The Development and Validation of a Mathematical Model of Output in A Manufacturing System

Ho Kok Hoe
PhD Graduate, Department of Industrial Engineering
Faculty Engineering and Green Technology
Universiti Tunku Abdul Rahman
Kampar, Perak, Malaysia
hokokhoe1980@gmail.com

Joshua Prakash
Assistant Professor, Department of Industrial Engineering
Faculty of Engineering and Green Technology
Universiti Tunku Abdul Rahman
Kampar, Perak, Malaysia
joshuaj@utar.edu.my

Ong Kok Seng
Professor, Department of Industrial Engineering
Faculty of Engineering and Green Technology
Universiti Tunku Abdul Rahman
Kampar, Perak, Malaysia
skong@utar.edu.my

Abstract

The construction of mathematical models includes a variety of strategies for closing the gap between the model and real-time data. This study uses a data mining technique to expand mathematical model construction to include simulation optimization. At the same time, this procedure is used to verify the mathematical model. In addition to dynamic property validation, the paper suggests mathematical programming as a static property validation method. The validated mathematical model is compared and examined with real-time data, and the results show that the mathematical model applies to real-time systems. The mathematical model has a low error margin estimate and can be used to forecast. The paper's main contribution is to investigate simulation optimization, mathematical programming, and mathematical models to handle the increasing complexity of variables to improve prediction capabilities and lay a solid foundation for tackling factory planning issues.

Keywords
Simulation model validation, simulation optimization, mathematical programming validation, mathematical model validation, output's mathematical model

1. Introduction
Much research has addressed the mathematical model in manufacturing systems (Abazari et al., 2012; Bagheri and Bashiri, 2014; Leitão and Restivo, 2008; X. Li et al., 2010; Shen and Yao, 2015). The issues of mathematical model development relate to integrating multiple variables and various constraints into one mathematical model and model effectiveness under a real-time system's unpredictable and dynamic nature. To find a solution to the issue, many researchers studied various methods to develop a mathematical model that narrows the gap between the model and the real-time data. The existing methods are data mining (Arif et al., 2013; Chincholkar and Herrmann, 2008; Kim

and Lee, 1997; Luo et al., 2015) and model enhancement (Al-Zuheri et al., 2012; Georgiadis and Michaloudis, 2012; Jacomino, 2014; Kim & Morrison, 2014; Usubamatov et al., 2015; Wu, 2014).

In (Arif et al., 2013), the authors studied a quality prediction model to deal with complex variable relationships. The authors used Cascade Quality Prediction Method and SP-ID3 and SP-PCA+ID3. The finding showed that the developed model performed better than other models with relatively high accuracy. Luo et al. (2015) analyzed a multiple regression model to predict equipment degradation and maintenance schedule optimization in manufacturing using four data mining techniques. The data mining techniques were artificial neural network, genetic algorithm, restricted Boltzmann machine and deep belief network. The effectiveness of the model is tested with industrial case achieving 74.1% testing accuracy. Model enhancement from previous literature is another mathematical model development method alternative. Model enhancement in research works is driven by analytical methods to derive, expand and improve existing mathematical models. Kim and Morrison (2014) revised literature models to develop a mathematical model on throughput using the Markov Chain concept. The results are acceptable through the comparison of the model and simulation data. Usubamatov et al. (2015) referred to previous research models to enhance the proposed mathematical model for the productivity rate of manufacturing systems with different failure rates. The finding indicates the mathematical model accuracy is close to actual production values. Wu (2014) revised the mathematical model from past literature in batch queuing models. The author discussed the validation result between the mathematical model and simulation, which is acceptable under certain conditions. Future research on the inclusion of batching size is proposed. Numerous validation methods of mathematical models are discussed in the literature (Jagdev et al., 1995; Kleijnen, 1995; Sargent, 2011). On validation method is simulation comparison with mathematical model (Almehdawe and Jewkes, 2013; Cao et al., 2012; Chincholkar and Herrmann, 2008; Zuheri et al., 2014). In general, the accuracy of the mathematical model is compared and validated with simulation data to evaluate the model's effectiveness (Almehdawe and Jewkes, 2013; Cao et al., 2012; Chincholkar and Herrmann, 2008; Zuheri et al., 2014). There are also works on real-time data validation methods (Defersha and Chen, 2006; Eickemeyer et al., 2014; Georgiadis and Michaloudis, 2012; W. Li, 2007; J.S. Lin, 2012). The two less-seen validation methods are the mathematical proof technique (Chang et al., 2012; Kim and Morrison, 2014) and the comparison with previous literature studies (Abazari et al., 2012; Brown, 2014; X. Li et al., 2010; Mahdavi et al., 2012).

The primary motivation behind this study is the challenge posed by the effectiveness of a mathematical model in comparing with real-time data. The first contribution is introducing a simulation optimization tool to support current mathematical model development in an attempt to draw nearer between theoretical and real-time data. The second contribution is introducing an additional validation method using mathematical programming and simulation to support the mathematical model's prediction when tested with real-time data. The paper is organized into four sections. Section 1 introduces the background of the topic and the paper's objective. Section 2 describes the methodology used in this paper. The findings of the study are discussed in Section 3. The conclusion of the study is drawn in Section 4.

2. Methodology

The simulation model is developed based on an assembly semiconductor manufacturing system. The simulation model is built using Pro-Model. The input variables from real-time data are inserted into the model to collect the data on the response variable. The simulation model is validated using data from a real-time system. A statistical t-test was used to validate the simulation model with real-time data on each process. If the validation fails, the simulation model is revised. The simulation model is incorporated into the simulation optimization software, Sim Runner to optimise the response variable. During the simulation optimization, the study defines a suitable replication. Once suitable replication is selected, the simulation optimization performs each experimental run to find the optimum output. The simulation optimization employs genetic and evolutionary algorithms to optimize the simulated system (Chau et al., 2014; Harrell and Price, 2003; Mason et al., 2008). The mathematical model is developed on a spreadsheet using simulation optimization experimental data. Mathematical programming is developed in the spreadsheet to collect the output data using the defined objective function and its constraints. The result of the mathematical model and mathematical programming is analysed using statistical t-test analysis. The mathematical model and programming will require further revision if the validation results are not statistically proven. Once the established mathematical model is validated, the model is further analysed with real-time data. The error acceptance between the mathematical model and real-time data enables it to be applied in manufacturing performance monitoring. Section 2.1, 2.2, 2.3, 2.4, 2.5 and 2.6 illustrates the data analysis and findings based on Figure 1 using the IDEF0 methodology. Section 2.1 lists the annotation used in the study. Section 2.2 to Section 2.7 shows the mathematical model development procedure and its integration with simulation optimization and mathematical programming.

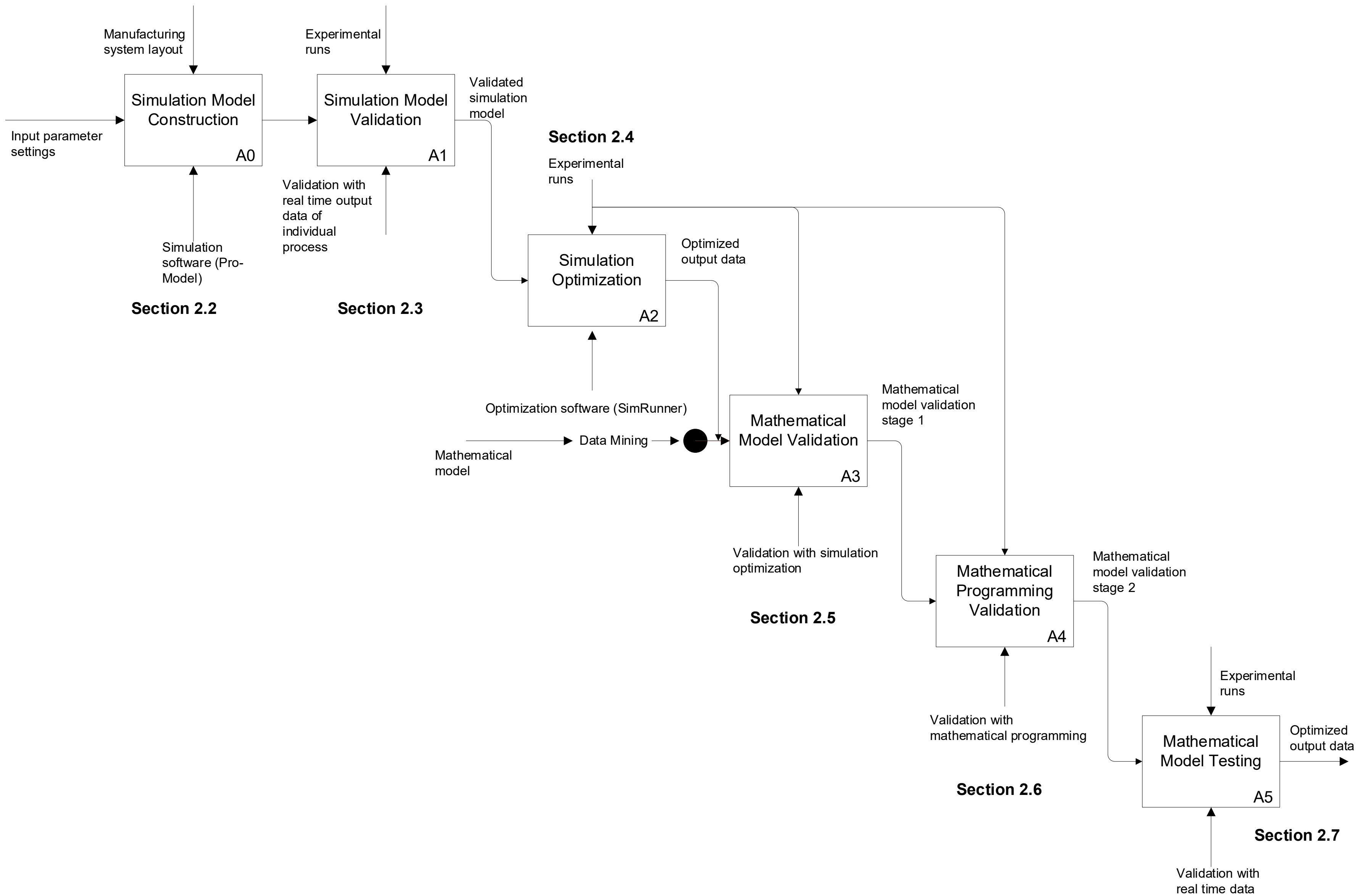

Figure 1. IDEF0 methodology from section 2.2 to section 2.7 for the mathematical model development and its concurrent validation framework

2.1 List of Annotations

Some annotations are pre-defined below:

CT_{DA}	cycle time per unit (Die Attach) / unit = seconds
CT_{WB}	cycle time per unit (Wire Bond) / unit = seconds
CT_{PC}	cycle time per unit (Pre-Cap Inspection) / unit = seconds
CT_{OC}	cycle time per batch (Oven Cure) / unit = seconds
$MTTR_{DA}$	mean time to repair at Die Attach
$MTBF_{DA}$	mean time between failure at Die Attach
$MTTR_{WB}$	mean time to repair at Wire Bond
$MTBF_{WB}$	mean time between failure at Wire Bond
ST_{DA}	setup time duration (Die Attach) / unit = seconds
ST_{WB}	setup time duration (Wire Bond) / unit = seconds
A_{DA}	availability at Die Attach
A_{WB}	availability at Wire Bond
BQ	batch quantity
T_b	average completion time per batch, b
T_{max}	maximum number of days used in the study
T_D	total time per day
ST_{DA}	setup time at Die Attach process
ST_{WB}	setup time at Wire Bond process
M	number of machines available at process (based on least capacity in the system)
O_S	optimized output per day from simulation optimization method
O_M	optimized output per day from predicted mathematical model

2.2 Simulation Model Construction

We examined a case study from assembly semiconductor manufacturing (Figure 2). Figure 3 shows using semiconductor manufacturing the simulation model of assembly Pro-Model. The configuration of the process is as follows: 3 die attach machines, continuous available oven cure machine, 9 wire bond machines, and 3 pre-cap inspection machines.

Die Attach → Oven Cure → Wire Bond → Pre-cap Inspection

Figure 2. Semiconductor manufacturing process flow

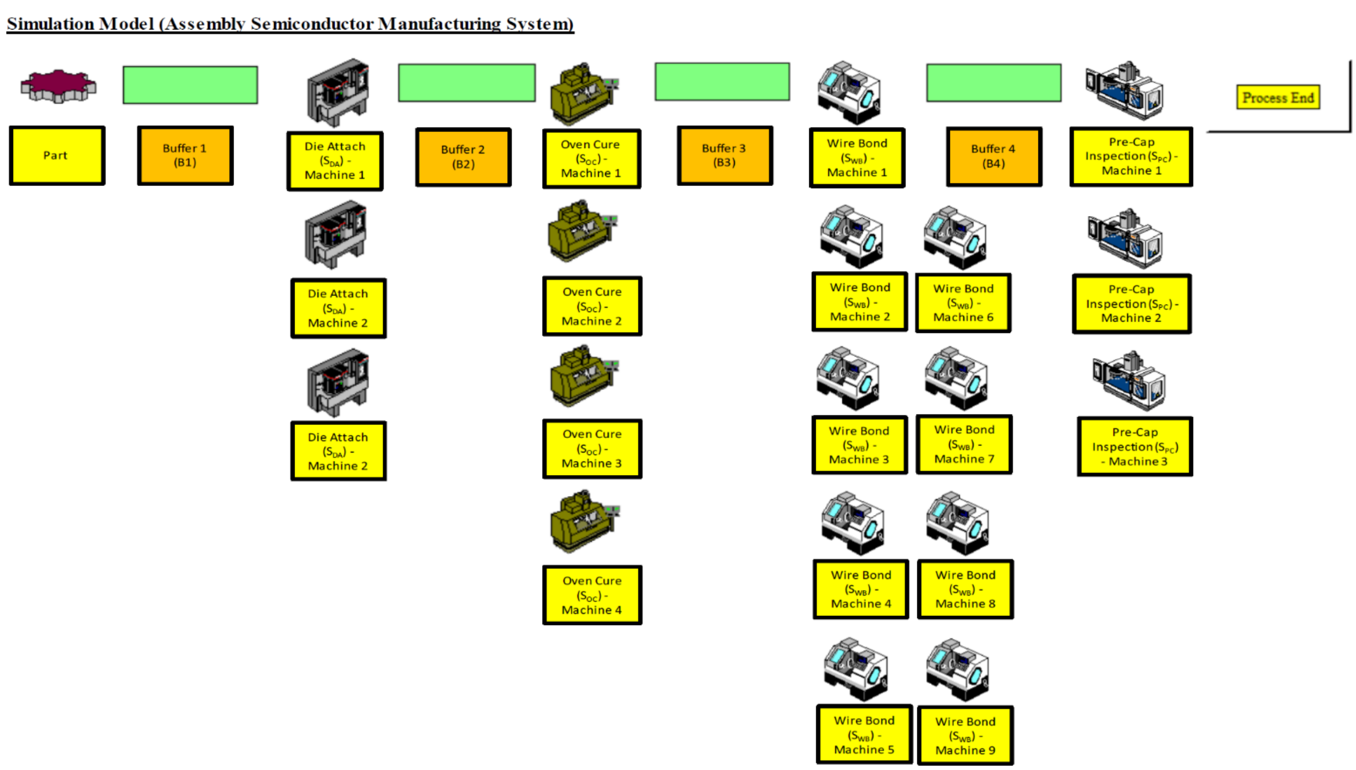

Figure 3. Simulation model of assembly semiconductor manufacturing using Pro-Model

2.3 Simulation Model Validation

Machine availability is defined below:

Machine availability, A

$$= \frac{MTBF}{MTBF+MTTR} \tag{1}$$

Table 1 represents the raw data from a real-time system which serves as input variables value. Each input variable's low and high settings are based on a 95% confidence interval of the distribution of input variables except BQ and CT_{OC}. Table 2 shows the raw data between simulation and real-time output for two individual processes.

Table 1. Input variable data for the simulation model

Input variable	Data		
	Low setting	Middle setting	High setting
CT_{DA} (seconds)	2.8072	-	2.9460
CT_{WB} (seconds)	6.0902	-	6.4609
CT_{PC} (seconds)	0.9882	-	1.0498
$MTTR_{DA}$	6883	-	6887
$MTTR_{WB}$	3619	-	2724
$MTBF_{DA}$	50006	-	109826
$MTBF_{WB}$	53115	-	84831
ST_{DA} (seconds)	2957	-	6063
ST_{WB} (seconds)	1324	-	2714
A_{DA}	0.8790	-	0.9410
A_{WB}	0.9362	-	0.9680
BQ (unit)	2200	3080	11264
CT_{OC} (seconds)	7200 (fixed)		

Note: Input variables values are randomly selected from real-time data

Table 2. Raw data between simulation and real-time for both die attach and pre-cap inspection on output

Low Setting (DA)		High Setting (DA)		Low Setting (PCI)		High Setting (PCI)	
Simulation	Real-Time	Simulation	Real-Time	Simulation	Real-Time	Simulation	Real-Time
78848	89458	78848	82709	40480	80379	38280	80379
95392	89289	87912	81987	81928	83724	78848	83724
81928	89624	84128	82899	92312	78552	78848	78552
92312	87127	87912	78949	84128	83793	90112	83793
93192	86084	78848	78114	95392	76586	79728	76586
84128	95195	90112	87949	90112	68069	92312	68069
90112	87054	78848	79861	84128	91621	87912	91621
95392	84264	90112	78076	93192	88828	78848	88828
81928	104389	81928	97405	92312	95241	90112	95241
92312	100457	90112	91937	81928	92379	78848	92379
95392	90168	76648	83468	95392	91345	93192	91345
78848	91749	92312	80734	90112	84828	78848	84828
93192	94255	76648	83696	84128	86172	90112	86172
95392	96612	90112	89620	93192	86483	78848	86483
78848	98199	81928	87190	81048	81552	93192	81552
95392	87354	90112	78835	93192	89310	76648	89310
84128	99419	78848	89316	95392	85138	92312	85138
90112	85780	93192	77658	81048	77414	76648	77414
95392	95096	78848	87873	93192	76207	90112	76207
81928	78492	87912	71772	92312	83345	81928	83345
92312	89378	81048	82177	81928	90172	90112	90172
93192	90409	87912	82177	93192	78552	78848	78552
84128	89238	90112	80696	95392	84586	90112	84586
90112	88015	81928	79367	81048	88966	78848	88966
95392	82134	78848	75190	93192	92897	87912	92897
81928	89377	90112	77696	92312	91966	84128	91966
92312	85252	90112	79367	81928	87931	87912	87931
93192	82366	78848	75987	95392	94414	78848	94414
84128	98009	90992	89278	78848	90759	93192	90759
90112	89979	81048	82177	95392	77621	78848	77621

Note: DA = Die Attach; PCI = Pre-cap Inspection

Table 3 illustrates the validation result of the die attach and pre-cap inspection from Table 2. In this paper, α = 0.05. The p-value for low and high input variables setting for die attach is 0.0961 and 0.3490, respectively. The p-value for low and high input variables setting for pre-cap inspection is 0.3462 and 0.3426, respectively. Since the p-value is larger than α, it accepts the null hypothesis indicating no significant difference between the simulation model and real-time data.

Table 3. Validation result of die attach and pre-cap inspection from table 2 using statistical t-test analysis

Die Attach		Pre-Cap Inspection	
Low Input Variables Setting	High Input Variables Setting	Low Input Variables Setting	High Input Variables Setting
P value = 0.0961	P value = 0.3490	P value = 0.3462	P value = 0.3426

Once the simulation model is validated, the simulation model is incorporated into simulation optimization software, SimRunner, to perform experimental optimization runs.

2.4 Simulation Optimization

Five replications are used to perform simulation optimization for 25 random experiments. The study has tested 1, 5, 10, and 100 replications. There is no significant different comparison among these replications as the model is deterministic.

The validated simulation model is integrated into the simulation optimization software, SimRunner. The input variables' low and high settings are inserted into SimRunner with an objective function to maximize the output. The SimRunner performs optimization for 25 random experiment runs, as shown in Table 4. The values of each variable in simulation optimization constitute the boundary between low and high settings, which are defined in Table 1. The SimRunner searches the maximum output for 25 random experiments within the boundary of these settings of input combination. Table 4 shows the optimized output from 25 random experiments with Sim Runner simulation software.

Table 4. Optimized output from 25 random experiments with simulation optimization

Run	CT_{DA}	CT_{WB}	CT_{DC}	$MTTR_{DA}$	$MTTR_{WB}$	$MTBF_{DA}$	$MTBF_{WB}$	A_{DA}	A_{WB}	BQ	O_S
1	2.810	6.090	0.988	6887	2724	84831	109826	0.9410	0.9689	2200	73333
13	2.826	6.391	1.049	9704	3615	59031	100665	0.9121	0.9423	2287	76233
23	2.908	6.112	1.020	6860	4100	64555	92420	0.9309	0.9403	2490	83000
2	2.000	6.000	1.000	6885	3027	74058	85548	0.9255	0.9607	4466	123262
25	2.931	6.232	1.039	8869	2836	72167	85742	0.9063	0.9622	3226	83876
14	2.831	6.279	1.008	5817	2589	61732	61162	0.9131	0.9597	3618	87194
20	2.836	6.168	1.040	6068	2844	58663	54829	0.9003	0.9538	4245	87023
6	2.887	6.203	1.010	4525	2837	63209	57228	0.9267	0.9570	4253	86761
9	2.870	6.239	0.997	6086	4784	82377	79319	0.9287	0.9451	4658	85707
3	2.000	6.000	1.000	6886	3266	65586	69630	0.9100	0.9526	6732	122522
10	2.908	6.136	1.017	6981	2714	67692	70876	0.9103	0.9615	6071	83780
4	2.000	6.000	1.000	6886	3459	58749	58390	0.8945	0.9444	8998	121473
22	2.884	6.163	1.022	5235	4038	61128	65848	0.9264	0.9380	7091	85801
15	2.935	6.334	1.021	5394	3016	54070	62208	0.9202	0.9472	7255	84158
7	2.849	6.305	1.028	5163	2941	54960	57330	0.9174	0.9492	8033	85150
11	2.938	6.313	0.995	5648	3820	74237	57113	0.9100	0.9511	8663	83165
8	2.906	6.423	1.050	9187	4867	79758	76621	0.8929	0.9425	8591	82474
24	2.940	6.249	1.010	9969	4249	72484	86945	0.8971	0.9446	8857	82370
19	2.859	6.107	1.023	8709	3930	75810	70905	0.8906	0.9507	9366	84294
12	2.935	6.418	1.043	9295	3698	78107	82014	0.8982	0.9548	10102	80816
17	2.900	6.418	0.994	7917	3119	74532	77301	0.9071	0.9598	10624	82867
21	2.909	6.275	1.046	5782	3416	64102	65130	0.9185	0.9494	10723	83639
16	2.894	6.092	1.026	9759	2656	72339	95121	0.9069	0.9646	10703	81343
18	2.891	6.455	1.031	9326	3705	65053	74533	0.8888	0.9461	11041	81703
5	2.946	6.461	1.050	6883	3619	53115	50006	0.8790	0.9362	11264	81101

2.5 Mathematical Model Validation

In Section 2.5, the study develops a mathematical output model as a function of input variables using simulation optimization data from 25 random experiments.

In order to find O_M, the paper defines the total time available in the system, $T_{sys,}$ as the sum of two terms (a) individual process time available minus setup time for day 1 and (b) individual process time available, then are multiplied by the balance number of days used in the study. T_{max} is used to achieve steady-state conditions to apply the mathematical model in a real-time system (Gavriel 2007; Khan 2005; Martand 2010). ST_{DA} and ST_{WB} are defined as setup time for Die Attach and Wire Bond once at the start of the experimental runs. A_{DA} and A_{WB} are defined as Die Attach and Wire Bond availability using Equation (1). M is the number of machines available at the process based on the least capacity in the system. TD is the total time per day.

Thus, the total time available in the system, T_{sys}:
= total time available in the system for day 1 that includes setup time + total time available in the system for remaining of the days that excludes setup time
= total time available for Die Attach includes setup time for day 1 + total time available for Die Attach exclude setup time for balance of the days (T_{max}-1) + total time available for oven cure for T_{max} + total time available for Wire Bond includes setup time for day 1 + total time available for Wire Bond excludes setup time for balance of the days (T_{max}-1) + total time available for pre-cap inspection for T_{max}.

$$= ((T_D-ST_{DA})(A_{DA})+(T_D)(1)+(T_D-ST_{WB})(A_{WB})+(T_D)(1))(M) +(((T_D)(A_{DA})+(T_D)(1)+(T_D)(A_{WB})+ (T_D)(1))(M))x(T_{max}-1) \quad (2)$$

The mathematical model to find output is defined as total time available in the system, T_{sys}, divided by completion time, T_b and multiplied by batch quantity, BQ (Gavriel 2007; Khan 2005; Martand 2010). This is because completion time is obtained based on batch quantity. O_M equation is shown below.

$$\text{Average output per day, } O_M = \frac{\frac{Tsys\ x\ BQ}{Tb}}{Tmax} \quad (3)$$

Table 5 shows O_M data for 25 random experiments, and T_b is collected from the simulation.

Table 5. O_M data collection from 25 random experiments

Run	$MTTR_{DA}$	$MTBF_{DA}$	A_{DA}	$MTTR_{WB}$	$MTBF_{WB}$	A_{WB}	T_{sys}	T_b	O_M
1	6887	109826	0.9410	2724	84831	0.9689	30391064	27991	79622
13	9704	100665	0.9121	3615	59031	0.9423	29951939	30954	73764
23	6860	92420	0.9309	4100	64555	0.9403	30086196	31825	78466
2	6885	85548	0.9255	3027	74058	0.9607	30204209	46115	97503
25	8669	85742	0.9063	2836	72167	0.9622	30062486	40159	80498
14	5817	61162	0.9131	2589	61732	0.9597	30101914	43687	83098
20	6068	54829	0.9003	2844	58663	0.9538	29952015	49661	85343
6	4525	57228	0.9267	2837	63209	0.9570	30184513	50045	85507
9	6086	79319	0.9287	4784	82377	0.9451	30107451	54227	86206
3	6886	69630	0.9100	3266	65586	0.9526	30017168	69186	97359
10	6981	70876	0.9103	2714	67692	0.9615	30092258	68334	89116
4	6886	58390	0.8945	3459	58749	0.9444	29830491	92256	96981
22	5235	65848	0.9264	4038	61128	0.9380	30031335	78789	90094
15	5394	62208	0.9202	3016	54070	0.9472	30055465	81644	89026
7	5163	57330	0.9174	2941	54960	0.9492	30052033	88972	90443
11	5648	57113	0.9100	3820	74237	0.9511	30004094	95743	90494
8	9187	76621	0.8929	4867	79758	0.9425	29802304	95729	89152
24	9969	86945	0.8971	4249	72484	0.9446	29850928	97461	90426
19	8709	70905	0.8906	3930	75810	0.9507	29850034	101470	91842
12	9295	82014	0.8982	3698	78107	0.9548	29942562	111171	90695
17	7917	77301	0.9071	3119	74532	0.9598	30053172	116024	91729
21	5783	65130	0.9185	3416	64102	0.9494	30056319	116596	92140
16	9759	95121	0.9069	2656	72339	0.9646	30089675	115184	93198
18	9326	74533	0.8888	3705	65053	0.9461	29798827	120643	90904
5	6883	50006	0.8790	3619	53115	0.9362	29643539	123367	90220

The data between OS and OM from Table 4 and Table 5 are compared using statistical t-test analysis to validate the mathematical model using simulation optimization results. The study uses $\alpha = 0.05$ at 95% confidence level for validation. The p-value between O_M and O_S is 0.7787. Since the p-value is larger than α, the null hypothesis is accepted, indicating no significant difference between the simulation optimization method and the predicted mathematical model. The result shows that the mathematical model development based on simulation optimization is validated statistically using 25 random experiments.

2.6 Mathematical Programming Validation

In Section 2.6, the study develops a mathematical programming model and collects the data on response variables using experimental runs from simulation optimization.

Some annotations for mathematical programming are pre-defined below.

x_1 : output Die Attach
x_2 : output Oven Cure

x_3 : output Wire Bond
x_4 : output Pre-Cap Inspection
T_{DA} : time available for T_{max} days at Die Attach
T_{OC} : time available for T_{max} days at Oven Cure
T_{WB} : time available for T_{max} days at Wire Bond
T_{PC} : time available for T_{max} days at Pre-Cap Inspection
T_D : total time per day
M_{DA} : number of machines available at Die Attach
M_{OC} : number of machines available at Oven Cure
M_{WB} : number of machines available at Wire Bond
M_{PC} : number of machines available at Pre-Cap Inspection
O_{OR} : optimized output per day from mathematical programming method

Equation (4)-(7) represents the total time available for T_{max} days at each process, including the setup time, availability and number of machines available. The total time available for T_{max} is defined as the sum of two-term (a) total time per day minus setup time multiplied by availability and number of machines available at each process and (b) total time per day multiplied by availability, the number of machine available at each process and (T_{max}-1) (Gavriel 2007; Khan 2005; Martand 2010).

Similar to section D, T_{max} is used in Equation (4)-(7) to match the T_{sys}, Equation (2) and also used for calculation on steady-state conditions.

$$T_{DA} = (T_D\text{-}ST_{DA})\ (A_{DA})(M_{DA})+(T_D x A_{DA} x M_{DA} x (T_{max}\text{-}1)) \quad (4)$$
$$T_{OC} = T_D \ x\ M_{OC}\ x\ T_{max} \quad (5)$$
$$T_{WB} = (T_D\text{-}ST_{WB})\ (A_{WB})(M_{WB})+(T_D x A_{WB} x M_{WB} x (T_{max}\text{-}1)) \quad (6)$$
$$T_{PC} = T_D \ x\ M_{PC}\ x\ T_{max} \quad (7)$$

Equation (8) represents the optimized output per day in mathematical programming. O_{OR} is defined as the average output at the end of the system multiply batch quantity and divided by the maximum number of days used in the study.

The O_{OR} is given below:

$$O_{OR} = \frac{\left[\left(\frac{x1+x2+x3+x4}{4}\right) x\ BQ\right]}{Tmax} \quad (8)$$

From Equation (4)-(8), the mathematical programming model is developed as below.

The objective function is to maximize the output at the end of the system.
Objective function:

$$\text{Maximize } \frac{\left[\left(\frac{x1+x2+x3+x4}{4}\right) x\ BQ\right]}{30} \text{ or Maximize } O_{OR} \quad (9)$$

Equations (10)-(15) ensure that the optimized output O_{OR} does not exceed the total time used by the O_{OR} than the overall time available in the system or at each process. Equation (10) states that the sum of each process's total processing time must not exceed the total time available in the system, T_{sys}. Equation (10) can be rewritten as Equation (11), where Equation (11) states the average output in the system multiplied by completion time, t_b must not exceed the total time available in the system, T_{sys}. Equation (12) states the sum of processing time at Die Attach must not exceed the time available at the process. Equation (13) states the sum of processing time at Oven Cure must not exceed the time available at the process. Equation (14) states the sum of processing time at Wire Bond must not exceed the time available at the process. Equation (15) states the sum of processing time at Pre-Cap Inspection must not exceed the time available at the process.

Constraints:

$$(CT_{DA}\ x\ BQ)\ (x_1) + 7200\ (x_2) + (CT_{WB}\ x\ BQ)\ (x_3) + (CT_{PC}\ x\ BQ)\ (x_4) \leq T_{sys} \quad (10)$$

or $\left(\frac{x1+x2+x3+x4}{4}\right)(Tb) \leq T_{sys}$
(11)

$$(CT_{DA}\ x\ BQ)(x_1) \leq T_{DA} \quad (12)$$
$$7200\ (x_2) \leq T_{OC} \quad (13)$$
$$(CT_{WB}\ x\ BQ)\ (x_3) \leq T_{WB} \quad (14)$$
$$(CT_{PC}\ x\ BQ)\ (x_4) \leq T_{PC} \quad (15)$$

Table 6 shows the data obtained from the mathematical programming method using 25 random experiments from simulation optimization.

Table 6. Result from mathematical programming using the same 25 random experiments from simulation optimization

Run	O_{OR}
1	79622
13	73764
23	78466
2	97503
25	80498
14	83098
20	85343
6	85507
9	86206
3	97359
10	89116
4	96981
22	90094
15	89026
7	90443
11	90494
8	89152
24	90426
19	91842
12	90695
17	91729
21	92140
16	93198
18	90904
5	90220

In order to validate the mathematical programming using simulation optimization results, the data between O_S and O_{OR} from Table 4 and Table 6 are compared using statistical t-test analysis. The study uses $\alpha = 0.05$ at 95% confidence level for validation. The p-value between O_M and O_S is 0.7787. Since the p-value is larger than α, it accepts the null hypothesis indicating no significant difference between the simulation optimization method and mathematical programming. The result shows that the mathematical programming based on simulation optimization is validated statistically using 25 random experiments.

In order to validate the mathematical model with mathematical programming, the data of O_M and O_{OR} are compared using statistical analysis. The study uses $\alpha = 0.05$ at 95% confidence level for validation. The p-value between O_M and O_{OR} is 1. Since the p-value is larger than α, it accepts the null hypothesis indicating no significant difference between the simulation optimization method and the predicted mathematical model. Furthermore, the result indicates both models show similar data from the 25 random experiments. Thus, there is no significant difference between the mathematical model and mathematical programming.

2.7 Mathematical Model Testing

The validated mathematical model (Equation (3)) is further compared with real-time data. The real-time data on output and input variables are collected from the system.

Table 7 compares 9 sample results between the mathematical model and the real-time data on the output response variable. O_M shows the data from the mathematical model using Equation (4). O_{RT} shows the data from the real-time system. The relative error was 1.57%–10.36%, with an average value of 7.28%.

Table 7. Relative error analysis between O_M and O_{RT}

O_M	O_{RT}	Relative error	% error
70156	71254	1098	1.57
65384	68678	3294	5.04
74680	72829	1851	2.48
102810	113458	10648	10.36
112808	105555	7253	6.43
101455	109689	8234	8.12
132519	119911	12608	9,51
138868	150216	13348	9,75
126634	117763	8871	7.01
Average % error			7.28

3. Finding and Discussion

The study shows that validation of the simulation model with real-time data plays a very important role in improving the mathematical model application in a real-time system. The main reason for the improvement in model prediction in this study is due to the revised methodology that envelopes the mathematical model development through data mining using simulation optimization data and its concurrent mathematical programming validation

The study shows the importance of validating the model's static and dynamic properties. The mathematical model development through data mining on simulation optimization has performed validation of dynamic properties as the mathematical model is producing the acceptance statistical analysis data as a simulation optimization result (see Tables 4 and 5). The dynamic properties of the model reflect the actual environment that is affected by time and describe the change from one state to another. The integration between mathematical model development and its validation using simulation optimization has advantages to reducing procedure in this proposed methodology and a higher probability of producing closer results to real-time data since the simulation is a reflection tool of the actual manufacturing system. Simulation replicates the behavior and exhibits the changes in nature, representing the real-time manufacturing system. While it is a popular validation tool among researchers, the study argues the limitation and less effectiveness of having one validation due to a lack of prediction effectiveness.

In order to improve the mathematical model's effectiveness, it is proposed to validate the static properties. The static properties calculate the model performance in a time-invariant steady-state, and the change takes place instantaneously. Using the input setting from simulation optimization, the mathematical model and mathematical programming comparison show similar data for each experiment of various input settings (see Tables 5 and 6). This method reinforces the validation of static properties as evidence to improve mathematical model robustness for prediction effectiveness. The importance of this validation reveals that the manufacturing system contains both dynamic and static properties, which are required in the proposed mathematical model.

Based on the results, we conclude that the model has a high satisfactory quality estimate and is acceptable for performing prediction. The error between the mathematical model and sampling real-time data in the study is in the range 0%–9.9% marginally, which is a relatively high-quality estimate (Eickemeyer et al., 2014). Based on the findings, it is recommended that simulation optimization is the better alternative data mining when compared with other methods as it exhibits the dynamic properties of the system. It is also recommended to use additional static properties of the model for validation to demonstrate prediction effectiveness to fit into the system's behavior. The inclusion of static properties of validation in parallel with dynamic properties enhances the effectiveness of the proposed mathematical model to produce closer results in manufacturing systems exhibiting both types of behavior.

The effectiveness of the mathematical model includes its capability to adapt the changes in various input values of higher number of variables and at the same time, it can produce acceptable error of the output variable. Two other key elements that led to this contribution are randomness in 25 experiments and the acceptance of other real-time input variables in the simulation model (validation of individual processes). Due to this result, the study proposes this enhanced methodology to establish a more effective mathematical model in manufacturing systems to address the limitations that the previous studies faced.

4. Conclusion

In this paper, an enhanced methodology that involves mathematical model development through simulation optimization and its concurrent static validation properties is proposed to improve the mathematical model in manufacturing systems. The main goal of the mathematical model is to minimize the gap between theory and practice. The data mining on simulation optimization is used for the mathematical model development. The developed mathematical model is further validated with static properties of mathematical programming. The paper emphasizes the quantitative analysis during the mathematical model development and validation. The effectiveness of the developed mathematical model is evaluated with real-time data. The mathematical model and real-time data result show that the model is satisfactory and acceptable for prediction analysis in the manufacturing system. Future studies must focus on enhancing the proposed methodology for the mathematical model development and its validation to narrow the gap between the mathematical model and real-time data. The findings can be used to continuously enhance the developed mathematical model and apply the model in real-time manufacturing systems in the future. Although this methodology is an improved version of current literature, the mixture of data mining on dynamic properties of the simulation model and static properties of mathematical programming in this proposed methodology provides a new platform for researchers to address further these limitations for future enhancement of the mathematical model.

References

Abazari, A. M., Solimanpur, M., and Sattari, H., Optimum loading of machines in a flexible manufacturing system using a mixed-integer linear mathematical programming model and genetic algorithm, *Computers and Industrial Engineering*, *62*(2), 469–478. doi:10.1016/j.cie.2011.10.013, 2012.

Almehdawe, E., and Jewkes, E. Performance analysis and optimization of hybrid manufacturing systems under a batch ordering policy. *International Journal of Production Economics*, *144*(1), 200–208. (2013).doi:10.1016/j.ijpe.2013.02.005

Al-Zuheri, a., Luong, L., and Xing, K., The role of randomness of a manual assembly line with walking workers on model validation, *Procedia CIRP*, *3*, 233–238. doi:10.1016/j.procir.2012.07.041, 2012.

Arif, F., Suryana, N., and Hussin, B., Cascade quality prediction method using multiple PCA+ID3 for multi-stage manufacturing system, *IERI Procedia*, *4*, 201–207. doi:10.1016/j.ieri.2013.11.029, 2013.

Bagheri, M., and Bashiri, M., A new mathematical model towards the integration of cell formation with operator assignment and inter-cell layout problems in a dynamic environment, *Applied Mathematical Modelling*, *38*(4), 1237–1254. doi:10.1016/j.apm.2013.08.026, 2014.

Brown, J. R., A capacity constrained mathematical programming model for cellular manufacturing with exceptional elements, *Journal of Manufacturing Systems*, 1–6. doi:10.1016/j.jmsy.2014.09.005, 2014.

Cao, Z., Deng, J., Liu, M., and Wang, Y., Bottleneck prediction method based on improved adaptive network-based fuzzy inference system (ANFIS) in semiconductor manufacturing system, *Chinese Journal of Chemical Engineering*, *20*(6), 1081–1088. doi:10.1016/S1004-9541(12)60590-4, 2012.

Chang, H. J., Su, R. H., Yang, C. Te, and Weng, M. W., An economic manufacturing quantity model for a two-stage assembly system with imperfect processes and variable production rate, *Computers and Industrial Engineering*, *63*(1), 285–293. doi:10.1016/j.cie.2012.02.011, 2012.

Chau, M., and Fu, Michael C, Huashuai Qu, I. O. R., Simulation optimization: A tutorial overview and recent developments in gradient-based method, *Proceedings of the 2014 Winter Simulation Conference*, 21–35, 2014.

Chincholkar, M., and Herrmann, J. W., Estimating manufacturing cycle time and throughput in flow shops with process drift and inspection, *International Journal of Production Research*, *46*(24), 7057–7072. doi:10.1080/00207540701513893, 2008.

Defersha, F. M., and Chen, M., A comprehensive mathematical model for the design of cellular manufacturing systems, *International Journal of Production Economics*, *103*, 767–783. doi:10.1016/j.ijpe.2005.10.008, 2006.

Eickemeyer, S. C., Steinkamp, S., Schuster, B., Bodenhage, F., and Nyhuis, P., Reliable capacity planning despite uncertain disassembly, regeneration and reassembly workloads by using statistical and mathematical approaches – Validation in subsidiaries of a global MRO company with operations in Asia, Europe and North America, *Procedia CIRP*, *23*, 252–257. doi:10.1016/j.procir.2014.10.097, 2014

Gavriel Salvendry, *Handbook of Industrial Engineering: Technology and Operations Management, Third Edition*, Wiley Online Library, ISBN: 9780471330578, doi: 10.1002/9780470172339, 2007.

Georgiadis, P., and Michaloudis, C., Real-time production planning and control system for job-shop manufacturing: A system dynamics analysis, *European Journal of Operational Research*, *216*(1), 94–104. doi:10.1016/j.ejor.2011.07.022, 2012.

Harrell, C. R., and Price, R. N., Simulation modeling using ProModel technology, *Proceedings of the 2003 Winter Simulation Conference, 2003.*, *1*, 175–181. doi:10.1109/WSC.2003.1261421, 2003.

Jacomino, A step toward capacity planning at finite capacity in semiconductor manufacturing. *Proceedings of the 2014 Winter Simulation Conference*, 2239–2250, 2014.

Jagdev, H. S., Browne, J., and Jordan, P., Verification and validation issues in manufacturing models, *Computers in Industry*, *25*(3), 331–353. doi:10.1016/0166-3615(94)00045-R, 1995.

Khan, *Industrial Engineering*. New Age International, ISBN 8122420591, 9788122420593, 2007.

Kim,S.H., and Lee, C.M, Non linear prediction of manufactuiring systems through explict and implicit data mining, *Computer Industrial Engineering*, 33(3-4), pp. 461-464, 1997.

Kim, W., and Morrison, J. R., The throughput rate of serial production lines with deterministic process times and random setups: Markovian models and applications to semiconductor manufacturing, *Computers & Operations Research*, *53*, 288–300. doi:10.1016/j.cor.2014.03.022, 2014.

Kleijnen, J., Verification and validation of simulation models, *European Journal of Operational Research*, *82*(1), 145–162. doi:10.1109/WSC.1998.744907, 1995.

Leitão, P., and Restivo, F., A holonic approach to dynamic manufacturing scheduling, *Robotics and Computer-Integrated Manufacturing*, *24*, 625–634. doi:10.1016/j.rcim.2007.09.005, 2008.

Li, W., Manufacturing process diagnosis using functional regression, *Journal of Materials Processing Technology*, *186*(April 2004), 323–330. doi:10.1016/j.jmatprotec.2006.12.052, 2007.

Li, X., Gao, L., Shao, X., Zhang, C., and Wang, C., Mathematical modeling and evolutionary algorithm-based approach for integrated process planning and scheduling, *Computers & Operations Research*, *37*(4), 656–667. doi:10.1016/j.cor.2009.06.008, 2010.

Lin, J. S., A novel design of wafer yield model for semiconductor using a GMDH polynomial and principal component analysis, *Expert Systems with Applications*, *39*(8), 6665–6671. doi:10.1016/j.eswa.2011.09.146, 2012.

Luo, M., Yan, H.-C., Hu, B., Zhou, J.-H., and Pang, C. K., A Data-driven two-stage maintenance framework for degradation prediction in semiconductor manufacturing Industries. *Computers & Industrial Engineering*, *85*, 414–422. doi:10.1016/j.cie.2015.04.008, 2015.

Mahdavi, I., Aalaei, A., Paydar, M. M., and Solimanpur, M., A new mathematical model for integrating all incidence matrices in multi-dimensional cellular manufacturing system, *Journal of Manufacturing Systems*, *31*(2), 214–223. doi:10.1016/j.jmsy.2011.07.007, 2012.

Martand Telsang, *Industrial Engineering and Production Management*, S. Chand & Company LTD, New Delhi, 2010.

Mason, S. J., Hill, R. R., Mönch, L., Rose, O., Jefferson, T., Fowler, J. W., … Chen, C., Some topics for simulation optimization, *Proceedings of the 2008 Winter Simulation Conference*, 27–38, 2008.

Sargent, R. G., Verification and validation of simulation models, *Proceedings of the 2011 Winter Simulation Conference S. Jain, R.R. Creasey, J. Himmelspach, K.P. White, and M.*, 183–198, 2011.

Shen, X.-N., and Yao, X., Mathematical modeling and multi-objective evolutionary algorithms applied to dynamic flexible job shop scheduling problems, *Information Sciences*, *298*(219), 198–224. doi:10.1016/j.ins.2014.11.036, 2015.

Usubamatov, R., Sin, T. C., and Ahmad, R., Mathematical models for the productivity rate of automated lines with different failure rates for stations and mechanisms, *The International Journal of Advanced Manufacturing Technology*. doi:10.1007/s00170-015-7005-6, 2015.

Wu, K., Taxonomy of batch queueing models in manufacturing systems, *European Journal of Operational Research*, *237*(1), 129–135. doi:10.1016/j.ejor.2014.02.004, 2014.

Zuheri, A. Al, Luong, L., and Xing, K., Using simulation in verification of a mathematical model for predicting the performance of manual assembly line occupied with flexible workforce. *International Journal of Simulation and Process Modelling*, *9*(4), 270. doi:10.1504/IJSPM.2014.066369, 2014.

Biography

Ho Kok Hoe is a PhD graduate (Doctor of Philosophy in Engineering) recently at University Tunku Abdul Rahman. He received a Master of Science in Engineering degree from Open University Malaysia in 2012 and a Bachelor of Engineering degree in Electrical from University Tun Hussein Onn in 2002. His major fields of study are based on Industrial Engineering, Manufacturing, Operations Research and Process Engineering. He was a former engineer in the polymer and semiconductor industries. Currently, he works as a senior lecturer at Monash University Malaysia under the School of Engineering.

Joshua Prakash is an assistant professor in the Department of Industrial Engineering, Faculty of Engineering and Green Technology, Universiti Tunku Abdul Rahman, Malaysia. He holds a B.Eng and PhD in manufacturing engineering from Universiti Sains Malaysia. His recent research projects include production planning and control, manufacturing systems and lean manufacturing.

Ong Kok Seng is a professor in the Department of Industrial Engineering, Faculty of Engineering and Green Technology, Universiti Tunku Abdul Rahman, Malaysia. He graduated with a PhD in Mechanical Engineering from the University of London. His research interests include heat pipes, heat exchangers, heat pumps, solar energy (thermal applications) and thermoelectric.

.