

AUTOMAÇÃO E CONTROLE DE PROCESSOS NA NUVEM: PROPOSTA E ESTUDO DE CASO

JEFERSON A. BIGHETI, MARCOS R. CALDIERI

SENAI – Lençóis Paulista - SP

E-mails: jefersonbigheti@terra.com.br, mrcaldieri@gmail.com

EDUARDO P. GODOY

UNESP – Univ. Estadual Paulista, Sorocaba - SP

E-mails: epgodoy@sorocaba.unesp.br

Abstract—New manufacturing applications such as the Industry 4.0 have focused on jointly using industrial automation and information technology in order to obtain greater efficiency, quality and productivity. For the development and implementation of this new approach, it is necessary the application of some recent technologies such as the Internet of Things (IoT), the wireless networked control systems (WNCS) and cloud computing. This paper presents a cloud-based architecture for automation and process control based on the application of WNCSs integrated with a Web server. The control systems communicate via Modbus TCP/IP and the Web server is implemented using the open source tool ScadaBR. A case study presents a network control of a flow process through a controller with communication via Web service with the data server.

Keywords – Cloud computing, Web server, Modbus, PIDPlus and WNCS

Resumo — Novas aplicações de manufatura como a Indústria 4.0 têm focado no uso conjunto de tecnologias de automação industrial e tecnologia da informação visando a obtenção de maior eficiência, qualidade e produtividade. Para o desenvolvimento e implementação dessa nova abordagem, é necessário a aplicação de algumas tecnologias recentes como a Internet das Coisas (IoT), os sistemas de controle via rede sem fio (WNCS) e a Computação em Nuvem. Este trabalho apresenta uma arquitetura de automação e controle de processos em nuvem baseada na aplicação de WNCS integrados a um servidor Web. Os sistemas de controle se comunicam via Modbus TCP/IP e o servidor Web é implementado usando a ferramenta open source ScadaBR. Um estudo de caso apresenta o controle via rede de uma malha de vazão através de um controlador com comunicação via Web Service com o servidor de dados.

Palavras-chave — Computação em nuvem, Servidor Web, Modbus, PIDPlus e WNCS

1. Introdução

Devido a sua alta capacidade em acelerar processos de produção e manufatura, a automação industrial tornou-se essencial para as indústrias que planejam melhorar seus processos e obter assim uma maior eficiência e qualidade dentro de menores custos e tempos. Dessa forma, a indústria tem evoluído de forma a incorporar novas tecnologias e consequentemente obter maior produtividade (SAUTER et al., 2011).

A Indústria 4.0, chamada de Quarta Revolução Industrial, é um novo conceito que representa uma evolução dos sistemas produtivos atuais a partir da convergência entre novas tecnologias de automação industrial e tecnologia da informação (TI) (BERGER, 2014). No contexto da Indústria 4.0, os sistemas de controle rígidos e centralizados das fábricas cedem seu lugar para inteligência descentralizada, num ambiente onde todos os equipamentos e máquinas estão conectadas em redes e disponibilizando informações de forma única. A maioria das tecnologias necessárias à implementação da Indústria 4.0 já existem. Entre elas podem-se citar o protocolo IPV6, os sistemas ciber-físicos (CPS – *Cyber Physical System*), o RFID, o uso de virtualização de sistemas e serviços a partir de softwares, a Internet das Coisas (IoT – *Internet of Things*), a Computação em nuvem (*Cloud Computing*) e Big Data (ZUEHLKE, 2010; STANKOVIC, 2014, LEE et al.,

2015, HEGAZY & HEFEEDA, 2015). O grande desafio, portanto, é promover a integração entre essas tecnologias, visando a obtenção de uma nova realidade produtiva, onde tudo estará conectado para que as melhores decisões de produção, custo e segurança sejam tomadas, tudo sob demanda e em tempo real.

Nesse sentido, muitos trabalhos têm focado no desenvolvimento de soluções que agreguem essas tecnologias e permitam a integração dos equipamentos da automação a serviços de TI que estão armazenados na nuvem. Hegazy et al (2015) apresenta um novo serviço na nuvem, automação industrial onde os controladores são hospedados em dois servidores em nuvem, fisicamente separados. Nikolaidis et al. (2015) propõem a interação entre dois conceitos: a utilização de controladores nas nuvens com dispositivos baseados em IoT. Chen et al. 2010 apresenta a ideia de *Robot as a Service* ou RAAS que é um serviço na nuvem para acesso a hardware e software de um robô. Wu et al. (2013) explora o conceito de *Cloud Manufacturing*, um modelo de produção baseado em nuvem.

Este artigo apresenta uma proposta de arquitetura de automação e controle de processos em nuvem. A arquitetura é baseada na aplicação de sistemas de controle via rede (NCS) usando o protocolo Modbus TCP/IP com comunicação com um servidor Web implementado no ScadaBR (SCADABR, 2015). As variáveis do processo são disponibilizadas na nuvem e podem ser acessadas no servidor para diferentes finalidades. Um estudo de caso de uma malha de vazão é

discutido, onde um controlador fisicamente separado do processo acessa as variáveis no servidor via Web Service para implementação do controle em rede.

2. Automação em Nuvem

Soluções baseadas em Ethernet Industrial permitem ao usuário a integração dos equipamentos de campo a serviços de TI fornecidos por servidores (DECOTIGNIE, 2005). Uso de protocolos de rede suportados a nível industrial expandiu-se rapidamente possibilitando a comunicação de dados dos sistemas de automação de chão de fábrica e de dados armazenados em bancos de dados de servidores. Dessa forma tornou-se possível disponibilizar informações provenientes desses sistemas de automação através de protocolos com suporte ao Ethernet TCP/IP em aplicações Web. Essa interconexão de dados e infraestruturas de rede de informação e rede de automação em alto nível e de forma segura e confiável representa uma tendência para o desenvolvimento de soluções inovadoras e integradoras de automação e controle na área industrial com a Indústria 4.0 (BERGER, 2014).

A comunicação sem fio é de fundamental importância para que a implementação do conceito de Indústria 4.0 se torne viável. Uma das tecnologias base para este conceito é a Internet das Coisas (IoT) e o Máquina para Máquina (M2M – *Machine to Machine*) (STANKOVIC, 2014). A IoT contempla a conexão lógica de todos os dispositivos e meios relacionados ao ambiente produtivo em questão como os sensores, controladores, computadores, células de produção, sistema de planejamento produtivo, sendo todas as informações compartilhadas através de bancos de dados. A M2M representa a interconexão entre células de produção, onde tais sistemas podem trocar informações entre si, de forma autônoma, e tomar decisões de produção e segurança relacionadas ao processo através de um modelo de inteligência gerenciado pela IoT.

Atualmente, as pesquisas em torno da IoT têm atraído interesse e focado em diferentes aplicações (STANKOVIC, 2014). A IoT é a definição de uma rede, como a Internet, que conecta ao invés de pessoas, coisas. Coisas essas que podem trocar informações entre si, todas essas informações são armazenadas em nuvem e podem ser processadas, utilizadas e acessadas por qualquer coisa ou pessoa que tenha acesso a rede, independentemente de sua localização geográfica. Há três elementos essenciais para a IoT. O hardware (sensores, atuadores e hardware de comunicação), a interface de apresentação dos dados e o middleware, que faz a mediação entre os outros dois elementos. O middleware na IoT consiste em uma interface de armazenagem por demanda e ferramentas computacionais, para a análise dos dados (GUBBI et al., 2013).

A computação em nuvem (*Cloud Computing*) se refere, essencialmente, à noção de utilizarmos, em qualquer lugar e independente de plataforma, as mais variadas aplicações por meio da Internet com a mesma facilidade de tê-las instaladas em computadores locais (DIVYA & JEYALATHA, 2012). Kushida & Pingali

(2014) apresentam uma revisão sobre os requisitos para a aplicação da computação em nuvem na indústria, descrevendo potenciais vantagens dessa tecnologia e discutindo casos de estudo com resultados reais desse tipo de implementação.

Seguindo essa tendência de computação em nuvem, recentemente tem aparecido discussões a respeito do potencial dessa tecnologia para virtualização de sistemas e criação de serviços de automação e controle na nuvem. Hegazy & Heffeda (2015) apresentam uma nova proposta de arquitetura de automação na nuvem. Nesta proposta, diversas aplicações como controle em malha fechada, monitoramento de processos, telemetria e otimização de processos seriam realizadas através de serviços na nuvem. Os autores afirmam que a proposta traria inúmeros benefícios em termos de redução de custo e tempo de desenvolvimento através da virtualização de controladores, redução de hardware utilizado e tempo de desenvolvimento de aplicações com o uso de serviços disponibilizados.

3. Tecnologias Aplicadas

3.1. Protocolo Modbus TCP/IP

O Modbus é um protocolo aberto que define uma estrutura de mensagens de comunicação usadas para transferir dados discretos e analógicos entre dispositivos com detecção e informação de erros de transmissão. Dispositivos Modbus comunicam usando um mestre-escravo (cliente-servidor), técnica em que apenas um dispositivo (mestre/cliente) pode iniciar operações (chamadas consultas). Os outros dispositivos (escravos/servidor) respondem, fornecendo os dados solicitados para o mestre, ou tomar as medidas solicitadas na consulta. Um escravo é qualquer dispositivo periférico (transdutor, válvula, unidade de rede ou outro dispositivo de medição) que processa a informação e envia sua saída para o mestre usando o Modbus.

A camada de aplicação Modbus define uma unidade de protocolo de dados (PDU) simples, mostrada na Figura 1, independente das camadas inferiores. E o mapeamento do protocolo em portas específicas ou redes se dá por meio da unidade de aplicação de dados (ADU). O modelo cliente/servidor estabelecido pelo serviço de mensagens do Modbus é baseado em três tipos de mensagens utilizadas para a troca de informações: Modbus Request PDU (consulta), Modbus Response PDU (resposta), Modbus Exception Response PDU (erro). Neste processo o cliente envia uma mensagem de consulta (*request*), o servidor indica o recebimento e envia uma mensagem de resposta para o cliente (*response*) ou uma mensagem de erro (*exception response*).

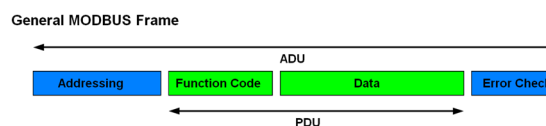


Figura 1: Estrutura de uma Mensagem Modbus (MODBUS, 2015).

Uma mensagem Modbus é composta pelos campos:

- ✓ **Address:** Tamanho de 1 byte. Número que define o endereço do dispositivo (escravo);
- ✓ **Function Code:** Tamanho de 1 byte. Define a ação a ser executada pelo escravo (ler dado, aceitar dado, reportar estado, reporte de erros);
- ✓ **Data:** Tamanho 0 a 252 bytes. Contém informações que o Escravo/Servidor usa para executar a ação definida pela função requisitada pelo Mestre/Cliente (endereços de memória, quantidade de itens transmitidos, quantidade de bytes do campo Data);
- ✓ **Errorcheck:** Tamanho de 2 bytes representando a checagem de erro na comunicação (enviado por um e checado pelo outro). Em caso de erro, é solicitada a retransmissão da mensagem.

O protocolo Modbus é implementado usando três versões: Modbus TCP/IP, o Modbus Serial (RTUI ou ASCII) e o Modbus Plus. É importante citar que todas as mensagens Modbus são transmitidas no mesmo formato. A única diferença entre as três versões está em como as mensagens são codificadas.

Modbus TCP/IP é basicamente o Modbus com uma interface Ethernet TCP/IP. O Modbus TCP/IP utiliza o TCP/IP e Ethernet para transportar os dados da estrutura de mensagem Modbus entre dispositivos compatíveis. De acordo com a Figura 2, o encapsulamento TCP/IP não modifica a estrutura básica da mensagem Modbus. As diferenças estão na interpretação do endereço (IP) e na verificação de erro (CRC-32).

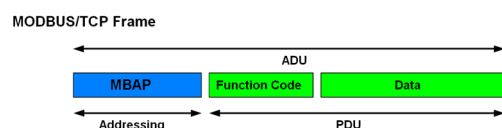


Figura 2: Estrutura de uma Mensagem Modbus TCP (MODBUS, 2015).

3.2. Protocolo IEEE 802.11 (Wi-Fi)

Em automação de processos, as redes Wi-Fi podem servir para a concentração de dados de dispositivos de campo de baixa potência a partir de um gateway. As vantagens de WLAN em automação de processos incluem: padrões abertos, robustez, boa relação custo-benefício, fácil acessibilidade para processar o controle por parte dos operadores móveis, alta taxa de transmissão de dados e segurança (WPA - *Wi-Fi Protected Access* e WPA2).

O protocolo Wi-Fi possui ainda uma característica importante entre os dispositivos sem fio que deve ser considerado quando o controle é implementado através de uma WLAN. Embora o protocolo Wi-Fi não tenha sido concebido para aplicações em WNCs, a sua implementação pode ser útil quando define o DCF (*Distributed Control Function*). Neste protocolo, a decisão da estação que pode transmitir é tomada entre os nós da WLAN.

3.3. Servidor ScadaBR

O ScadaBR é um software open source do tipo SCADA (*Supervisory Control and Data Acquisition*),

desenvolvido em plataforma Web, usado para automatizar processos de medição e automação (SCADABR, 2015). Através do ScadaBR é possível acessar e controlar dispositivos físicos, onde os dados desses dispositivos são salvos continuamente em uma base de dados. Além disso é possível visualizar os históricos das variáveis, receber alarmes, controlar o processo por meio de scripts. A Figura 3 apresenta a arquitetura interna do ScadaBR.

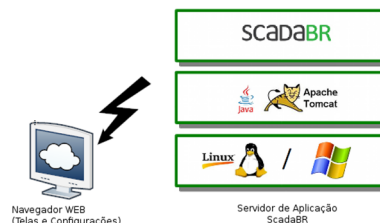


Figura 3 – Arquitetura do ScadaBR (SCADABR, 2015).

O equipamento onde o ScadaBR é instalado é conhecido como o "servidor da aplicação". Tanto as telas "IHM" quanto as configurações do ScadaBR são acessadas através de um navegador de Internet. O aplicativo do ScadaBR roda sobre um servidor Java, geralmente o Apache Tomcat. Rodando o servidor de aplicação, o aplicativo do ScadaBR pode ser acessado via navegador de Internet (endereço IP ou endereço de Internet). O acesso pode ser feito do mesmo computador onde o servidor da aplicação estiver instalado, ou a partir de computadores que estiverem na mesma rede.

3.4. Controlador PIDPLUS

A introdução de sensores sem fio na indústria origina novos desafios quando a medição da variável é utilizada em aplicações de controle. Para lidar estes desafios, o controlador PID pode ser modificado para operar de forma adequada com grandes períodos de amostragem do sensor, atualizações não periódicas da variável e perda de mensagens transmitidas (BLEVINS et al., 2014). A chave para se compreender como o PID deve ser modificado é realizando o reset do PID implementado utilizando uma rede de realimentação e um filtro onde a constante de tempo é um reflexo direto da resposta dinâmica do processo.

A implementação PIDPlus (SONG et al., 2006) para WNCs é ilustrada na Figura 4. O PIDPlus mantém o sinal de controle no último nível calculado até que uma nova medida seja recebida. É importante notar que sua sintonia é independente do período de amostragem, depende apenas das características físicas da planta.

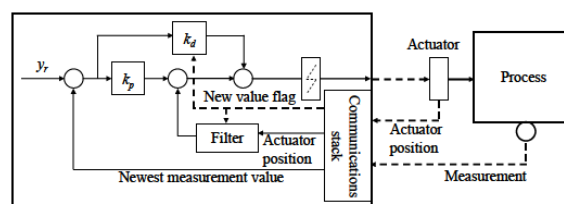


Figura 4: Estrutura do controlador PIDPlus (SONG et al., 2006).

A realimentação (*newest measurement value*) e o filtro de 1ª ordem (*filter*) são modificados para criar a contribuição de reposição com o seguinte comportamento:

- ✓ manter a última saída do filtro calculado (F_{N-1}), até uma nova medição ser informada (*new value flag*);
- ✓ quando uma nova medição é recebida (*new value flag*), utilize a nova saída do filtro como contribuição da realimentação (F_N).

A principal diferença do PID e PIDPLUS está na parte integrativa que foi substituído por um filtro de 1ª ordem. A saída do filtro é calculada conforme (1):

$$F_N = F_{N-1} + (O_{N-1} - F_{N-1}) \left(1 - e^{\frac{-\Delta T}{T_{reset}}} \right) \quad (1)$$

Onde: F_N = nova saída do filtro, F_{N-1} = saída do filtro na última execução, O_{N-1} = saída do controlador na última execução e ΔT = intervalo de tempo desde que o último valor medido foi recebido e T_{reset} = constante de tempo da planta somado ao tempo morto.

A parte derivativa é substituída conforme (2):

$$O_D = K_D \frac{e_N - e_{N-1}}{\Delta T} \quad (2)$$

Onde: e_N = erro atual, e_{N-1} = último erro, O_D = termo derivativo do controlador e K_D = ganho derivativo.

Considere a contribuição da parte derivativa quando as entradas são perdidas durante vários períodos. Para o algoritmo PID tradicional, o divisor na parte derivativa seria o período (discretização do controlador), enquanto que, no algoritmo PIDPlus é o tempo decorrido entre duas medições recebidas com sucesso (ΔT). É óbvio que o algoritmo modificado produz uma ação derivativa menor do que o algoritmo de controle PID. (SONG et al, 2006).

Na implementação do PIDPlus, o cálculo de reposição compensa automaticamente a alteração da medição e taxa de atualização da medição. Os cálculos do termo derivativo para um novo valor de medição não estão

disponíveis a cada execução do PID. Assim, não há necessidade de modificar a sincronização para o controle sem fio, ou seja, o ajuste é baseado estritamente no ganho e dinâmica do processo.

4. Arquitetura Proposta

A implementação do sistema de controle em estudo, foi baseada em uma Planta Piloto de Instrumentação da escola SENAI de Lençóis Paulista como mostrado na Figura 5. A planta piloto foi construída em escala industrial para ensino e pesquisa em automação e controle com os principais processos utilizados na indústria, como malhas de controle de nível, pressão, vazão, massa e temperatura. Todos os sensores e atuadores utilizados na planta são industriais. A variável que foi utilizada para controle nesse artigo foi a vazão de fluido na tubulação.



Figura 5 – Planta Piloto de Instrumentação.

A arquitetura de automação e controle em nuvem proposta neste trabalho é mostrado na Figura 5, onde temos o bloco Servidor que utiliza o ScadaBR como um Servidor Web que armazena em nuvem os dados de controle utilizados no processo num banco de dados MySQL.

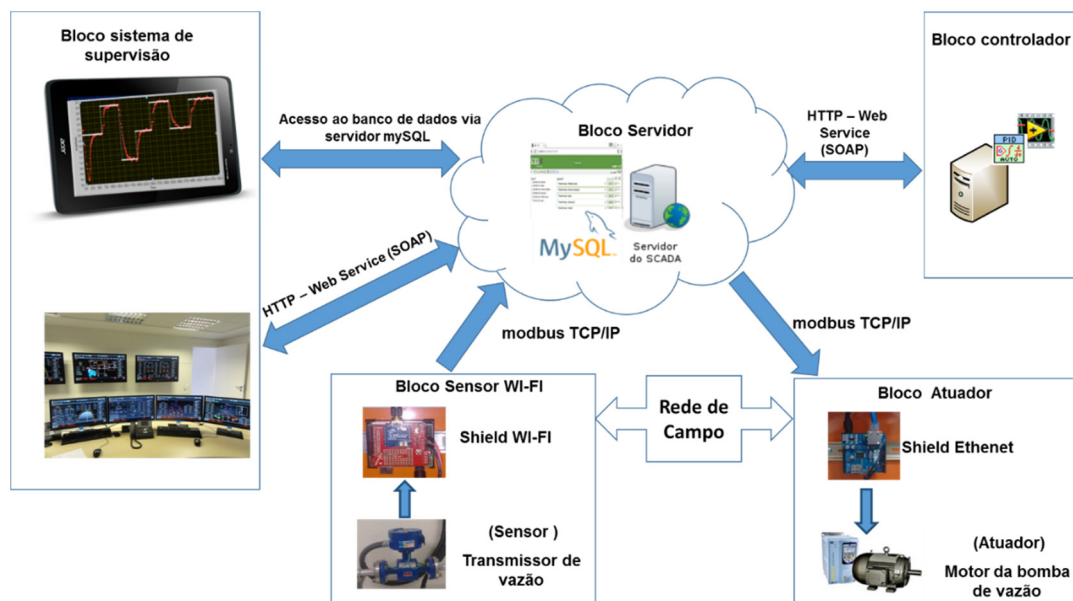


Figura 5. Arquitetura proposta para Automação e Controle em Nuvem.

O bloco Controlador é responsável pelo controle do processo, através do acesso às variáveis de controle utilizando um serviço web (Web Service) do ScadaBR. Esta web service utiliza a tecnologia SOAP, numa arquitetura cliente-servidor acessível via rede IP. A arquitetura também permite o acesso aos dados do servidor usando conexão direta e comunicação com o banco de dados MySQL. Além disso, também seria possível a implementação do controlador como um serviço do servidor web (ex: via Java script). O bloco Sensor Wi-Fi da Figura 5 é composto por um medidor de vazão magnético da planta piloto, que faz a aquisição da variável do processo (PV). Essa variável é utilizada por um conversor de sinal para Wi-Fi composto por um Arduino com Shield Wi-Fi que implementa o protocolo Modbus TCP/IP para se comunicar com o bloco Servidor. O sinal de controle (MV) no bloco Servidor, atualizado pelo bloco Controlador, é transmitido ao bloco Atuador via Modbus TCP/IP usando uma conexão Ethernet cabeada. O bloco Atuador utiliza uma placa Arduino com shield Ethernet, ligado a um inversor de frequência que comanda uma moto-bomba para variação da vazão como mostrado no bloco atuador da Figura 5. O sistema proposto habilita a aplicação da IoT para automação e controle de processos ao disponibilizar online, as informações de realimentação dos sensores e o sinal de controle para os atuadores, para qualquer tipo de plataforma, como smartphones, tablets ou computadores, tornando os dados de monitoramento e controle acessíveis em tempo real, para qualquer ponto conectado à rede e/ou Internet.

4.1. Estudo de Caso: Controle de Vazão

O estudo de caso deste artigo é baseado no controle de vazão da planta piloto (Figura 4), usando a arquitetura da Figura 5. O diferencial é que o controlador se encontra fisicamente separado do processo, obtendo os dados da malha de vazão no servidor ScadaBR. Através de Web Service SOAP, os dados em nuvem são utilizados e o controlador é capaz de controlar o processo. O serviço web do ScadaBR é padronizado via WSDL (*Web Services Description Language*), que é uma linguagem baseada em XML utilizada para descrever serviços web. Isto permite estender o ScadaBR através da arquitetura cliente-servidor, onde o ScadaBR age como servidor, e um módulo externo (bloco Controlador) atua como cliente. O ScadaBR e o Aplicativo Cliente podem executar na mesma máquina, ou em computadores separados, desde que conectados via rede IP.

Uma rede Wi-Fi entre o medidor de vazão e um Access Point foi criada. O servidor ScadaBR (dados do processo) com o banco de dados MySQL armazena os dados de controle do processo. Para o caso de estudo de vazão, são disponibilizadas as seguintes variáveis pelo servidor ScadaBR: Planta Piloto Vazão – PV, Planta Piloto Vazão – MV, Planta Piloto Vazão – Setpoint, Planta Piloto Vazão – P (Proporcional, Planta Piloto Vazão – I (Integral), Planta Piloto

Vazão – D (Derivativa). O sinal controle MV analógico (0 a 5Vcc) gerado a partir de um sinal PWM do Arduino comanda um inversor de frequência responsável pela variação da vazão. O bloco Controlador acessa os dados do medidor de vazão (PV) no servidor Web na nuvem e implementa o algoritmo de controle. O sinal de controle calculado é atualizado no servidor. Toda a comunicação entre Controlador e servidor Web é feita via Web Service SOAP. O controlador utilizado foi o PIDPlus (BLEVINS et al., 2014). Todo o ciclo de controle em malha fechada implementado na arquitetura proposta possui uma taxa de atualização de 2s.

7. Análises dos resultados

Neste trabalho foi estudado a aplicação do protocolo Modbus TCP/IP (Wi-Fi) para comunicação com um servidor Web implementado no ScadaBR. Usando uma comunicação via Web Service com o ScadaBR, foi implementado um controlador PID-Plus (LabVIEW) que acessa as variáveis de controle e comanda o processo através da nuvem.

O comportamento do controlador PIDPlus para o controle de vazão através do servidor em nuvem pode ser visto na Figura 6. Os resultados apresentados validam o desenvolvimento da arquitetura de automação e controle na nuvem proposta neste artigo. Verifica-se na Figura 6 que o controle via rede da malha de vazão, usando comunicação Modbus TCP/IP e web service SOAP se mostrou adequado e pode ser aplicado ao estudo de caso da planta piloto.

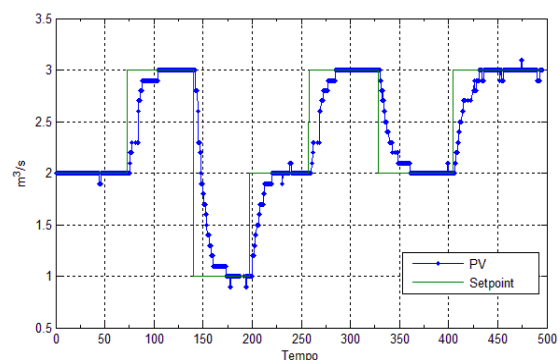


Figura 6. Comportamento do Controlador PIDPlus para o controle de vazão através do servidor em nuvem.

8. Conclusões e Trabalhos Futuros

Um WNCS via rede Modbus TCP/IP integrado a um servidor Web (ScadaBR) para uma malha de controle de vazão foi implementado e avaliado neste trabalho. A utilização do controlador PIDPlus se mostrou efetiva para o caso de estudo pois incorpora modificações na estrutura do PID para compensar o impacto de amostragens ou medições da variável de processo (ciclo de atualização do controle) não periódicas e variáveis no tempo, além de compensar

possíveis perdas de dados transmitidos. O desempenho do controle de vazão se mostrou adequado em face as características do processo. O uso da ferramenta open source ScadaBR como servidor web de dados cumpriu com os requisitos da aplicação, além de proporcionar a possibilidade de uso de outras funcionalidades da ferramenta na arquitetura proposta como supervisão, gerenciamento de alarmes e geração de relatórios. A comunicação via Web Service implementada entre o controlador e o servidor web se mostrou funcional e simples, mostrando potencial de uso para outras finalidades.

A integração das tecnologias de automação e informática como a IoT através da arquitetura de automação e controle em nuvem apresentada neste trabalho permitirá a análise e tomada de decisões com base nos dados disponíveis em nuvem, possibilitando o monitoramento e melhoria da eficiência do processo. Este trabalho apresenta resultados preliminares, mas que permitem vislumbrar potenciais de aplicação da arquitetura desenvolvida, que serão estudados em trabalhos futuros, como:

- ✓ Aplicações de controle em nuvem;
- ✓ Fornecimento de serviços de identificação de sistemas e controle avançado usando dados históricos disponíveis no servidor web do processo;
- ✓ Monitoramento, telemetria e otimização de processos através de serviços na nuvem;
- ✓ Virtualização de controladores e oferecimento de serviços de redundância ou backup de controladores com custo reduzido;
- ✓ Possibilidade de redução de custo e hardware utilizado, além de tempo de desenvolvimento de aplicações com o uso de serviços disponibilizados como serviços de comissionamento virtual.

9. Agradecimentos

Os autores agradecem ao apoio para este trabalho: 2015/01303-2, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

10. Referências Bibliográficas

- Berger, R. (2014). Industry 4.0 – The new industrial revolution, Strategy Consultants, 2014. <https://www.rolandberger.com/media/pdf/Roland_Berger_TAB_Industry_4_0_20140403.pdf>. Acesso em: Novembro, 2015.
- Blevins, T.; Nixon, M and Wojsznis, W. (2014). PID Control Using Wireless Measurements American Control Conference (ACC) June 4-6, 2014. p.1-6.
- Chen Y., Du, Z. and Garcia-Acosta, M. (2010). Robot as a service in cloud computing. in Proc. IEEE Int. Symp. Serv. Oriented Syst. Eng., Jun. 2010, pp. 151–158.
- Decotignie, J. D. (2005). Ethernet-based real-time and industrial communications. Proceedings of the IEEE, v. 93, n. 6, pp. 1102-1117.
- Divya, K. and Jeyalatha, S. (2012). Key technologies in cloud computing," in 2012 International Conference on Cloud Computing Technologies, Applications and Management (ICCTAM), pp.196-199, 8-10 Dec.
- Gubbi, J.; Buyya, R.; Marusic, S. and Palaniwami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems Vol. 29, pp. 1645-1660.
- Hegazy, T. and Hefeeda, M. (2015). Industrial Automation as a Cloud Service, IEEE Transactions on Parallel and Distributed Systems, Vol.26, No.10, pp.2750-2763, Oct. 1.
- Kim, W.J., Ji, K., and Ambike, A. (2006). Real-Time Operating Environment for Networked Control Systems, IEEE Transactions on Automation Science and Engineering, Vol. 3, No. 3, pp. 287–296, July.
- Kushida, T. and Pingali, G.S. (2014). Industry Cloud - Effective Adoption of Cloud Computing for Industry Solutions," in 2014 IEEE 7th International Conference on Cloud Computing, pp.753-760.
- Lee, J.; Bagheri, B. and Kao, H. (2015) A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems", Manufacturing Letters, Vol. 3, January 2015, pp. 18-23.
- Modbus (2015). Modbus Organization. Disponível em: <http://www.modbus.org/>. Acesso em: Dezembro, 2015.
- Nikolaidis, P.; Didic, A.; Mubeen, S.; Pei-Breivold, H.; Sandstrom, K. and Behnam, M. (2015). Applying Mitigation Mechanisms for Cloud-based Controllers in Industrial IoT Applications, Malardalen Real-Time Research Centre (MRTC), Malardalen University, Sweden ABB Corporate Research.
- Sauter, T.; Soucek, S.; Kastner, W. and Dietrich, D. (2011). The Evolution of Factory and Building Automation," IEEE Transactions in Industrial Electronics Magazine, Vol.5, No.3, pp.35-48.
- Scadabr (2015). Sistema supervisorio Web em licença Open Source. Disponível em: <<http://www.scadabr.com.br/>>. Acesso em: Novembro, 2015.
- Song, J.; Mok, A. K.; Chen, D.; Nixon, M.; Blevins, T. and Wojsznis, W. (2006). Improving PID control with unreliable communications. In ISA EXPO Technical Conference.
- Stankovic, J.A., (2014). Research Directions for the Internet of Things, Internet of Things Journal, IEEE , Vol.1, No.1, pp.3-9.
- Wu D., Greer M., Rosen D., and Schaefer D. (2013). Cloud manufacturing: Strategic vision and state-of-the-art, J. Manuf. Syst., Vol. 32, No. 4, pp. 564–579.
- Zuehlke, D. (2010). Smart Factory—Towards a factory-of-things, Annual Reviews in Control, Vol. 34, No. 1, pp. 129-138.