

PENGEMBANGAN *HARDWARE-IN-THE-LOOP* UNTUK PRAKTIKUM SISTEM KENDALI

Muhammad Rif'an
DIII Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta
E-mail: m.rifan@unj.ac.id

Abstract

This paper describe about develop a hardware simulation with a hardware-in-the-loop principle for control system analysis dedicated to the practice of diploma students. The real system is simulated using a neural network based system model and the controller uses digital control. The system model is built based on real system input-output data. meanwhile, control of the system is built with digital PID control. Students can tune PID parameters and provide disturbance to the system model. The results of this study indicate that the system model is able to represent real systems and PID controllers are able to improve system performance when there is disturbance.

Keywords: *Hardware-in-the-loop, Control System, Neural Network.*

Abstrak

Penelitian ini mengembangkan media pembelajaran untuk sistem kendali kontinyu menggunakan *hardware-in-the-loop* sehingga diharapkan dapat dihasilkan sistem proses di industri yang serupa dan identik yang dapat diujicoba secara langsung oleh mahasiswa. *Hardware-in-the-loop* yang dikembangkan merupakan sebuah rangkaian elektronika dan komputer dengan model sistem berbasis pada neural-network. Selain itu, *hardware-in-the-loop* yang dikembangkan memiliki fitur sistem kendali PID yang dapat diubah-ubah sesuai dengan kebutuhan. Hasil dari penelitian ini menunjukkan bahwa model yang dibangun dapat merepresentasikan sistem sebenarnya dan Kendali PID dapat bekerja meningkatkan performa sistem.

Kata kunci: *Hardware-in-the-loop, Sistem Kendali, Neural Network.*

PENDAHULUAN

D3 Bidang Teknologi Elektronika pada umumnya memiliki visi menghasilkan lulusan siap kerja di dunia industri yang trampil dan bersertifikasi serta berkompeten pada bidang elektronika. Untuk itu salah satu kompetensi yang harus disemai dan dikembangkan kepada lulusan D3 Bidang Teknologi Elektronika adalah mampu menguasai dan menerapkan sistem kendali atau sistem elektronika dengan pendekatan analitis dan mempertimbangkan standar teknis, aspek kinerja, keandalan, kemudahan penerapan, keberlanjutan, serta memperhatikan faktor-faktor ekonomi, kesehatan dan keselamatan publik, kultural, sosial dan lingkungan. Kompetensi ini sangat sejalan dengan kerangka kualifikasi nasional Indonesia (KKNI) yang menyatakan bahwa lulusan D3 atau level 5 harus memiliki tingkat kemampuan menyelesaikan pekerjaan berlingkup luas, memilih berbagai metode.

Sejalan dengan hal tersebut, Kompetensi ini dicapai melalui beberapa mata kuliah seperti sistem kendali kontinyu. Pengendalian sistem kontinyu telah banyak digunakan pada industri dalam rangka menjaga kualitas produksi seperti pada pembangkit listrik, pembuatan mobil, tekstil, dan lain sebagainya. Kesulitan dan keterbatasan dalam menyampaikan dan menanamkan kompetensi ini adalah kurang mendukungnya fasilitas laboratorium karena sangat mahal, tidak efisien dan dapat membahayakan mahasiswa jika dalam setiap laboratorium disediakan sistem kontinyu yang sama dengan industri sesungguhnya.

Untuk itu, dalam penelitian ini akan dikembangkan media pembelajaran untuk sistem kendali kontinyu menggunakan *hardware-in-the-loop* sehingga diharapkan dapat dihasilkan sistem proses di industri yang serupa dan identik yang dapat diujicoba secara langsung oleh mahasiswa. *Hardware-in-the-loop* yang dikembangkan merupakan sebuah rangkaian elektronika dan komputer dengan model sistem berbasis pada neural-network dan model matematika. Neural network digunakan karena kemampuannya dalam identifikasi sistem proses baik yang bersifat linier maupun non linier.

METODE

Dalam penelitian ini metode yang digunakan adalah metode eksperimen dengan langkah-langkah sebagai berikut:

1. Identifikasi model dengan mengambil data *input-output* sistem sesungguhnya, dalam percobaan ini digunakan *Pressure Process RIG System*.
2. Membangun model sistem menggunakan algoritma Neural network.
3. Menanamkan hasil model sistem ke dalam Raspberry pi.
4. Membangun algoritma PID pada Arduino UNO.
5. Menyusun rangkaian *Hardware-In-the-loop*.
6. Uji-coba pengendalian *Hardware-In-The-Loop*.

DASAR TEORI

Pressure Process Rig

Pressure Process Rig (PPR) seperti ditunjukkan pada Gambar 1 merupakan instrument yang

digunakan untuk proses pengaturan presure (tekanan) dan kecepatan aliran (*flow*) udara. sistem ini terdiri dari:

- Sebuah Pneumatic Control Valve,
- Sebuah I/P converter,
- Sebuah Blok Orifice,
- Sebuah Flowmeter,
- Enam buah pengukur tekanan (*Gauge*),
- Dua buah regulator tekanan,
- Sebuah sensor tekanan,
- Sebuah sensor perbedaan tekanan,
- Sebuah Air Receiver,
- Tujuh buah Valve, dan
- Beberapa buah saluran pipa yang menghubungkan komponen – komponen di atas



Gambar 1. *Pressure Process RIG*

Sistem ini merupakan *Single Input Single Output* (SISO) dengan masukan berupa *Air Compressor* dan *Output* berupa *Pressure*.

Raspberry Pi

Raspberry piditunjukkan pada Gambar 2 adalah kartu mini-kredit dengan prosesor kelas ARM11. Raspberry pi memiliki *built in video*, *audio*, dan *input* USB yang dapat dihubungkan ke keyboard dan mouse serta kemampuan koneksi jaringan melalui Ethernet, nirkabel dan USB.

Penyimpanan data yang digunakan adalah kartu SD dengan OS yang saat ini beredar yaitu Noobs dan Raspbian. Papan ini memiliki 40 pin GPIO yang dapat dihubungkan ke sensor dan aktuator dan mudah diprogram menggunakan bahasa yang kuat seperti C, python dll. Selain itu, paket ini menyediakan Apache Server yang sangat familiar dan kuat.



Gambar 2. *Raspberry Pi*

Arduino UNO

Arduino UNO ditunjukkan pada Gambar 3 adalah board mikrokontroler berbasis ATmega328 (datasheet). Memiliki 14 pin *input* dari *output* digital dimana 6 pin *input* tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, jack power, ICSP header, dan tombol reset. Arduino UNO diprogram menggunakan bahasa berbasis C dengan IDE dari Arduino yaitu Sketch.

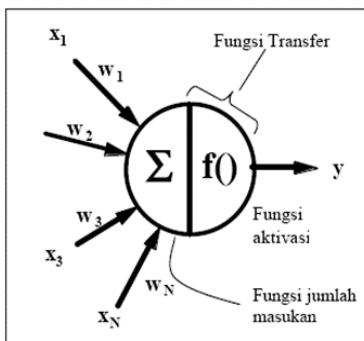


Gambar 3. *Raspberry Pi*

Neural Network

Neural Network atau Jaringan Syaraf Tiruan dalam dunia komputer merupakan sebuah model saraf pusat (khususnya otak) yang mampu belajar dan mengenal pola.

Pemrosesan informasi dalam Neural Network dapat diterangkan sebagai berikut: Sinyal (baik berupa aksi ataupun potensial) muncul sebagai masukan unit (sinapsis) $[x_1, x_2, \dots, x_N]$; efek dari tiap sinyal ini dinyatakan sebagai bentuk perkalian dengan sebuah nilai bobot $[w_1, w_2, \dots, w_N]$ untuk mengindikasikan kekuatan dari sinapsis. Semua sinyal yang diberi pengali bobot ini kemudian dijumlahkan satu sama lain untuk menghasilkan unit aktivasi. Jika aktivasi ini melampaui sebuah batas ambang tertentu maka unit tersebut akan memberikan keluaran dalam bentuk respon terhadap masukan. Unit aktivasi ini kemudian dibandingkan dengan sebuah nilai ambang, dan hasilnya dimasukkan kedalam fungsi transfer (fungsi non-linier) yang akan menghasilkan sebuah keluaran. Secara ringkas proses tersebut dapat digambarkan dalam gambar 4.

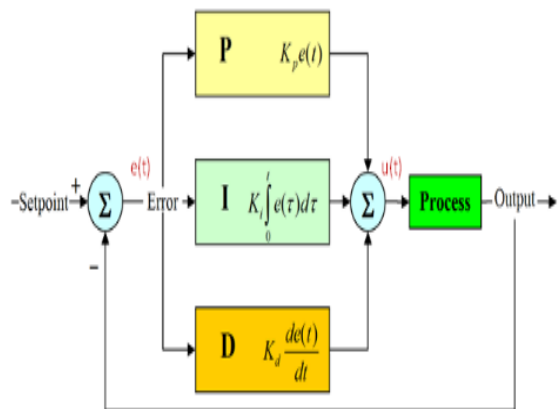


Gambar 4. Neuron sebagai Operator Matematis

Pengendali PID

Pengendali PID adalah pengendali yang paling klasik. Keuntungan pengendali ini adalah tidak membutuhkan model matematis yang akurat dari sistem yang dikendalikan dan telah digunakan secara luas di pelbagai bidang kendali proses. Untuk sistem *time invariant linear*, pengendali PID konvensional selalu dapat memiliki efek antisipasi kendali melalui parameter tuning, terutama untuk sistem orde rendah.

Konfigurasi PID konvensional ditunjukkan seperti Gambar 5.



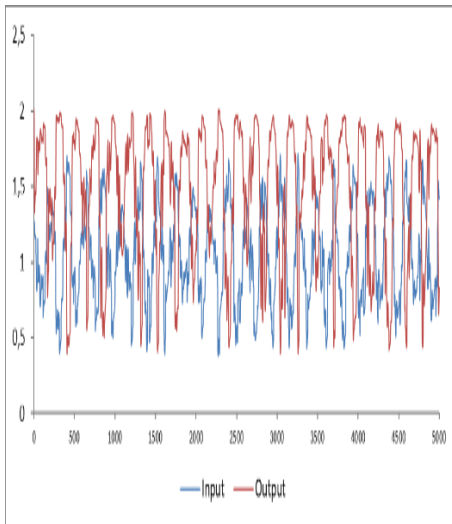
Gambar 5. Konfigurasi PID

Dalam konfigurasi ini, pengendali ditempatkan secara seri dengan plant. Maka konfigurasi sistem seperti ini biasa disebut *series* atau *cascade compensation*. Konfigurasi ini yang paling banyak dipakai di dalam desain sistem kontrol. Seperti yang telah diketahui, manfaat penggunaan *feedback* yaitu untuk meningkatkan performansi sistem kontrol itu sendiri, memperbaiki performansi response atau mengatur transien respons (seperti mereduksi *time constant*), mengurangi *error*, dan

mengurangi sensitivitas atau mendapatkan *robustness*. Di dalam Prakteknya, bisa saja hanya menggunakan term proporsional saja (P), term integral dan proporsional (PI) atau term proporsional dan derivative (PD). Hal itu disesuaikan dengan kebutuhan dan sangat tergantung dari karakteristik sistem yang akan dikendalikan.

HASIL DAN PEMBAHASAN

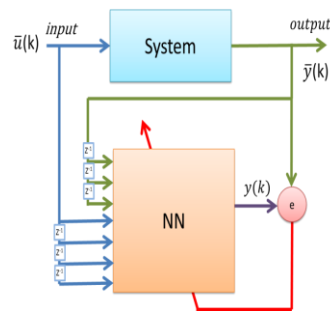
Identifikasi dengan Neural Network untuk Sistem Dinamik *Pressure process rig* (PPR) yang merupakan simulator proses pengendalian dan pengukuran tekanan. Data *input* dan *Output* sistem PPR dipresentasikan pada gambar 6.



Gambar 6. *Input-Output*

Sistem PPR merupakan sistem satu masukan satu keluaran (SISO) dan dapat direpresentasikan dalam bentuk *finite difference equation*. Terdapat 5.000 pasang data *input* dan *output* yang dapat digunakan untuk indentifikasi menggunakan Neural

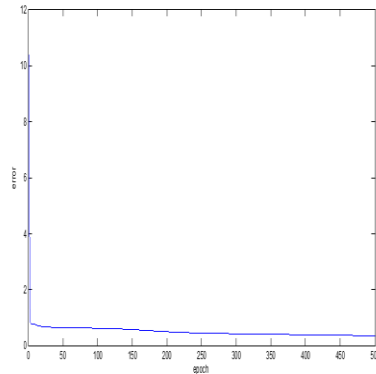
Network. Blok Diagram identifikasi disajikan pada Gambar 7.



Gambar 7. Blok Diagram Identifikasi

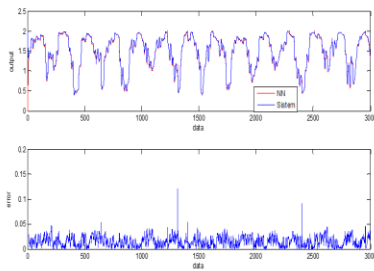
Neural Network memiliki 7 masukan dan 1 keluaran. Masukan terdiri dari $u(k)$, $u(k-1)$, $u(k-2)$, $u(k-3)$, $y(k-1)$, $y(k-2)$, dan $y(k-3)$, sedangkan keluaran adalah $y(k)$. Nilai keluaran $y(k)$ akan dibandingkan dengan nilai $\bar{y}(k)$ [output sistem PRR] sehingga didapat nilai *error e* yang digunakan untuk pembelajaran neural network selanjutnya.

Penurunan *Error* dari pembelajaran dapat dilihat pada gambar 8.



Gambar 8. Penurunan *Error*

Hasil akhir model identifikasi sistem PPR yang didapat disajikan pada Gambar 9.



Gambar 9. Hasil Model Identifikasi

Gambar 9 menunjukkan bahwa sinyal keluaran Identifikasi sistem dengan Neural network dengan masukan 3.000 sinyal uji telah mampu mendekati keluaran sistem PPR dengan SSE sebesar 0.9827 dan MSE sebesar $3.2258e-04$. *Error* untuk masing-masing pasangan data keluaran sistem dan keluaran Neural Network diperlihatkan pada bagian bawah.

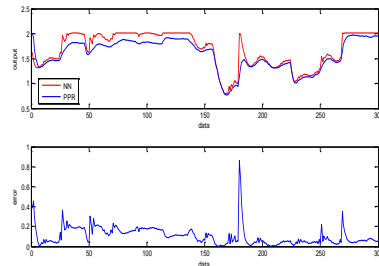
Setelah mendapatkan model, langkah berikutnya adalah menanamkan model identifikasi tersebut ke dalam mini komputer yang pada penelitian ini digunakan Raspberry pi 3 Model B dan menanamkan algoritma PID pada Arduino UNO.

Experimental Setup disajikan pada Gambar 10.



Gambar 10. *Experimental Setup*

Hasil ujicoba dalam *Hardware-In-The-Loop* disajikan pada Gambar 11.



Gambar 11. Hasil Ujicoba HIL

Gambar 11 bagian atas memperlihatkan kinerja sistem secara keseluruhan, dengan masukan Y topi (garis biru), didapat keluaran Y (garis merah) dengan *error* untuk masing-masing data seperti gambar pada bagian bawah. *Error* yang didapat relatif kecil dengan nilai < 0.05 .

KESIMPULAN

Berdasarkan hasil pengujian yang dilakukan pada penelitian ini, maka dapat disimpulkan bahwa:

1. Dalam penelitian ini prototipe simulasi *hardware-in-loop* telah dirancang dan dibangun.
2. Prototipe simulasi *hardware-in-loop* dengan Neural-Network adalah cara yang efisien untuk memodelkan sistem kontrol industri.
3. Hasil eksperimen menghasilkan kesalahan yang relatif kecil dan sistem dapat berfungsi sesuai rencana dengan *error* yang minimal sebesar < 0.05 .

DAFTAR RUJUKAN

Ronsheng L, Satina M (2010). *Aerospace real-time control system and software*. In: Lewis (ed) *The control handbook*:

- control system applications. CRC Press, Boca Raton, FL
- National Aeronautics and Space Administration (2007). *NASA systems engineering handbook*. NASA, Washington
- European Cooperation for Space Standardization (2010). *ECSSE-HB-60A: control engineering handbook*.
- European Cooperation for Space Standardization (2001). *ECSSE60: control engineering Standard*.
- Burbank JL, Kasch W, Ward J (2011). *An introduction to network modeling and simulation for the practicing engineer, 1st edn*. John Wiley, Hoboken, NJ
- Isermann R (1999). *Hardware-in-the-loop simulation for the design and testing of engine-control systems*. *Control Eng Pract* 7: 643–65.