Fuzzy Logic Controller for Low Temperature Application

Inseob hahn, A. Gonzalez[+], and M. Barmatz

Jet Propulsion Laboratory, California Institute of Technology,
4800 Oak Grove Dr., Pasadena, CA 91109

We describe the development of a new computer-based temperature controller using fuzzy logic algorithms. Fuzzy logic rules and reasoning are applied in-situ for determining PID controller parameters based on an error signal and its first derivative. This controller design will be used to stabilize several isothermal stages of a cryostat for studying thermodynamic properties near the ^{3}He liquid-gas critical point. Simulations of temperature control near the ^{3}He critical point (T_c = 3.3 K) show that faster settling, smaller overshoot, and easier control can be achieved using the fuzzy logic temperature controller in comparison with the traditional PID controller.

1. INTRODUCTION

The most common temperature controller used in low temperature experiments is the proportional- integral derivative (PID) controller due to its simplicity and robustness. However, the performance of temperature regulation using the PID controller depends on initial parameter setup, which often requires operator's expert knowledge on the system. Furthermore, low temperature application of this controller requires additional care because the cooling power of a low temperature cryostat is very small compared to the heating power. Also, when the temperature controller operates over a wide temperature range, it is non-trivial to adjust the parameters of the PID controller. In this paper, we present a computer-assisted temperature controller based on the well known "fuzzy logic" concept used in industrial control processes[1]. Although this concept has been utilized for many years in adaptive control systems, it has not been as popular in temperature controller applications.

2. PRINCIPLE OF OPERATION

The standard integral form of a PID-controller can be descretized for small sampling time T_0 obtaining[2],

$$u(n) = G_P\left[e(n) + \frac{T_0}{T_I}\sum_{i=0}^{n-1} e(i) + \frac{T_D}{T_0}\left(e(n) - e(n-1)\right)\right],$$

where $u(n)$ is the control signal, G_P is gain, T_I is integration time, T_D is derivative time, and e(n) is the error signal between the set point and current value of the process.

The parameters G_P, T_I, and T_D can be manipulated for optimal control, however finding the optimal value is non-trivial. Our temperature controller uses the same PID algorithm to calculate the control signal $u(n)$. Fuzzy logic rules based upon knowledge of the system are used to find the optimal values of the parameters. A schematic drawing of this concept is shown in Fig 1.

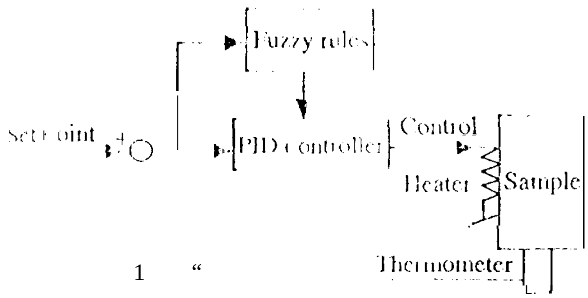

Figure 1: Schematic drawing of fuzzy logic controller.

In this fuzzy controlled PID, the gain, integration time and derivative times used by the PID are constantly varied using fuzzy logic in order to improve the response of the system. Tables of fuzzy logic rules are constructed which give different values of the control parameters based on real time error signals (e, and δe).

This process is fuzzy in the sense that a given set of input parameters corresponds to multiple values of a predefined membership function. This membership function quantifies to what degree an input parameter corresponds to each category (fuzzy set) in the function.

* This work was supported by NASA.

+ Current address: UC Santa Cruz, Astronomy Board, 1156 High Street, Santa Cruz, CA 95064.

3. IMPLEMENTATION AND RESULT

Membership coefficients μ_i (i= zero, small, medium large) are determined based on the values of e(n). Figure 2 shows the membership function (order 2 B-splines[3]) that was used. This function is probably the simplest possible membership function since only two categories overlap linearly for a given e(n). It is plausible to obtain better performances using different types of membership functions depending on the system.

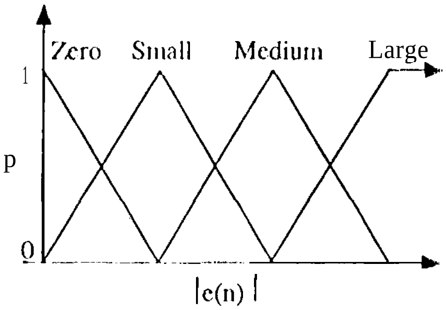

Figure 2: Membership function for e(n).

The interval size is a user defined variable, in the present case it was chosen to be several millidegrees. The membership coefficients for $\delta e(n)$ are handled in an identical fashion using a different interval size.

We constructed tables using fuzzy logic rules of the form: if e(n) is Small and $\delta e(n)$ is Small, then G_p is *Small, etc.* G_p' is a normalized gain ($0 < G_p' < 1$) obtained from predefined values of maximum and minimum gain. These rules are derived from conventional wisdom in PID control theory. For instance, when the error is near Zero, and δe is near Zero, we expect a ***Small*** control signal [0 avoid overshoot, etc. A possible logic table for G_p' is shown in the Table 1.

Table 1. Fuzzy tuning rules for G_p'.

δe \ e	Zero	Small	Medium	Large
Zero	*Small*	*Small*	*Big*	*Big*
Small	*Small*	*Small*	*Big*	*Big*
Medium	*Small*	*Small*	*Big*	*Big*
Large	*Small*	*Small*	*Small*	*

* If $\delta e(n)$ is greater than e(n), then this takes on the value *Big*. Otherwise, it takes on the value *Small*.

Small and *Big* are members of fuzzy set for G_p'. We used the following G_p' membership function[4]:

$\mu_{small}(G_p') = -1/4 \ln(G_p')$ for *Small,*

$\mu_{large}(G_p') = -1/4 \ln(1 - G_p')$ for *Big*.

Once e(n) and $\delta e(n)$ have been determined, the membership coefficients, $\mu_i(e)$, and $\mu_j(\delta e)$, are calculated using the function shown in Fig 2. The product of these values are used to determine G_p [4]. Other tuning parameters, T_I (integration time), and α (the ratio of the integration time to the derivative time) can also be determined in the same fashion. In the present work these parameters are derived from simple numeric logic tables. Because the system can easily heat but not cool, the positive and negative error sides are designed to be asymmetric. This unique feature of tuning tables helps to avoid accidentally overheating the low temperature system. Figure 3 shows one example of simulation using two different methods starting with same initial (not optimized) PID parameters

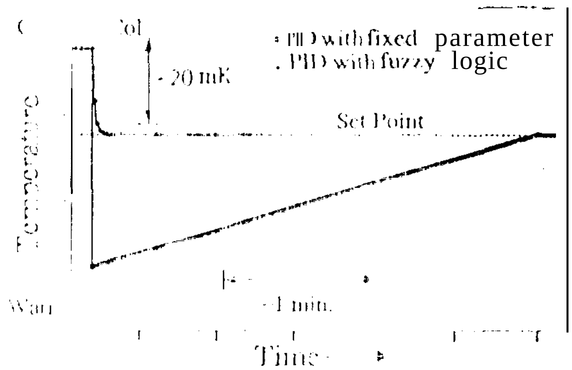

Figure 3: Result of simulation. When PID paramaters are optimized, both methods show similar performance.

4. CONCLUSIONS

We have used fuzzy logic to determine the parameters in an in-situ PID temperature controller based on an error signal and its first derivative. Simulations show that this controller (logic tables and membership functions) can be designed to perform better than a typical PID controller using fixed parameters.

REFERENCES

[1] E H. Mamdani, Proc IEE, Vol 121, No.12, 1585 (1974).

[2] R Isermann, *Digital Control Systems*, Vol 1 (1989).

[3] M. Brown and C. Harris, *Neurofuzzy Adaptive Modeling and Control*, pp. 370 (1994).

[4] Z. Zhao, M. Tomizuka, and S. Isaka, IEEE Transations on System, Man, and Cybernetics, Vol 23 No. 5, 1392 (1993).