Turret Punch Time Estimate for Sheet Metal Parts using Machine Learning Approach

Yearn-Tzuo (Andrew) Hwang
Department of Mechanical Engineering
National Yang Ming Chiao Tung University
Hsinchu, Taiwan 300093
yearnhwang@nycu.edu.tw; ahifcbc@gmail.com

Ting-Yi Zhang, and Hung-Chun Hsu
Graduate Students, Department of Mechanical Engineering
National Yang Ming Chiao Tung University
Hsinchu, Taiwan 300096
Tingyizhang22@gmail.com; a3744089@gmail.com

Abstract

Turret punch machines have been used to produce parts in sheet metal shops since the 1950s. It is common to place multiple parts of different sizes and designs from multiple customers on one sheet blank to manufacture. However, this practice poses challenges in accurately calculating the fabrication time and subsequent manufacturing cost of each part. This is because turret punch machines only display the total machine run time per entire sheet blank, whereas the punching process depends not only on the parts themselves but also on the punching tools used and punching routes undertaken. This paper proposes a Machine Learning approach to solve this longstanding punching time distribution problem by using the punching tools' hitting counts, the moving distances of sheet blanks and the number of tool changes during production time. 425 sheet blanks of parts with various designs and sizes are processed, and machine run times for each sheet blank are recorded from a turret punch simulation program. Five common types of turret punching operations ("*hit*"), such as nibbling, single hit, line shearing and etc., along with transverse and longitudinal moving distances of sheet blanks and tool changes are considered in this research. Every part on the sheet blanks is broken down into its component geometric entities (e.g. line, arc, hole, etc.). Geometric entities are then grouped by the kind of punching hit and/or assigned punching tool itself. Three supervised Machine Learning models, namely Linear Regression, Ridge Regression, and Lasso Regression are used and compared using 10-fold cross-validation process each consisting of 383 (90%) sheet blanks for training and the remaining (10%) for validation. The preliminary results indicate that the proposed Machine Learning approach is viable and consistent in determining the fabrication time for every part on sheet blanks manufactured by turret punch machines. The approach can be integrated with real-time data collection of the counts of punching hits, parts' geometry, sheet blank's motion and machine run time on any turret punch machines to determine the fabrication time and subsequent manufacturing cost of each part in the Industry 4.0 era.

Keywords

Machine Learning, Turret Punch Machine, Time Estimate, Sheet Metal Fabrication Cost, Automation, Industry 4.0

1. Introduction

Sheet metal parts are frequently manufactured using turret punch machines and laser cutting machines. A common practice is to place multiple parts, with different geometries from various customers, on a single sheet blank to enhance material utilization. However, this practice makes it challenging to accurately calculate the machine run time and the subsequent manufacturing cost of each part on the sheet blanks (Hwang and Yang, 2023). Inaccurate machine run time estimation poses challenges in determining operational profit or loss of machine shops and affects the quoting ability for customer orders. Currently, the industry relies on subjective estimation or simple calculation (Bargelis and Rimasauskas, 2007) based on part geometry to assign each part a machine run time, which lacks accuracy and consistency. Furthermore, if a different machine model is used, part's machine run time must be recalculated, which can lead to confusion, accounting errors, and delays.

The main obstacle in resolving this issue for turret punch machines is that machines only display the total run time for the entire sheet blank, and there are infinite fabrication possibilities for the parts on the sheet blank. Although some machines may output "time stamp" data, it requires significant efforts to analyze data log files in order to relate punching tools, punching routes and time stamp data for individual parts.

A turret punching work example is presented here to illustrate the issue. On an aluminum sheet blank of 0.081" thickness and 40" by 80" size, six parts are arranged to be manufactured as shown in Figure 1. A numerical control program ("*NC"*) of G-code commands is created by a CAM software. The corresponding G-code simulation for the first example part is given in Figure 2, and a snap shot of punching operations in Figure 3. This example part contains rectangular holes, circular holes, obround holes and octagon holes. Furthermore, the part requires drill-screw operations on several circular holes. This process involves initially punching these holes using a 'round' tool (circular shape) and subsequently inserting a drill-screw tool into the holes. For rectangular holes, the material covering the holes can be either punched only along the boundary of the holes or by a series of line-by-line hits across the rectangular holes. Parts' outer shapes are usually processed in the last phase of fabrication. By doing so, we can avoid the finished parts to roam on the machine table while fabrication is still in place. It is also common to leave small material attachment (called "*micro-joint*") between the parts and sheet blank. If should be noted that if drill-screw or other operations would take place after part's punching is complete, it is imperative to have micro-joints.

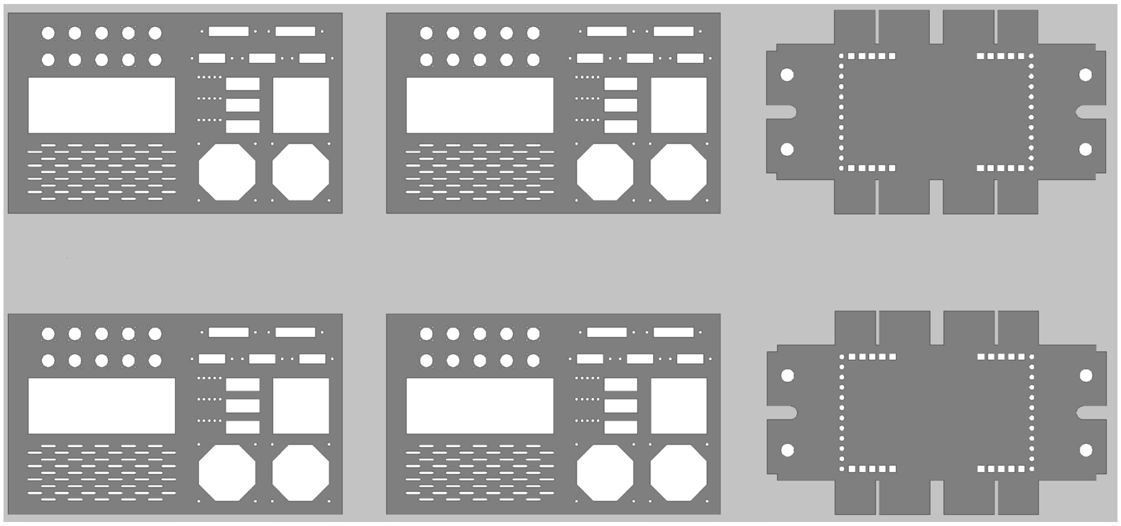


Figure 1. Various parts on a sheet blank

Similar consideration will be given to the other example part. After punching tools and punching sequences of the sheet blank are assigned to all parts, the actual turret punching operations take place. All operations undertaking the same punching tool are executed before switching to another turret punch tool in order to minimize turret tool changes. On the other hand, each part associates all geometric entities that it is composed of, and each geometric entity is recorded with all punching operations that it has received.

Machine run time for fabricating a sheet blank is displayed on the turret punch machine controller after the fabrication process is complete. However, this research uses a simulation software rather than a turret punch machine to collect machine run time data for 425 training sheet blanks and 1 test sheet blank.

Three supervised Machine Learning methods are chosen to distribute the machine run time of a sheet blank to the parts on the sheet blank. The "features" used in the three Machine Learning models are the punching tools' hit counts, moving lengths of the sheet blank, number of tool changes, and number of special commands. Punching tools' hit counts are the sum of turret stokes based on the types of punching operations (G-code). The sheet blank's motion is discomposed into the transverse and longitudinal moving lengths. Our preliminary result demonstrates that the proposed approach is a viable method to systematically derive a machine run time for each part on sheet blank.

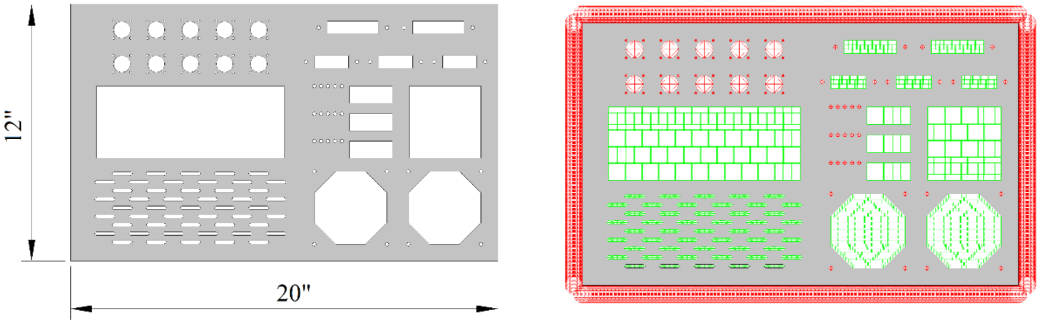


Figure 2. Punching process simulation for one example part


Courtesy of Takachi Electronics Enclosure Co. video at YouTube

Figure 3. Actual turret punch machine operations for the example part

1.1 Objectives

The main objective of this research is to find a more reliable approach to accurately calculate machine run time for each part on sheet blanks fabricated using turret punch machines. This objective is achieved by applying Machine Learning methods onto collected turret punch machine operation data, such as counts of punching tool hits, moving lengths, special commands and machine run time of sheet blanks. The second objective is to develop a punching simulation program that will introduce random fluctuation of operational time for every punching operation and material movement, and provide simulated machine run time for sheet blanks. The third objective is to train three Machine Learning models, namely the Linear Regression, Ridge Regression and Lasso Regression, compare their results and verify whether the proposed approach is a viable solution or not. The added random fluctuation onto data can make the simulated machine run time not deterministic, thus closer to a true turret punch machine. In the meantime, randomly created parts in various geometries and sizes are randomly placed on the sheet blanks before the relevant punching tools, operations, routes and sequences data are collected. 425 training sheet blanks are executed to train the Machine Learning models, and 1 test sheet blank to verify the viability and consistency of the proposed approach.

2. Turret Punch Review

In contrast to laser cutting machines, where the laser head moves during fabrication, turret punch machines operate differently. On a turret punch machine, it is the sheet blank that moves on the machine table while an active punching tool remains fixed at the "hit" position. The punching tools, which come in various shapes and sizes, are housed inside the tool stations of the turret. Based on the geometric characteristics of the parts' entities, the appropriate punching tools, routes, and sequences are determined. Initially, the turret station with the required tool is positioned at the machine's "hit" position. Subsequently, the sheet blank aligns itself to ensure that the punching location coincides precisely with the selected tool in the turret station. To fabricate the parts, the punching tool is pressed onto the sheet blank, removing material and creating holes. After each punch, the sheet blank moves to the next "hit" position, awaiting the next strike of the punching tool. This punch-move process continues until all geometric entities that require the same punching tool are processed. Then, a "tool change" occurs as the turret station holding the next necessary tool moves into position. The remaining geometric entities of the parts are processed until all the entities on the sheet blank have been addressed. Typically, the outer boundaries of the parts are processed last to ensure stability during punching operations. To prevent collisions between the sheet blank and material scraps on the machine table, small scrap pieces are punched out one by one across the holes of each part. Given the infinite possibilities for part sizes, shapes, orientations, and locations, there are numerous options for punching tools, routes, and sequences for the parts on the sheet blanks, regardless of their similarity. Currently, the industry relies on experienced workers to assign a part's machine run time, but this approach lacks reliability and consistency. Furthermore, it is not uncommon to switch turret punch machines within machine shops, rendering machine run times calculated based on previous machines obsolete in such cases.

3. Model Formulation

A sheet blank's movement on turret punch machines is dictated by two motors controlling clamping bars in X and Y directions respectively. It is assumed their accelerations in X and Y directions are different for the clamping bars. Furthermore, it is observed that different type of punching operation exhibits different punching speed. All these factors ultimately affect the machine run time for fabricating sheet blanks. With these presumptions, our objective of calculating the machine run time using Machine Learning method is achieved by treating these factors as Machine Learning model "*features*", and collect features' data and machine run time after executing sheet blanks. The Machine Learning model "features" and associated G-codes are discussed below.

3.1 Slots/holes by "single hit"

A *one-stroke-down* punching operation is categorized as the "*single hit*" type regardless the shape of tools and holes. The machine presses down the punching tool to remove material and make a hole on the sheet blank. User can simply specify the X & Y coordinates in the NC program to initiate a "*single hit*" operation. Three other G-code commands - *punch along a line* (G28), *punch along an arc* (G29), and *punch by two-dimensional grid* (G36) are also considered this type as shown in Figure 4. If the punching tool happens to be a rectangular tool, *punch along a line* may be referred to as "*slotting*". For standard-shape holes, the geometric shape of the tool must match that of the hole in "single hit" operations. In other words, the geometric center of the punching tool coincides with that of the hole on the sheet blank. In this research, the count of single hits, the transverse and longitudinal moving ("*travel*") distances are "features" to be collected. Take the two-dimensional grid (G36) case in Figure 4 as an example, the collected "feature" data would be 24 single hits, 36" transverse travel distance, and 5" longitudinal travel distance.

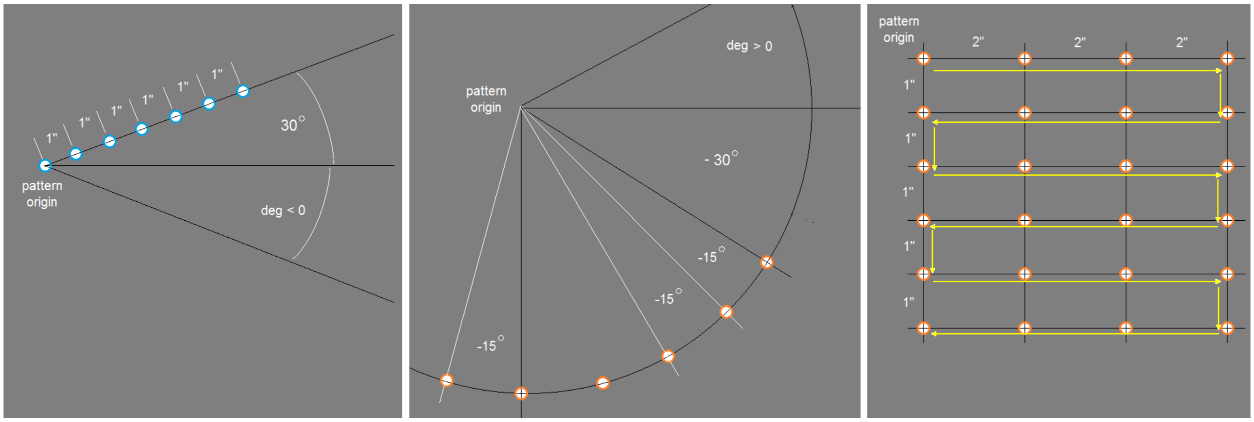


Figure 4. "Single hit" punching type on a sheet blank

3.2 Rectangular holes punched on boundary

Rectangular holes that align to the machine table's coordinate system can be fabricated by punching along the 4 sides of rectangular holes using G67 command. Generally speaking, "square" tool is preferred for this operation due to its simplicity in arranging the tool to punch along the horizontal and longitudinal sides of rectangular holes (Figure 5). It should be noted that there is no restriction as to choosing a rectangular tool for this operation, except that more calculation is needed in order to cope with rectangular tool's width and height. The data to be collected for this operation is the count of tool hits of this type (G67), the sheet blank's transverse and longitudinal travel distances exhibited in this operation.

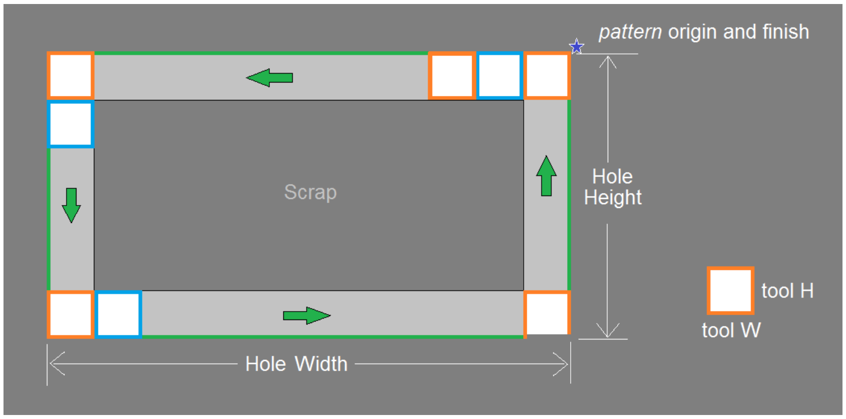


Figure 5. Rectangular hole by "boundary" type punching

3.3 Rectangular holes punched on area (*SHEAR*)

It is not uncommon to completely punch out material inside the rectangular holes of parts small piece by small piece. By doing so, there would not be a large metal scrap piece left on the machine table when the process of fabricating a rectangular hole is finished. The G-code command for this operation is G66 and the operation is sometime referred to as *Shearing* operation. The *Shearing* command (G66) can be executed on inclined rectangular holes as shown in Figure 6. The data to be collected for *Shearing* operations are the count of *Shearing* hits, sheet blank's transverse and longitudinal moving distances during the *Shearing* operation. When the rectangular holes are inclined, however, it is only necessary to collect the larger movement between the transverse and longitudinal distances of two consecutive *Shearing* operations and the count of *Shearing* hits. This is because the sheet blank moves simultaneously in both transverse and longitudinal directions, and only the longer distance of X or Y direction contributes to the machine run time because it takes more time. Take Figure 6 as an example, there would be 42 *Shearing* hits (assuming 7 per line), 36" of horizontal travel distance and 4" of vertical travel distance provided the inclined angle θ is 45° or less.

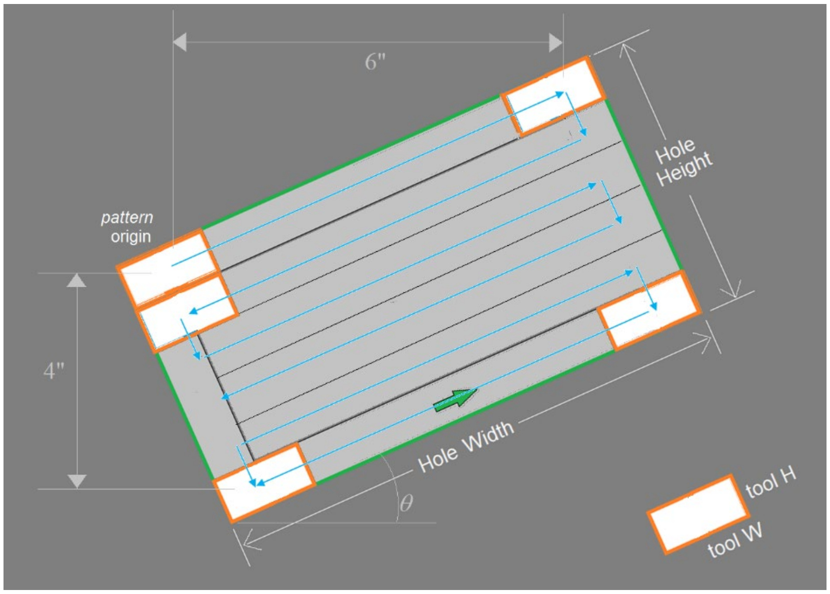


Figure 6. Rectangular hole by "shear" type punching

3.4 Nibbling on arc

"Nibbling" punching operation for turret punch machines refers to a process that removes material by making holes along a profile through multiple punching strokes. A nibbling process consists of fast movement of the sheet blank and fast stroke punching of the punching tool. It should be noted that nibbling operation is not suitable for thick metal sheet, such as Stainless steel of 0.12” thickness (*gauge 11*) or thicker. The G68 command is used for nibbling an arc or a circle. For circular holes without a “round” tool of same size, this command is particularly useful because “single hit” operation is not suitable. Due to the fact that the nibbling operation is performed as continuous punching hits by a small length offset (“*pitch”*) between two consecutive hits, only the count of “*nibbling hits on arc*” hits is collected as feature in this research. The transverse and longitudinal travel distances are excluded. As shown in Figure 7 where the profile arc length is 1.0” long and the nibbling *pitch* is 0.01”, the count of “*nibbling on arc*” hits is 101.

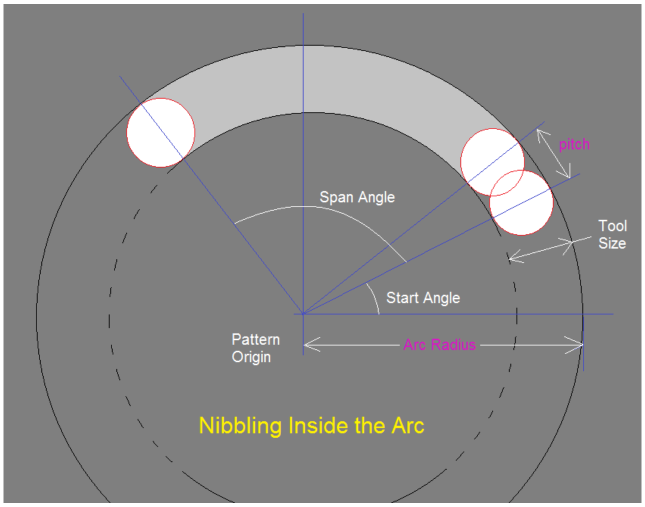


Figure 7. Nibbling operation on the inside of an arc (G68)

3.5 Nibbling on line

Nibbling on line operation (G69) is similar to *nibbling on arc* operation except that its profile is a straight line. Combining with *nibbling on arc* operation, more complex geometry can be formed by nibbling operations. For “*nibbling on line*” operations, the count of nibbling hits is collected for Machine Learning model training. In this research, the combined counts of G68 and G69 nibbling hits is treated as the “*nibbling*” feature (Figure 8).

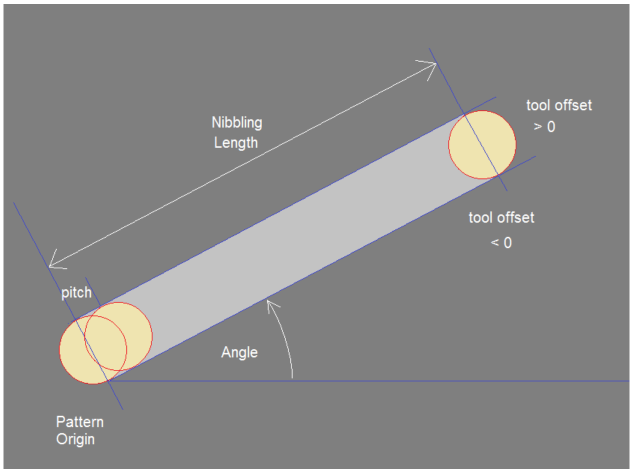


Figure 8. Nibbling operation along a line (G69)

3.6 Two-hit/three-hit operations

Multiple-hit punching operations refer to those that exhibit consecutive hits using tool(s) placed in one turret station and punching at the same location on sheet blanks. They require no transverse or longitudinal movements. It should be noted that non-consecutive punching hits at the same location is not considered to be multiple-hit operations and treated as separate punching operations. As shown in Figure 9(a), a *countersink forming* is fabricated by a special tool which first punches a circular hole with a "round" tool, followed by hard-pressed ("*coining*") of a *cone-shape* tool. This would be collected as one count of two-hit "feature" in this research. By comparison, the *shape* in Figure 9(b) would be collected as three *single hits* by three standard tools - "round" tool, "horizontal rectangular" tool and "vertical rectangular" tool. Because the tool change operation is needed to fabricate this hole, three "single hit" counts is collected.

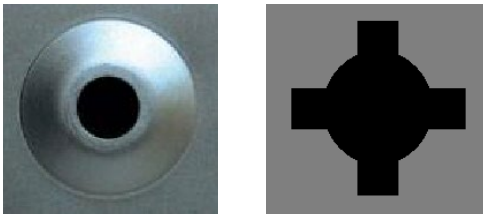

(a) (b)

Figure 9. *Countersink* and a cut-off shape of two-hit and three-hit operations

3.7 Special tool punching operation

Special tools can be used to make nonstandard holes or forms, referred to as *shape* or *forming*, as shown in Figure 10. Punching operations using the special tools are similar to the "*single hit*" operations or multiple-hit operations. However, due to the fact that the geometric characteristics of special tools can be very different, operational time involving special tools may quite differ from that for the standard tools. Therefore, "punching by special tool" operations are treated as another "feature" to be collected in this research. The count of special tool punching, the transverse and longitudinal travel distances are collected for this category. It should be noted that *weighting* can be assigned to certain special tools' operations in order to reflect the different processing time among special tools themselves in this research.

Figure 10. Examples of punching operation with special tools

3.8 Turret tool change

Punching tools needed for manufacturing sheet metal parts are placed inside turret stations before the fabrication starts. When a different tool is needed during fabrication, the sheet blank will stop its motion while the machine turret rotates to bring the correct tool to the punching location of the machine. The tool change operation will add processing time to the machine run time for the sheet blank. When tool change takes place, the rotating degrees of the turret station is collected as a feature for the proposed Machine Learning approach.

3.9 Special commands

Special commands are used to perform specific activities that are necessary for manufacturing using turret punch machines. Operational times to execute these special commands are included in the machine run time of sheet blanks. For example, the *sheet-loading* operation, *trap-door-open* operation, and *ending-return-origin* operation are common for many turret punch machines. Special commands are collected as a Machine Learning model feature in terms of *second* because each special command is pre-assigned with an operational time in *second*. If 2.5 seconds for *sheet-loading*, 3 seconds for *trap-door-open*, and 1.5 seconds for *ending-return-origin* are pre-assigned for a turret punch machine, the feature value of 10.0 will be collected for 2 *trap-door-open* commands with loading/unloading operations for one sheet blank using this machine.

3.10 Machine run time

Machine run time is the time period that a turret punch machine takes to manufacture all metal parts on one sheet blank. It starts from the loading of sheet blank to its returning to the machine table's origin, and can be found on the machine controller screen after the fabrication is complete. Small difference in machine run time may be observed for fabricating two identical sheet blanks sometimes. Because this research uses a simulation software rather than a turret punch machine to collect machine run time data for sheet blanks, random variation in operational time is intentionally added to every punching operation, sheet blank movement, as well as *Shearing pitch* to make machine run time less deterministic. The ranges of percentage for the random variations in simulation program are pre-assigned and adjustable based on the type of turret punching operations. By doing so, the simulated machine run time for two identical sheet blanks would be less likely identical. The simulation software will sum up all operational time of every tool hits, sheet blank's moves, tool changes, and special command wait time resulting from the parts on a sheet blank, and produce the machine run time for that executed sheet blank.

4. Data and Regression Model

4.1 Machine run time data

Machine run time of a sheet blank can be found from the machine controller after the fabricating process is complete. Instead of using a turret punch machine, we use a simulation software to provide machine run times for the 425 training sheet blanks in this research (Table 1). Each sheet blank contains automatically generated test parts in various parameterized shapes and sizes, and placed randomly on the blank sheet as illustrated in Figure 11. The simulation software sums up all operation times for each of the 425-training sheet blanks and produces 425 machine run times.

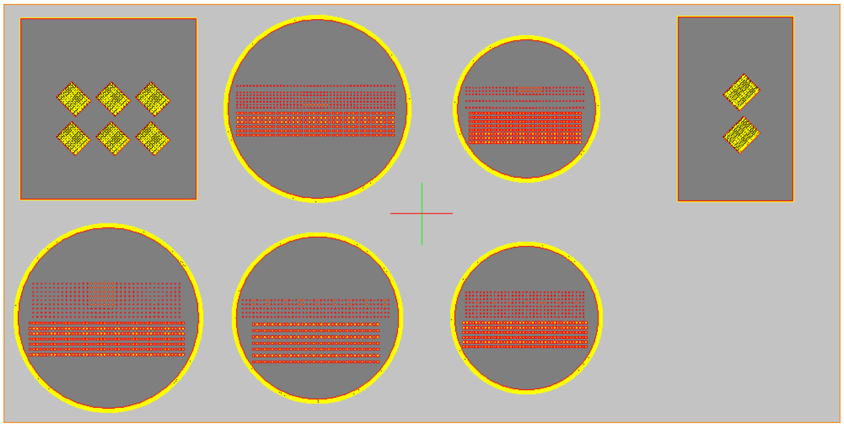


Figure 11. Sample of training sheet blanks with arbitrary shapes and sizes at random locations

4.2 Punching operation data

Punching operation data contains the counts for five types of punch hits, sheet blank's horizontal and longitudinal moving distances, tool changes, and special commands' wait time. These data are collected after punching tools, routes and sequences are determined for the parts on sheet blanks. A graphics simulation utility is also developed to view the punching routes and sequences and verify the accuracy of punching operation data.

4.3 Regression Models

Three regression models— Linear Regression, Ridge Regression and Lasso Regression— are chosen to be trained with the simulated machine run time and punching operation data collected from the 425 sheet blanks. The objective of training is to minimize the error functions defined in three models using the data collected. The error functions for the Linear Regression, Ridge Regression and Lasso Regression models are given in the Eq. 1, 2, and 3 respectively. A 10-fold cross-validation training procedure is adopted in this research. In addition, permutation of data of 425 sheet blanks are performed before carrying out the 10-fold cross-validation procedure. The training process is repeated many times until it is determined that three models have converged. The accuracy scores (99%) of the three regression models based on data of the 425 sheet blanks are shown in Table 2. Note that the accuracy scores listed in Table 1 are the average value of scores of the 10-fold cross-validation training procedure.

$$\boldsymbol{E}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sum_i (\hat{y}_i - y_i)^2 = \sum_i (wx_i + b - y_i)^2 \tag{1}$$

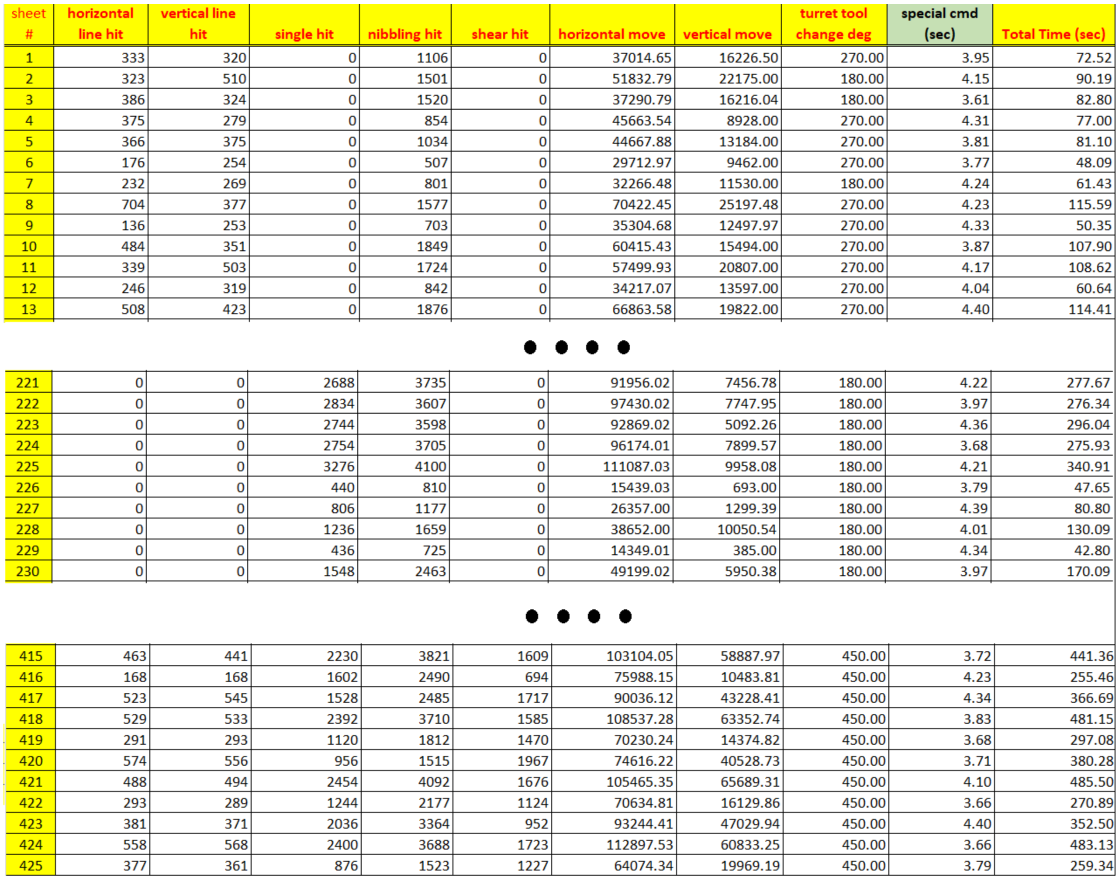
$$\boldsymbol{E}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sum_i (wx_i + b - y_i)^2 + \alpha \sum_i {w_i}^2 \tag{2}$$

$$\boldsymbol{E}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sum_i (wx_i + b - y_i)^2 + \alpha \sum_i |w_i| \tag{3}$$

5. Results and Discussion

With the data collected from 425 sheet blanks, three chosen Machine Learning models with 10-fold cross-validation have resulted in the same level of accuracy (99%) as shown in Table 2. It indicates that sufficient amount of data is collected because all three models have converged to same level of accuracy. Using the same modeling parameters found in the 10-fold cross-validation training process, we create three new models using the whole data of 425 sheets. The three new Machine Learning models are applied to the test sheet blank to predict the machine run time. Because the test sheet blank is not used in training process, the accuracy score from the test sheet blank is a good indication of training outcome. The punching data of the test sheet blank is shown in Table 3, and the accuracy scores of 425 training sheet blanks and test sheet blank are given in Table 4 respectively. Again, it shows that three new models reach the same level of accuracy (99%). Comparing the accuracy levels of training dataset and test dataset (99% vs 88%), it indicates that "overfitting" may have occurred using the data generated by the simulation program. The accuracy levels, however, confirmed that our proposed Machine Learning approach is a consistent approach to estimate the machine run time even though it is using the psuedo data from a simulation program. Further investigation is still in progress to check if the data accuracy or consistency from the simulation software is maintained.

Table 1. Machine run time and punching operation data of 425 training sheet blanks

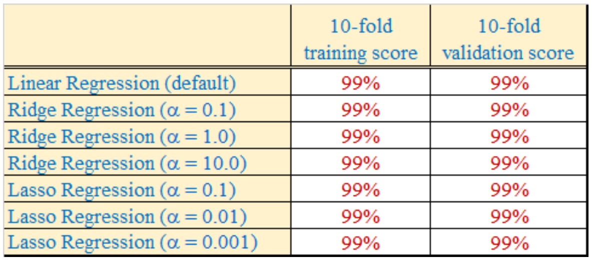


sheet #	horizontal line hit	vertical line hit	single hit	nibbling hit	shear hit	horizontal move	vertical move	turret tool change deg	special cmd (sec)	Total Time (sec)
1	333	320	0	1106	0	37014.65	16226.50	270.00	3.95	72.52
2	323	510	0	1501	0	51832.79	22175.00	180.00	4.15	90.19
3	386	324	0	1520	0	37290.79	16216.04	180.00	3.61	82.80
4	375	279	0	854	0	45663.54	8928.00	270.00	4.31	77.00
5	366	375	0	1034	0	44667.88	13184.00	270.00	3.81	81.10
6	176	254	0	507	0	29712.97	9462.00	270.00	3.77	48.09
7	232	269	0	801	0	32266.48	11530.00	180.00	4.24	61.43
8	704	377	0	1577	0	70422.45	25197.48	270.00	4.23	115.59
9	136	253	0	703	0	35304.68	12497.97	270.00	4.33	50.35
10	484	351	0	1849	0	60415.43	15494.00	270.00	3.87	107.90
11	339	503	0	1724	0	57499.93	20807.00	270.00	4.17	108.62
12	246	319	0	842	0	34217.07	13597.00	270.00	4.04	60.64
13	508	423	0	1876	0	66863.58	19822.00	270.00	4.40	114.41
						● ● ● ●				
221	0	0	2688	3735	0	91956.02	7456.78	180.00	4.22	277.67
222	0	0	2834	3607	0	97430.02	7747.95	180.00	3.97	276.34
223	0	0	2744	3598	0	92869.02	5092.26	180.00	4.36	296.04
224	0	0	2754	3705	0	96174.01	7899.57	180.00	3.68	275.93
225	0	0	3276	4100	0	111087.03	9958.08	180.00	4.21	340.91
226	0	0	440	810	0	15439.03	693.00	180.00	3.79	47.65
227	0	0	806	1177	0	26357.00	1299.39	180.00	4.39	80.80
228	0	0	1236	1659	0	38652.00	10050.54	180.00	4.01	130.09
229	0	0	436	725	0	14349.01	385.00	180.00	4.34	42.80
230	0	0	1548	2463	0	49199.02	5950.38	180.00	3.97	170.09
						● ● ● ●				
415	463	441	2230	3821	1609	103104.05	58887.97	450.00	3.72	441.36
416	168	168	1602	2490	694	75988.15	10483.81	450.00	4.23	255.46
417	523	545	1528	2485	1717	90036.12	43228.41	450.00	4.34	366.69
418	529	533	2392	3710	1585	108537.28	63352.74	450.00	3.83	481.15
419	291	293	1120	1812	1470	70230.24	14374.82	450.00	3.68	297.08
420	574	556	956	1515	1967	74616.22	40528.73	450.00	3.71	380.28
421	488	494	2454	4092	1676	105465.35	65689.31	450.00	4.10	485.50
422	293	289	1244	2177	1124	70634.81	16129.86	450.00	3.66	270.89
423	381	371	2036	3364	952	93244.41	47029.94	450.00	4.40	352.50
424	558	568	2400	3688	1723	112897.53	60833.25	450.00	3.66	483.13
425	377	361	876	1523	1227	64074.34	19969.19	450.00	3.79	259.34

Because the accuracy level is almost identical among three models, Ridge Regression model is chosen to distribute the machine run time of a sheet blank to each individual part on the sheet blank. Note that the machine run time for a sheet blank is calculated based on the Machine Learning model "features", namely hit counts, moving distances, tool changes and special commands. The features of a sheet blank are the summation of features from the parts on the sheet blank. Therefore, if we apply feature values of a part in the Ridge Regression model, it gives the fair share of the machine run time for that part.

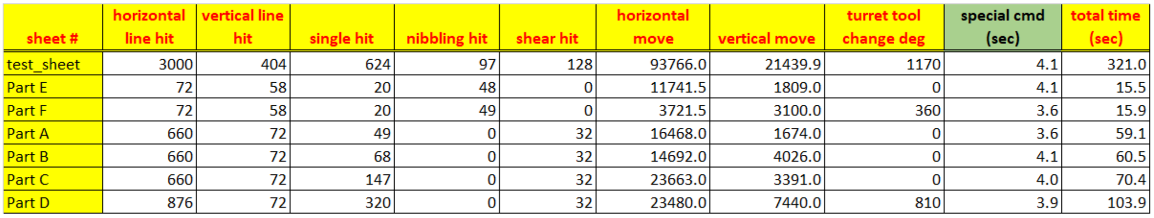
In practice, the time difference of the recorded machine run time and calculated regression model time can be furtherly distributed among all parts on the sheet blank. A common way to divide this time difference is based on the ratio of the regression model time of a part versus that of the sheet blank. The sum of machine run time of all parts will eventually equate to the machine run time of a sheet blank collected from the turret punch machine.

Table 2. 10-fold cross-validation training results



	10-fold training score	10-fold validation score
Linear Regression (default)	99%	99%
Ridge Regression (α = 0.1)	99%	99%
Ridge Regression (α = 1.0)	99%	99%
Ridge Regression (α = 10.0)	99%	99%
Lasso Regression (α = 0.1)	99%	99%
Lasso Regression (α = 0.01)	99%	99%
Lasso Regression (α = 0.001)	99%	99%

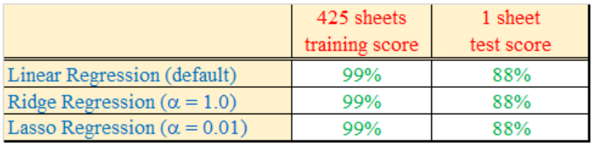
Table 3. Machine run time and punching operation data of test 1 sheet blank



sheet #	horizontal line hit	vertical line hit	single hit	nibbling hit	shear hit	horizontal move	vertical move	turret tool change deg	special cmd (sec)	total time (sec)
test_sheet	3000	404	624	97	128	93766.0	21439.9	1170	4.1	321.0
Part E	72	58	20	48	0	11741.5	1809.0	0	4.1	15.5
Part F	72	58	20	49	0	3721.5	3100.0	360	3.6	15.9
Part A	660	72	49	0	32	16468.0	1674.0	0	3.6	59.1
Part B	660	72	68	0	32	14692.0	4026.0	0	4.1	60.5
Part C	660	72	147	0	32	23663.0	3391.0	0	4.0	70.4
Part D	876	72	320	0	32	23480.0	7440.0	810	3.9	103.9

A part's manufacturing cost versus that of the entire sheet blank is proportional to its machine run time versus the sheet blank's machine run time when fabricated by turret punch machines. All expenses associated with manufacturing activities, including utilities, people's pecuniary compensation, equipment, finance, machine efficiency, etc. are directly or indirectly linked to the machine run time. The average machine run time for the same parts is therefore the basis for calculating the manufacturing cost of that part.

Table 4. Test scores of the test sheet vs. 425 training sheets



	425 sheets training score	1 sheet test score
Linear Regression (default)	99%	88%
Ridge Regression (α = 1.0)	99%	88%
Lasso Regression (α = 0.01)	99%	88%

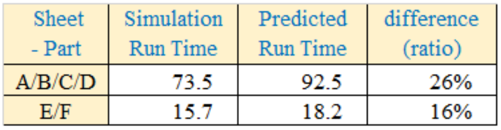
5.1 Numerical Results

Two example parts, one of 4 pieces and the other 2 pieces, are placed on the test sheet blank. The comparison of simulated machine run time for the = part of 4 pieces and the other of 2 pieces, and the predicted run time from the Ridge Regression model is listed in Table 5. The average difference is 22% for the 2 parts which is considered too high. More test sheets are being prepared and further investigation is in progress as of today.

5.2 Proposed Improvements

Simulation software is used to generate the machine run time of sheet blanks and punching operation data of parts on sheet blanks. It is preferable to collect the real machine run time from the turret punch machines if possible. On the other hand, the punching operation data are readily available in many industrial CAM software. It would be beneficial to gain access to machine shop(s) that run turret punch machines and form a joint research project. Some efforts were made without success so far. Continuing effort is necessary.

Table 5. Simulated machine run time versus predicted run time of Ridge Regression model



Sheet - Part	Simulation Run Time	Predicted Run Time	difference (ratio)
A/B/C/D	73.5	92.5	26%
E/F	15.7	18.2	16%

The velocity-time and acceleration-time relationships for the clamping bars' motors could be nonlinear when fabricating parts on sheet blanks. We may consider introducing higher order term(s), e.g. quadratic term, of the transverse and longitudinal moving distances to be the additional features in order to capture the nonlinearity aspect of motor behaviors.

The *nibbling on arc* and *nibbling on line* are regarded as the same type of operation in this research. It may be worthwhile to treat them as different features and see if the predictability of machine run time improves.

Only commonly used punching operations are considered in this research to predict the machine run time of sheet blanks. Judging by the research outcome, it indicates that the proposed Machine Learning method is a viable and consistent approach. It is advised to expand and include more punching tools, operations, sequences and commands that are executed by turret punch machines.

The 425 training sheet blanks contain randomly placed parts, and each part is generated using prearranged geometric shapes but with arbitrary parametric values. It appears that these parameter-driven parts are too coherent and similar, which may have caused the "overfitting" result using the data from 425 training sheet blanks. More studies can be done to diversify the parameter-driven, auto-generated test parts to enlarge the range and scope of punching tool assignments and punching sequence alternatives.

6. Conclusion

A new approach is proposed to distribute the machine run time of sheet blanks to the parts on sheet blanks for all turret punch machines. It adopts the Machine Learning method and incorporates the counts of punching operations, transverse and longitudinal moving distances of sheet blanks, tool changes, and machine's special commands as the modeling features. A punching simulation program is developed to provide a non-deterministic simulated machine run time for the sheet blanks. It also relates the punching tools, operations, routes and sequences to the parts on sheet blanks and retrieves applicable modeling features. 425 training sheet blanks are executed to train the Linear Regression, Ridge Regression, and Lasso Regression models. Using the 10-fold cross-validation procedure, the training result shows that all three models can achieve 99% accuracy scores. Furthermore, one test sheet blank is executed to verify the viability and consistency of the proposed approach.

Our results demonstrate that the proposed approach is viable and consistent to provide machine run time and subsequent manufacturing cost for parts of various geometries and sizes on sheet blanks. It can replace the subjective estimation or simple calculation currently used in industry, and can be automated with real-time data collection in the era of Industry 4.0. The approach can be expanded to incorporate laser cutting machines and develop a more comprehensive and versatile model for the laser-turret combination machines.

Acknowledgements

The authors would like to thank NSTC (National Science and Technology Council) of Taiwan for the financial support of this research.

References

Bargelis, A. and Rimasauskas, M., *Cost forecasting model for order-based sheet metalworking*, Journal of Mechanical Engineering Science, vol. 221, no. 1, pp. 55-65, 2007.

Caristan, C., *Laser Cutting Guide for Manufacturing*, illustrated edition, Society of Manufacturing Engineers, 2003.

Chang, Y., *Lung Nodule Classification via Deep Learning*, Master Degree Dissertation, National Taiwan University of Science and Technology, 2014.

Geiger, M., Knoblach, J. and Backes, F., *Cost Estimation for Large Scale Production of Sheet Metal Parts Using Artificial Neural Networks*, Production Engineering, vol. 2, pp. 81-84, 1998.

Géron, A., *Hands-On Machine Learning with Scikit-Learn, Keras, and Tensor Flow*, 3rd edition, O'Reilly Media, 2022.

Hsieh, J., Cheng J., Kuo Y., Gong, J. and Chen, S., *Practical Deep Learning*, 1st edition, Tsanghai, 2018.

Hwang, Y. and Yang, J., *Laser cutting time estimate for sheet metal parts of various geometries by Machine Learning approach*, 13th International Conference on Industrial Engineering and Operations Management, March 14, 2023.

Lantz, B., *Machine Learning with R: Learning data cleansing to modeling from the tidyverse to neural networks and working with big data,* 4th edition, Packt Publishing, 2023.

Lasso vs Ridge vs Elastic Net | ML, Available: https://www.geeksforgeeks.org/lasso-vs-ridge-vs-elastic-net-ml/

Liam, C., *Estimating Sheet Metal Fabrication Costs*, ETM Manufacturing, Available: http://www.etmmfg.com

Matel, M., Vahdatikhaki, F., Hosseinyalamdary, S., Evers, T. and Voordijk, H., *An artificial neural network approach for cost estimation of engineering services*, International Journal of Construction Management, vol. 22, no. 7, pp. 1274-1287, 2022

Müller, A. and Guido, S., *Introduction to Machine Learning with Python*, 1st edition, O'Reilly Media, 2016.

Powell, J., *CO2 Laser Cutting*, 2nd edition, Springer, 1993.

Scikit-learn, Available: http://www.scikit-learn.org/stable/modules/linear_model.html

Smith, A. and Mason, A., *Cost Estimation Predictive Modeling: Regression versus Neural Network*, The Engineering Economist, vol. 42, no. 2, pp. 137-161, 1997.
Available: https://doi.org/10.1080/00137919708903174, 2010.

Solidworks Web Help, Available: https://www.dallan.com/en/news/four-steps-to-calculate-the-manufacturing-cost-of-sheet-metal-products/

Soroush, M., Baldea, M. and Edgar, T., *Smart Manufacturing: Concepts and Methods*, 1st edition, Elsevier, 2020.

Steen, W., Mazumder, J. and Watkins, K., *Laser Material Processing*, 4th edition, Springer, 2010.

Taiwan 3AXLE Technology, Inc., Available: https://www.3axle.com

Wang, X. and Bi, Z., *Computer Aided Design and Manufacturing*, 1st edition, Wiley and Sons Ltd., 2020.

Biographies

Yearn-Tzuo (Andrew) Hwang is an Assistant Professor in Design and Manufacturing in the Department of Mechanical Engineering at National Yang Ming Chiao Tung University, Hsinchu, Taiwan. He earned his Ph.D. in Mechanical Engineering from the University of California, Los Angeles, and M.S. in Mechanical Engineering from the University of Texas at Austin, in USA, and B.S. in Mechanical Engineering from the National Chiao Tung University in Taiwan. He founded a Los Angeles based consulting company which was acquired by Amada America Inc., and where he then worked in the R&D division. He holds several patents in both the USA and Japan. His research interests include Machine Learning and Artificial Intelligence in manufacturing, Industry 4.0, Automation in sheet metal fabrication, Design Optimization, and Work Order Scheduling.

Ting-Yi Zhang is a graduate student in Design and Manufacturing in the Department of Mechanical Engineering at National Yang Ming Chiao Tung University, Hsinchu, Taiwan. He earned his B.S. in Mechanical Engineering from National Taiwan University of Science and Technology, and had worked as a design engineer in Taiwan industry for 4 years before entering the graduate program.

Hung-Chun Hsu is currently a senior Mechanical Engineering student, and will proceed to the graduate program in Design and Manufacturing in the Department of Mechanical Engineering at National Yang Ming Chiao Tung University, Hsinchu, Taiwan.