ao ciclo de controle FOCALE.

Para prover os requisitos das aplicações da CI de acordo com o seu domínio, foi incorporado à arquitetura o *middleware* centrado em dados DDS da OMG[10]. Como é possível observar na Figura 5, o controlador SDN publica as notificações de chegada de pacote através do DDS DataReader (DR) listener quando há pacotes em seu manipulador de pacotes. Já o encaminhador de pacotes SDN envia os pacotes armazenados no DDS DataWriter (DW) listener. Esses pacotes são criados pelas aplicações IoT. Por fim, o serviço de programação de fluxo é utilizado para definir as regras de fluxo nos switches Openflow através das políticas definidas pelo administrador da infraestrutura da CI. A definição de políticas específicas para um determinado domínio de aplicação da

[10]http://portals.omg.org/dds

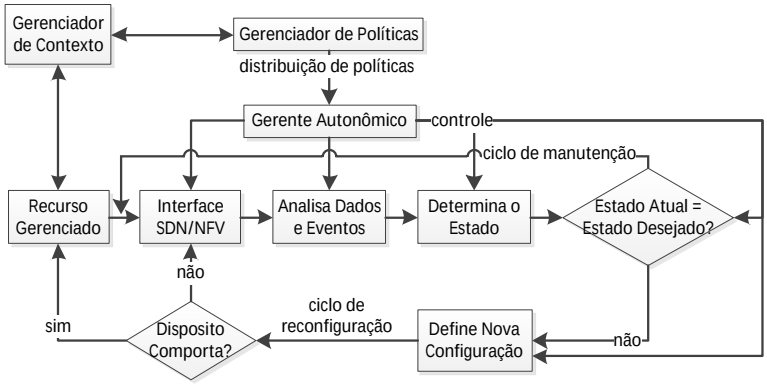

Figura 4. Ciclo de controle FOCALE e SDN/NFV

IoT é possível através do mecanismo de **partição** provido pelo DDS. Desta forma, cada domínio da IoT pertence a uma determinada partição DDS que terá uma programação específica do controlador SDN.

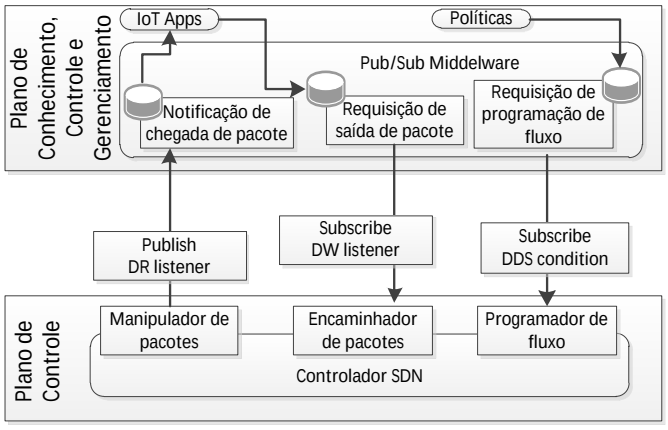

Figura 5. Middleware DDS (Adaptado de [Hakiri et al. 2015])

5. Conclusão

Este trabalho propõem a arquitetura AutoManIoT para gerenciamento autonômico de rede no contexto da IoT e das CIs. Esta arquitetura é construída empregando os recursos de SDN e NFV e considera a disposição eficiente dos dispositivos de controle e gerência autonômica, observando a natureza distribuída e de grande escala da IoT e CIs. A arquitetura se encontra na etapa de validação com a elaboração de experimentos com conjuntos de dados reais, simulando o comportamento da IoT e CIs, e verificação da eficiência da gerência autonômica considerando a utilização dessas tecnologias.

5.1. Trabalhos Futuros

Com o objetivo de validar e avaliar a arquitetura proposta pretende-se executar a experimentação do *framework* proposto. Serão utilizados conjuntos de dados reais de diversos domínios de aplicação de CIs (vide Tabela 2 [Rathore et al. 2016]). Esses dados irão alimentar o tráfego gerado para o *framework* que estará sendo executado no PlanetLab. Neste ambiente, será avaliado o custo computacional e a escalabilidade do *framework*. Ainda, serão elaborados diferentes ambientes para avaliar a eficiência e custo dos mecanismos de autonomia e a capacidade de provisão dos requisitos das aplicações IoT.

#	Conjunto de Dados	Tamanho	Qtd. de Parâmetros
1	Enchente	16MB	30
2	Uso de Água	5MB	11
3	Tráfego de veículos (Madrid)	450MB	5
4	Medição de Mobilidade Veicular	4.03GB	5
5	Estacionamentos	2.94KB	7
6	Poluição	32GB	8
7	Redes Sociais	16MB	7
8	Tráfego de veículos (Aarhus)	33GB	9
9	Clima	3MB	7

Tabela 2. Detalhamento dos Conjuntos de Dados

Além disso, pretende-se implementar uma ferramenta de caracterização de tráfego para IoT, de forma que ela possa prover informações para o plano de conhecimento, como novos requisitos de rede de acordo com o domínio da aplicação.

Referências

Agoulmine, N. (2011). *Autonomic Network Management Principles: From Concepts to Applications*. Elsevier, Burlington, MA.

Angelidou, M. (2014). Smart city policies: A spatial approach. *Cities*, 41, Supplement 1:S3 – S11. Current Research on Cities.

Ashton, K. (2009). That 'internet of things' thing. `http://www.rfidjournal.com/articles/view?4986`. Accessed: 2016-03-20.

Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Computer Networks*, 54(15):2787 – 2805.

Behringer, M., Pritikin, M., Bjarnason, S., Clemm, A., Carpenter, B., Jiang, S., and Ciavaglia, L. (2015). Autonomic networking: Definitions and design goals. Technical Report 7575, IETF.

Borgia, E. (2014). The internet of things vision: Key features, applications and open issues. *Computer Communications*, 54:1 – 31.

Case, J., Fedor, M., Schoffstall, M., and Davin, J. (1990). Simple network management protocol (snmp). Technical Report 1157, IETF.

Chaparadza, R., Menem, T. B., Strassner, J., Radier, B., Soulhi, S., Ding, J., and Yan, Z. (2014). Industry harmonization for unified standards on autonomic management

amp; control (amc) of networks and services, sdn and nfv. In *2014 IEEE Globecom Workshops (GC Wkshps)*, pages 155–160.

Clayman, S. and Galis, A. (2011). Inox: A managed service platform for inter-connected smart objects. In *Proceedings of the Workshop on Internet of Things and Service Platforms*, IoTSP '11, pages 2:1–2:8, New York, NY, USA. ACM.

Dobson, S., Denazis, S., Fernández, A., Gaïti, D., Gelenbe, E., Massacci, F., Nixon, P., Saffre, F., Schmidt, N., and Zambonelli, F. (2006). A survey of autonomic communications. *ACM Trans. Auton. Adapt. Syst.*, 1(2):223–259.

Enns, R., Bjorklund, M., Schoenwaelder, J., and Bierman, A. (2011). Network configuration protocol (netconf). Technical Report 6241, IETF. Updated by RFC 7803.

ETSI (2013). Network functions virtualization (nfv); architecture framework. Technical report, ETSI GS NFV 002 v1.1.1.

Farhady, H., Lee, H., and Nakao, A. (2015). Software-defined networking: A survey. *Computer Networks*, 81:79 – 95.

Gama, K., Touseau, L., and Donsez, D. (2012). Combining heterogeneous service technologies for building an internet of things middleware. *Comput. Commun.*, 35(4):405–417.

Hakiri, A., Berthou, P., Gokhale, A., and Abdellatif, S. (2015). Publish/subscribe-enabled software defined networking for efficient and scalable iot communications. *IEEE Communications Magazine*, 53(9):48–54.

Han, B., Gopalakrishnan, V., Ji, L., and Lee, S. (2015). Network function virtualization: Challenges and opportunities for innovations. *IEEE Communications Magazine*, 53(2):90–97.

Horn, P. (2001). Autonomic computing: Ibm\'s perspective on the state of information technology. *Manifesto*.

Kephart, J. O. and Chess, D. M. (2003). The vision of autonomic computing. *Computer*, 36(1):41–50.

Klas, G., Rodermund, F., Shelby, Z., Akhouri, S., and Höller, J. (2014). Lightweight m2m: Enabling device management and applications for the internet of things. Technical report, White Paper.

Li, S., Xu, L., Wang, X., and Wang, J. (2012). Integration of hybrid wireless networks in cloud services oriented enterprise information systems. *Enterprise Information Systems*, 6(2):165–187.

Li, S., Xu, L. D., and Zhao, S. (2015). The internet of things: A survey. *Information Systems Frontiers*, 17(2):243–259.

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). Openflow: Enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74.

Movahedi, Z., Ayari, M., Langar, R., and Pujolle, G. (2012). A survey of autonomic network architectures and evaluation criteria. *IEEE Communications Surveys Tutorials*, 14(2):464–490.

Nam, T. and Pardo, T. A. (2011). Conceptualizing smart city with dimensions of technology, people, and institutions. In *Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times*, dg.o '11, pages 282–291, New York, NY, USA. ACM.

Naphade, M., Banavar, G., Harrison, C., Paraszczak, J., and Morris, R. (2011). Smarter cities and their innovation challenges. *Computer*, 44(6):32–39.

Nunes, B. A. A., Mendonca, M., Nguyen, X. N., Obraczka, K., and Turletti, T. (2014). A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications Surveys Tutorials*, 16(3):1617–1634.

Pan, G., Qi, G., Zhang, W., Li, S., Wu, Z., and Yang, L. T. (2013). Trace analysis and mining for smart cities: issues, methods, and applications. *IEEE Communications Magazine*, 51(6):120–126.

Qi, Q., Wang, W., Gong, X., and Que, X. (2014). A sdn-based network virtualization architecture with autonomie management. In *2014 IEEE Globecom Workshops (GC Wkshps)*, pages 178–182.

Rathore, M. M., Ahmad, A., Paul, A., and Rho, S. (2016). Urban planning and building smart cities based on the internet of things using big data analytics. *Computer Networks*, 101:63 – 80. Industrial Technologies and Applications for the Internet of Things.

Savaglio, C. and Fortino, G. (2015). *Autonomic and Cognitive Architectures for the Internet of Things*, chapter 1, pages 39–47. Springer International Publishing, Cham.

Sehgal, A., Perelman, V., Kuryla, S., and Schonwalder, J. (2012). Management of resource constrained devices in the internet of things. *IEEE Communications Magazine*, 50(12):144–149.

Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B., Lake, D., Finnegan, J., Viljoen, N., Miller, M., and Rao, N. (2013). Are we ready for sdn? implementation challenges for software-defined networks. *IEEE Communications Magazine*, 51(7):36–43.

Strassner, J., Agoulmine, N., and Lehtihet, E. (2006). Focale: A novel autonomic networking architecture. In *Latin American Autonomic Computing Symposium (LAACS)*, Campo Grande, MS, Brazil.

Waldbusser, S. (2006). Remote network monitoring management information base version 2. Technical Report 4502, IETF.

Wendong, W., Yannan, H. U., Que, X., and Xiangyang, G. (2012). Autonomicity design in openflow based software defined networking. In *2012 IEEE Globecom Workshops*, pages 818–823.

Yovanof, G. S. and Hazapis, G. N. (2009). An architectural framework and enabling wireless technologies for digital cities & intelligent urban environments. *Wireless Personal Communications*, 49(3):445–463.

Zanella, A., Bui, N., Castellani, A., Vangelista, L., and Zorzi, M. (2014). Internet of things for smart cities. *IEEE Internet of Things Journal*, 1(1):22–32.