Redes Bayesianas para a Detecção de Violação de SLA em Infraestrutura como Serviço

Fernando Schubert[1], **Rafael Mendes**[1], **Carlos Becker Westphall**[1]

[1]Laboratorio de Redes e Gerência – Universidade Federal de Santa Catarina (UFSC)
Caixa Postal 476 - 88040-970 - Florianópolis - SC - Brasil

{schubert,mendes,westphal}@inf.ufsc.br

Abstract. *In this paper an approach for virtual instances monitoring and service level violation detection is proposed based on bayesian networks in the infrastructure as a service context.*

Resumo. *Neste trabalho propõe-se uma abordagem para o monitoramento e detecção de falhas no nível de serviço em instâncias virtuais a partir de uma rede bayesiana no contexto de infraestrutura como serviço.*

1. Introdução

A computação em nuvem está transformando a indústria de tecnologia da informação. Ao invés das empresas e centros de pesquisa necessitarem grandes parques de máquinas com super computadores e *clusters*, a computação em nuvem possibilita a locação de recursos em provedores que possuem capacidade aparentemente infinita sob a perspectiva do usuário.

A vantagem da Nuvem em relação ao *data center* tradicional é a capacidade de expandir seus recursos e otimizar a sua utilização, característica conhecida como elasticidade [Mell 2011]. Esta elasticidade possibilita ao usuário da nuvem obter recursos e reciclar estes recursos quando não mais utilizados, pagando apenas pelo período em que efetivamente os utilizou.

De acordo com uma das definições mais aceitas de computação em nuvem, [Mell 2011] define a nuvem como um modelo para habilitar o acesso à rede de forma ubíqua, conveniente e sob demanda, a um conjunto de recursos computacionais compartilhados (por exemplo, redes, servidores, armazenamento, aplicações, serviços) que podem ser rapidamente provisionados e removidos com um esforço mínimo de gerenciamento e interação dos provedores de serviços. Define também os modelos de serviço, brevemente descritos na Figura 1.

Os serviços computacionais necessitam ser altamente confiáveis, escaláveis, e autonômicos para suportar o acesso ubíquo, descoberta e capacidade de composição de serviços dinâmica [Buyya et al. 2008].

Diversos desafios de pesquisa emergem da computação em nuvem. Entre eles, a manutenção da qualidade de serviço (QoS, *Quality of Service*) e detecção de violações nos contratos de nível de serviço (SLA, *Service Level Agreement*) surgem como aspectos relevantes de pesquisa e desenvolvimento [Zhang et al. 2010]. A qualidade de serviço possui uma função crucial nos sistemas orientados a serviço, como a nuvem. Os contratos

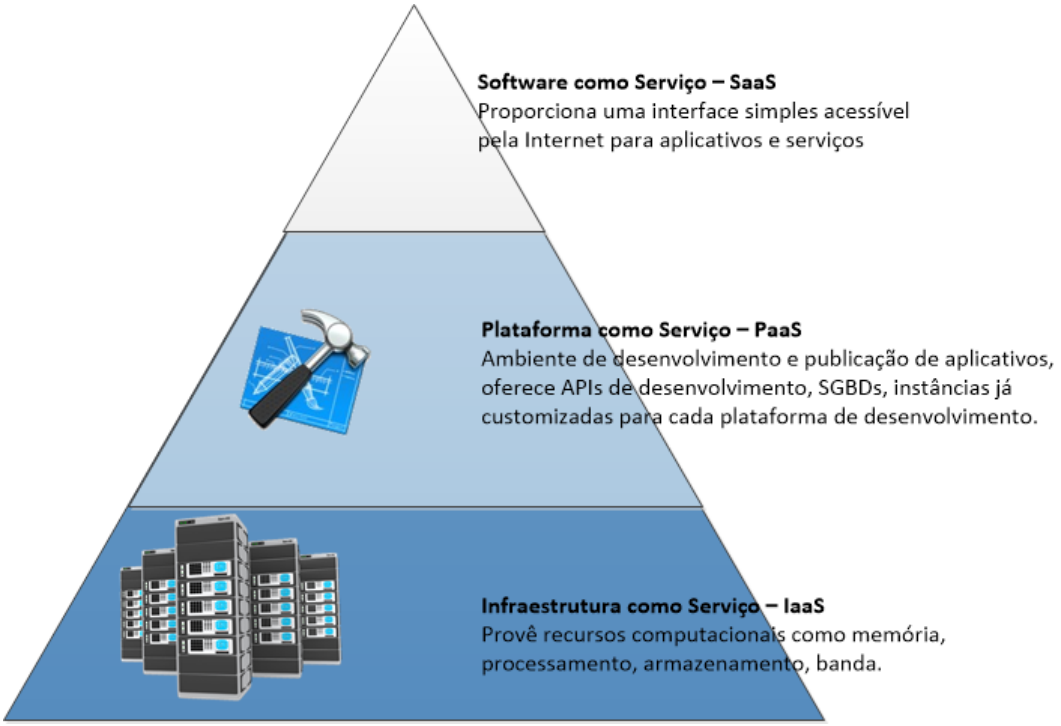

Figura 1. Modelos de serviço da computação em nuvem

de nível de serviço são necessários para definir a qualidade de serviço esperada entre o provedor e o consumidor dos recursos [Michlmayr et al. 2009].

O presente trabalho visa endereçar o problema da manutenção da qualidade de serviço através da detecção das violações no SLA acordado entre provedor e consumidor. A partir de um modelo constituído por uma base de monitoramento e um sistema estatístico probabilístico, construiu-se uma rede *bayesiana* sobre a qual inferências são efetuadas para verificar a manutenção dos contratos de nível de serviço de uma dada instância virtual em um ambiente de infraestrutura como serviço.

A principal contribuição do trabalho é a utilização de sistemas probabilísticos na inferência e detecção de violações nos contratos de nível de serviço. Esta avaliação e detecção se dá através do aprendizado da rede bayesiana a partir de dados históricos coletados de variáveis de uma instância.

O trabalho está organizado da seguinte forma: na seção 2 o problema de pesquisa é fundamentado; na seção 3 trabalhos correlatos abordando detecção de SLAs na nuvem são analisados, na seção 4 o modelo proposto é apresentado, na seção 5 os resultados são demonstrados, na seção 6 a conclusão do trabalho é apresentada.

2. Contextualização do Problema

A escalabilidade de recursos pode ser facilmente efetuada por um operador humano, porém esta opção deixou de ser prática devido ao crescente tamanho das nuvens e o compartilhamento da infraestrutura por várias aplicações e consumidores. A automação da detecção de variações na utilização dos recursos e violação de SLA tornou-se uma necessidade. Esta automação necessita levar em conta o histórico de performance, gargalos de performance, os SLAs vigentes e a conservação de recursos [Hormozi et al. 2012].

O objetivo do trabalho é detectar mudanças na qualidade de serviço de uma instância virtual em uma nuvem, a partir do monitoramento dos seus recursos básicos, como memória, carga, armazenamento e consumo de banda. O objetivo está dividido nos seguintes itens:

a) Identificar os objetos de nível de serviço (SLO, *Service Level Objects*);
b) Definir os limites aceitáveis para a correta operação da instância para cada SLO.
c) Modelar um sistema especialista com base no histórico de monitoração dos SLOs.
d) Apresentar resultados de inferências sobre o sistema especialista de acordo com uma matriz de confusão que representa as regras do SLA.

Tipicamente, os SLAs garantem boa parte dos aspectos da entrega de serviços, incluindo tanto aspectos tecnológicos quanto aspectos referentes aos serviços prestados ao cliente. As garantias de serviços ao cliente geralmente incluem itens como disponibilidade dos recursos de suporte e tempo de resposta para requisições de atendimento. As garantias tecnológicas podem incluir garantias de tempo máximo para a resolução de erros, tempo de resposta do sistema, e quase sempre incluem garantias da disponibilidade do sistema e *uptime* [Hormozi et al. 2012].

Os contratos de nível de serviço são definidos de acordo com as necessidades do cliente, que podem ser mais estritas em relação a um parâmetro, por exemplo, necessidade de poder de processamento para atender a uma carga de trabalho elevada *jobs* extensos, ou grande quantidade de memória para adicionar tabelas de um banco de dados em memória.

O problema apontado está na necessidade de se avaliar os dados de monitoramento coletados nas instâncias ou máquinas virtuais e avaliar de forma robusta e eficiente estes dados extraindo informações capazes de detectar flutuações nos contratos de garantia de serviço contratados entre o provedor de computação em nuvem e o usuário.

3. Trabalhos Correlatos

Detecção e monitoração das variações de um SLA tem sido objetivo de pesquisas na área de redes e sistemas distribuídos, recentemente também objeto de estudo na computação em nuvem, que adicionou novos desafios à monitoração e na definição de SLAs.

A abordagem proposta por [Michlmayr et al. 2009] é constituída pela monitoração em dois escopos diferentes, no cliente e no servidor dos atributos de QoS que representam os SLOs definidos para aplicações Web, em nivel de SaaS. A proposta trabalha no nível de SaaS utilizando o arcabouço de controle de SLAs e QoS VRESCo (detalhes sobre o arcabouço VRESCo estão contidos em [Michlmayr et al. 2010]), responsável pelo controle de eventos, seleção e composição de serviços baseado nos requisitos de QoS, entre outros, sendo utilizado para centralizar em banco de dados as informações coletadas e auferir com base nos dados de monitoração e SLAs acordados a violação ou não das restrições de QoS. A monitoração do servidor e feita com base nos contadores de performance do sistema operacional Microsoft Windows, enquanto o cliente é monitorado a partir do *toolkit* QUATSCH, que efetua requisições de prova ao servidor e avalia atributos de QoS como tempo de resposta e disponibilidade do serviço. A infraestrutura necessaria é relativamente complexa, com pelo menos duas instâncias dedicadas para monitoração e análise dos dados.

Em [Eyraud-Dubois 2013], o problema da alocação de recursos e gerenciamento de violações de SLA é abordado com o desenvolvimento de uma variação do problema

do empacotamento, onde cada máquina ou instância virtual é modelada como um ítem apresentando um tamanho (ou peso), variando enre (0,1] e cada maquina física é modelada como uma caixa com capacidade total igual a 1. O problema trabalha a alocação de recursos para controlar as oscilações de demanda e possibilitar a melhor relação entre os recursos contratados e alocados, visando maior lucratividade para o provedor de recursos, reduzindo de forma consistente a margem de segurança de cada instância sem ferir o SLA contratado.

A proposta em [Andreolini et al. 2010] apresenta a utilização de perfis de carga utilizando a carga (processamento) das máquinas físicas para a realocação das instâncias, satisfazendo assim o SLA. Os perfis de carga utilizam o processamento como entrada, avaliados conforme um modelo estatístico baseado em CUSUM (veja mais em [Page 1957]). Nesta abordagem os limiares (*thresholds*) tradicionalmente definidos em SLOs não são utilizados.

As técnicas avaliadas abordam aspectos da monitoração e em alguns casos predição de cargas de trabalho. A proposta do trabalho se sobressai pela sua simplicidade e baixa complexidade em relação a [Michlmayr et al. 2009] e [Eyraud-Dubois 2013], maior representatividade dos elementos monitorados em relação à [Andreolini et al. 2010] e um foco especifico na detecção de falhas de nível de serviço.

4. Modelo Proposto

Os SLAs contém geralmente objetivos relacionados com a performance estatística que a aplicação ou serviço deve entregar. É importante que o tempo de resposta do modelo de detecção responda de maneira realista. Isto requer que a noção de distribuição de probabilidade esteja presente neste modelo [Zhang 2007]. As redes bayesianas são um arcabouço de probabilidade e satisfazem naturalmente este critério.

Tendo em vista a necessidade de um modelo que avalie o estado atual da instância e detecte flutuações no SLA, propõe-se um modelo que utilize redes bayesianas como mecanismo de diagnóstico/detecção de violações das cláusulas dos SLAs. As redes bayesianas foram escolhidas pois, neste trabalho, será abordada a aleatoriedade contida no problema das violações de SLAs que, efetivamente, dependem da atenção e disposição humana em diagnosticar e dar consequência às violação. Ainda, as redes bayesinas disponibilizam um robusto ferramental matemático para propagação das crenças, bem como, servem para implementação para sistemas autonômicos, permitindo inclusive, a utilização de técnicas de aprendizagem de máquina [Friedman et al. 1999] aprendizagem da topologia e probabilidades *a priori*.

A escolha pelas redes bayesianas deve-se também à sua facilidade de interpretação e modelagem intuitiva, aumentando a confiança na correção do modelo e das funções de gerenciamento baseadas no mesmo [Zhang 2007].

Os nós da rede representam os itens monitorados e nós de saída, sendo os itens monitorados:

- Memória: utilização de memória da instância em um dado momento.
- Processamento (CPU): utilização de processamento da instância em um dado momento.

Tabela 1. Nós da rede.

Nó	Estado	Limiar
MEMÓRIA	OK	menor que 80% da capacidade
	WARNING	maior igual a 80% e menor que 90% da capacidade
	CRITICAL	maior igual a 90% da capacidade
PROCESSAMENTO	OK	menor que 90%
	WARNING	maior igual a 90% e menor que 100%
	CRITICAL	igual a 100%
ARMAZENAMENTO	OK	menor que 90%
	WARNING	maior igual a 90% e menor que 95%
	CRITICAL	maior igual a 95%
CONSUMO DE BANDA	OK	menor que 90%
	WARNING	maior igual a 90% e menor que 95%
	CRITICAL	maior igual a 95%
HOST	UP	Host Ativo
	DOWN	Host Inativo
SLA	VIOLADO	Contrato Quebrado
	MANTIDO	Contrato Preservado
SLA GERAL	VIOLADO	Contrato Quebrado
	MANTIDO	Contrato Violado

- Armazenamento: utilização do espaço disponível de armazenamento em um dado momento.
- Consumo de banda: utilização da largura de banda da instância em um dado momento.
- Host: disponibilidade da instância.

Já os nós calculados dinamicamente pela rede bayesiana são:

- SLA: contrato de nível de serviço dos itens de monitoração da instância.
- SLA Geral: contrato de nível de serviço levando em conta a disponibilidade da instância e parâmetros de monitoração dos recursos.

A Tabela 1 apresenta os limiares definidos para cada item monitorado, que representa um SLO a ser monitorado e garantido.

A rede bayesiana foi montada conforme a Figura 2 de forma a inicialmente considerar os nodos básicos de monitoramento de uma máquina ativa. Após o resultado do SLA básico o SLA GERAL é calculado utilizando como base o SLA dos itens ativos de monitoramento e o estado atual da instância.

5. Resultados

A rede bayesiana foi construída utilizando o arcabouço de desenvolvimento de redes bayesianas Netica, uma das principais ferramentas para a construção de sistemas de inferência probabilísticos, segundo [Hope et al. 2002].

O treinamento da rede se deu a partir da utilização de um conjunto de 100 casos que representam o monitoramento constante de uma instância virtual, contendo os dados

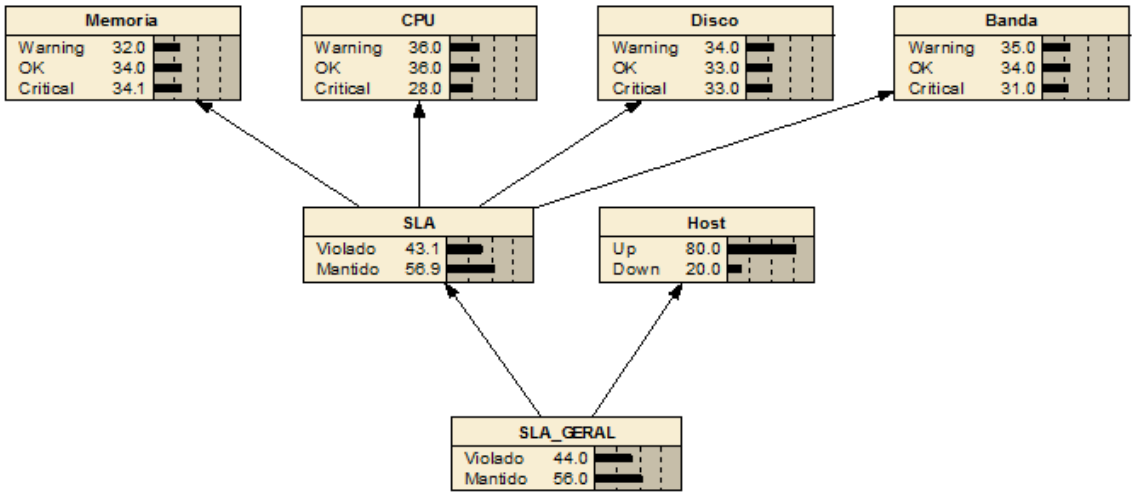

Figura 2. Representação da rede bayesiana e suas conexões

em fatias de tempo de 5 minutos e em situações simuladas de processamento e disponibilidade. Relatórios obtidos a partir do treinamento baseado em Gradiente Descendente [Baldi 1995] e apresentados na Tabela 2.

A diferença no cálculo de probabilidades atingiu 0.0 pontos após 22 ciclos de treinamento, ou seja, o estado ótimo. A partir deste ponto a continuação no treinamento não trará melhorias na performance da rede. Probabilidades iniciais calculadas com base no treinamento estão detalhadas na Tabela 3.

Após a construção e treinamento da rede bayesiana, um conjunto de inferências sera realizado para validar a rede. A Tabela 4 representa a matriz de regras contendo as regras de inferência e o resultado esperado dados os valores de entrada nos nós. As regras de inferência da Tabela 4 representam as definições dos SLAs para a instância.

Com base na matriz de confusão apresentada na Tabela 4, foram efetuadas inferências sobre a rede bayesiana implementada. A Tabela 5 apresenta os resultados obtidos a partir da simulação dos valores da matriz de confusão.

Pode-se verificar que os resultados apresentaram correspondência em todos os itens avaliados, de acordo com as regras. Para verificar a robustez da rede foi adicionado um ruido de 10 % à rede, enfraquecendo as relações para avaliar a resposta da rede às inferências. Inicialmente adicionou-se um ruído de 10% - reduzindo a probabilidade da opção OK e ampliando a probabilidade de CRITICAL em todos os nós da rede. A nova configuração das probabilidades nos nós pode ser visualizada na Figura 3.

A tabela 5 traça um comparativo entre o resultado esperado para a rede e o resultado obtido com 10% de ruído. Utilizou-se para tal os casos presentes na matriz de confusão (Tabela 4).

Pode-se observar a partir da Tabela 5 que mesmo com 10% de enfraquecimento nos nós de entrada de monitoração, mais especificamente enfraquecendo as probabilidades para os valores classificados na categoria ”OK”em todos os nós, a detecção final do SLA não sofre degradação correspondente. Considerou-se para os testes a saída da rede com o nó HOST sempre em UP. Após o enfraquecimento em 10% foi realizado um en-

Tabela 2. Iterações de treinamento da rede.

Iteração	Probabilidade	Mudança %
0	6,50737	
1	5,89169	9,4613
2	5,73948	2,9229
3	5,58664	2,3226
4	5,56284	0,4260
5	5,53936	0,4221
6	5,52387	0,2797
7	5,51191	0,2165
8	5,51007	0,0335
9	5,5082	0,0338
10	5,50659	0,0294
11	5,5058	0,0142
12	5,50542	0,0069
13	5,50533	0,0016
14	5,50526	0,0013
15	5,50522	0,0007
16	5,5052	0,0003
17	5,50516	0,0007
18	5,50516	0,0001
19	5,50516	0,0001
20	5,50516	0,0001
21	5,50514	0,0000
22	5,50515	-0,0000

fraquecimento de 20% em OK e fortalecimento em CRITICAL para verificar a robustez da rede.

A Tabela comparativo entre o resultado esperado para a rede e o obtido com 20% de ruído. Utilizou-se para tal os casos presentes na matriz de regras.

Pode-se observar a partir da Tabela 7 que mesmo com 20% de enfraquecimento nos nós de entrada de monitoração, o SLA não sofre degradação correspondente, na maioria dos casos, tendo apresentado melhora na manutenção do SLA nos casos 1 e 4, e uma degradação de aproximadamente 10% nos casos 3 e 5, ficando inalterado no caso 2. Isto demonstra a capacidade e robustez das redes bayesianas para a detecção probabilística de casos a partir do conteúdo aprendido.

6. Conclusão

A detecção e predição de variações dos atributos de QoS, impactando consequentemente no SLA, em instâncias virtuais, tem sido foco de pesquisa na computação em nuvem, devido à relevância do tópico tanto para provedores quanto consumidores da nuvem.

Para provedores, a detecção prematura de possíveis variações na performance das

Tabela 3. Probabilidades dos nós da rede.

Nó	Estado	Limiar
MEMÓRIA	OK	0.34
	WARNING	0.32
	CRITICAL	0.34
PROCESSAMENTO	OK	0.36
	WARNING	0.36
	CRITICAL	0.28
ARMAZENAMENTO	OK	0.33
	WARNING	0.34
	CRITICAL	0.33
CONSUMO DE BANDA	OK	0.34
	WARNING	0.35
	CRITICAL	0.31
HOST	UP	0.8
	DOWN	0.2
SLA	VIOLADO	0.4311
	MANTIDO	0.5890
SLA GERAL	VIOLADO	0.4400
	MANTIDO	0.5600

Tabela 4. Matriz de confusão.

Caso	Memoria	Processamento	Armazenamento	Banda	Host	SLA Geral
1	WARNING	CRITICAL	WARNING	WARNING	UP	MANTIDO
2	OK	OK	OK	OK	DOWN	VIOLADO
3	OK	WARNING	CRITICAL	CRITICAL	UP	VIOLADO
4	CRITICAL	CRITICAL	WARNING	OK	UP	MANTIDO
5	WARNING	WARNING	WARNING	WARNING	UP	MANTIDO

Tabela 5. Resultados dos casos de teste da Matriz de Confusão

Caso	SLA Geral	Mantido %	Violado %
1	MANTIDO	66,2	33,8
2	VIOLADO	19,6	80,4
3	VIOLADO	28,2	71,8
4	MANTIDO	84	16
5	MANTIDO	75,5	24,5

instâncias pode gerar uma grande economia em penalidades e custos de imagem, a partir da tomada de decisão e de ações que minimizem o impacto da violação do SLA e recuperem o estado normal de funcionamento no menor tempo possível. Além disso, a análise

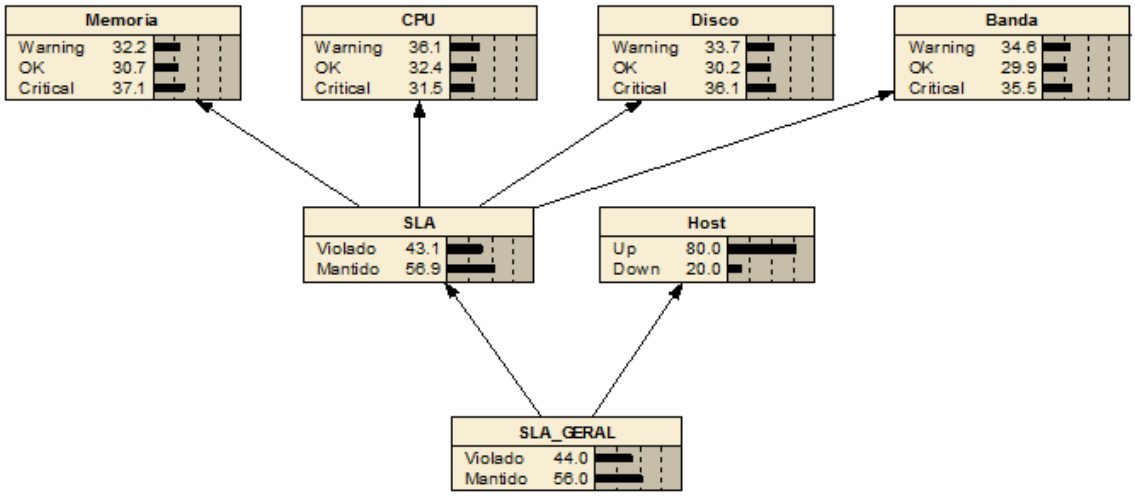

Figura 3. Representação da rede bayesiana com ruido de 10 %

Tabela 6. Resultados dos casos de teste com ruído de 10%

Caso	SLA Geral	Mantido %	Violado %
1	MANTIDO	67,4	32,6
2	VIOLADO	19,9	80,1
3	VIOLADO	33,4	66,6
4	MANTIDO	84,6	15,4
5	MANTIDO	75,2	24,8

e detecção dos padrões de comportamento das instâncias cliente possibilita ao provedor utilizar técnicas de migração de máquinas virtuais e computação verde para reduzir custos de energia e operação, gerando com isso economia em termos monetários. Os consumidores de serviços na nuvem se beneficiam com a monitoração e análise constante dos seus recursos e disponibilidade dos mesmos, possibilitando enviar cargas de trabalho que serão avaliadas e detectadas pelo provedor, que por sua vez informará o consumidor ou tomara de forma autônoma medidas para mitigar o impacto.

As redes bayesianas são uma modelagem rápida e eficiente de predição e detecção de variações no SLA de acordo com as regras e valores treinados, com pouco custo computacional conseguem prever satisfatoriamente a violação ou não de um SLA. Mesmo

Tabela 7. Resultados dos casos de teste com ruído de 20%

Caso	SLA Geral	Mantido %	Violado %
1	MANTIDO	68,8	31,2
2	VIOLADO	19,9	80,1
3	VIOLADO	38,1	61,9
4	MANTIDO	85,2	14,8
5	MANTIDO	70,9	29,1

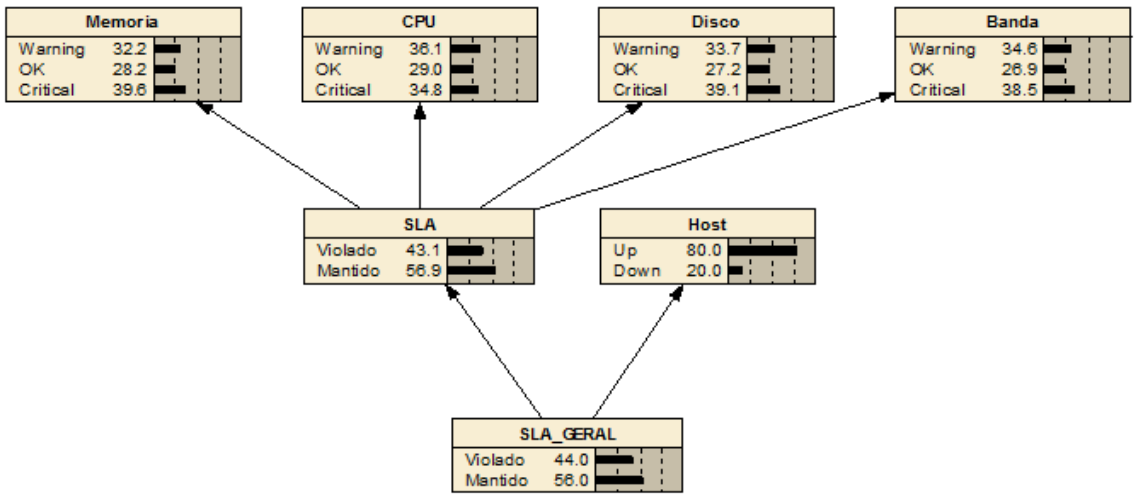

Figura 4. Representação da rede bayesiana com ruido de 20 %

com enfraquecimento nas entradas de monitoramento, o SLA consegue ser previsto com sucesso e pequena variabilidade nas probabilidades de saída.

Como trabalhos futuros a adição de hibridismo a partir de uma integração com sistemas difusos e também a integração em um sistema autônomo que efetue não apenas a detecção de violações, mas também recomende e reconfigure os recursos para atender à nova demanda.

Referências

Andreolini, M., Casolari, S., Colajanni, M., and Messori, M. (2010). Dynamic load management of virtual machines in cloud architectures. In Avresky, D., Diaz, M., Bode, A., Ciciani, B., and Dekel, E., editors, *Cloud Computing*, volume 34 of *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering*, pages 201–214. Springer Berlin Heidelberg.

Baldi, P. (1995). Gradient descent learning algorithm overview: a general dynamical systems perspective. *Neural Networks, IEEE Transactions on*, 6(1):182–195.

Buyya, R., Yeo, C. S., and Venugopal, S. (2008). Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In *High Performance Computing and Communications, 2008. HPCC '08. 10th IEEE International Conference on*, pages 5–13.

Eyraud-Dubois, L. e Larcheveque, H. (2013). Optimizing resource allocation while handling sla violations in cloud computing platforms. In *Proceedings of the "IPDPS - 27th IEEE International Parallel and Distributed Processing Symposium (2013)*.

Friedman, N., Nachman, I., and Peér, D. (1999). Learning bayesian network structure from massive datasets: the «sparse candidate« algorithm. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, UAI'99, pages 206–215, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Hope, L., Nicholson, A., and Korb, K. (2002). Knowledge engineering tools for probability elicitation.

Hormozi, E., Hormozi, H., Akbari, M., and Javan, M. (2012). Using of machine learning into cloud environment (a survey): Managing and scheduling of resources in cloud systems. In *P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2012 Seventh International Conference on*, pages 363–368.

Mell, Peter e Grance, T. (2011). The nist definition of cloud computing. Information Technology Laboratory.

Michlmayr, A., Rosenberg, F., Leitner, P., and Dustdar, S. (2009). Comprehensive qos monitoring of web services and event-based sla violation detection. In *Proceedings of the 4th International Workshop on Middleware for Service Oriented Computing*, MWSOC '09, pages 1–6, New York, NY, USA. ACM.

Michlmayr, A., Rosenberg, F., Leitner, P., and Dustdar, S. (2010). End-to-end support for qos-aware service selection, binding, and mediation in vresco. *IEEE Trans. Serv. Comput.*, 3(3):193–205.

Page, E. S. (1957). Estimating the point of change in continuous process. In *Biometrika, Vol. 44*.

Zhang, Rui e Bivens, A. J. (2007). Comparing the use of bayesian networks and neural networks in response time modeling for service-oriented systems. In *Proceedings of the 2007 workshop on Service-oriented computing performance: aspects, issues, and approaches*, SOCP '07, pages 67–74, New York, NY, USA. ACM.

Zhang, Q., Cheng, L., and Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of Internet Services and Applications*, 1:7–18.