1-2014

Manufacturing Assembly Time Estimation Using Structural Complexity Metric Trained Artificial Neural Networks

Michael G. Miller
Clemson University, mm3@clemson.edu

James L. Mathieson
Clemson University, jmathie@clemson.edu

Joshua D. Summers
Clemson University, jsummer@clemson.edu

Gregory M. Mocko
Clemson University, gmocko@clemson.edu

MANUFACTURING ASSEMBLY TIME ESTIMATION USING STRUCTURAL COMPLEXITY METRIC TRAINED ARTIFICIAL NEURAL NETWORKS

Michael G. Miller
Research Assistant
Department of Mechanical Engineering
Clemson University
Clemson, SC 29634-0921
mm3@clemson.edu

James L. Mathieson
Research Assistant
Department of Mechanical Engineering
Clemson University
Clemson, SC 29634-0921
jmathie@clemson.edu

Joshua D. Summers
Professor
Department of Mechanical Engineering
Clemson University
Clemson, SC 29634-0921
jsummer@clemson.edu

corresponding author

Gregory M. Mocko
Associate Professor
Department of Mechanical Engineering
Clemson University
Clemson, SC 29634-0921
gmocko@clemson.edu

Originally Submitted as: IDETC 2012-71337

ABSTRACT

Assembly time estimation is traditionally a time-intensive manual process that requires detailed geometric and process information, which is often subjective and qualitative in nature. As a result, assembly time estimation is rarely applied during early design iterations. In this paper, the authors explore the possibility of automating the assembly time estimation process while reducing the level of design detail required. In this approach, they train artificial neural networks (ANNs) to estimate the assembly times of vehicle sub-assemblies using either assembly connectivity or liaison graph properties, respectively as input data. The effectiveness of estimation is evaluated based on the distribution of estimates provided by a population of ANNs trained on the same input data using varying initial conditions. Results indicate that this method can provide time estimates of an assembly process with +/-15% error while relying exclusively on the geometric part information rather than process instructions.

Keywords: graphs, DFA, artificial neural networks

1 ASSEMBLY TIME ESTIMATION

With the evolution of the Ford manufacturing system and the Toyota Production System in the 1950's, managers focused their efforts on eliminating waste by emphasizing a streamlining of the work sequence to reduce the duration of that employees, machines and materials were idle. This focus became the foundation for lean manufacturing, which resulted in an emphasis on identifying and eliminating waste, while increasing production and quality [1–3]. This trend, which required the standardization of assembly time estimation procedures in turn led to the development of many methods for predicting product assembly times [4–7].

Assembly time estimation is the first step in most design-for-assembly methods, by which designers can predict, with varying degrees of accuracy, not only how long it takes to assemble a product but also to compare assembly times between different design solutions [8,9]. The marked increase in global competition has forced manufacturing enterprises to compete in terms of quality, cost, and time to market of their products [10,11]. Consequently, in order to operate efficiently and competitively, manufacturers must understand all the costs they incur with the manufacture of their goods.

It is increasingly apparent elucidating these costs must be achieved earlier in the design phase. The possibility of influencing a product's cost substantially decreases early in the product's life cycle while the cost of modifications increases drastically as the project advances. Thus, although decisions made early in a product's life cycle significantly impact the overall costs of the product, the bulk of such expenses are incurred upon completion of the design phase as shown in Figure 1 [12]. Though nearly 80% of the costs a product incurs throughout its life cycle occur within the design phase, this phase of production consumes less than 15% of the budget [8,12–14].

As a product moves from the design phase to production, more information about the product and the assembly process becomes available, which in turn increases in certitude. The availability of more information, in turn results in more accurate assembly time estimates. For example, little information is available during conceptual design, often only requirements and conceptual sketches. As the design process continues, more data is collected, including information regarding the connectivity of parts to each other.

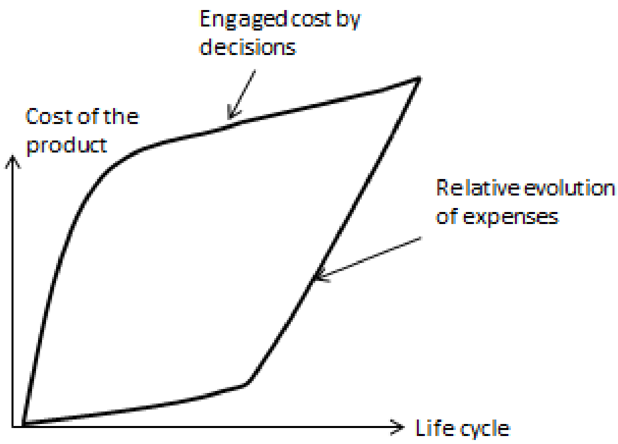

Figure 1: Cost engagements and expense occurrences throughout lifecycle [12]

Previous research has shown that assembly time estimates for consumer products can be predicted to within 16% of Boothroyd and Dewhurst assembly time estimates using only information about the part connectivity of the product [15]. Here, the authors seek to determine if this approach can be adapted and applied to the automotive industry as illustrated in Figure 2. Here, a typical V-diagram with system level design and requirements are evolved to the component level with subsequent validation and detailing, resulting in assembly process planning. Often, the assembly time estimation is only done late in the process after all the design decisions have been determined, limiting engineers in justifying design decisions based on assembly time improvements.

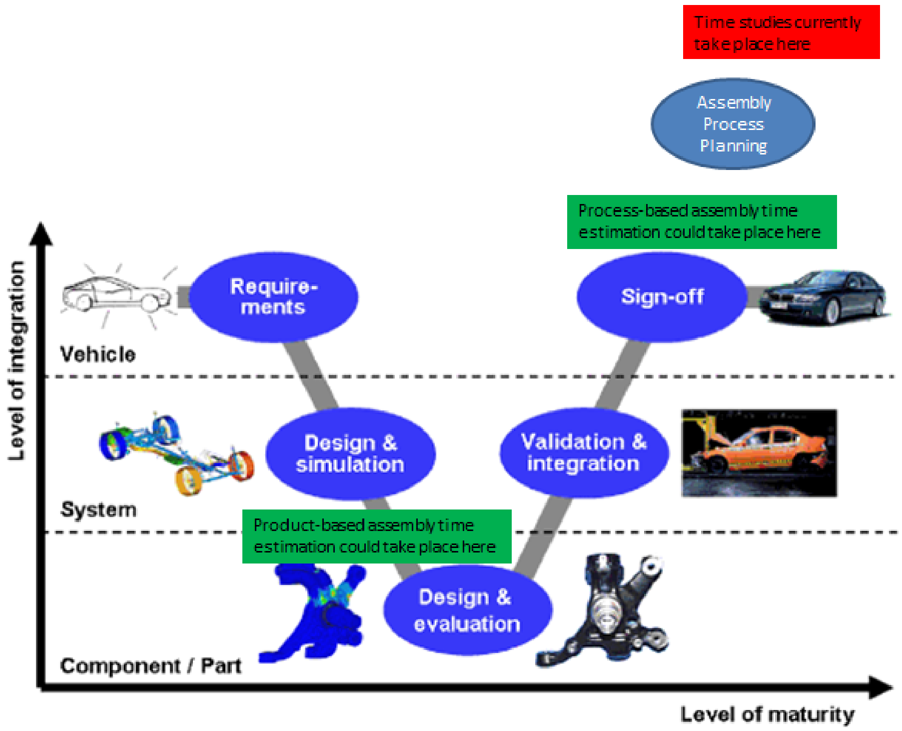

Figure 2: Automotive manufacturing product life-cycle, adapted from [16]

Assembly time estimates made at early stages of a product life-cycle are not expected to be as accurate as that conducted during production because there is substantially less information available to analyze. As a result, the estimates sought in this research are not intended to replace full scale assembly time studies which occur during production, but to supplement them by providing information earlier in the product life cycle. Figure 3 illustrates a typical vehicle development timeline, starting with the solid models of parts for the vehicle being defined approximately 20 months before launch, progressing to pre-launch assembly plans where the work instructions are defined approximately 16 months before launch, to prototype build nearly six months before launch when final time studies with both assembly fixtures and assembly associates are defined. The work presented in this paper addresses the connectivity assistance where the parts and assemblies are defined. In a parallel research effort, the authors report on using the process instruction sets to predict assembly times [17].

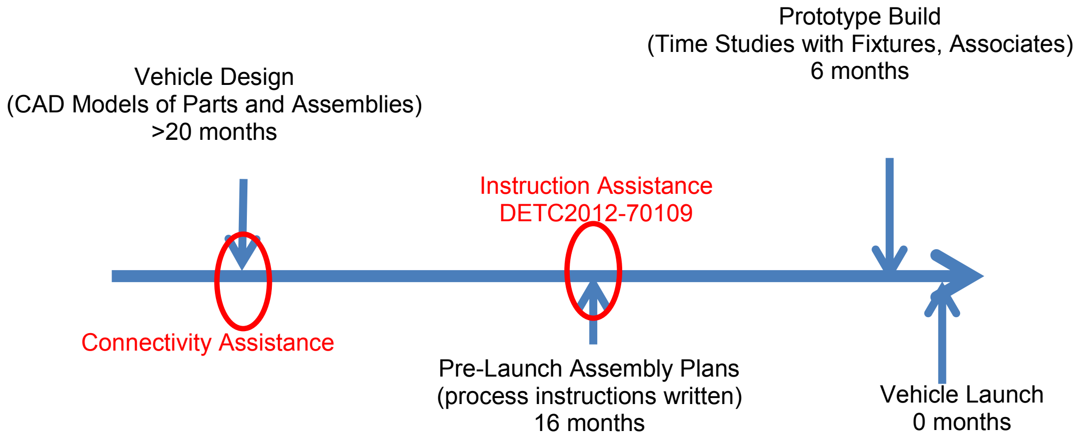

Figure 3: Typical Vehicle Development Timeline

1.1 Assembly Time Estimation Methods

In any effort to develop an assembly time estimation method, a means of assessing the assembly time of a system must be employed. In most traditional, late-stage assembly time estimation methods this takes the form of time studies requiring the direct observation of physical system assembly processes to a statistically significant replication count. For the development of an early-stage assembly time estimation method with an acceptable degree of inaccuracy, however, such exhaustive methods are both impractical and unnecessary. Rather, existing assembly time estimation methods can be used to acquire assembly time information which does not vary significantly from the observed assembly time to skew results. In this paper, three assembly time estimation methods are discussed: methods-time measurement (MTM), Boothroyd and Dewhurst (B&D), and connectivity.

1.1.1 Methods-Time Measurement

Methods-time measurement, MTM, is a predetermined time study system in which operations are described by MTM "elements" [5,18]. Though originally time-consuming, this MTM technique has since evolved to reduce the time required for analysis, with the first iteration MTM-2 followed by MTM-3, MOST and MTM-UAS [19].

MTM analysis requires the analysis of each motion of an operation, which on this level allows the user to easily identify obvious problems and non-value added motions [20]. The pre-determined times associated with specific motions have been determined in advance by statistical analysis. As an example, MTM-UAS defines seven basic motions: take and place, place, helper equipment use, running (machine or equipment), motion cycles, body motions, and visual control [20].

Though an MTM style method is effective and relatively objective, the application of the method is labor intensive, requiring a significant amount of time to discretize an operation into its individual motions. On the other hand, the objectivity and accuracy of the method lend MTM and MTM derivatives to application in industry. The majority of assembly time estimates used as targets in this research is the result of a custom MTM derivation developed by an automotive OEM.

1.1.2 Boothroyd and Dewhurst

The Boothroyd and Dewhurst method examines the difficulties of handling and insertion a part when assigning an appropriate assembly time. A series of tables is used to determine an appropriate time for both handling and insertion of each part. The individual times within the tables have been developed from numerous time studies. The handling and insertion times selected from each set of tables are summed for each of the parts to determine the estimated assembly time for the product. Though beneficial, this time consuming method includes subjective inputs [8,21]. Additionally, the information required for this analysis is often only present after the design is complete or near completion.

For example, to determine a handling time, the user must decide if the part is easy or difficult to handle as well as the freedom of the part to rotate both parallel and perpendicular to its insertion point [8]. For insertion time, the user must determine if holding down is required to maintain orientation, if the part is easy or difficult to align, and if resistance is present when inserting the part. Additionally, users must select if the part and tool can easily access the desired location, as well as if access is either obstructed or vision is restricted.

The major deficiencies of both the tabular and software based Boothroyd and Dewhurst assembly time estimation methods include the i) need for significant amounts of information about both the product and the process and ii) the number of subjective inputs required for analysis.

1.1.3 Connectivity Assembly Time Estimation Method

The connectivity based assembly time estimation technique allows for the estimation of assembly time relying strictly on part connectivity information [15,22]. Though tested only on consumer appliances, it is likely that this method can be expanded to the automotive manufacturing industry. For example, it may be possible to extend connectivity based assembly time to an automotive sub-assembly and its part connectivity graph (Figure 4). Detailed examples of the bi-partite graph and tabular representations used can also be seen in Figure 5 and Figure 6.

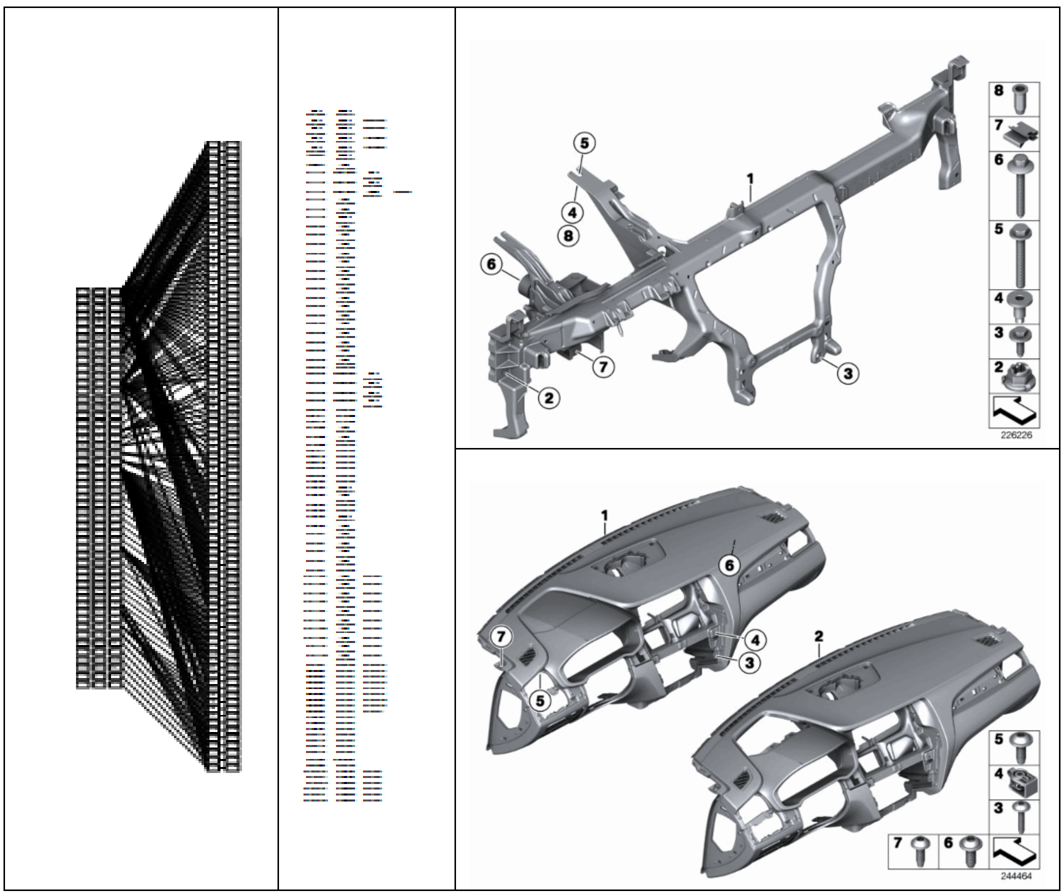

Figure 4: Bi-partite graph and tabular equivalent of automotive sub-assembly

The current connectivity method predicts an assembly time based on a surrogate function on the properties of the part connectivity graph. This proposed model was developed using a manual pattern recognition approach, by comparing and combining regression trends of average path length, the number of elements, and the path length density plotted against Boothroyd and Dewhurst assembly time estimates [15]. Though successful for consumer appliances as compared to Boothroyd and Dewhurst predicted assembly times, this method has yet to be fully applied to products in different industries. The method could be fully objective with further definition of connections and has the potential to be fully automated.

1.1.4 Summary of Time Estimation Methods

With the exception of the connectivity method, currently available assembly time estimation methods rely heavily on information that is not available until late in the design phase (e.g. required body movements; difficulty of handling and inserting parts; part size, weight and stickiness; and required order of connection). This poses a problem when users need information about assembly times earlier in the product's life cycle. For this reason, the authors propose that assembly time estimation can be obtained using part connectivity which is available earlier in the product life-cycle. Hence a major purpose of this research is to perform the analysis using information more readily available early in a product's lifecycle. Table 1 summarizes the information required in each of the assembly time estimation methods discussed thus far. The objective information (in shaded boxes) is obtained through automated and algorithmic software, whereas the subjective information requires human interpretation and judgment.

Table 1: Questions asked of the designers for existing assembly time estimation techniques

MTM	Boothroyd and Dewhurst	Connectivity
What movements are necessary to perform assembly actions?	What order are parts connected in?	Which parts are connected to each other?
How difficult are parts to handle?	How difficult are parts to handle?	How many connection instances are present between parts?
How difficult are parts to insert?	How difficult are parts to insert?	
Part attributes such as envelope size, weight, stickiness, etc.	Part attributes such as envelope size, weight, stickiness, etc.	

The connectivity method has shown promise in estimating assembly times for consumer products based on time estimates derived from Boothroyd and Dewhurst time analysis. The research presented in this paper addresses the possibility of refining this method for use in the automotive manufacturing industry.

2 PRODUCT CONNECTIVITY INFORMATION

Information about the part connectivity of a product can be generated manually by reverse engineering a product or studying 3-D models and 2-D drawings [10,23,24]. Significant research has also been conducted on the ability to extract part connectivity information from computer-aided design models [25,26]. Connectivity may also have various connotations, however.

One such connotation, that used for analysis in the connectivity method, entails dealing with parts in contact with other parts and the number of contact points between them [15]. Another form of part connection information is mating relationships defined in CAD models. Though related, these two information types are not identical. Consequently, the authors seek to elucidate connectivity in terms of physical part connections which may not have explicit mate conditions defined in the CAD environment. In that there are many permutations for fully constraining parts in an assembly, different designers, or even the same designer, may constrain assemblies differently. For these reasons, we emphasis the connections of physical parts here rather than the modeling of mates in CAD, which is reserved for other work [22].

3 DATA COLLECTION

To begin the process of evaluating and refining the connectivity method, the connectivity graphs and associated MTM-based time estimates for a sample of systems are required. As discussed earlier, there are currently no automated means of generating connectivity information although current research in this field is ongoing [21,22,27]. Therefore, we obtain the connectivity graphs using a combination of observation of the assembly process, informal interviews with process associates, and information contained within CAD models.

Information collected for this analysis originates from the successive tasks performed in the assembly of one sub-assembly area for one specific vehicle. The assembly tasks are all within the instrument panel and take place between the attachment of the first part to a fixture and the completed sub-assembly's marriage to the vehicle.

One observation noted during information collection was that the graph did not change after the completion of every assembly task. Though a perceived limitation, it also has obvious advantages. The limiting factor is obvious; the method is incapable of estimating an assembly time for tasks that do not result in changes in the connectivity of the parts. However, those tasks not captured by the connectivity graphs are not truly value added activities. This finding, which may be the subject of our future research, may allow for automated connectivity analysis to identify non-value added activities. Some examples of these tasks types are:

- Place cockpit sub-assembly broadcast sheet to the AGC with magnet;
- Remove two transport covers from upper flaps of the Heater/Aircon Low;
- Place in the recycle bin.

In that these three tasks do not affect the connectivity graph of the product, an assembly time cannot be estimated for these steps using an analysis of the connectivity graph.

The first process in the sub-assembly that changes the connectivity graph is the attachment of the second part to the base part. The connectivity graph upon completion of this step is shown in Figure 5. This process involves attaching Part B to Part A and securing with two bolts (Bolt A and Bolt B) and two screws (Screw A and Screw B).

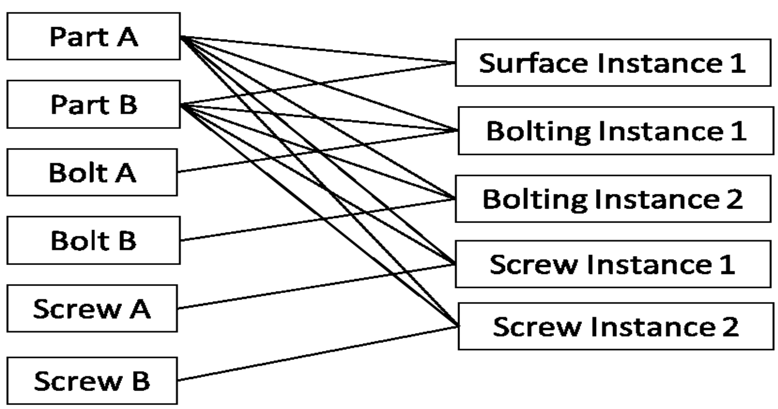

Figure 5: Connectivity graph after first set of assembly tasks

As the process evolves, the connectivity graph continues to grow in size with the inclusion of more parts to the system and the incorporation of more connection instances. The properties of the graph also change with the completion of more assembly processes. In this paper, the authors attempt to elucidate this trend for purposes of mapping the properties of connectivity graphs to assembly times. The growth of the connectivity graph is shown in that tables in Figure 6 and Figure 7.

IPC - 9201698	IHKA-9225727	
IPC - 9201698	IHKA-9225727	9906200A
IPC - 9201698	IHKA-9225727	9906200B
IPC - 9201698	IHKA-9225727	7142046A
IPC - 9201698	IHKA-9225727	7142046B

Figure 6: Tabular view of connectivity graph after one process

Figure 7 shows an increase in graph size after the completion of the first three assembly processes.

IPC - 9201698	IHKA-9225727		
IPC - 9201698	IHKA-9225727	9906200A	
IPC - 9201698	IHKA-9225727	9906200B	
IPC - 9201698	IHKA-9225727	7142046A	
IPC - 9201698	IHKA-9225727	7142046B	
7060601	IHKA-9225727		
7654321	IPC - 9201698		
9177178	9242146A	IPC - 9201698	
9177178	9242146B	IPC - 9201698	
9177178	9242146C	IHKA-9225727	7060601
9177178	IPC - 9201698		
9177178	IPC - 9201698		
9177178	IHKA-9225727		

Figure 7: Tabular view of connectivity graph after three processes

The data collection process continued for each of the non-option related tasks associated with the sub-assembly of the instrument panel. In total, 24 connectivity graphs and the associated assembly times for each activity were collected to be used in the analysis.

4 GRAPH PROPERTY ANALYSIS

The original connectivity method proposes that the properties of product connectivity graphs could be used to estimate assembly times for a given product [15]. The properties of a bi-partite graph, such that shown in Figure 4, are the basis for the analysis proposed here and in the authors' previous research [28–30]. The graph properties used here are based on the same as that proposed in the original assembly time estimation research based on connectivity [15]. These have been expanded to twenty nine properties, each falling into one of four main categories: size, interconnectivity, centrality, and decomposition.

5 PERFORMANCE EVALUATION OF EXISTING CONNECTIVITY METHOD

The original connectivity method, designed and mapped to model assembly times for consumer products, was mapped to connectivity graphs to Boothroyd and Dewhurst assembly time estimates to within 16% [15]. Prior to establishing a new mapping scheme that can be applied to automotive industry assembly processes a study was conducted to determine the extent to which the original method could predict these times. The results of this study on twenty-four connectivity graphs shows an average error of twenty six percent with a range in errors between -134% and 352%. The plot of MTM-based assembly time estimates for these twenty-four graphs and the original connectivity estimates is shown in Figure 8. As the function developed in the original model is not capable of accurately estimating assembly times in the automotive industry, the authors propose exploring other, more complex techniques to develop an appropriate mapping for this application.

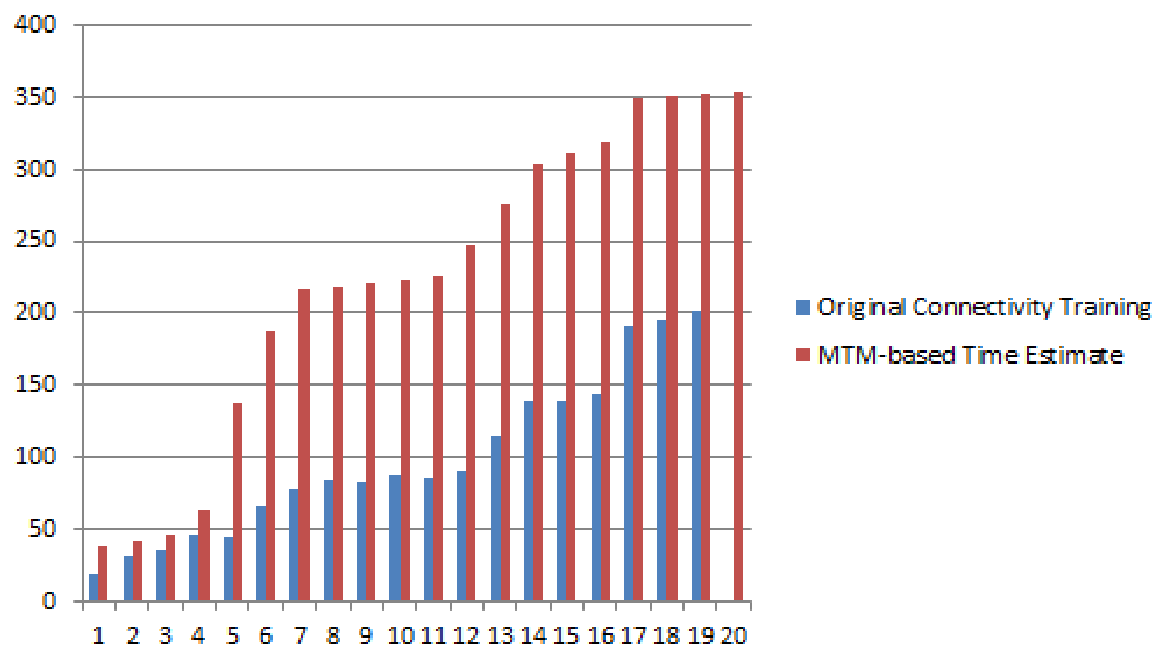

Figure 8: MTM-based estimates and original connectivity-based estimates

6 ARTIFICIAL NEURAL NETWORKS

The next step involves mapping the connectivity graph properties to the MTM-based assembly time estimates. Artificial neural networks were chosen to explore this relationship due to their ability to perform nonlinear statistical modeling [31]. Other machine learning approaches, such as support vector machines and decision trees are ill-suited to this problem as they primarily perform a classification or clustering function and therefore do not provide for a continuous differentiable output. The advantages of neural networks include requiring less formal statistical training, the ability to detect complex nonlinear relationships between independent and dependent variables, the ability to discover all possible interactions between predictor variables, and the ability to use multiple training algorithms [31]. Artificial neural network analysis also has its disadvantages, most notably in its "black box" nature, the greater computational expense, the tendency to "over-fit", and the empirical nature of the model development [32]. As the purpose of this research entails developing a model which reliably generalizes between the input graph properties and the assembly time, thus obviating the need for physical meaning between them, the "black-box" nature is acceptable. The issue of over-fitting is addressed in training by instituting an early stopping algorithm as well as withholding samples from training entirely to test generalization on non-training data. Table 2 summarizes the results of several studies concerning the applicability and performance of artificial neural networks as compared to other analysis methods. For a more comprehensive assessment of the literature see [32].

Table 2: Artificial neural network comparisons in literature

Application of ANN	What ANN was compared to	Conclusions	Ref.
Predict dynamic nonlinear systems	Statistical Models	ANNs provide satisfactory performance in forecasting	[32]
Forecasting	Box-Jenkins automatic forecasting expert system	Similar Results	[33]
Nonlinear statistical modeling of medical outcomes	Logistic regression	Neural Networks preferred when primary goal is outcome prediction	[31]
Cost Estimation of Steel Pipe Bending	Linear regression	Neural Network	[34]
Prediction of Commodity Prices	Logistic regression	ANNs are consistently better and find more turning points	[35]

Two critical factors for this analysis are the inputs and the targets. The input for the analysis is the vector of graph properties for each connectivity graph. It should be noted that the connectivity graph at any time represents all of the connections made up to that point in time. This includes the execution of all assembly tasks that are required to make all of the connections present in the connectivity graph. Therefore, the graph property vector for a connectivity graph is to be mapped to the total assembly time up to that point. To determine the time for an isolated assembly step or steps the estimated assembly time prior to that step must be subtracted from the total estimated time including the step.

The target for the mapping is the MTM-based assembly time estimate provided by an automotive OEM. These estimates, the result of a formal study performed by time study personnel, were conducted using a company specific adaptation of MTM-UAS. Again, the aim of the methods proposed here is not to replace the formal, late-stage time study, but rather to provide an assembly time estimate much earlier in the product life cycle to enable the automation of such a method. In that formal time studies come late in the product life-cycle, after production has begun, the added level of detail and analysis time yields a more accurate estimate. Therefore, the assembly time estimates provided by the OEM implementation of MTM-UAS are used as target values in this study.

The process of building the model scheme is shown in more detail in Figure 9. The graph property vectors representing 19 connectivity graphs and their associated assembly times are used as inputs and targets. Graph property vectors for 5 connectivity graphs and their associated assembly times are withheld from training for later validation, however. One hundred simulations are then performed for each of 189 different ANN architectures, followed by a probability density function generated for each of the architectures. The architectures are then evaluated based on the probability of predicting assembly times to within fifteen percent of the target time. This is calculated by integrating the area under the probability density plot between the upper and lower fifteen percent bounds. Finally, a combination of the 100 predicted assembly times from the five best performing neural networks is used to generate a probability density plot. It is expected that the combination of the top performing architectures will enable the model to more accurately predict assembly times for different vehicle areas.

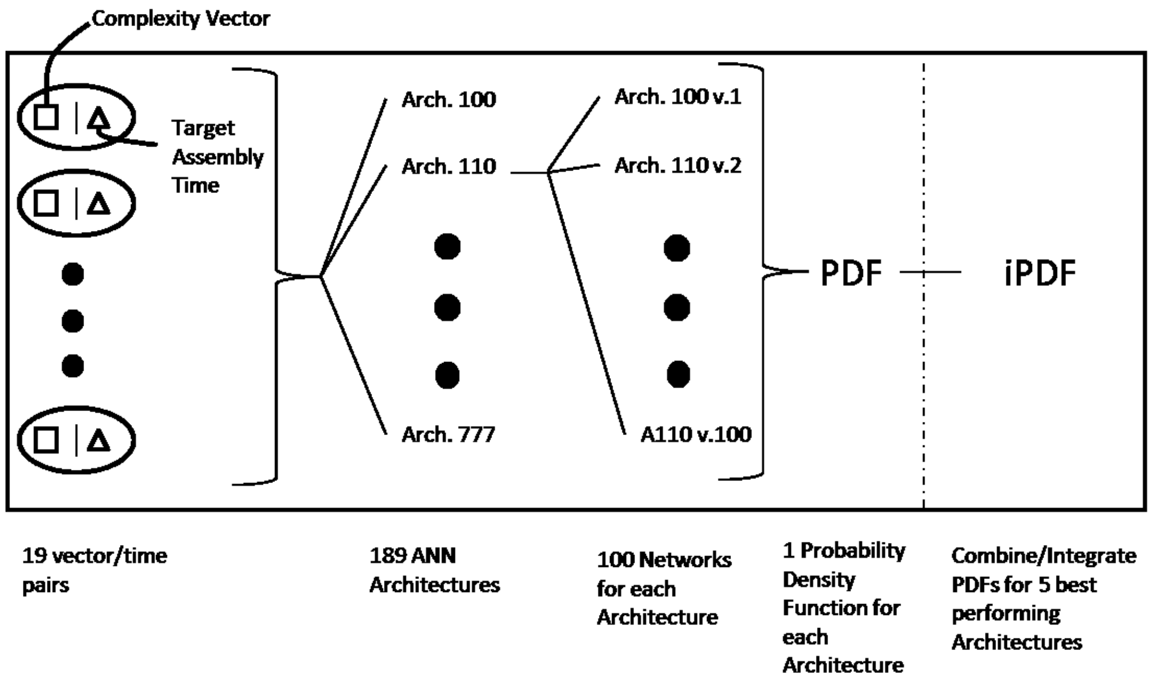

Figure 9: Model building process

Once the model is built and tested to understand the degree of accuracy, it can be used to estimate assembly times. The process of using the model is illustrated in Figure 10. Here, a graph property vector representing a connectivity graph is used as the input to the model. Then, the model is simulated using the five top-performing ANN architectures. Finally, a probability density function is generated using 500 ANN replications (100 from each of the five architectures). This probability density function can be used to gain an understanding of the predicted assembly time.

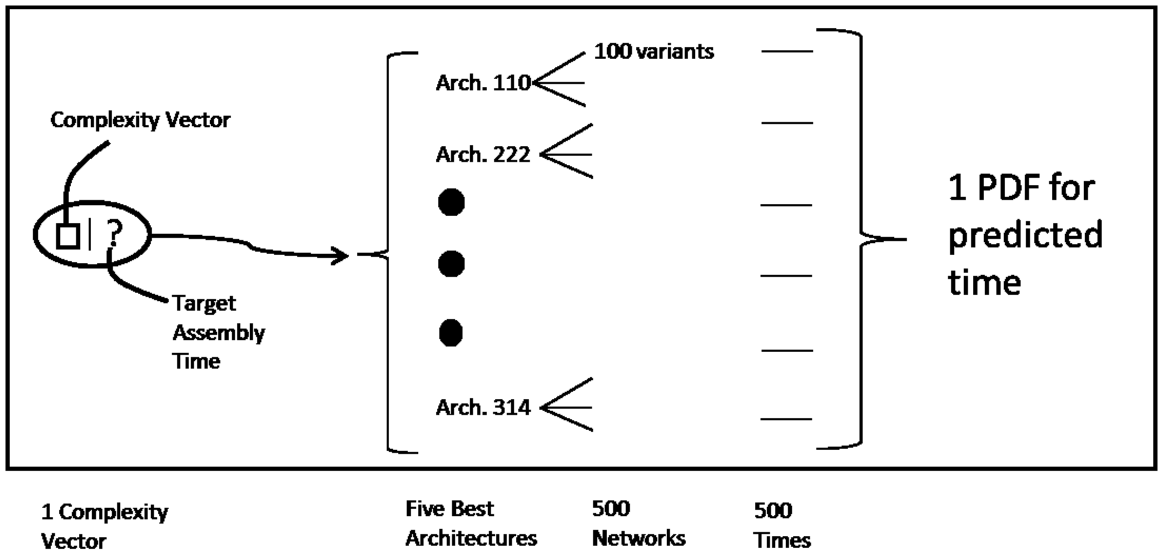

Figure 10: Process of using the model

Roughly 20%, or 5 of the 24 graphs, are omitted from training for external testing, leaving 19 of the 24 data points for use as inputs and targets for the analysis. The remaining five are used after the appropriate ANN architectures are selected to determine the accuracy to which the network was capable of mapping connectivity graphs to MTM-based assembly time estimates.

One hundred and eighty nine different ANN architectures, ranging from a single layer with a single neuron to three layers with five neurons in each layer, were simulated to identify the most appropriate for this mapping. Architectures with one layer were simulated with a neuron count ranging from one to fifteen and architectures with two layers were simulated with one neuron, each of which had up to seven neurons in each layer. Finally, architectures with three layers were simulated with combinations of up to five neurons. This capping of the number of hidden layer units at 15, places the hidden unit count between the input unit count and output unit count, thus promoting generalization of the output [36].

An ANN will not generate the exact same mapping even when given the same inputs and outputs due to the different initial conditions in each training instance. The most notable of these variations is the use of an early-stopping validation algorithm, which withholds a 15% subset of the training data for use in testing that while training progresses generalization also continues to improve. Without this measure, the network could be expected to consistently over-fit to the training data. As the subset of the training data to be used for early-stopping is selected at random, each architecture is trained for 100 replications of the ANN, enabling the generation of probability density functions describing the typical behavior of the 189 ANN architectures. A probability density plot for an ANN architecture consisting of three layers with five neurons in the first two layers and one neuron in the third layer evaluated for one connectivity graph is shown in Figure 11. An equivalent plot is generated for all combinations of ANN architectures and connectivity graphs.

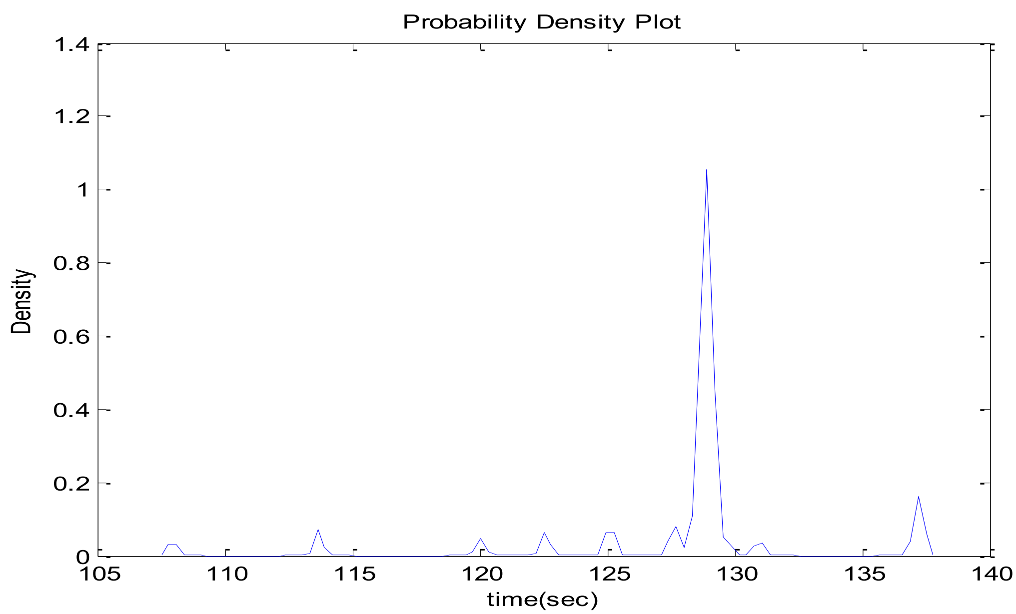

Figure 11: Probability density plot for connectivity graph #5 and ANN structure 134

The next step entails analyzing where the probability function lies in relation to the target value and acceptable range of assembly times. Figure 12 shows the probability density plot as well as the target value and associated 15% range of the target values for the assembly. The mean predicted value is shown by the red dotted line while the black dotted line shows the target assembly time value. Note that the predicted values fall well within the 15% acceptable range.

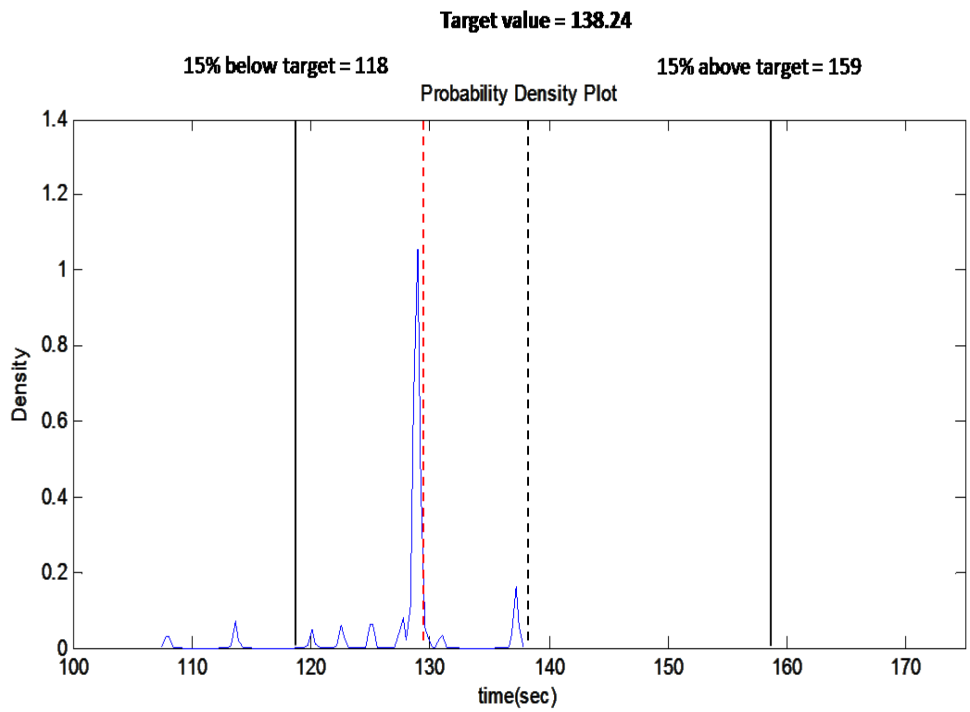

Figure 12: Probability density plot with target value and 15% range

Next, the artificial neural network architecture with the highest probability of estimation to within fifteen percent of the target is identified for use in future assembly time estimation. This probability is based on the assembly time estimation for the five connectivity graphs which are omitted from training. Note that four of the ANN architectures resulted in probabilities of greater than one, the result of integration errors in computation. To determine the probability, the area under the probability density plot between the upper and lower limits was calculated using trapezoidal approximate integration. Also note that some of the architectures result in a probability of zero, suggesting that none of the simulated assembly times fall within a fifteen percent range of the target times.

The most appropriate ANN architecture was determined based on the data presented in Table 3. Columns two and three represent the probability that the minimum and mean of the estimation will be within 15% of the target value for the five validation sets. In the first column identifying the ANN architecture, it is clear that each of these have a high probability of estimating the assembly time to within 15% of the MTM-based assembly time estimate. Since there is no significant difference for any of the cases between performance on the predicted mean and minimum, the ANN architectures with the highest mean probability are selected. The top ANN architecture consists of three layers with three neurons in the first layer, four in the second, and five neurons in the third.

Table 3: Evaluation of ANN structure performance

ANN Architecture	Minimum Probability of Estimation to within 15% of Target	Mean Probability of Estimation to within 15% of Target
134-[3,4,5]	0.99991	0.99994
188-[5,5,4]	0.99990	0.99992
153-[4,3,4]	0.99983	0.99990
157-[4,4,3]	0.99983	0.99990
77-[1,3,3]	0.99985	0.99989
32-[3,3]	0.99977	0.99987
170-[5,2,1]	0.99966	0.99986
69-[1,1,5]	0.99970	0.99983

7 RESULTS

As previously mentioned, five of the 24 data sets are omitted from training for validation separate from any validation set generated by the training algorithm. These data points are used to test the ability of the selected ANN architecture to generalize new data. The top network is trained two additional times using the selected architecture while omitting different sets of five data points. During the original training and testing, the five largest connectivity graphs are omitted for later use as test points, for purposes of determining the forward prediction of the model. The second training omits every fifth data point (when ranked in terms of connectivity graph size) beginning with the smallest connectivity graph. The third validation set consists of every fifth connectivity graph starting with the fifth. The second and third validation sets are used to determine the applicability of the model to a wide range of graph sizes after training on a representative sample. The final validation set consists of using a combination of the top performing architectures.

7.1 <u>First Validation Set (Last Five Omitted)</u>

The ANN population is trained using the first 19 data points as inputs and targets and the top ANN architecture. This network was then simulated with the input being the graph property vectors of the final five data points. The results of the ANN validation simulation, shown in Table 4, indicates the estimated time from the MTM-based time study from the automotive OEM and the estimated time by the ANN model for each data point. The estimate from the model is the mean output from a population of 100 ANN replications. The error represents the percentage error of the model as compared to the formal time study estimate. The final three columns show the probability of each of the 100 ANN replications predicting a time to within a specified percentage of the formal time study estimate, indicating the forward predicting capability of this model within an area of the vehicle. In other words, the ANN trained on a set of smaller connectivity graphs can predict assembly times for graphs larger than those used in the training.

Table 4: Prediction results for first validation set

Graph	MTM [s]	ANN [s]	Error	Probability of prediction within:		
				10%	5%	1%
20	352.92	350.30	-0.7%	1	1	.52
21	362.64	370.18	2.1%	1	.99	.27
22	366.96	384.45	4.8%	1	.53	0
23	376.62	386.09	2.5%	1	.72	.18
24	392.94	386.94	-1.5%	.99	.58	.12

The results of this validation set show a successful mapping between the complexity of connectivity graphs and MTM-based assembly time estimates for a specialized case. This validation, however, is only applicable to forward prediction of assembly times within a specific vehicle area, and does not imply any mapping of assembly times in other parts of a vehicle. This validation also does not indicate the model's ability to predict assembly times for connectivity graphs that are smaller than the input graphs or for a range of sizes of connectivity graphs. As such, the authors performed a second validation, which is discussed in Section 7.2.

7.2 Second Validation Set (Every Fifth Omitted Starting with Graph 1)

The second validation set seeks to explore the model's ability to predict assembly times for a wider range of connectivity graph sizes for cases in which the model is trained on a representative sample of the population. The ANN population for this case was trained using 19 of the 24 collected data points as inputs and targets and top performing ANN architecture. This ANN population was then simulated with the input being the graph property vectors of every fifth data point, or connectivity graph, starting with the smallest.

The results of the second neural network validation simulation, shown in Table 5 using the format established in Table 4, indicate that the only connectivity graph which was not predicted with an error of less than 15% was the first. This connectivity graph is the smallest graph in any of the collected data, and is consequently smaller than the training set. This error, along with the low probability of estimation to within 10% highlights this procedure's lack of ability to predict assembly times for graphs smaller than those used in the training.

Table 5: Prediction results for second validation set

Graph	MTM [s]	ANN [s]	Error	Probability of prediction within:		
				10%	5%	1%
1	39.3	45.73	16.4%	.23	.11	.03
6	187.38	202.29	8.0%	1	0	0
11	226.26	225.10	-0.5%	1	.96	.02
16	318.78	315.40	-1.1%	1	1	.47
21	362.64	372.90	2.8%	1	1	0

Though the results of the second validation set show great potential for the deployment of the method, it indicates a weakness in the model's ability to predict assembly times for graphs smaller than those present in the network's training. As a result, the goal of the third validation set is to determine the model's applicability when trained on a representative sample of the population with upper and lower bounded data.

7.3 Third Validation Set (Every Fifth Omitted Starting With Graph 5)

The third validation set omits every fifth graph in terms of size, but starts with graph 5. This validation seeks to examine the capabilities of the method when trained on a representative sample which contains the upper and lower bounded graph sizes. Here, the graphs omitted for validation are between the largest and smallest graphs used in the training of the ANN population.

The results of the third ANN validation set, shown in Table 6 in a format similar to the previous validation sets, shows that 100% of the predictions for this validation fall within 25% of the target value. Similarly, for every graph except graph 5, all simulations are within 15%.

Table 6: Prediction results for third validation set

Graph	MTM [s]	ANN [s]	Error	Probability of prediction to within:				
				25%	15%	10%	5%	1%
5	138.24	122.67	-11.3%	1	.67	.44	.22	.04
10	223.20	228.28	2.3%	1	1	1	.95	.29
15	311.04	311.43	0.1%	1	1	1	1	.98
20	352.92	351.65	-0.4%	1	1	1	1	1
24	392.94	408.98	4.1%	1	1	1	.66	0

The third and final validation set demonstrates that an ANN population can successfully predict assembly times when trained on a representative sample of the population and when the extreme data points in terms of graph size are included in the training.

7.4 Fourth Validation Set (Every Fifth Held Back Starting with Graph 5) using best five architectures

In the final validation set within the instrument panel area of the vehicle a combination of the top five performing artificial neural networks (134, 188, 153, 157, and 77) is used. An ANN population with equal members of each of the five architectures is simulated for the test cases to obtain the analytical results. For each architecture, 100 replications exist within the population, resulting in 500 predicted times for each test case. Finally, a probability density plot is generated using the 500 assembly times. Table 7 shows a summary of the results of the fourth validation set, the results of which are quite accurate. The mean predicted values for each of the test points are within 2.5 percent of the target assembly time, indicating that the use of a combination of top performing architectures is helpful in successfully mapping the complexity of connectivity graphs to assembly times.

Table 7: Prediction results for fourth validation set

Graph	**MTM [s]**	**ANN [s]**	**% Error**	**Probability of prediction to within:**		
				10%	**5%**	**1%**
5	138.24	138.15	-0.1%	.51	.08	.02
10	223.20	223.20	0.9%	1	1	.37
15	311.04	311.04	1.1%	1	1	.46
20	352.92	352.92	-0.3%	1	.94	.94
24	392.94	392.94	-2.2%	1	.81	.01

Table 8 shows a summary of the validation results, the results of which within a particular vehicle area are also accurate. The best results are obtained when the five top-performing neural networks are used to simulate the assembly times, however, as shown in the last column where the maximum error is less than 2.5 percent.

Table 8: Summary of validation results

Test	Last 5 graphs	Every 5th graph, starting with #1	Every 5th graph, starting with #5	Every 5th graph, best 5 architectures
1	-0.7%	16.4%	-11.3%	-0.1%
2	2.1%	8.0%	2.3%	0.9%
3	4.8%	-0.5%	0.1%	1.1%
4	2.5%	-1.1%	-0.4%	-0.3%
5	-1.5%	2.8%	4.1%	-2.2%

8 EXTERNAL GENERALIZATION

The results in Section 7 show that the capability of the connectivity method in predicting assembly times when tested on the vehicle area used for the ANN training. The application of this mapping to other areas of the vehicle or to non-automotive assemblies has yet to be explored, however.

8.1 Application to vehicle sub-systems

The first question to be addressed in the external generalization is if the ANN trained only on the instrument panel is capable of predicting assembly times for other vehicle parts. The insulating panel will serve as the other vehicle area for this analysis.

The ANN developed in Section 7 was simulated to predict assembly times for the graph property vectors of the insulating panel's connectivity graphs, the results of which are shown in Table 9. The errors in this validation set range from negative forty-two percent to 435%, a lack of consistency indicating the difficulty of using a model trained on one specific vehicle area to estimate assembly times for another.

Table 9: External Generalization Results

Graph	MTM [s]	ANN [s]	% Error
1	20	69	231
2	27	15	-42
3	30	34	15
4	34	184	435
5	47	181	284
6	60	201	235
7	80	259	223
8	95	301	214

The results shown in Table 9 elucidated the second question which the authors used to determine if a neural network trained on complexity vectors and times from multiple vehicle areas can produce accurate estimates for the different vehicle areas. An ANN population 7 is trained on the instrument panel as performed in Section 7 and three of the eight connectivity graphs collected from the insulating panel.

Table 10: Connectivity results for insulating panel assembly

Graph	MTM [s]	ANN [s]	% Error
2	27.18	47.67	75%
4	34.56	97.97	183%
5	47.34	84.52	79%
7	80.28	54.66	-32%
8	95.94	67.12	-30%

The results of this analysis, presented in Table 10, clearly show the possibility of determining with some degree of accuracy, assembly times for multiple vehicle areas. The results are not as accurate as those within the same vehicle area as the training set, however. This lower level of accuracy is likely because a much larger number of instrument panel processes was used than insulating panel processes.

Since the training set now includes data from the insulating panel, the accuracy of the estimation of instrument panel assembly times may have been reduced. The assembly time estimation for the instrument panel processes both before and after the addition of insulating panel processes are presented in Table 11. When the ANN population is trained on insulating and instrument panel assemblies, the accuracy of the assembly time estimates for the instrument panel is not decreased. This stasis may suggest that a higher number of processes used in the neural network results in a greater accuracy in assembly time prediction regardless of the vehicle areas analyzed.

Table 11: Connectivity results for instrument panel assembly

Graph	ANN [s]	MTM [s]	% Error	% Error (IP-only Training)
IPA-5	199	187	6%	12%
IPA-10	244	226	8%	-3%
IPA-15	313	318	-2%	-3%
IPA-20	361	362	0%	-2%

8.2 Application to Consumer Products

The next step involves determining the applicability of the newly developed ANN model to assembly time estimation outside of automotive assembly processes. The original connectivity training was developed for use on consumer product assemblies [15]. To determine the new model’s ability to predict assembly times for consumer products, the model is tested on three products used in the initial connectivity research including a mixer, a chopper, and a Tweel™ prototype.

The results of this analysis, presented in Table 12, clearly indicate the necessity of training the model on data specific to the application for which it's used. Furthermore, it is likely that different automotive OEMs must train the model specifically for application in the respective company.

Table 12: Results of model application to non-automotive assemblies

Graph	B&D [s]	ANN [s]	% Error
Mixer	136	180	32%
Tweel™	13561	228	-98%
Chopper	228	35	-84%

9 CONCLUSIONS & FUTURE WORK

The authors explored the possibility of creating an automatable, early-stage assembly time estimation model based on part connectivity for the automotive industry. Though the existing regression based assembly time estimation methods using connectivity information were not extendable to automotive assemblies, an artificial neural network mapping approach incorporating populations of ANNs was proposed and evaluated. Subsequent evaluation indicated the suitability of this ANN approach in predicting assembly times for intermediate steps when trained on a representative sample with upper and lower bounds, suggesting its possible use as an accelerating supplement for formal time studies. This ANN approach also performed moderately in predicting the assembly time of steps beyond the upper bound, indicative progress towards creating an automated early-stage assembly time estimation tool.

The authors recommend that a model used for assembly time estimation be trained on a set of graph property vectors representing the upper and lower bounded connectivity graphs, and a representative sample of intermediate graphs. Additionally, a population of at least five of the top performing network architectures should be used to generate a probability density plot representing the estimated assembly time. Predictions generated from this model are only applicable to vehicle areas on which it has been trained and is not viable for direct use in other industries. The authors further hypothesized that the company-specific nature of the model would prevent its use across automotive OEMs.

Research results also indicate that a higher number of training data points representing a sample of each of the vehicle areas may result in a model that can accurately predict assembly times for all vehicle areas. The development of such a model will require significantly more effort due to the large number of samples required, however. There is as yet no definite rule does for sample sizes; the size of the training set depends on the network structure, training method, and the complexity of the problem [32]. It is possible, however, that the result would be a product-based assembly time estimation model capable of providing accurate results early in the automotive product life-cycle.

ACKNOWLEDGEMENTS

The authors gratefully recognize the support provided by BMW in the product-process integration project through both student support and data availability. The work presented here does not necessary represent the views of BMW.[6]

REFERENCES

[1] Sharma U., 2003, "Implementing lean principles with the Six Sigma advantage: how a battery company realized significant improvements," J. Organ. Excell., **22**(3), pp. 43–52.

[2] Gunasekaran A., 1999, "Agile manufacturing: a framework for research and development," Int. J. Prod. Econ., **62**(1), pp. 87–105.

[3] Ameri F., and Dutta D., 2005, "Product lifecycle management: closing the knowledge loops," Comput. Des. Appl., **2**(5), pp. 577–590.

[4] Boothroyd G., and Alting L., 1992, "Design for Assembly and Disassembly," CIRP Ann. Technol., **41**(August), pp. 625–636.

[5] Choi C. K., and Ip W. H., 1999, "A comparison of MTM and RTM," Work Study, **48**(2), pp. 57–61.

[6] Mathieson J., Wallace B., and Summers J. D., "Estimating Assembly Time with Connective Complexity Metric Based Surrogate Models," Int. J. Comput. Integr. Manuf., **on-line**(in press).

[7] Mohd Naim Z., 2009, "Design for assembly and application using Hitachi assemblability evaluation method," Universiti Malaysia Pahang.

[8] Boothroyd G., Dewhurst P., and Knight W. A., 2011, Product Design for Manufacture and Assembly, CRC Press, Boca Raton.

[9] Whitney D. E., 2004, Mechanical assemblies, Oxford University Press, Cambridge, MA.

[10] Otto K. N., and Wood K. L., 1998, "Product evolution: a reverse engineering and redesign methodology," Res. Eng. Des., **10**(4), pp. 226–243.

[11] Ben-Arieh D., and Qian L., 2003, "Activity-based cost management for design and development stage," Int. J. Prod. Econ., **83**(2), pp. 169–183.

[12] Duverlie P., and Castelain J. M., 1999, "Cost estimation during design step: parametric method versus case based reasoning method," Int. J. Adv. Manuf. Technol., **15**(12), pp. 895–906.
[13] Hoover C. W., and Jones J. B., 1991, Improving Engineering Design: Designing for Competitive Advantage, National Academies Press, Washington, DC.
[14] Jiao J., and Tseng M. M., 1999, "A pragmatic approach to product costing based on standard time estimation," Int. J. Oper. Prod. Manag., **19**(7), pp. 738–755.
[15] Mathieson J. L., Wallace B. A., and Summers J. D., 2013, "Assembly time modelling through connective complexity metrics," Int. J. Comput. Integr. Manuf., **on-line**(in press), pp. 1–13.
[16] Weber J., 2009, Automotive development processes: Processes for successful customer oriented vehicle development, Springer, New York, NY.
[17] Miller M., Griese D., Summers J. D., Peterson M., and Mocko G. M., 2012, "Representation: Installation Process Step Instructions as an Automated Assembly Time Estimation Tool," ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, ASME, Chicago, IL, p. CIE–70109.
[18] Maynard H., Stegemerten G. J., and Schwab J., 1948, Methods-Time Measurement, McGraw-Hill, New York, NY.
[19] Laring J., 2002, "MTM-based ergonomic workload analysis," Int. J. Ind. Ergon., **30**(3), pp. 135–148.
[20] Cakmakci M., and Karasu M. K., 2007, "Set-up time reduction process and integrated predetermined time system MTM-UAS: A study of application in a large size company of automobile industry," Int. J. Adv. Manuf. Technol., **33**(3), pp. 334–344.
[21] Owensby E., Shanthakumar A., Rayate V., Namouz E. Z., Summers J. D., and Owensby J. E., 2011, "Evaluation and Comparison of Two Design for Assembly Methods: Subjectivity of Information," ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, ASME, Washington, DC, pp. DETC2011–47530.
[22] Owensby J. E., Namouz E. Z., Shanthakumar A., and Summers J. D., 2012, "Representation: Extracting Mate Complexity from Assembly Models to Automatically Predict Assembly Times," ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, ASME, Chicago, IL, pp. DETC2012–70995.
[23] Snider M., Summers J. D., Teegavarapu S., and Mocko G. M., 2008, "Database Support for Reverse Engineering, Product Teardown, and Redesign as Integrated into a Mechanical Engineering Course," Comput. Educ. J., **18**(4), pp. 9–21.
[24] Snider M., Teegavarapu S., Hesser D. S., and Summers J. D., 2006, "Augmenting tools for reverse engineering methods," ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, ASME, Philadelphia, PA, p. DAC–99676.
[25] Mathew A., and Rao C. S. P., 2010, "A CAD system for extraction of mating features in an assembly," Assem. Autom., **30**(2), pp. 142–146.
[26] Sambhoos K., Koc B., and Nagi R., 2009, "Extracting Assembly Mating Graphs for Assembly Variant Design," J. Comput. Inf. Sci. Eng., **9**, p. 34501.

[27] Namouz E. Z., and Summers J. D., 2013, “Complexity Connectivity Metrics – Predicting Assembly Times with Low Fidelity Assembly CAD Models,” Springer, Bochum, Germany.

[28] Mathieson J. L., Shanthakumar A., Sen C., Arlitt R., Summers J. D., and Stone R., 2011, “Complexity as a Surrogate Mapping between Function Models and Market Value,” ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. DETC2011–47481.

[29] Mathieson J. L., Sen C., and Summers J. D., 2009, “Information Generation in the Design Process,” ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference Computers and Information in Engineering Conference, ASME, San Diego, CA, pp. DETC2009–87359.

[30] Mathieson J. L., Miller M., and Summers J. D., 2011, “A Protocol for Connective Complexity Tracking in the Engineering Design Process,” International Conference on Engineering Design 2011, Copenhagen, Denmark, p. No–657.

[31] Tu J. V., 1996, “Advantages and Disadvantages of Using Artificial Neural Networks versus Logistic Regression for Predicting Medical Outcomes,” J. Clin. Epidemiol, **49**(11), pp. 1225–1231.

[32] Zhang G., Patuwo B. E., and Hu M. Y., 1998, “Forecasting with Artificial Neural Network: The State of the Art,” Int. J. Forecast., **14**(1), pp. 35–62.

[33] Sharda R., and Patil R. B., 1992, “Connectionist Approach to Time Series Prediction: An Empirical Test,” J. Intell. Manuf., **3**, pp. 317–323.

[34] Shtub A., and Versano R., 1999, “Estimating the Cost of Steel Pipe Bending: A Comparison between Neural Networks and Regression Analysis,” Int. J. Prod. Econ., **62**, pp. 201–207.

[35] Kohzadi N., Boyd M. S., Kermanshahi B., Kaastra I., and Khozadi N., 1996, “A Comparison of Artificial Neural Networks and Time Series Models for Forecasting Commodity Prices,” Neurocomputing, **10**(2), pp. 161–181.

[36] Blum A., 1992, Neural networks in C++: an object-oriented framework for building connectionist systems, John Wiley & Sons, Inc.