Automation, Control and Intelligent Systems
2016; 4(2): 35-41
http://www.sciencepublishinggroup.com/j/acis
doi: 10.11648/j.acis.20160402.15
ISSN: 2328-5583 (Print); ISSN: 2328-5591 (Online)

Fuzzy Fine Tuning Therapies for Intelligence of High Speed Electronic Packaging Equipment

He Yunbo*, Hu Yongshan, Chen Xin, Gao Jian, Yang Zhijun, Chen Yun, Tang Hui, Ao Yinhui, Zhang Yu

Guangdong Provincial Key Lab. of Computer Integrated Manufacturing Systems, Key Laboratory of Mechanical Equipment Manufacturing and Control Technology of Ministry of Education, School of Electromechanical Engineering, Guangdong University of Technology, Guangzhou, P.R. China

Email address:
heyunbo@gdut.edu.cn (He Yunbo)
*Corresponding author

To cite this article:
He Yunbo, Hu Yongshan, Chen Xin, Gao Jian, Yang Zhijun, Chen Yun, Tang Hui, Ao Yinhui, Zhang Yu. Fuzzy Fine Tuning Therapies for Intelligence of High Speed Electronic Packaging Equipment. *Automation, Control and Intelligent Systems*. Vol. 4, No. 2, 2016, pp. 35-41. doi: 10.11648/j.acis.20160402.15

Received: March 26, 2016; **Accepted**: April 6, 2016; **Published**: April 19, 2016

Abstract: It's always difficult to satisfy both strict requirements of dynamic performance and settling performance of high-precision and high-acceleration point-to-point motions on electronic packaging equipment, i.e., Die bonders, Wire bonders, Flip-chip bonders, Wafer bumping machines, and so on. Focusing on this difficulty, Fuzzy logic is promoted in this paper to fine tune feed forward coefficient parameters of motion systems on packaging equipment to satisfy multiple targets of dynamic and settling performance requirements. This intelligent fine tuning approach, with achievable multiple motion targets, provides a therapy to automate and optimize motion system performance on packaging equipment, help on development and mass production of semiconductor packaging equipment, and improve machine intelligence.

Keywords: Dynamic Performance, Settling Performance, Electronic Packaging Equipment, Fuzzy Logic, Fine Tuning, Machine Intelligence

1. Introduction

Electronic packaging machines (i.e., Die bonders, Wire bonders, Flip-chip bonders, Wafer bumping machines, and so on) require their moving axes to move with high speed and high precision during mass production of electronic and semiconductor manufacturing. Due to the large production volume, similar movements are repeated by thousands of hundreds of times 24 hours a day. Even a small improvement in the motion control performance brings faster and finer therapies in manufacturing, which result in significant economic gains. To achieve optimized dynamic tracking and settling motion control performance, maximum feedback system bandwidth and feedforward control (FFC) are quoted. FFC is important to provide large control output in dynamic tracking, so that motors and actuators can be driven according to fast motion profile to achieve and maintain high acceleration. With FFC in dynamic stages, settling performance can be affected and shaped accordingly. To achieve optimum dynamic tracking and settling performance, performance fine tuning is important and frequently addressed [1-8].

People, especially for those control engineers and scientists who are familiar with control systems and experienced in fine tuning, prefer manual fine tuning of motion performance. Yet, electronic packaging machinesare frequently used by machine operators, technicians and maintenance engineers in production. Above users normally have no pre knowledge of motion systems and fine tuning. Auto tuning or self tuning is a must for them. Currently in semiconductor equipment industry, speed command and acceleration command of a motion profile are selected as FFC resources (or inputs) to simplify implementation of FFC modeling for convenience. And, two proportional coefficients (proportional coefficient of velocity command kvel and that of acceleration command kacc) are utilized to shape dynamic tracking performance by searching or fine tuing.

Searching mechanism can be described as below: The default FFC coefficients Kvel and Kacc, and their up/low searching ranges are initially set, as well as a combined performance target, which includes dynamic and static performance indicators and their individual weightings. Then, a best Kacc is found by fixing a default Kvel and searching for a minimum combined performance target with captured motion performances through machine running round by round with incremental Kacc between its low and up ranges; then fix the best Kacc and repeat above machine running and performance searching process to search for a best Kvel. Searching mechanism is far worse than tuning. Drawbacks of searching mechanism are described as follows:

1. Much time consumption and small range of Up/Low limits of FFC coefficients Kvel/Kacc. Obviously, much time is needed when searching step by step between low and up ranges of Kvel/kacc. Low/up ranges cannot be wide, otherwise much time should be consumed.

2. Strict default values of Kvel/Kacc. Before searching starts, an experienced engineer or expert is needed to set the default Kvel/Kacc values to around the best. Tuning result may not be good if default Kvel/Kacc values are not good enough.

3. Uncontrollable searching result. There are too many objectives and weightings to build up the final integrated single performance target. After searching, we just know that the final Kvel/Kacck may be good as the final integrated performance target is the best. But for every individual performance objective, such as DPE (max Dynamic Position Error) and PUS/POS (static Position UnderShoot/OverShoot), and DAC in settling, we don’t know whether they are in specs when the searching result, the final set of Kvel/Kacc is applied. In other words, we cannot separately control these multiple objectives. We have to integrate all these objectives to one single performance target with different weightings for every performance index. Different Kvel/Kacc and different individual performances may lead to the same integrated performance target. This searching method may lead to uncontrollable Kvel/Kacc, as well as non-repetitive performance.

4. Fixed tuning resolution/step. The searching method cannot identify whether the default value is near or far from the best value during searching. It cannot accelerate or decelerate the incremental steps in searching process. For example, given a default value for Kacc 270 and suppose the best value is 272, by searching method, the program must search from the low limit (i.e. 0) to up limit (i.e. 400) step by step with a fixed incremental step, say 3. After over 100 times of machine running, performance capturing, calculations and comprisions, with much time consumed, it can just find a value (either 270 or 273) as the best searching value. It's still a little different from 272, which is actually the best.

5. Little flexibility to apply to new products/users/purposes. The up/low limits, default values, objectives, and weightings must be carefully set by an experienced engineer before the searching mechanism is submitted for use in mass production. This method is not convenient to apply to new products and users directly. And also, it’s hard to expand this method for other purposes such as to find the best DPE, Amax (maximum Acceleration) or DAC settling range in a motion.

An intelligent tuning method is needed to actually tune the FFC coefficients toward controllable motion performance, and to save tuning time. Stearns H. (Dept. of Mech. Eng., Univ. of California, Berkeley, Berkeley, CA) published a paper on feedforward controller tuning. A fixed-structure feedforward controller of a wafer stage of a photolithography machine is tuned. The feedforward controller is tuned with the objective of minimizing a cost function that is a quadratic function of tracking error. Simulation and experimental results are presented which show that the iterative tuning method effectively reduces tracking error. [4] An automation technique is presented by Timothy P. K. (Drexel University), which yields high-performance, low-cost optoelectronic alignment and packaging through the use of intelligent control theory and system-level modeling. The approach is based on model-based control, to build an a priori knowledge model, specific to the assembled package's optical power propagation characteristics, and use this to set the initial "feed-forward" conditions of the automation system. The research was preliminary, algorithms were only verified through simulation and the author was starting hardware implementation and testing, with donated equipment from Kulicke & Soffa, Willow Grove, PA. [9] Some publications involved in Fine tuning. [1-2, 4-8, 10-11] Fuzzy logic feedforward fine tuning was roughly introduced by the authors. [12]

Before fuzzy logic is introduced, we must find the rules how FFC parameters influence motion performance while motion profile (i.e., motion distance, motion maximum velocity, and motion maximum acceleration) and other motion conditions are pre-determined. It’s difficult to theoretically map FFC parameters to the dynamic motion tracking system. Fortunately we can find the trend that FFC parameters affect actual motion after careful study on quite a lot of motions with different FFC parameters and different profile movement conditions. Fuzzy logic should tune FFC parameters along with the correct direction towards expected motion tracking performance. It’s also important to appropriately adjust the tuning incremental steps under different circumstances. That is, according to tracking performance, if the present FFC value is far from the best value (too big or too small), the tuning program shall automatically accelerate tuning to approach the best value quickly; and if the present value is around the best one, the tuning program shall decrease steps to catch the nearby best value accurately. In a word, an “intelligent” tuning method is needed to keep the tuning process fast and accurate, besides, tuning shall be smooth and robust enough. Fuzzy logic suits this tuning process a lot because it is intelligent enough to deal well with complexity of this dynamic tuning process without modeling.

2. Fuzzy Logic Fine Tuning

2.1. How Is Fuzzy Logic Introduced to FFC Tuning

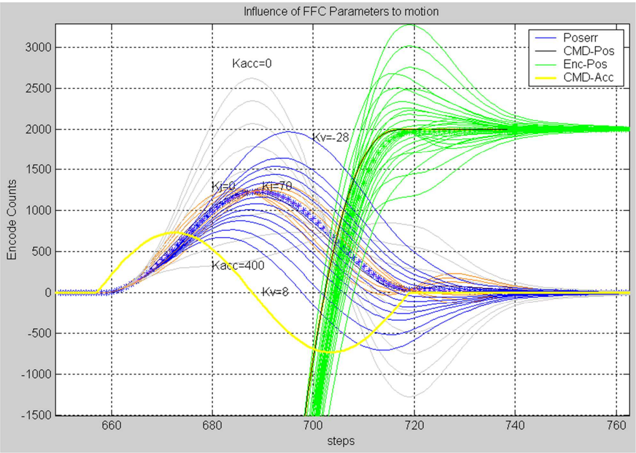

Figure 1. *Influence of FFC Parameters to Motion Performance.*

Figure 1 is the simulation result of FFC coefficient influence to a typical motion control system on electronic packaging equipment. Simulation illustrates the typical influence of FFC coefficient parameters to motion performance. The dynamic segment of the Position Error curve shifts upper or lower if Kvel is gradually increased or decreased, shown as the blue curves in Fig. 2. Kacc (shown as gray curves in Fig. 2) is like a sine wave to make both the curve's maximum and minimum values larger or smaller, while performances at the beginning and end of the dynamic stage of motion keep no change.

The tuning idea is to simultaneously tune Kvel and Kacc by fuzzy logic according to DPE (Dynamic tracking Position Error) and PUS (Position UnderShoot).

Fuzzy rules of tuning Kvel and Kacc are introduced in Table 1 from watching DPE (Dynamic Position Error) and PUS (Position UnderShoot). It must be pointed out that here PUS is a little bit different from usual PUS. PUS may also refer to negative POS if it's mainly overshoot at the end of the motion. And here DPE means the difference of the actual maximum dynamic position error to expected DPE.

This is a typical double input (fuzzy variables DPE and PUS) double output (fuzzy variables Kvel and Kacc) fuzzy inference problem. Fuzzy subsets of DPE, PUS, Kvel and Kacc quote {NB NM NS ZO PS PM PB}. Where, N means Negative; P means Positive; B refers to Big; M refers to Medium; S refers to Small. The 49 fuzzy inference rules are expressed in the following and Table 1.

If DPE is NB and PUS is NB, then Kvel is PB and Kacc is ZO;

If DPE is NB and PUS is NM, then Kvel is PB and Kacc is ZO;

…

If DPE is PB and PUS is PB, then Kvel is NB and Kacc is ZO.

Table 1. *Fuzzy rules of Kvel and Kacc tuning.*

Kvel/Kacc		**PUS**						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
DPE	NB	PB/ZO	PB/ZO	PB/ZO	PB/PS	ZO/PM	ZO/PB	ZO/PB
	NM	PM/NS	PM/ZO	PM/ZO	PS/PS	ZO/PS	ZO/PM	ZO/PB
	NS	PS/NB	PS/NS	PS/ZO	PS/PS	ZO/PS	ZO/PM	ZO/PB
	ZO	ZO/NB	ZO/NS	ZO/NS	ZO/ZO	ZO/PS	ZO/PS	NS/PB
	PS	ZO/NB	ZO/NS	ZO/NS	NS/ZO	NS/ZO	NS/ZO	NM/PM
	PM	ZO/NB	ZO/NM	ZO/NS	NS/NS	NM/ZO	NM/ZO	NM/PM
	PB	ZO/NB	ZO/NB	NM/NM	NM/NM	NM/NS	NB/ZO	NB/ZO

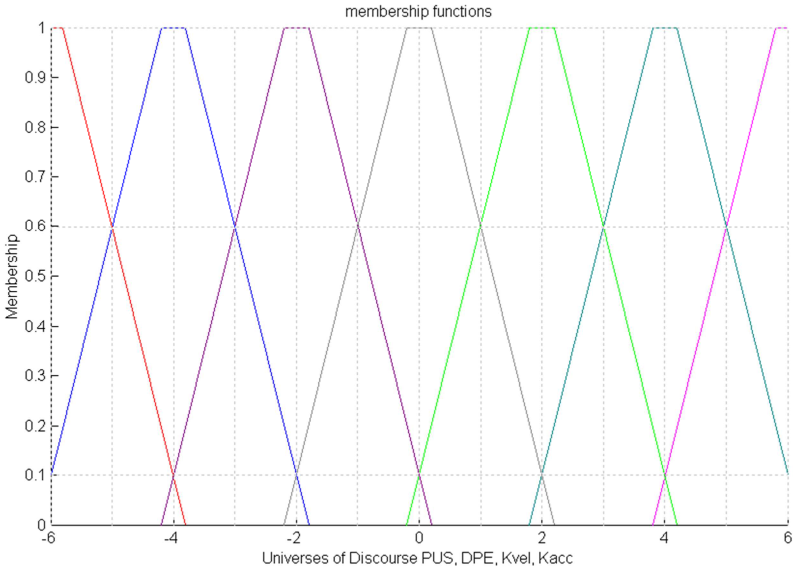

Figure 2. *Trapezoid Memberships.*

To simplify fuzzy inference process, the universes of discourses PUS, DPE, Kvel and Kacc quote {-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5 ,6}. Differences of these four discourses are arranged in quantification factors for PUS and DPE and scaling factors for Kvel and Kacc. Trapezoid membership functions for NB, NM, NS, ZO, PS, PM and PB are quoted and shown in Figure 2.

2.2. Fuzzy Inference

49 Fuzzy relations are as follows:

$$R_{Kvel1} = (NB_{DPE} \times PB_{Kvel}) \circ (NB_{PUS} \times PB_{Kvel}) = R_{DPE2Kvel1} \cap R_{PUS2Kvel1} \tag{1}$$

$$R_{Kacc1} = (NB_{DPE} \times ZO_{Kacc}) \circ (NB_{PUS} \times ZO_{Kacc}) = R_{DPE2Kacc1} \cap R_{PUS2Kacc1} \tag{2}$$

$$\cdots$$

$$R_{Kvel49} = (PB_{DPE} \times NB_{Kvel}) \circ (PB_{PUS} \times NB_{Kvel}) = R_{DPE2Kvel49} \cap R_{PUS2Kvel49} \tag{3}$$

$$R_{Kacc49} = (PB_{DPE} \times ZO_{Kacc}) \circ (PB_{PUS} \times ZO_{Kacc}) = R_{DPE2Kacc49} \cap R_{PUS2Kacc49} \tag{4}$$

The memberships of two fuzzy inputs $\overrightarrow{DPE_i}$ and $\overrightarrow{PUS}_j$ quote [0, …, 0, 1(i), 0, …, 0] and [0, …, 0, 1(j), 0, …, 0], respectively. Then after fuzzification (or fuzzy quantification),

$$\vec{Kvel}_1 = (\overrightarrow{DPE}_i \circ R_{DPE2Kvel1}) \cap (\overrightarrow{PUS}_j \circ R_{PUS2Kvel1}) \tag{5}$$

$$\vec{Kacc}_1 = (\overrightarrow{DPE}_i \circ R_{DPE2Kacc1}) \cap (\overrightarrow{PUS}_j \circ R_{PUS2Kacc1}) \tag{6}$$

$$\cdots$$

$$\vec{Kvel}_{49} = (\overrightarrow{DPE}_i \circ R_{DPE2Kvel49}) \cap (\overrightarrow{PUS}_j \circ R_{PUS2Kvel49}) \tag{7}$$

$$\vec{Kacc}_{49} = (\overrightarrow{DPE}_i \circ R_{DPE2Kacc49}) \cap (\overrightarrow{PUS}_j \circ R_{PUS2Kacc49}) \tag{8}$$

Then the 2 fuzzy output variables, $\vec{Kvel}$ and $\vec{Kacc}$, are deduced in the following:

$$\vec{Kvel}\Big|_{\substack{\overrightarrow{DPE}_i \\ \overrightarrow{PUS}_j}} = \bigcup_{k=1}^{49} \vec{Kvel}_k \tag{9}$$

$$\vec{Kacc}\Big|_{\substack{\overrightarrow{DPE}_i \\ \overrightarrow{PUS}_j}} = \bigcup_{k=1}^{49} \vec{Kacc}_k \tag{10}$$

Centroid Defuzzification Technique (the center of gravity procedure) is quoted for computing the crisp value of Kvel and Kacc,

$$\text{Kvel}\Big|_{\substack{\overrightarrow{DPE}_i \\ \overrightarrow{PUS}_j}} = \sum_{k=1}^{length(\vec{Kvel})} \frac{\mu(k)\vec{Kvel}(k)}{\mu(k)} \tag{11}$$

$$\text{Kacc}\Big|_{\substack{\overrightarrow{DPE}_i \\ \overrightarrow{PUS}_j}} = \sum_{k=1}^{length(\vec{Kacc})} \frac{\mu(k)\vec{Kacc}(k)}{\mu(k)} \tag{12}$$

From above inference, fuzzy tables of Kvel and Kacc are given as Table 2 and Table 3.

Table 2. *Fuzzy table of Kvel.*

Kvel		**PUS**
		{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6}
DPE	-6	{5.2, 5.0, 5.2, 5.0, 4.8, 4.5, 4.2, 2.1, 0.6, 0.0, 0.0, 0.0, 0.0}
	-5	{4.4, 4.4, 4.4, 4.4, 4.1, 3.4, 3.0, 2.2, 0.8, 0.0, 0.0, 0.0, 0.0}
	-4	{3.7, 3.7, 3.7, 3.7, 3.7, 3.0, 2.0, 1.3, 0.6, 0.0, 0.0, 0.0, 0.0}
	-3	{3.0, 3.0, 3.0, 3.0, 3.0, 3.0, 2.0, 1.0, 0.3, 0.0, 0.0, 0.0, 0.0}
	-2	(2.0, 2.0, 2.0, 2.0, 2.0, 2.0, 2.0, 1.0, 0.3, 0.0, -0.6, -0.7, -0.6}
	-1	{1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 0.3, 0.0, -0.7, -2.1, -2.1}
	0	{0.3, 0.3, 0.3, 0.3, 0.0, 0.0, 0.0, 0.0, 0.0, -0.3, -0.6, -2.1, -4.2}
	1	{0.0, 0.0, 0.0, 0.0, -0.3, -1.0, -1.0, -1.0, -1.0, -1.0, -1.3, -2.4, -3.7}
	2	{0.0, 0.0, 0.0, 0.0, -0.3, -1.0, -2.0, -2.0, -2.0, -2.0, -2.0, -2.7, -3.4}
	3	{0.0, 0.0, 0.0, 0.0,-0.3, -1.0,-2.0,-3.0, -3.0, -3.0, -3.0, -3.0, -3.7}
	4	{0.0, 0.0,-0.6,-0.8, -0.6, -1.3,-2.0,-3.0, -3.7, -3.7, -3.7, -3.7, -3.7}
	5	{0.0, 0.0,-0.8, -2.0, -2.0, -2.0,-2.7,-3.0,-3.7,-4.4, -4.4, -4.4,-4.4}
	6	{0.0, 0.0,-0.6,-2.0,-3.4,-3.3,-3.4,-3.7, -3.7, -4.4, -5.2, -5.0, -5.2}

Table 3. *Fuzzy table of Kacc.*

Kacc		PUS
		{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6}
DPE	-6	{-0.3, -0.3, -0.3, 0.0, 0.3, 1.0, 2.0, 3.0, 3.7, 4.1, 4.8, 5.0, 5.2}
	-5	{-1.0, -1.0, -0.3, 0.0, 0.3, 1.0, 2.0, 3.0, 3.0, 3.4, 4.1, 4.4, 5.0}
	-4	{-2.0, -1.3, -0.6, -0.3, 0.0, 1.0, 2.0, 2.3, 2.3, 3.0, 3.7, 4.4, 5.2}
	-3	{-3.0, -2.2, -1.3, -1.0, 0.0, 1.0, 1.7, 2.0, 2.3, 3.0, 3.7, 4.4, 5.0}
	-2	(-4.2, -3.0, -2.0, -1.0, 0.0, 0.7, 1.4, 1.7, 2.0, 3.0, 3.7, 4.1, 4.8}
	-1	{-4.5, -3.3, -2.0, -1.0, -0.7, 0.0, 0.7, 1.0, 2.0, 3.0, 3.0, 3.4, 4.5}
	0	{-4.8, -3.3, -2.0, -1.7, -1.4, -0.7, 0.0, 1.0, 2.0, 2.0, 2.0, 3.0, 4.2}
	1	{-4.5, -3.3, -2.3, -2.0, -1.7, -1.0, 0.0, 1.0, 1.0, 1.0, 1.3, 2.4, 3.7}
	2	{-4.8, -3.3, -2.3, -2.3, -2.0, -1.0, 0.0, 0.0, 0.0, 0.3, 0.6, 2.0, 3.4}
	3	{-4.5, -3.4, -3.0, -3.0, -2.0, -1.0, -1.0, -1.0, -0.3, 0.0, 0.8, 2.0, 3.3}
	4	{-4.8, -4.1, -3.7, -3.0, -2.0, -2.0, -2.0, -1.3, -0.6, -0.3, 0.4, 2.0, 3.4}
	5	{-5.0, -4.4, -4.1, -3.4, -3.0, -3.0, -2.7, -2.0, -1.3, -1.0, 0.4, 2.0, 2.0}
	6	{-5.2, -5.0, -4.8, -4.1, -3.7, -3.7, -3.4, -2.7, -2.0, -1.0, 0.4, 0.8, 0.6}

Quantification factors for PUS and DPE and scaling factors for Kvel and Kacc are important and shall be carefully selected. In FFC tuning, Variable Step Size Method is quoted in order to acquire a fast and accurate tuning result.

2.3. Experimental Results and Analysis

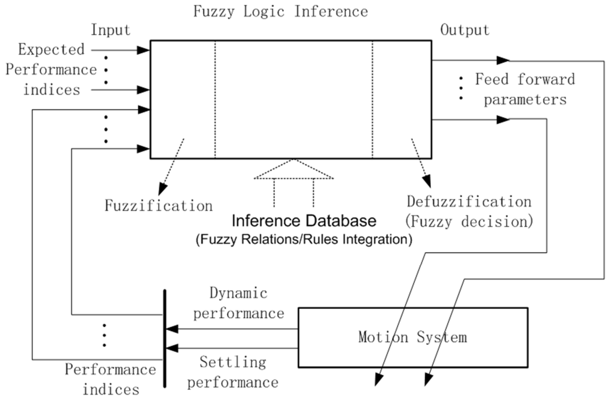

Figure 3. *Illustration of fuzzy logic feed forward tuning process.*

The Fuzzy FFC Tuning process is illustrated in Figure 3. And the tuning algorithm is programmed and tested on wire bonding machines.

2.3.1. Time Consumption Test

Table 4. *Time consumption comparison of two FFC tuning methods.*

	MC1	MC2	MC3	MC4
Fuzzy FFC Tuning	1 min	1 min 14 sec	1 min 5 sec	1 min 50 sec
Traditional Auto Tuning	10-12 minutes			

As shown in Table 4, Fuzzy FFC tuning saves 80% tuning time.

2.3.2. Repeatability Test

Repeatability tests were conducted on MC1 (3 times) and MC2 (5 times), as shown in Table 5 and Table 6. Standard Deviations are within 10% of mean values. The results show good repeatability of fuzzy FFC tuning.

Table 5. *Repeatability Test of Fuzzy FFC Tuning on MC1.*

	Motion 1		Motion 2		Motion 3		Motion 4		Motion 5		Motion 6	
	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc
Test 1	19	173	17	145	15	183	14	230	15	152	18	171
Test 2	19	174	19	154	15	183	14	233	15	150	18	175
Test 3	19	175	17	145	15	183	14	230	15	149	18	174
Mean	19	174	18	148	15	183	14	231	15	150	18	173
STDev	0	1	1.2	5.2	0	0	0	1.7	0	1.5	0	2.1
STD%	0	0.6	6.5	3.5	0	0	0	0.7	0	1	0	1.2

Table 6. *Repeatability Test of Fuzzy FFC Tuning on MC2.*

	Motion 1		Motion 2		Motion 3		Motion 4		Motion 5		Motion 6	
	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc	Kvel	Kacc
Test 1	18	254	18	226	14	261	14	358	17	234	17	260
Test 2	18	256	18	222	13	266	14	360	16	228	18	259
Test 3	18	252	18	222	14	261	14	356	17	228	18	256
Test 4	18	246	18	223	14	260	14	356	17	234	18	250
Test 5	19	249	17	212	14	263	14	355	17	230	19	251
Mean	18	251	18	221	14	262	14	357	17	231	18	255
STDev	0.4	4	0.4	5.3	0.4	2.4	0	2	0.4	3	0.7	4.5
STD%	2.5	1.6	2.5	2.4	3.2	0.9	0	0.6	2.7	1.3	3.9	1.8

2.3.3. Performance Test

Performances are captured on above 4 machines. As an example, Table 7 shows the performance indices tuned on MC1. Extracted from both dynamic and static performances, motion performance indices (DPE: Maximum Dynamic Position Error; Sserr: Static settling position error; Ts: Settling time; Pos: Position overshoot; Pus: Position undershoot; Dve: Maximum Dynamic velocity error; Vo: Velocity overshoot; Vu: Velocity undershoot.) meet with packaging process requirements for every specific motion of six typical motions.

Table 7. *Performance indices of Fuzzy Tuning on MC1.*

	Kvel/Kacc	DPE	Sserr	Ts	Pos	Pus	Dve	Vo	Vu
Motion 1	19/173	1542	-29.2	0.13	20.4	2.1	77	6.7	3.2
Motion 2	17/145	225.5	-23	0.13	9.5	6.5	32.6	8.5	2.2
Motion 3	15/183	806	-20.6	0.13	5.1	23.7	70.4	6.7	1.6
Motion 4	14/226	251.5	-26.4	0.13	13.8	5.8	15.4	3.4	2.2
Motion 5	15/151	185.8	-23.7	0.13	7.3	6.8	30.2	7.1	2.1
Motion 6	15/178	1528.9	-9.3	0.13	11.9	21.8	87.6	7.3	1.6

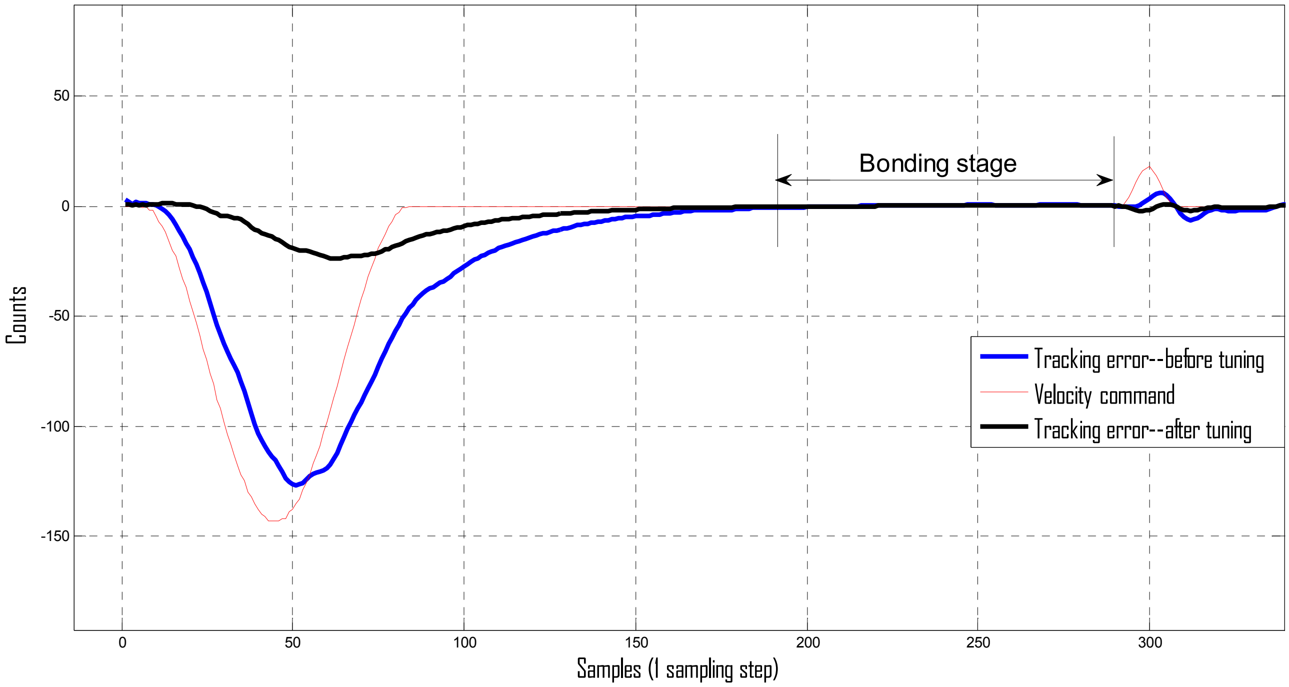

Figure 4. *Tracking errors before and after tuning.*

Figure 4 shows tuning results of tracking position errors. Compared to position tracking errors before tuning, they are improved (diminished) a lot at both dynamic tracking and settling stages. Tracking performance tuned helps to reduce offsets along XY direction during bonding, and save waiting time for settling before bonding.

Experiments show that Fuzzy logic approach is a reliable and time-saving FFC tuning method. And, it's robust enough.

3. Conclusion

Feed forward coefficient parameters of motion systems on packaging equipment are tuned by fuzzy logic to satisfy multiple targets of dynamic and settling performance requirements. The major contributions of Fuzzy FFC Tuning are to save tuning time and get controllable dynamicmotion performance. This intelligent tuning therapy is used to shape dynamic tracking, so as to help on performance in the settling stage. Experimental results demonstrate that fuzzy logic is a feasible and effective solution to motion performance fine tuning on electronic packaging equipment.

Acknowledgements

Authors show their acknowledgments to supporters of this paper: the National Basic Research Program of China (973 Program No.2011CB013104), the Applied Science and Technology Research and Development Project of Guangdong Province (No. 2015B010133005), and the Scientific Innovation Key Project from Guangdong Educational Department (No. 2012CXZD0020, No. 2012A080303004).

References

[1] Iuliana R., Maarten S., Rogier E., "Adaptive Iterative Learning Control for High Precision Motion Systems," IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 16, NO. 5, SEPTEMBER 2008, pp. 1075-1082.

[2] Aurelio P., Antonio V., "An Iterative Approach for Noncausal Feedforward Tuning", Proceedings of the 2007 American Control Conference, Marriott Marquis Hotel at Times Square, New York City, USA, July 11-13, 2007, pp. 1251-1256.

[3] Marcel H., Daan H., Maarten S., "MIMO feed-forward designin wafer scanners using a gradient approximation-based algorithm" Control EngineeringPractice18(2010), pp. 495–506.

[4] Stearns H., Mishra S., Tomizuka M., "Iterative Tuning of Feedforward Controller with Force Ripple Compensation for Wafer Stage," 10th IEEE International Workshop, AMC '08, Trento, Italy, 2008, pp. 234-239.

[5] Morteza M., "Self-tuning PID controller to three-axis stabilization of a satellite with unknown parameters", International Journal of Non-Linear Mechanics 49 (2013), pp. 50–56.

[6] Meric C., SerdarI., "A novel auto-tuning PID control mechanism for nonlinear systems", ISA Transactions 58 (2015), pp. 292–308.

[7] Onur K., MujdeG., IbrahimE., EnginY., TufanK., "Online tuning of fuzzy PID controllers via rule weighing based on normalized acceleration", Engineering Applications of Artificial Intelligence 26 (2013) , pp. 184–197.

[8] Osama E., Mohammad E., Nabila M., "Development of Self -Tuning Fuzzy Iterative Learning Control for Controlling a Mechatronic System," International Journal of Information and Electronics Engineering, Vol. 2, No. 4(2012), pp. 565-569.

[9] Timothy P. K., Allon G., Shubham K. B., "Model-based Optoelectronic Packaging Automation," IEEE Journal of Selected Topics in Quantum Electronics, Vol. 10, No. 3 (2004), pp. 445-454.

[10] Frank B., Tom O., Maarten S., "Accuracy Aspects in Motion Feedforward Tuning," *2014 American Control Conference (ACC)*, Portland, Oregon, USA, 2014, pp. 2178-2183.

[11] Dennis B., Niels V. D., "Combined Input Shaping and Feedforward Control for Flexible Motion Systems," 2012 American Control Conference, Fairmont Queen Elizabeth, Montréal, Canada, 2012, pp. 2473-2478.

[12] Yunbo H., Xin C., etc., Invited paper, 17th Electronics Packaging Technology Conference, EPTC 2015 Singapore, "Development of High Performance Bonding Machines with Improved Motion Control and Intelligent Fine Tuning Algorithm," 2-4 December 2015.