

Detección de Fallos en inyección de Plástico.

**Nombres: Shelsea Stephanie Melgara Mendoza
Manuel Alejandro Juarez Juarez
Josue Noel López Guevara**

Fecha: [18/07/2022]

Tabla de contenido

Introducción	3
Desarrollo	4
Conclusión	14

Información del Proyecto

Empresa / Organización	Plasticosa S. A
Proyecto	Detección de Fallos
Fecha de preparación	15 Julio 202
Cliente	PLASTICOSA S.A

Introducción

El Presente trabajo describe un informe técnico del diseño de un Sistema para la detección de fallas en el proceso de producción de la empresa PLASTICOSA S.A en el departamento de Inyeccion de Plastico.

La inyección de plástico es un proceso semicontinuo que consiste en inyectar un polímero en estado fundido (o ahulado) en un molde cerrado a presión y frío, a través de un orificio pequeño llamado compuerta.

Dentro de las fases del proceso de inyección se realizan 4 de manera sincronizada:

- Cierre del molde. Se suministra el polímero en la unidad de inyección y se cierra el molde por presión.
- Inyección:
 - Se inyecta el plástico a través de una boquilla dentro del molde.
 - Para lograr que la pieza tenga las dimensiones deseadas, se mantiene la presión.
- Enfriamiento. La pieza se mantiene en el molde hasta enfriarse.
- Apertura y expulsión de la pieza: El molde se abre y se libera la pieza.

Durante el proceso, las variables más importantes son:

Temperatura del material

Es primordial que se alcance la temperatura idónea para lograr una fusión correcta del polímero, y mantenerla durante el tiempo necesario antes de enfriarse.

Presión de inyección

A través de la boquilla, la máquina debe inyectar el material a la presión justa, de tal manera que el llenado del molde se realice adecuadamente.

Velocidad de inyección

En el momento en que se inyecta el material dentro de la cavidad del molde, comienza a enfriarse. Por eso es muy importante que el proceso se realice rápidamente.

Presión y tiempo de mantenimiento

En la fase de mantenimiento, tanto la presión ejercida como el tiempo son claves para que el resultado sea satisfactorio.

Tiempo y velocidad de enfriamiento

Ya hemos visto que en cuanto el material es inyectado comienza a enfriarse. Se requiere de un tiempo variable para conseguir que la pieza sea estable y pueda expulsarse; este tiempo dependerá de la pieza y del material empleado.

Dentro de los principales problemas en la inyección de plástico tenemos varias como:

1. Contracción Volumétrica causada por la escasez de materia prima o por parámetros de temperatura incorrectos en la fabricación de la pieza. Debido a ello el material se contrae en el centro y se produce el fenómeno conocido como ‘halado’.
2. Líneas de Soldadura Ejerciendo control sobre la ubicación del punto de inyección y los parámetros del proceso se puede reducir el debilitamiento.
3. El encogimiento de la cristalización determina cuánto encogerá el polímero, que por naturaleza tiende a solidificarse. Si el enfriamiento del material es súbito, se producirá un bajo grado de cristalización dentro del molde, y será un grado alto si la tasa de enfriamiento es lenta.

De estos factores que mencionamos anteriormente, seleccionamos la problemática número 1 para implementar el sistema de detección de fallos en el proceso de producción.

Desarrollo

Existen numerosos estudios sobre cómo controlar el proceso de inyección. La mayoría de ellos se basan en la sensorización de las fases críticas del proceso, colocar sensores de presión y de temperatura en el interior del molde permite caracterizar las curvas que permiten identificar la calidad de la pieza final antes de abrir el molde, disminuyendo la inspección visual como medio de control de calidad.

La metodología desarrollada para resolver el problema anteriormente descrito, sigue las fases básicas de los sistemas basados en el conocimiento: (1) adquisición de conocimiento implícito y explícito para el aprendizaje y (2) extracción del conocimiento para generar soluciones.

Razonamiento Basado en Casos/Conocimientos (CBR)

El razonamiento basado en casos reutiliza información de situaciones pasadas. Al igual que en muchos algoritmos de aprendizaje automático, un caso está formado por varios atributos independientes que dan una descripción del problema y una solución para el caso. Así, la contribución de un atributo para una clase se puede determinar de forma independiente de los otros atributos. Este requisito simplifica los algoritmos de aprendizaje y genera un rendimiento robusto.

El CBR es un proceso que consta de cuatro pasos:

1. RECUPERAR el caso o casos más similares.
2. REUTILIZAR ese caso para resolver el problema.
3. REVISAR la solución propuesta.
4. GUARDAR las partes de esta experiencia que se consideren útiles para resolver futuros problemas.

La principal ventaja de CBR es la gran capacidad de adaptación a nuevos casos ya que se basa en extrapolar la solución. Aunque también existen algunos inconvenientes:

- Precisión: Los sistemas CBR necesitan una gran cantidad de datos para ser precisos. La curva de aprendizaje de este tipo de sistemas tarda en ser estable.
- Puesta en marcha: Tal y como se ha comentado anteriormente, la necesidad de tener una base de datos significativa hace que el tiempo de puesta en marcha sea algo mayor que otras soluciones.

Para poder realizar el software se hizo uso de los siguientes programas:

☐ PYTHON y las librerías :

→ Pandas: es un popular paquete de Python para la ciencia de datos y Machine Learning, las razones son muchas y es que ofrece estructuras de datos poderosas, expresivas y flexibles que facilitan la manipulación y

análisis de datos. Entre las estructuras más utilizadas se encuentra el DataFrame.

- NumPy: Es una librería de Python especializada en el cálculo numérico y el análisis de datos, especialmente para un gran volumen de datos.
 - Incorpora una nueva clase de objetos llamados arrays que permite representar colecciones de datos de un mismo tipo en varias dimensiones, y funciones muy eficientes para su manipulación.
 - Scikit-learn es uno de los open-source y bibliotecas de aprendizaje automático más populares en Python. La biblioteca scikit-learn contiene muchas herramientas eficientes para aprendizaje automático y modelado estadístico, incluyendo clasificación, regresión, agrupación, y reducción de dimensionalidad.
- ☐ Excel: hoja de cálculo que nos permite manipular datos numéricos y de texto, analizar información, generar reportes.
- ☐ Interfaz Gráfica:
- JAVASCRIPT: Es el lenguaje de programación encargado de dotar de mayor interactividad y dinamismo a las páginas web. Con este lenguaje de programación del lado del cliente (no en el servidor) podemos crear efectos y animaciones sin ninguna interacción, o respondiendo a eventos causados por el propio usuario tales como botones pulsados y modificaciones del DOM (document object model).
 - PYTHON: Python es un lenguaje de programación de alto nivel, orientado a objetos, con una semántica dinámica integrada, principalmente para el desarrollo web y de aplicaciones informáticas.
 - PHP: es un lenguaje de programación que permite el desarrollo web o aplicaciones web dinámicas, el cual es apto para incrustar el lenguaje HTML,

Funcionamiento del Programa:

Lo primero que se hizo fue buscar información acerca de los parámetros normales para las máquinas de inyección de tinta, acá encontramos 4 variables con los diferentes rangos. Teniendo estos datos se procedió a copiarlos en archivos de excel para cada una de las variables.

MAT	Temp_rholve	Temp_secado	Tiempo_secado	P_inyeccion	P_mantenimient
1,80,80,239	1300,50,240	80,1,0,0,0,0,0,0			
1,77,78,240	1300,50,240	80,1,2.064307458	0,0,0,0,0,0		
1,79,79,234	1300,50,240	80,1,6.92994854	0,0,0,0,0,0		
1,78,76,237	1300,50,240	80,1,8.444770602	0,0,0,0,0,0		
1,75,73,233	1300,50,240	80,1,13.69455316	0,0,0,0,0,0		
1,72,75,235	1300,50,240	80,1,14.98106472	0,0,0,0,0,0		
1,71,71,236	1300,50,240	80,1,16.92895195	0,0,0,0,0,0		
1,66,74,238	1300,50,240	80,1,19.03584112	0,0,0,0,0,0		
1,70,70,231	1300,50,240	80,1,21.57579546	0,0,0,0,0,0		
1,67,77,229	1300,50,240	80,1,26.84777488	0,0,0,0,0,0		
1,69,67,232	1300,50,240	80,1,30.20810683	0,0,0,0,0,0		
1,64,68,230	1300,50,240	80,1,30.8993803	0,0,0,0,0,0		
1,60,65,228	1300,50,240	80,1,32.18589186	0,0,0,0,0,0		
1,73,69,227	1300,50,240	80,1,37.43431543	0,0,0,0,0,0		
1,61,72,222	1300,50,240	80,1,38.25786403	0,0,0,0,0,0		
1,59,66,226	1300,50,240	80,1,46.46662318	0,0,0,0,0,0		
1,58,63,223	1300,50,240	80,1,51.18413423	0,0,0,0,0,0		
1,51,64,225	1300,50,240	80,1,60.18971516	0,0,0,0,0,0		
1,45,62,221	1300,50,240	80,1,63.05401537	0,0,0,0,0,0		

Luego, se procedió a recopilar los datos almacenados en los últimos meses de los parámetros de las máquinas durante el proceso de inyección de plástico, haciendo uso del mismo programa, para seleccionar con exactitud los parámetros más significativos que usamos para detectar los fallos.

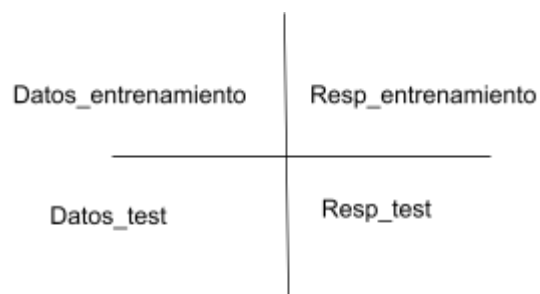
									P1	
80	80	239					0	0	0	0
77	78	240					0	4,109589	2,083333	2,064307
79	79	234					3,571429	10,27397	6,944444	6,929949
78	76	237					5,357143	11,64384	8,333333	8,444771
75	73	233					8,035714	20,54795	12,5	13,69455
72	75	235					9,821429	21,23288	13,88889	14,98106
71	71	236					14,28571	21,91781	14,58333	16,92895
66	74	238					17,85714	23,9726	15,27778	19,03584
70	70	231					17,85714	25,34247	21,52778	21,5758
67	77	229					26,78571	29,45205	24,30556	26,84777
69	67	232					28,57143	32,19178	29,86111	30,20811
64	68	230					28,57143	32,87671	31,25	30,89938
60	65	228					30,35714	33,56164	32,63889	32,18589
73	69	227					35,71429	35,61644	40,97222	37,43432
61	72	222					37,5	36,30137	40,97222	38,25786
59	66	226					51,78571	41,78082	45,83333	46,46662
58	63	223					53,57143	51,36986	48,61111	51,18413
51	64	225					72,32143	56,16438	52,08333	60,18972
45	62	221					76,78571	58,90411	53,47222	63,05402
47	60	224					76,78571	60,9589	57,63889	65,12784
48	61	220					77,67857	63,0137	61,11111	67,26779
46	59	219					77,67857	67,80822	61,80556	69,09745
44	57	217					78,57143	72,60274	64,58333	71,91917
49	55	218					81,25	73,28767	73,61111	76,04959
38	58	214					88,39286	73,9726	77,08333	79,81626
41	56	215					91,07143	74,65753	78,47222	81,4004
39	53	216					91,96429	83,56164	81,25	85,59198
76	60	213					95,71429	85,21644	87,63889	87,07076

El siguiente código que se explicara es el que se utiliza para entrenar la IA, es el mismo procedimiento para los 4 modelos, cambiando solamente los diferentes valores y asignando diferentes nombres a las variables; con él, también accedemos a los archivos de excel creados (datas), para poder hacer uso de su información, que nos permitirán calcular las fallas detectadas.

Dentro del código se hace uso de la regresión lineal, ¿qué es esto?

La regresión lineal se utiliza para predecir el valor de una variable según el valor de otra. La variable que desea predecir se denomina variable dependiente. La variable que se está utilizando para predecir el valor de la otra variable se denomina variable independiente. Esta forma de análisis estima los coeficientes de la ecuación lineal, involucrando una o más variables independientes que mejor predicen el valor de la variable dependiente. Los modelos de regresión lineal son relativamente sencillos y proporcionan una fórmula matemática fácil de interpretar que puede generar predicciones.

Usamos regresiones lineales porque como tenemos valores directamente proporcionales, y son pocos valores.



Dentro de este esquema se puede entender mejor para que sirve este código a la IA se le pasan los Datos y las respuestas cuando se hace esto ella se entrena, y es ahí donde después del entrenamiento se le manda los datos_test para que en base a eso genere una respuesta y comprueba el error del modelo.


```
import pandas as pd

datosPA66= pd.read_csv('./PA66/PA66.csv')
DatosPA66_entrenamiento=datosPA66.sample(frac=0.8, random_state=0)
DatosPA66_test=datosPA66.drop(DatosPA66_entrenamiento.index)
ResPA66P_entrenamiento= DatosPA66_entrenamiento.drop(labels=['MAT','Temp_molde','Temp_secado','Tiempo_secado','P_inyeccion','P_mantenimiento','Temp_barril','Temp_alimentacion','V_inyeccion'], axis= 1)
Resp_test=DatosPA66_test.drop(labels=['MAT','Temp_molde','Temp_secado','Tiempo_secado','P_inyeccion','P_mantenimiento','Temp_barril','Temp_alimentacion','V_inyeccion'],axis=1)
DatosPA66_entrenamiento=DatosPA66_entrenamiento.drop(labels=['P1','P2','P3','P4','P5','P6','P7'], axis= 1)
DatosPA66_test=DatosPA66_test.drop(labels=['P1','P2','P3','P4','P5','P6','P7'], axis= 1)
import sklearn.linear_model as sk
modeloPA66=sk.LinearRegression()
modeloPA66.fit(DatosPA66_entrenamiento,ResPA66P_entrenamiento)
x=modeloPA66.predict(DatosPA66_test)
```

Pero además de eso, quedan 2 data sheet donde tengo el 80 % de información para entrenar a la IA y donde tengo el 20% lo voy a usar para comprobar el porcentaje de error.

En rest test se van a guardar las respuestas de los valores es decir las respuestas que nos generará la IA que es lo que queremos que ella prediga. Se hace uso del Drop lo que hace es eliminar y guardar en otra variable.

El sklearn.linear_model es una librería que tiene ya varios modelos de Matching learning ya hechos modelo con el que trabaja la IA para entrenarse.

En esta parte se guardan las predicciones de los datos de prueba para probar los datos de error, se hace uso de una función matemática para poder calcular el error. Para una explicación más sencilla acá es donde se comparan los datos generados con los originales y devuelve un porcentaje de error que es del 10%.

```
x=modeloPA66.predict(DatosPA66_test)
import numpy as np
from sklearn.metrics import mean_squared_error
error=np.sqrt(mean_squared_error(Resp_test,x))
error
```

CVS es un array bidimensional donde se pasa todo a la IA. Es importante que la IA nos regrese un array bidimensional en números en decimales.

Se hace un data frame para poder guardar los valores en una lista bidimensional y se convierte en un arreglo.

```
prediccion=pd.DataFrame(np.array([[1,80,80,240,1300,50,240,80,1]]),columns=['MAT','Temp_molde','Temp_secado','Tiempo_secado','P_inyeccion','P_mantenimiento','Temp_barril','Temp_alimentacion','V_inyeccion'])
x=modeloPA66.predict(prediccion)
resultado=[]
for i in x:
    for j in i:
        if j <0:
            j=0
            resultado.append(j)
        elif j>=100:
            j=100
            resultado.append(j)
        else:
            resultado.append(j)

prediccion=pd.DataFrame(np.array([resultado]),columns=['P1','P2','P3','P4','P5','P6','P7'])
prediccion
```

Interfaz Gráfica



Página de inicio del sistema es lo que se va a poder observar los operadores encargados de vigilar el comportamiento de los sistemas.

Para realizar esto, se hace uso de Div, labels, head, y css para dar animaciones en las fotos mostradas.

SISTEMA BASADO EN CONOCIMIENTO SOBRE PRODUCCIÓN DE ARTICULOS DE PLÁSTICO

[Inicio](#) [Detección de errores](#) [Alerta y Soluciones](#)

Descripción

Material	Temperatura de molde (C°)	Presion de inyección (bar)	Presion de mantenimiento (%)
<input type="text" value="Elegir..."/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Temperatura de secado (C°)	Tiempo de secado (minutos)	Temperatura de barril (C°)	Temperatura de alimentación (C°)
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Velocidad de inyección (m/s)			
<input type="text"/>			
<input type="button" value="Aceptar"/>			

Este apartado de la interfaz es para declaraciones de los valores, es importante rellenar todos los campos porque sino, no se podrá obtener una respuesta de la aplicación, dado que se ha utilizado el atributo onsubmit para poder verificar que todos los campos estén llenos. Cuando se han llenado todos los campos agarra esa información y los ingresa en variables python para que se puedan procesar por python y la IA.

SISTEMA BASADO EN CONOCIMIENTO SOBRE PRODUCCIÓN DE ARTICULOS DE PLÁSTICO

Inicio Detección de errores Alerta y Soluciones

Apartado sobre detección de errores

Enchuecamiento	Flash	Lineas de flujo	Carbonizaciones	Piel de naranja	Parte incompleta	Parte con rebabas
100%	43.7501%	32.3805%	0%	0%	0%	100%

Enchuecamiento

La posible solución es: Incrementar el tiempo de enfriamiento dentro del molde

Flash

Debajo del 50% de probabilidad

Lineas de flujo

Debajo del 50% de probabilidad

Carbonizaciones

Debajo del 50% de probabilidad

Piel de naranja

Debajo del 50% de probabilidad

Parte incompleta

Debajo del 50% de probabilidad

Parte con rebabas

La posible solución es: Disminuir la temperatura del barril, aumentar la velocidad de inyección y reducir la temperatura del molde

Apartado de carga, una vez enviados los datos, python los procesa y en el momento que termina mete los valores dentro de un data frame; para mostrarnos cuales son las soluciones para evitar un daño.

Conclusión

Una vez realizada la programación para dar entrenamiento a la IA, y pasándole los valores correctos para su funcionamiento y luego los valores que no estarán dentro de su parámetro normal; se diseñó la aplicación que será manejada por los técnicos encargados de supervisar las máquinas inyectoras.

Trabajando de la siguiente manera; una vez introducidos los valores de entrada dentro de la IA dependiendo de los valores ella nos dará una salida. Esto se debe a que la IA ya está entrenada, ella ya conoce los valores que deberían ser los precisos y cuando se da la ocasión de que encuentra el porcentaje o la posibilidad que se pueda un error ella dará una respuesta basándose a lo que sabe.